# 7. Fuzzy Logic

We are in the process of discussing how automated systems can deal with uncertainty. In the last chapter we discussed a number of methods to do this, among others, probability theory, in particular Bayes's method. Instead of using crisp numbers, we used probabilities. As we know, in the real world, there is no certainty. Thus it is better to associate the probability of an event (say a disease) with a particular symptom (e.g. the result of a test). We saw that while Bayesian methods have their definite merits (they show us how we can deal with uncertainty in mathematically correct ways by including a priori probabilities), they also have their problems (some numbers are not available and hard to define, the real world changes). So, they do not represent the definite solution to deal with uncertainty in the real world. We briefly mentioned certainty factors. But the latter also have their problems when put to work in the real world, as the large number of failed expert systems projects demonstrates. Many of these projects in fact employed certainty factors (or something similar). So, it seams that we have not yet found the final solution. Many alternatives have been suggested. A prominent one is fuzzy logic.

As we will see, one of the reasons why fuzzy logic is so popular is that it has a highly appealing way to deal with the real world. Rather than trying to define how things "really are", fuzzy logic takes account of the fact that things in the real world are not either this way or the other way, but most of the relevant properties are in fact gradual ones.

Fuzzy logic has been very successful. It has been, and still is, especially popular in Japan, where fuzzy logic has been introduced into all types of consumer products with great determination. Nowadays, "Fuzzy", in Japanese

フアジイ

has become something like a quality seal. In table 7.1 a number of applications of fuzzy logic are given (more applications can be found in Dubois et al. — the latter also includes theoretical papers).

Table 7.1: Applications of fuzzy logic in Japan and Korea (*fielded products*) (1992). Based on Kosko, B. (1992). *Fuzzy thinking. The new science of fuzzy logic.* New York, NY.: Hyperion. Additional examples have been included.

| *Product* | *Company* | *Fuzzy logic role* |
|---|---|---|
| Air conditioner | Hitachi, Matsushita, Mitsubishi, Sharp | Prevents overshoot-undershoot temperature oscillation and consumes less on-off power |
| Anti-lock brakes | Nissan | Controls brakes in hazardous cases based on car speed and acceleration and on wheel speed and acceleration |
| Auto engine | NOK/Nissan | Controls fuel injection and ignition based on throttle setting, oxygen content, cooling water temperature, RPM, fuel volume, crank angle, knocking, and manifold pressure |
| Auto transmission | Honda, Nissan, Subaru | Selects gear ratio based on engine load, driving style, and road conditions |
| Chemical mixer | Fuji Electric | Mixes chemicals based on plant conditions |

| *Product* | *Company* | *Fuzzy logic role* |
|---|---|---|
| Copy machine | Canon | Adjusts drum voltage based on picture density, temperature, and humidity |
| Cruise control | Isuzu, Nissan, Mitsubishi | Adjusts throttle setting to set speed based on car speed and acceleration |
| Dishwasher | Matsushita | Adjusts cleaning cycle and rinse and wash strategies based on the number of dishes and on the type and amount of food encrusted on the dishes |
| Dryer | Matsushita | Converts load size, fabric type, and flow of hot air to drying times and strategies |
| Elevator control | Fujitec, Mitsubishi Electric, Toshiba | Reduces waiting time based on passenger traffic |
| Factory control | Omron | Schedules tasks and assembly line strategies |
| Golf diagnostic system | Maruman Golf | Selects golf club based on golfer's physique and swing |
| Health management | Omron | Over 500 fuzzy rules track and evaluate an employee's health and fitness |
| Humidifier | Casio | Adjusts moisture content to room conditions |
| Iron mill control | Nippon Steel | Mixes inputs and sets temperatures and times |
| Kiln control | Mitsubishi Chemical | Mixes cement |
| Microwave oven | Hitachi, Sanyo, Sharp, Toshiba | Sets and tunes power and cooking strategy |
| Palmtop computer | Sony | Recognizes handwritten Kanji characters |
| Paper industry | Cellulose do Caima, Portugal | Pulp production |
| Plasma etching | Mitsubishi Electric | Sets etch time and strategy |
| Refrigerator | Sharp | Sets defrosting and cooling times based on usage. A neural network learns the user's usage habits and tunes the fuzzy rules accordingly. |
| Rice cooker | Matsushita, Sanyo | Sets cooking time and method based on steam, temperature, and rice volume |
| Shower system | Matsushita (Panasonic) | Suppresses variations in water temperature |

| *Product* | *Company* | *Fuzzy logic role* |
|---|---|---|
| Still camera | Canon, Minolta | Finds subject anywhere in frame, adjusts autofocus |
| Stock trading | Yamaichi | Manages portfolio of Japanese stocks based on macroeconomic and microeconomic data |
| Subway control | LIFE Institute, Yokohama | Controls the subway during peak hours. |
| Television | Goldstar (Korea), Hitachi, Samsung (Korea), Sony | Adjusts screen color and texture for each frame and stabilizes volume based on viewer's room location |
| Translator | Epson | Recognizes, translates word in pencil-size unit |
| Toaster | Sony | Sets toasting time and heat strategy for each bread type |
| Vacuum cleaner | Hitachi, Matsushita, Toshiba | Sets motor-suction strategy based on dust quantity and floor type |
| Video camcorder | Canon, Sanyo | Adjusts autofocus and lighting |
| Video camcorder | Matsushita (Panasonic) | Cancels handheld jittering and adjusts autofocus |
| Washing machine | Daewoo (Korea), Goldstar (Korea), Hitachi, Matsushita, Samsung (Korea), Sanyo, Sharp | Adjusts washing strategy based on sensed dirt level, fabric type, load size, and water level. Some models use neural networks to tune rules to user's tastes. |

The goal of this chapter is to introduce fuzzy logic and to show how it can be applied to real-world problems. Moreover, we will put this technology into the broader context of real-world computing.

We will proceed as follows. First we will introduce some of the basic concepts and compare them to classical probabilistic concepts. Surprisingly enough we will discover some fundamental differences. Then we introduce the so-called "Kosko Cube", an instrument that greatly helps to visualize fuzzy sets. We then look at how fuzzy rule systems work and how they can be made adaptive. We then briefly look at hard- and software for fuzzy logic applications. Finally we discuss some of the success factors. We end we a note on cognitive science.

## 7.1 Fuzziness vs. randomness

Bart Kosko, one of the champions of fuzzy logic starts his book, "Fuzzy thinking: the new science of fuzzy logic" as follows:

> "Hold an apple in your hand. Is it an apple? Yes. The object in your hand belongs to the clumps of space-time we call the set of apples — all apples anywhere, ever. Now take a bit, chew it, swallow it. Let your digestive tract take apart the apple's molecules. Is the object in your hand still an apple? Yes or no? Take another bite. Is

the new object still an apple? Take another bite, and so on down to void." (Kosko, 1992, p. 4).

Initially the apple is clearly an apple. But as the number of pieces bitten off increases it gradually looses the property of "apple-ness". At the end, when the apple has been completely eaten, it is no longer a member of the class of apples. The basic idea of fuzzy logic is to associate a number with each object indicating the degree to which it belongs to a particular class of objects. Initially, for our apple, this number will be 1 or close to 1. At the end it will be zero, since the apple ceases to exist. In between it will be slowly decreasing. The function that associates a number with the object is called the *membership function*. In classical set theory this function is either 1 (the object belongs to the set) or 0 (the object does not belong to the set); it is also called the "characteristic" function.

In the last chapter we discussed probabilities as a way to deal with uncertainty in the real world. Probabilities are also numbers between 0 and 1, just like the membership functions in fuzzy logic. We might suspect that they are pretty much the same. But it turns out that this is by no means the case. Just think for a moment about the apple. Assume that you have eaten two thirds of it and it really begins to loose its "apple-ness". So, the value of the membership function might be something like .3. We would say, the degree to which it is an apple is .3. We could ask: "What is the probability that this is an apple?" Would that make sense? The probability always requires that an event takes place or does not take place, the only thing is that it is *not known* whether it has taken place or not.

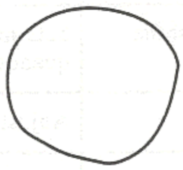

Figure 7.1: a fuzzy circle. It makes more sense to postulate that this is a circle to a certain degree (say, .6) than to say that the probability that this is a circle is .6.

Another example is the circle shown in figure 7.1. Is it a circle or is it not a circle? The interesting point about this example is that everything is given, all factors are known, but the uncertainty remains. Let us take an example from the real world: In American weather reports we often hear statements like "there is a 20% chance of light rain tomorrow". This includes probability and fuzziness. There is a certain probability that it will rain. If it were only that then the underlying assumption would be that it can be determined with 100% certainty whether it is raining or not. As we all know, there are many weather conditions where it is unclear if it is raining or not: heavy fog with small water particles slowly drifting towards the ground, something in between fog and drizzle, or just a few raindrops, but no "real" rain. Fuzziness is a kind of intrinsic property of elements of categories. Probability applies to any event in the real world. An event in the real world is never certain to happen (although the probability can be very high as in the case of dropping a rock). Thus, in the real world we have both, probability and fuzziness.

## 7.2 Basic concepts

The original ideas were developed by Lotfi Zadeh of University of California Berkeley in 1965 (Zadeh, 1965). We briefly introduce some formalism in order to show some interesting differences between classical Western and Eastern thinking. The fundamental concept is the fuzzy set. Fuzzy logic designates a particular kind of inference calculus based on fuzzy sets. Fuzzy systems employ fuzzy sets and fuzzy logic.

We always start with a universe of discourse, i.e. all the objects that we could possibly talk about. Let $X$ be the universe of discourse, and $A$ a set of elements. It can be defined as follows:

$$A: \{x, m_A(x) = 1\}; m_A(x) = \begin{cases} 1; x \in A \\ 0; x \notin A \end{cases} \quad (1)$$

If instead we have:

$m_A(x) \in [0,1]$

we have a fuzzy set $A$. Think again of the apple. The classical way of looking at the issue is that an object either IS an apple (the characteristic function has a value of 1) or IS-NOT an apple (the characteristic function has a value of 0). The membership function in a fuzzy set framework assigns a value between 0 and 1 to every element of the fuzzy set. This value indicates the degree of membership of the element to the set.

Let us look at an example of such a membership function. Assume we want to represent the concept of a "foreign car" (given that we live in the US) (see Mendel, 1995). For US citizens this is a very important concept since a large part of the US economy depends on the car industry. If foreign cars, especially Japanese cars, start dominating the market, the US economy is in trouble.

> "A car can be viewed as "domestic" or "foreign" from different perspectives. One perspective is that a car is domestic if it carries the name of a USA manufacturer, otherwise it is foreign. There is nothing fuzzy about this perspective; however, many people today feel that the distinction between a domestic and a foreign automobile is not as crisp as it once was, because many of the components for what we consider to be domestic cars (e.g. Ford, GM, and Chrysler) are produced outside of the USA. Additionally, some "foreign" cars are manufactured here in the USA. Consequently, one could think of the membership functions for domestic and foreign cars looking like $m_D(x)$ and $m_F(x)$ depicted in figure 7.2. Observe that a specific car (located along the horizontal axis by determining the percentage of its parts made in the USA) exists in both subsets simultaneously—domestic cars and foreign cars—but to different degrees of membership. For example, if our car has 75% of its parts made in the USA, then $m_D(75\%) = 0.9$ and $m_F(25\%) = 0.25$. (Mendel, 1995, pp. 348-349)

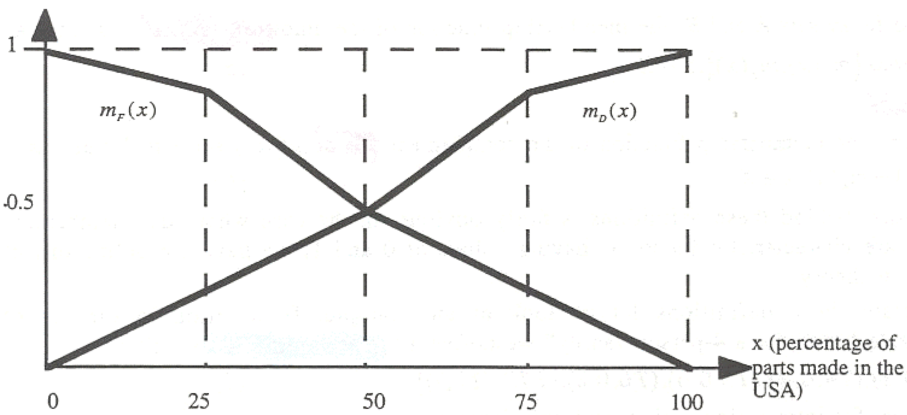


Figure 7.2: Membership functions for domestic and foreign cars, based on the percentage of parts in the car made in the USA. (Mendel, 1995, p. 349).

Even within the fuzzy logic community, there are different ways of viewing fuzzy sets. Let us call them the traditional one and the Kosko-interpretation. In order to provide some intuition we start with the traditional one. The latter is also the one which is most often used in the literature and corresponds largely to Zadeh's original paper. Then we introduce the Kosko-interpretation which has a lot of elegance and mathematical as well as intuitive appeal.

## The traditional view

Let us look at an example from natural language. A real estate agent wants to classify the house he offers to his clients (example taken from Zimmermann, 1991). How would we formalize the following phrase: "Comfortable house for a four-person family"? Let $X$ be the universe of discourse, consisting of the set of different types of house, i.e.:

$X: \{T1, T2, T3, \ldots, T8, T9, T10\}$

where the index in $Ti$ indicates the number of bedrooms in a house. Let us now assign the membership function to the house types in order to define the "comfortable house for a four-person family":

$$A: \{(T1, 0.2), (T2, 0.5), (T3, 0.8), (T4, 1), (T5, 0.7)(T6, 0.3)\} \quad (2)$$

For $T8$, $T9$, and $T10$ the membership function is assumed to be 0.

We can now define various operations.

*Cardinality*

In classical set theory this is the number of elements in a set. This is written as $M(A)$. For a finite fuzzy set we can write:

$$M(A) = \sum_{x \in X} m_A(x) \quad (3)$$

The intuition behind this formula is that the contribution of an element to the cardinality should be weighted with the membership function. An element that hardly belongs to the set (very small value of the membership function) contributes little to the cardinality of the set.

Applying formula (3) to the example (2) yields:

$M(A) = 0.2 + 0.5 + 0.8 + 1 + 0.7 + 0.3 = 3.5$

$= M$("comfortable house for 4-person family")

*Intersection*

Given two fuzzy sets $A$ and $B$, the membership function of the intersection $C = A \cap B$ is defined as

$$m_C(x) = \min\{m_A(x), m_B(x)\}, x \in X \quad (4)$$

*Union*

Given two fuzzy sets $A$ and $B$, the membership function of the union $D = A \cup B$ is defined as

$$m_D(x) = \max\{m_A(x), m_B(x)\}, x \in X \quad (5)$$

*Complement*

And finally, the membership function of the complement $A^c$ of a fuzzy set $A$ is defined as

$$m_{A^c}(x) = 1 - m_A(x), x \in X \quad (6)$$

The intuition behind these definitions is fairly obvious. In the case where the membership functions are characteristic functions having values of 0 and 1, we have the definitions of classical set theory.

To illustrate these definitions let us look at an example. In addition to the set of "comfortable house for a 4-person family" we define a "large house", $B$ as follows:

$$B: \{(T3, 0.2), (T4, 0.4), (T5, 0.6), (T6, 0.8), (T7, 1)(T8, 1)\} \quad (7)$$

Thus we get for intersection, union, and complement:

$A \cap B = \{(T3, 0.2), (T4, 0.4), (T5, 0.6), (T6, 0.3)\}$

$A \cup B = \{(T1, 0.2), (T2, 0.5), (T3, 0.8), (T4, 1), (T5, 0.7), (T6, 0.8), (T7, 1), (T8, 1)\}$

$B^C = \{(T1, 1), (T2, 1), (T3, 0.8), (T4, 0.6), (T5, 0.4), (T6, 0.2), (T9, 1), (T10, 1)\}$

We have defined fuzzy sets. But fuzzy sets in turn are not either fuzzy or not: they vary in degree of fuzziness. So, we can ask how fuzzy a fuzzy set is. Obviously the notion of fuzziness is fuzzy itself.

Assume that the set A is finite. We can define a measure of fuzziness E, that has the following properties:

(i) $E(A)$=0 if A is a regular (crisp) set. This is a sensible requirement: if for every element it is known whether it belongs to the set or not, there is no uncertainty and thus no fuzziness.

(ii) $E(A)$=max (i.e. $E$ has its maximum value) if $m_A(x) = \frac{1}{2} \; \forall x \in X$. This requirement makes sense since in this case the least is known about the elements in the universe of discourse.

(iii) $E(A) \geq E(A')$ if $A'$ is more crisp than $A$, i.e. $m_{A'}(x) \leq m_A(x)$ if $m_A(x) \leq \frac{1}{2}$ and $m_{A'}(x) \geq m_A(x)$ if $m_A(x) > \frac{1}{2}$. It makes sense that it is symmetric with respect to the "middle", and it captures the right intuition about the relative fuzziness of fuzzy sets.

(iv) $E(A) = E(A^C)$. Clearly, the fuzziness of a set should be the same as the fuzziness of its compliment.

There have been many suggestions as to how $E$ should be defined. One is to resort to classical information theory and use the definition of entropy (which is often called $H$). $E$ is defined as:

$$E(A) = H(A) + H(A^C), x \in X$$
$$H(A) = -k \sum_{i=1}^{n} m_A(x_i) \ln(m_A(x_i)) \qquad (8)$$

Entropy is a measure for the "disorder" or the degree of uncertainty in a system. It is, in a sense, the opposite of information content. This is why information content is sometimes called "negative entropy". Recall from information theory that entropy is measured as follows:

$$H = -\sum_{i=1}^{n} p_i \ln p_i \qquad (9)$$

where $p_i$ are the probabilities for an event i to happen, e.g. the probability that it will rain tomorrow. (8) is defined in analogy to (9). It is the standard way for measuring uncertainty or lack of information.

Let us look at the intuition behind definition (8). If $m_A(x_i) = 0$ then the contribution of that term to $H$ is obviously 0. If $m_A(x_i) = 1$ we have $\ln(m_A(x_i)) = \ln(1) = 0$. Thus, if we have a classical crisp set, its entropy and thus the measure of fuzziness $E$ will be 0, and our definition fulfills requirement (i) above.

Once again, the technical details of the formulas are not essential for our purposes. What counts is the intuition. We can see that as the values membership function move towards $\frac{1}{2}$, E will increase, reaching a maximum if all $m_A(x_i) = \frac{1}{2}$. The remaining properties can easily be verified by the reader. There are other measures that satisfy conditions (i) through (iv). We showed a common one, and we will add another one below.

This calculus can be extended in many ways. There is a whole literature on the topic. The purpose of this introduction was to provide an intuition of the basic ideas and to show how they can be formalized in a straightforward manner. In what follows, we will present a different way of viewing fuzzy sets, a very elegant one. We will call it the "geometrical" view. It is due to Bart Kosko and is beautifully explained in his textbook on "Neural networks and fuzzy systems" (Kosko, 1992).

## The geometrical view

What is the "geometry" of a fuzzy set? What does that mean? The classical way that we have just seen, interprets fuzzy sets as generalized classical (crisp) set, where the membership functions assumes values in the interval [0,1] rather than from the binary set {0,1} only. Normally, we can visualize the membership functions by having an x-axis with the elements of a fuzzy set, and a y-axis with the value of the membership function. There are better ways of visualization, for example, the "Kosko cube". Before starting, just one note on terminology. Kosko uses the term "fit" value to designate the value of the membership function. This is to contrast the term "bit" which is used for "binary" values — the f stands for "fuzzy".

### *The "Kosko cube"*

This section is largely based on Kosko's presentation. In order to understand the "geometry" of fuzzy sets, let us ask a strange question: What does the fuzzy power set $F(2^X)$ look like? Remember (from your school days) that the power set is the set of all subsets. Thus, $2^X$ is the set of all subsets of $X$ and $X$ is our universe of discourse. Thus $F(2^X)$ is the set of all fuzzy subsets of $X$. This set looks like a cube. We will call it the "Kosko cube". What does a fuzzy set look like? It is a point in a cube. The set of all fuzzy subsets equals the unit hypercube $I^N = [0,1]^n$. This notation means that we have an n-dimensional cube with one corner in its origin. A 2-dimensional version is visualized in figure 7.3.

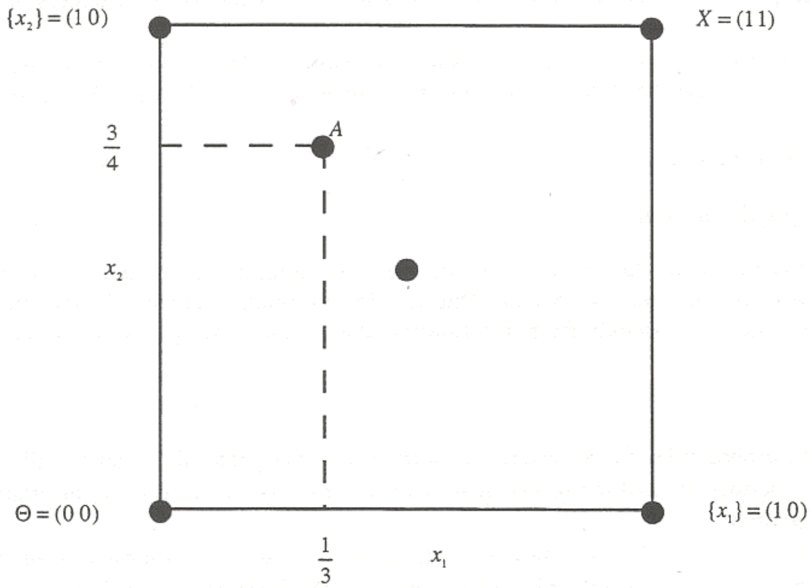


Figure 7.3 Illustration of the geometry of fuzzy sets. The fuzzy subset $A$ is a point in the 2-dimensional unit cube with coordinates $(\frac{1}{3}\ \frac{3}{4})$. The first element of $A$, $x_1$ fits in or belongs to $A$ to degree $\frac{1}{3}$, the second element, $x_2$, to degree $\frac{3}{4}$. The cube consists of all possible fuzzy subsets of two elements $\{x_1, x_2\}$. The four corners represent the power set of the classical set, consisting of 2 elements $\{x_1, x_2\}$. (from Kosko, 1992, p. 270).

Vertices (corners) of the cube $I^N$ define nonfuzzy sets. The membership functions of the elements are given in parentheses "()". The power set of the set consisting only of 2 elements $\{x_1, x_2\}$, $2^X$ is $\{\Theta, \{x_1\}, \{x_2\}, \{x_1, x_2\}\}$. These four sets correspond respectively to the four bit vectors (0 0), (1 1), (1 0), and (0 1). More precisely, they can be defined by the two-valued membership functions $m_A : X \rightarrow \{0,1\}$.

Now consider the fuzzy subsets of X. We can view the fuzzy subset $A = (\frac{1}{3}\ \frac{3}{4})$ as one of the membership functions $m_A : X \rightarrow [0,1]$ with continuous values. The bit-vector representation for crisp sets can be replaced by fit vector representation $(\frac{1}{3}\ \frac{3}{4})$ of the fuzzy set $A$.

Now look at the midpoint. All its membership values, or fuzzy values, are $\frac{1}{2}$. From our discussion above we know that this corresponds to maximum fuzziness (requirement (ii)). This point has funny properties. It is equal to its own compliment. Moreover, it is equal to its intersection with its compliment, and it is equal to its union with its complement. Or formally,

$$A = A \cap A^C = A \cup A^C = A^C \qquad (10)$$

Just as before we can define the operations on the membership functions:

$$m_{A \cap B} = \min(m_A, m_B)$$
$$m_{A \cup B} = \max(m_A, m_B) \qquad (11)$$
$$m_{A^c} = 1 - m_A$$

To practice our notation a bit, look at the following example:

$A = (1\ .8\ .4\ .5)$

$B = (.9\ .4\ 0\ .7)$

$A \cap B = (.9\ .4\ 0\ .5)$

$A \cup B = (1\ .8\ .4\ .7)$

$A^c = (0\ .2\ .6\ .5)$

$A \cap A^c = (0\ .2\ .4\ .5)$

$A \cup A^c = (1\ .8\ .6\ .5)$

In normal, crisp sets, we always have $A \cap A^c = \Theta$ and $A \cup A^c = X$ which is not the case here. A set A is called properly fuzzy if it has non-degenerate overlap and underlap, i.e. $A \cap A^c \neq \Theta$ and $A \cup A^c \neq X$. This implies that Aristotle's principle of the excluded middle only holds at the corners of the Kosko cube.

In figure 7.4 we have illustrates the four sets $A = (\frac{1}{3}\frac{3}{4}), A^c = (\frac{2}{3}\frac{1}{4}), A \cap A^c = (\frac{1}{3}\frac{1}{4}), A \cup A^c = (\frac{2}{3}\frac{3}{4})$. As $A$ moves towards the center, all four sets become maximally fuzzy, if they move towards one of the corners, they converge to the crisp sets.

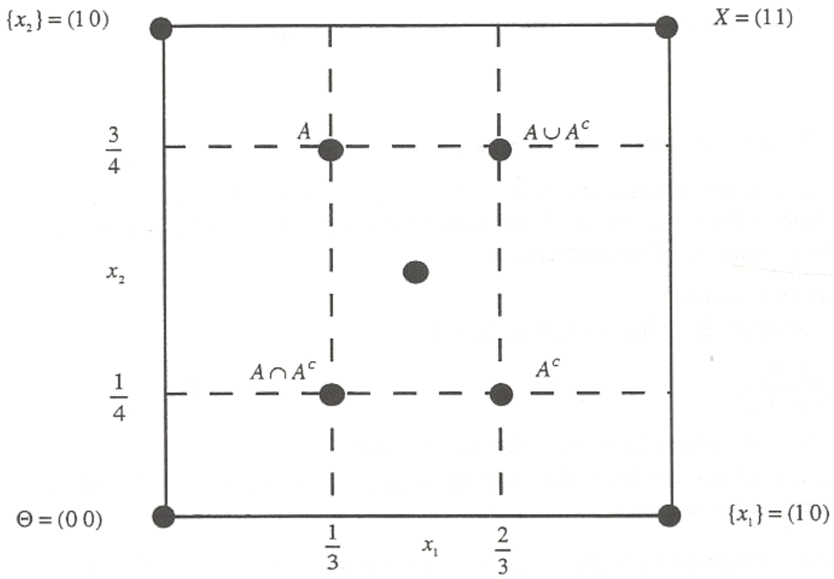


Figure 7.4: Completed 2-dimensional Kosko cube. Details, see text.

This leads to interesting considerations at midpoint. Kosko argues, that classical logic excludes this midpoint. At midpoint nothing is distinguishable, there are no contradictions. Elimination of this point leads to the paradoxes of which there is a lot in classical logic (like the liar from Crete who said that all Cretans are liars). Moreover, Kosko argues, that the paradoxes are the result of Western yes/no thinking, whereas Eastern Yin-Yang thinking has no problems with this.

***Cardinality***

When introduction the classical definitions we had defined cardinality $M(A)$ as

$$M(A) = \sum_{x \in X} m_A(x) \qquad (3)$$

For our fuzzy set $A = (\frac{1}{3}\frac{3}{4})$ we get $M(A) = \frac{1}{3} + \frac{3}{4} = \frac{13}{12}$. We can also interpret the fuzzy cardinality measure $M(A)$ geometrically. This is shown in figure 7.6. $M(A)$ equals the magnitude of the vector drawn from the origin to the fuzzy set $A$. But rather than taking the normal Euclidean metric (which in two dimension is $\sqrt{x_1^2 + x_1^2}$) we use a so-called $l^1$-norm:

$$M(A)=\sum_{i=1}^{n} m_A(x_i)=\sum_{i=1}^{n}|m_A(x_i)-0|=\sum_{i=1}^{n}|m_A(x_i)-m_\Theta(x_i)|=l^1(A,\Theta) \quad (12)$$

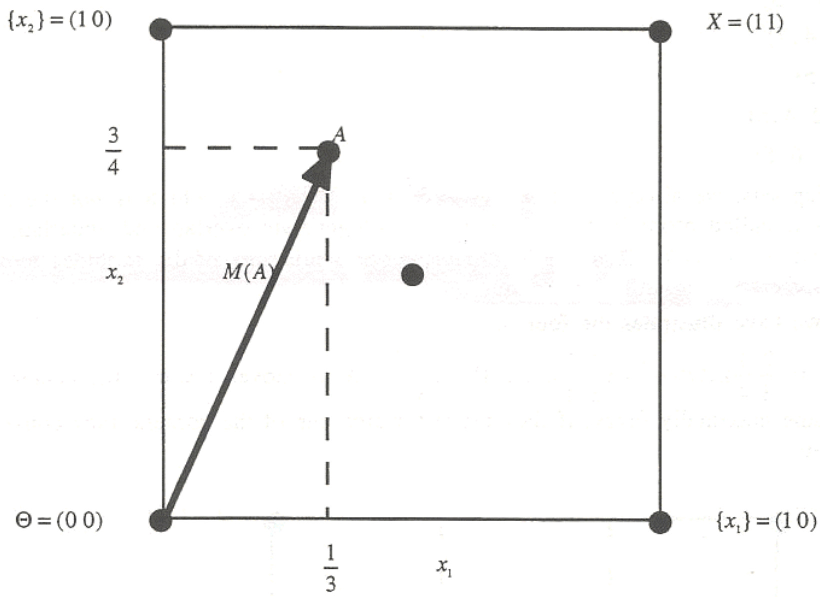


Figure 7.5: Illustration of the cardinality of a fuzzy set.

Measure (12) is sometimes called the "fuzzy Hamming distance". With this in mind we can have another look at the notion of fuzziness and see if we can find a geometric interpretation of our measure of fuzziness $E$ defined earlier.

***The fuzzy entropy theorem***

Let us define fuzziness $E$ of fuzzy set $A$ as follows:

$$E(A)=\frac{a}{b}=\frac{l^1(A,A_{near})}{l^1(A,A_{far})} \quad (13)$$

We can now check if criteria (i) through (iv) are fulfilled.

(i) In the corners where we have the crisp (non-fuzzy) sets we have a=0 and therefore $E(A)=0$ which is what we want.

(ii) In the middle, where all coordinates are $\frac{1}{2}$, we have a=b and therefore $E(A)=1$. This is the maximum, since at any other point in the square a will always be smaller than b. The mid-point is the only point where a is equal to b.

(iii) We always have $E(A')\geq E(A)$ if A is less fuzzy than $A'$, i.e. closer to a corner. This is obviously true.

(iv) Inspection of figure 7.6 shows that $E(A)=E(A^C)$

The fuzzy entropy theorem is as follows:

$$E(A)=\frac{M(A\cap A^c)}{M(A\cup A^c)} \quad (14)$$

Again, using the Kosko cube and geometric reasoning, the theorem is obvious (see figure 7.7). M is again the $l^1$-norm which is the measure for the "size" of a fuzzy set. Thus, the fuzzy entropy theorem states that the fuzziness of a fuzzy set is the ration between the overlap and the underlap between the set and its complement. $A$ lot of overlap of fuzzy set $A$ and its complement means high degree of fuzziness. The denominator is used for scaling purposes to make $E$ vary between 0 and 1.

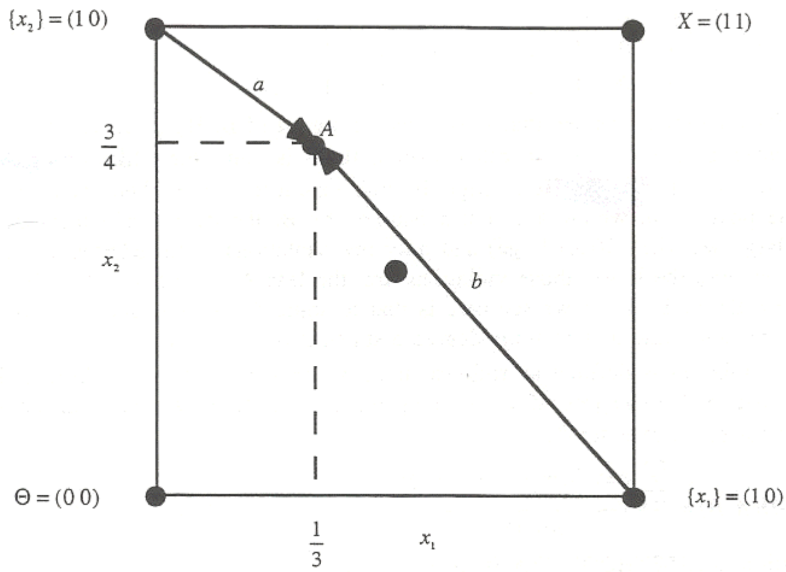


Figure 7.6: Illustration of the fuzzy entropy theorem

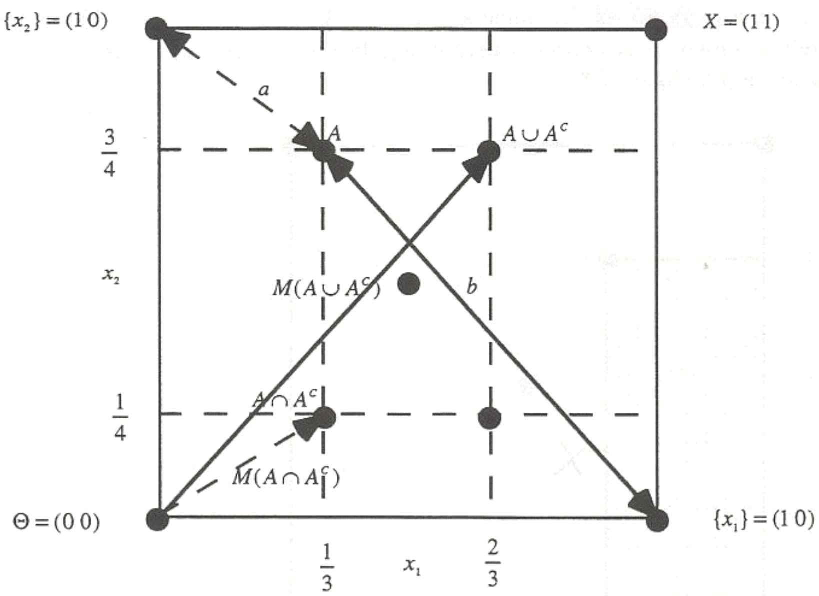


Figure 7.7: Geometric proof of the fuzzy entropy theorem.

***Fuzzy subsethood***

In classical set theory A is a subset of B, written as $A \subset B$ if and only if every element in A is an element of B. The power set $2^B$ contains all of B's subsets. Thus we can rewrite $A \subset B$ as $A \in 2^B$. Zadeh (1965) defined the subset relation for fuzzy sets A and B as follows:

$A \subset B$ if and only if $m_A(x) \leq m_B(x)$ for all x. (15)

Kosko (1992) calls this the *dominated membership function relationship*. If $A$=(.3 0 .7) and $B$=(.4 .7 .9), then $A$ is a fuzzy subset of $B$, but $B$ is not a fuzzy subset of $A$. If $A$=(.5 0 .7), then $A$ is no longer a subset of $B$. In other words, fuzzy set $A$ either is, or is not, a fuzzy subset of $B$. So Zadeh's definition of fuzzy subsethood is not fuzzy.

Let us look for a moment at the following definitions of crisp sets A and B. A and B are defined in terms of the characteristic function on $x_i \subset X$:

| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ | $x_7$ | $x_8$ | $x_9$ | $x_{10}$ | $x_{11}$ | $x_{12}$ | $x_{13}$ | $x_{14}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| B | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |

If we ignore $x_1$ for a moment, we see that A is indeed a subset of B. But even if we include $x_1$, A is still *almost* a subset of B. In other words, there is something like a degree of subsethood. Let us call it $S(A,B)$. It is high in this example. Since there is only one violation, i.e. one position in which A is not subsumed by B, the degree of subsethood is large. Thus, for large sets (i.e. $M(A)$ large) and only few violations, SUBSETHOOD is high. The more violations and the larger those violations are, the less A is a subset of B, but, the more A is a SUPERSET of B. What we see here is that to some degree one set is a subset of the other, but at the same time it is to some degree a superset of the other.

For fuzzy sets, a violation is defined as follows: if A is considered to be a subset of B, a violation means that $m_A(x) \geq m_B(x)$. If we sum up all the violations we get a measure of SUPERSETHOOD for set A over set B.

$$SUPERSETHOOD(A,B) = \frac{\sum_{x \subset X} \max(0, m_A(x) - m_B(x))}{M(A)} \qquad (16)$$

Using this definition, SUBSETHOOD $S(A,B)$ is defined as:

$$S(A,B) = 1 - \frac{\sum_{x \subset X} \max(0, m_A(x) - m_B(x))}{M(A)} \qquad (17)$$

If $m_A(x) - m_B(x)$ is always $\leq 0$, in the numerator there will always be a zero. This implies that A is entirely a subset of B (Zadeh's definition). In other words, set A is in the cube between B and the origin (figure 7.8).

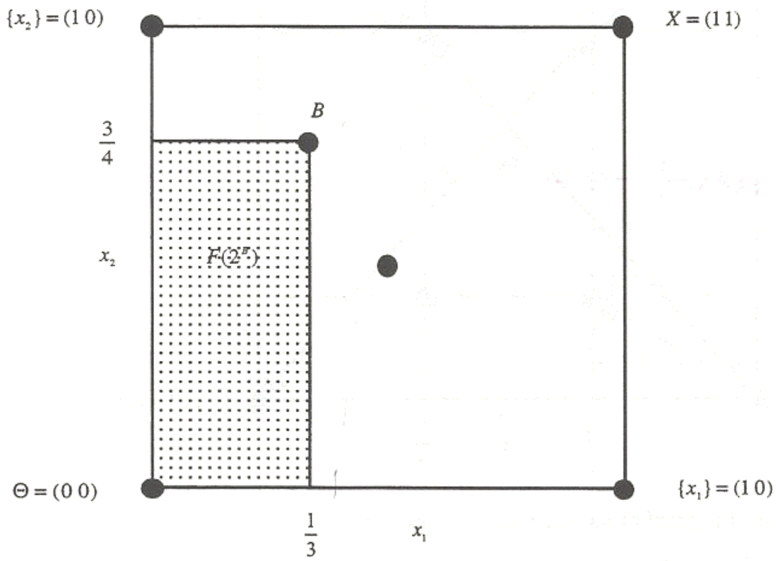


Figure 7.8: Illustration of the notion of a fuzzy subset.

But if we do it like that we have a binary, i.e. non-fuzzy definition of subsethood. But we would like to express the degree to which $A$ is a subset of $B$. We use our intuitions about violations that we developed above to define a measure of subsethood. It is called the fit-violation-strategy. We take into account the number and magnitude of the violations: The greater the number of violations (relative to the size $M(A)$ of $A$), the less $A$ is a *subset* of $B$, or, equivalently, the more $A$ is a *superset* of $B$.

When there is no violation, there should be no contribution to the supersethood of A (remember, we want to assess A with respect to B). If $m_A(x_i) - m_B(x_i)$ is positive we have a violation. In this case we want to take the magnitude of the violation into account. Thus the violation, normalized with respect to the size of a (i.e. $M(A)$), we have

$$\text{SUPERSETHOOD}(A,B) = \frac{\sum_x \max(0, m_A(x_i) - m_B(x_i))}{M(A)} \quad (18)$$

And for subsethood *S(A,B)* we have

$$S(A,B) = 1 - \frac{\sum_x \max(0, m_A(x_i) - m_B(x_i))}{M(A)} \quad (19)$$

Formula (19) has the properties we would like to have. If there are no violations *S(A,B)*=1. *S(A,B)*=0 if and only if *B* is the empty set.

What we have discussed so far in terms of basic concepts should be sufficient to provide an intuition of what fuzzy theory is about and the sorts of problems it tries to deal with. As we saw, there are a number of surprising implications. For example, the fact that a set A can at the same time be a subset and a superset of B, in fact — in the real world — it will always be both. We have focused on the notion of fuzzy sets. Now we want to look at fuzzy systems, i.e. systems that employ fuzzy logic to deal with fuzzy sets. In the field of knowledge-based systems we discussed rules to write programs, pattern-driven programs. We showed how conclusions can be drawn under uncertainty in the previous chapter. In this chapter will demonstrate how fuzzy logic works.

# 7.3 Fuzzy associative memories

Following Kosko's idea fuzzy sets can be viewed as points in hypercubes. Fuzzy systems map fuzzy sets to fuzzy sets. A fuzzy system S is a transformation $S: I^n \to I^p$. The n-dimensional hypercube $I^n$ houses all the fuzzy subsets of the domain space, or input universe of discourse, $X = \{x_1, \ldots, x_n\}$. $I^p$ houses all the fuzzy subsets of the range space, or output universe of discourse, $Y = \{y_1, \ldots, y_p\}$. In what follows we will focus on systems $S: I^n \to I^p$ that map "balls" of fuzzy sets in $I^n$ to "balls" of fuzzy sets in $I^p$ in such a way that close inputs are mapped to close outputs. Such systems behave as *associative memories*. They are called *fuzzy associative memories* or *FAMs*.

The simplest FAM encodes the FAM rule or association $(A_i, B_i)$, which associates the *p*-dimensional fuzzy set $B_i$ with the n-dimensional fuzzy set $A_i$. In general a FAM system $F: I^n \to I^p$ encodes and processes in parallel a FAM bank of m FAM rules $(A_1, B_1), \ldots, (A_m, B_m)$. Each input $A$ to the FAM system activates each stored FAM rule to a different degree. The minimal FAM that stores $(A_i, B_i)$ maps input $A$ to $B_i'$, a partially activated version of $B_i$. The more $A$ resembles $A_i$, the more $B_i'$ resembles $B_i$. This is the essentially property of associative memories.

Engineers sometimes call the fuzzy-set association $(A_i, B_i)$ a "rule". This is a bit misleading, since reasoning with sets is not the same as reasoning with propositions. If we know that we are talking about numbers and matrices then it is OK to view them as a kind of IF-THEN rule: IF $A_i$ THEN $B_i$. The general FAM system architecture is shown in figure 7.9. The FAM system $F$ maps fuzzy sets in the unit cube to fuzzy sets: $F(A) = B$. So, $F$ defines a fuzzy-system transformation which maps fuzzy sets in the unit cube $I^n$ to fuzzy sets in the unit cube $I^p$. If the input A is not a fuzzy set, but a given numerical value and the output is also a numerical value (rather than a fuzzy set), this is called a BIOFAM, a binary input-output fuzzy associative memory. In practical applications, BIOFAMs are most often used.

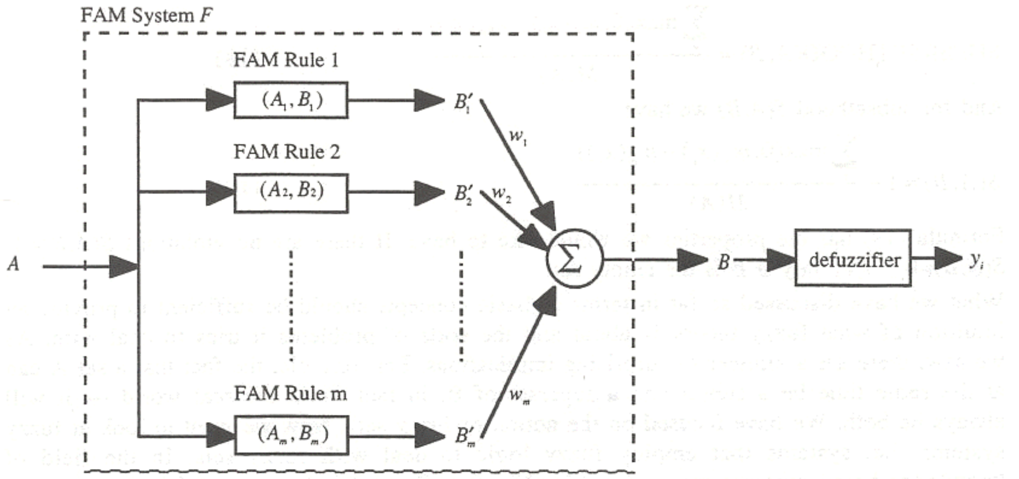


Figure 7.9: FAM system architecture. Details: See text.

*Control of a traffic light:* Consider a fuzzy association for the flexible control of a traffic light: "If the traffic is heavy in this direction, then keep the light green longer." The fuzzy association is (HEAVY,LONGER). The input fuzzy variable *traffic density* assumes the fuzzy-set value HEAVY. The output fuzzy variable *green light duration* assumes the fuzzy set value LONGER. Another fuzzy association might be (LIGHT,SHORTER).

The fuzzy system encodes each linguistic association or "rule" in a numerical *fuzzy associative memory* (FAM)-mapping. The FAM then numerically processes numerical input. This is very different from a rule in an expert system which, in essence, processes symbols. It may process some numbers such as certainty factors on the side, but the numbers are not essential. In the case of FAMs, the essence is the numerical processing.

What we have to do is to define, for example, what we mean by concepts like HEAVY or LONGER. Figure 7.10 shows a set of possible membership functions for traffic density (HEAVY, MEDIUM, and LIGHT). On the x-axis there is a numerical measure of traffic density and on the y-axis there is the degree of membership to the corresponding variable.

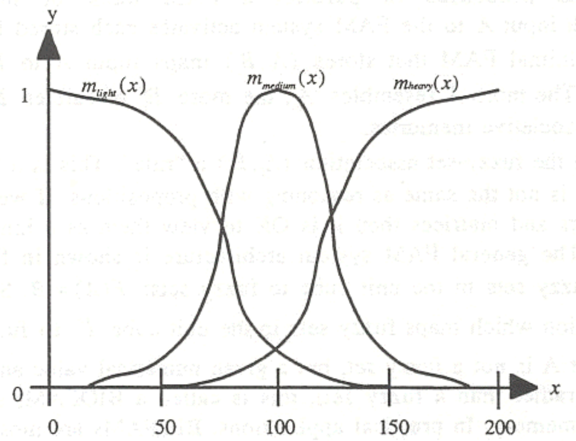


Figure 7.10: The membership functions for HEAVY, MEDIUM, and LIGHT traffic. On the horizontal axis there is a measure of traffic density.

# 7.4 Binary input-output fuzzy associative memories: BIOFAMs

As mentioned in practical applications BIOFAMs are the most widely used FAMs. They map numerical values to numerical values. For example, the traffic control BIOFAM maps traffic density to green light duration.

To make things a bit more concrete, let us look at a standard benchmark for any optimization technique, the inverted pendulum (also called the pole balancing problem). It has been discussed extensively in the literature on neural networks, genetic algorithms, control theory, and, of course, fuzzy control. The goal is to adjust a motor to balance an inverted pendulum in two dimensions. The adjustment is to be achieved by a BIOFAM.

There are a number of steps to be followed in the development of a BIOFAM:

(1) Identification of the essential (linguistic) variables. These variable are called linguistic because they are derived from natural language

(2) Definition of the membership functions.

(3) Construction of a set of fuzzy rules (the FAM rules).

(4) Combination of all outputs of the "rule set" into a geometrical output function.

(5) "Defuzzification" in order to get a non-fuzzy output value.

The experimental set-up is shown in figure 7.11.

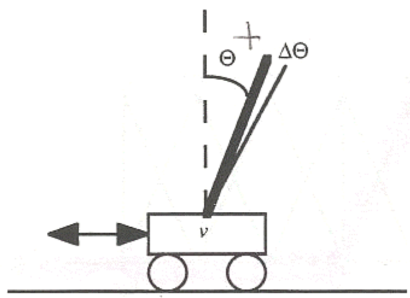


Figure 7.11: Basic set-up of the pole balancing problem.

Let us now go through the design of the system by looking at the various steps:

(1) *Identification of variables:* We choose the variables $\Theta$ and $\Delta\Theta$. $\Theta$ is the angle between the pendulum and the vertical. A zero angle corresponds to the vertical position. Positive angles are to the right of the vertical, negative angles to the left. $\Delta\Theta$ is the angular velocity. In practice we approximate the angular velocity as the difference between the current angle $\Theta_t$ and the previous angle $\Theta_{t-1}$: $\Delta\Theta_t = \Theta_t - \Theta_{t-1}$.

The variable that we want to control is the motor current $v$. This is actually not quite true. What we do want to control is the force applied to the cart. But we cannot influence the force directly. Thus, we assume that there is a linear, or at least monotonous relationship between motor current and force. We define the following directions: If the pendulum falls to the right, the motor current should be negative to compensate. If the pendulum falls to the left, it should be positive. If the pendulum successfully balances at the vertical, the motor velocity should be zero.

For each fuzzy variable we have to define the domain over which they can vary, i.e. the universe of discourse. For all three variables this is the real line $R$. Since angles greater than 90° do not exist, we can restrict ourselves to the interval [-90°,90°] for the (angle) variables $\Theta$ and $\Delta\Theta$.

(2) *Definition of membership functions:* Now we have to define the membership functions. In order to do that we have to quantize each universe of discourse (i.e. for $\Theta$, $\Delta\Theta$, and $v$) into overlapping fuzzy-set values. In our example, the fuzzy variables can be positive, zero, or negative. It is our own choice, whether we want to have only a coarse quantization or whether we request a more fine-grained one. The appropriate quantization depends on the

particular problem. Let us choose a coarse one consisting of *small*, *medium*, and *large*. This leads to seven fuzzy-set values, namely:

NL: Negative Large
NM: Negative Medium
NS: Negative Small
ZE: Zero
PS: Positive Small
PM: Positive Medium
PL: Positive Large

The fuzzy set values have labels, corresponding to the fuzzy-set values they can assume. The variable Θ can, for example, take *NL* as a fuzzy-set *value*. This is a very important step in the design of a fuzzy controller. Let us quote Kosko: "The expressive power of the FAM approach stems from these fuzzy-set quantizations. In one stroke we reduce system dimensions, and we describe a nonlinear numerical process with linguistic commonsense terms." (Kosko, 1992, p. 318).

Now we have to choose the membership functions. The shapes can vary. The set ZE may define a Gaussian curve for the pendulum angle Θ, a triangle for angular velocity ΔΘ, and a Gaussian curve for the motor current *v*. But all fuzzy sets center about the numerical value zero. The most common shapes are triangles or trapezoid shapes because the require very little computation. The exact shape does not matter very much. A typical choice is shown in figure 7.12.

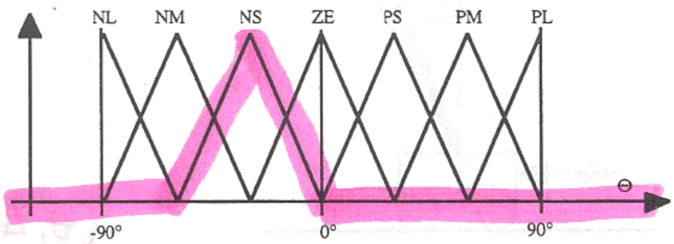


Figure 7.12: Example of a typical choice of membership functions.

The degree to which the contiguous fuzzy sets overlap is roughly 25%. This value has been determined empirically. Too much overlap blurs the distinction between the fuzzy set values, too little overlap tends to produce unstable results. Let us choose similar membership functions for the variables ΔΘ and *v*.

(3) Construction of the fuzzy set rules: We now have to design the rules. They can be written in various forms. A natural-language like formulation would be:

IF the angle is negative medium
AND the angular velocity is about zero
THEN the motor velocity should be positive medium

Using the labels for the fuzzy sets introduced above we can write them in a more compact way:

IF Θ=NM AND ΔΘ=ZE
THEN *v*=PM

An even more compact way to write the rules would be (NM,ZE;PM). In order to design the complete bank of rules we have to define 49 rules. For each entry in the rule bank (see figure 7.13) there are seven possible values, thus there are $7^3$ possible rules.

The idea in designing the rules is now that on the basis of the framework with the fuzzy set variables and the intuitive names given to them (about zero, negative small, etc.) that we can use our commonsense reasoning. Let us start at the middle. If the angle Θ is about zero (ZE) and the angular velocity $\Delta\Theta$ is about zero (ZE), then the motor current $v$ should be about zero (ZE) and we have our first rule: (ZE,ZE;ZE).

Now suppose that the angle $\Theta$ is zero but the pendulum moves. If the angular velocity is negative, the pendulum will overshoot to the left. So the motor velocity should be positive to compensate. The greater the angular velocity is in magnitude, the greater the motor velocity should be. This leads, for example to the following rules: (ZE,NS;PS), (ZE,NM;PM), etc. The symmetric cases of these rules are obvious.

The result of this commonsense analysis is shown in figure 7.13. As we see, only 15 of the 49 possible entries contain rules. In practice it is often the case that we do not need all the rules. But which ones should we take? There are, in essence three ways in which this can be done.

(a) We use engineering or commonsense knowledge, which is what we have done. The question then is, why there are empty positions on the rule bank (this is left as an exercise to the reader. Hint: there are cases that take care of themselves—nothing needs to be done, and then there are the "lost causes"— whenever these situations occur, the system will not be able to recover. If the rules are designed properly, these cases should never take place).

(b) We can perform a sensitivity analysis to see which rules are really required (see below).

(c) We can apply adaptive procedures which are based on empirical data.

| $\Delta\Theta$ \ $\Theta$ | NL | NM | NS | ZE | PS | PM | PL |
|---|---|---|---|---|---|---|---|
| NL | | | | PL | | | |
| NM | | | | PM | | | |
| NS | | | | PS | NS | | |
| ZE | PL | PM | PS | ZE | NS | NM | NL |
| PS | | | PS | NS | | | |
| PM | | | | NM | | | |
| PL | | | | NL | | | |

Figure 7.13: A FAM rule bank for the inverted pendulum problem.

(4) *Combination of all outputs of the "rule set" into a geometrical output function*: Now that we have the rules we have to say how to apply them and how to combine the results of the individual rules. The idea is illustrate in figure 7.14.

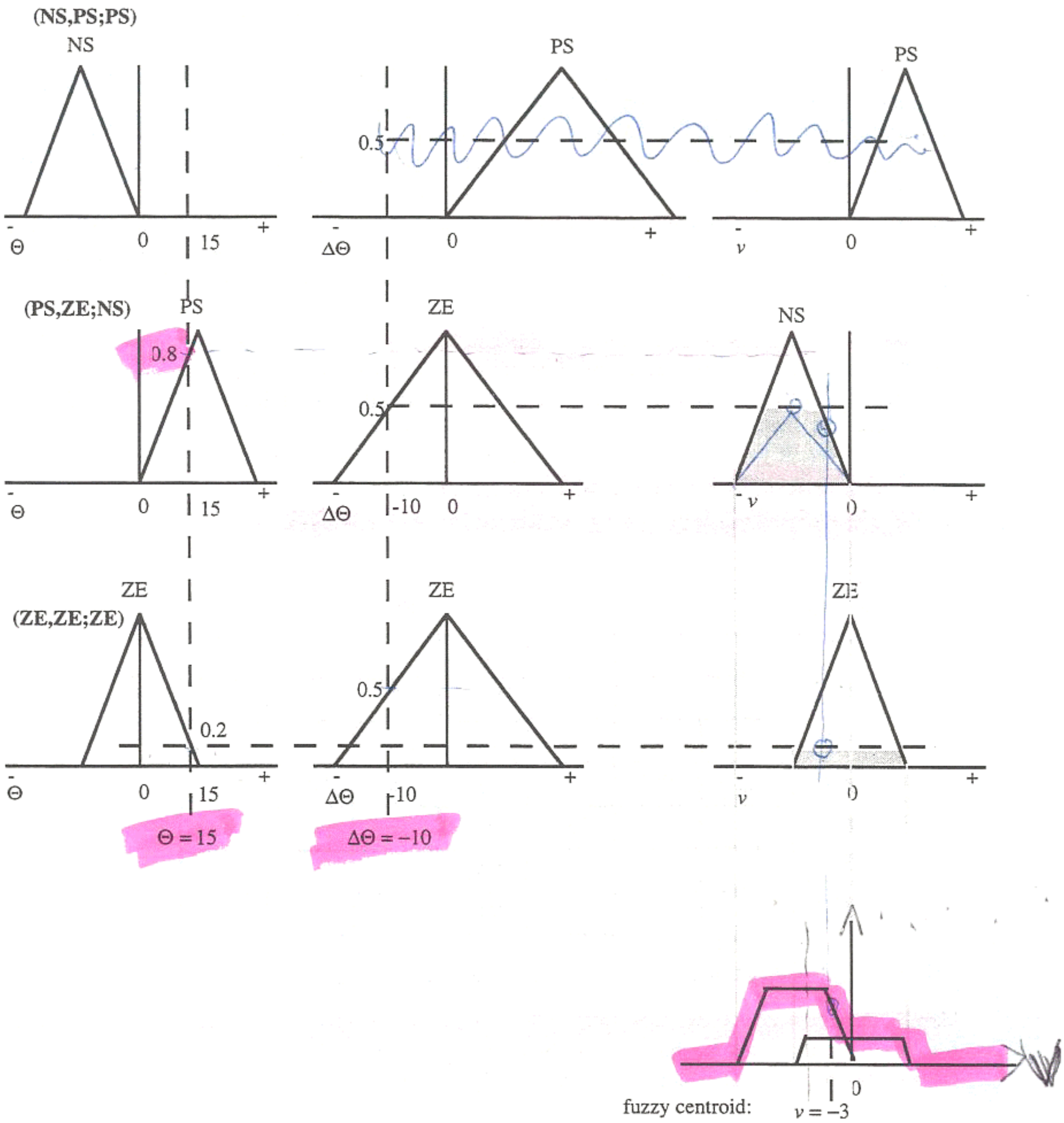


Figure 7.14: Illustration of the correlation-minimum fuzzy inference procedure. The first rule is not applied since the input values $\Theta = 15$ and $\Delta\Theta = -10$ do not intersect any of their membership functions.

The BIOFAM inference procedure activates in parallel the antecedents of all FAM rules in the bank. In figure 7.14 the input values $\Theta = 15$ and $\Delta\Theta = -10$ are shown. How do we determine the output? The rules are all activated in parallel, but to varying degrees. The degree depends on how well the input value matches the membership function on the left-hand side (antecedent side). For example, in the top rule shown in figure 7.14 the membership functions NS and PS do not intersect with the input values. The only two rules for which this is the case are the bottom two ones shown in the figure. Let us look at the first. The membership function for PS is intersected by $\Theta = 15$ at 0.8, the one for ZE is intersected by $\Delta\Theta = -10$ at 0.5. We can write $m^{\Theta}_{PS}(15) = 0.8$ and $m^{\Delta\Theta}_{ZE}(-10) = 0.5$.

These values are combined by using the minimum (logically speaking the AND) function: $\min(m^{\Theta}_{PS}(15), m^{\Delta\Theta}_{ZE}(-10)) = \min(0.8, 0.5) = 0.5$. Thus, this rule is applied to degree 0.5. An alternative would be to take the product instead of the minimum. This would be the *correlation-product* method. The correlation product method leads to smaller areas on the consequent side. Thus, in order to have the same weight, a rule in a correlation-product inference procedure has to match to a greater extent. In practice, both methods— correlation-minimum and correlation-product — are used.

The correlation-minimum inference procedure activates the angular-velocity fuzzy set ZE to degree 0.5 which is propagated to the consequent side:

$\min(m^{\Theta}_{PS}(15), m^{\Delta\Theta}_{ZE}(-10)) \wedge m^{v}_{NS}(v) = 0.5 \wedge m^{v}_{NS}(v)$

Similarly we find for the other rule:

$\min(m^{\Theta}_{ZE}(15), m^{\Delta\Theta}_{ZE}(-10)) \wedge m^{v}_{ZE}(v) = 0.2 \wedge m^{v}_{ZE}(v)$

If we now take the maximum over all the consequent sides of the rules (the shaded areas) we get the fuzzy centroid shown at the bottom. This is the geometrical output function that we were looking for. So, this is a min-max kind of inference procedure. Now we have to calculate a particular value for $v$, since the motors require one crisp single value for current. This is done by a process called "defuzzification".

(5)"*Defuzzification" in order to get a non-fuzzy output value*: The defuzzification procedure consists of calculating the center of gravity of the fuzzy centroid. In the general case this is done as follows:

$$\bar{v} = \frac{\int v \cdot m_{output}(v)dv}{\int m_{output}(v)dv} \qquad (20)$$

This is the general formula. It can be considerably simplified for the kinds of trapezoid shapes that we have seen in our example. Moreover, in standard fuzzy logic software, there are libraries of functions for the various shapes of the membership functions.

The BIOFAM inference procedure described here requires exact inputs (also called binary or delta pulses). The procedures can be generalized to the case where the input itself is a fuzzy set which is the case of (general) FAM inference system.

## Adaptive BIOFAM clustering

In the last section we have used our common-sense or engineering knowledge to design a fuzzy controller. Earlier we have seen that often, if enough data are available, systems can be generated automatically. This was illustrated in the Bayesian approach and is also intrinsic in neural network or induction-oriented designs. Similarly, the structure of fuzzy controllers can be adaptively "learned" if there are enough data available.

In control applications, humans or automatic controllers generate a continuous stream of—obviously appropriate—input-output data. Adaptive BIOFAM clustering converts this data to weighted FAM rules. In essence, BIOFAM clustering counts synaptic quantization vectors in FAM cells. The clustering procedure samples the nonfuzzy input-output stream $(x_1, y_1)(x_2, y_2),...$ An unsupervised clustering procedure distributes the k synaptic quantization vectors $m_1,...,m_k$ in $X \times Y$. Learning distributes them to different cells in the rule bank. The key idea is that each cluster equals a rule. This is illustrated in figures 7.15 and 7.16.

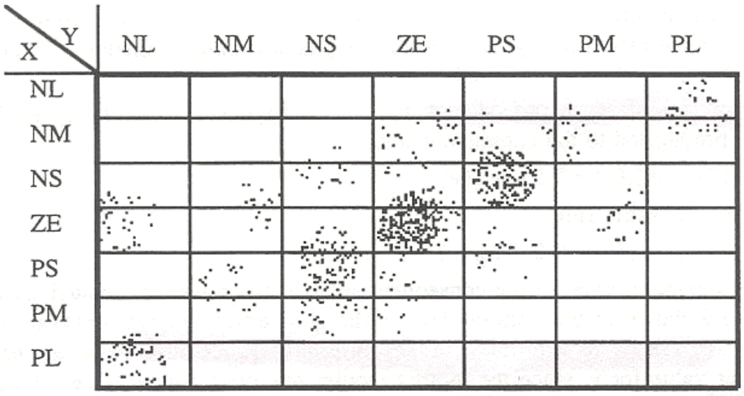


Figure 7.15: Distribution of input data $(x_i, y_i)$ in the input-output product space $X \times Y$.

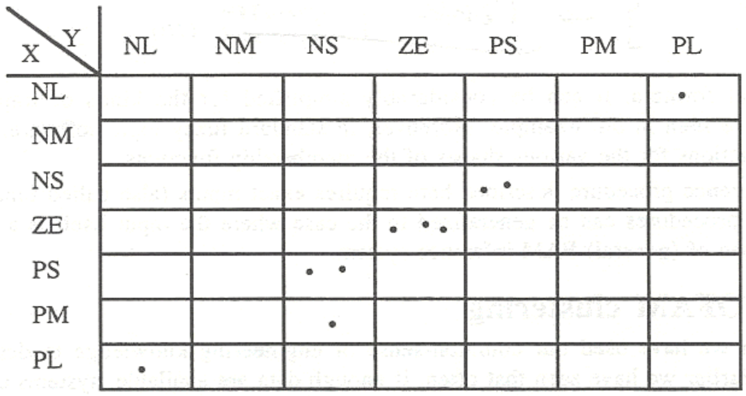


Figure 7.16: Distribution of the quantization vectors, given the data shown in figure 16.

The procedure to perform adaptive BIOFAM clustering can be described as follows:

- Identify the variables.
- Find the linguistic labels and define the membership functions.
- Take an existing, functioning controller, a human or a machine.
- Choose the number of quantization vectors (e.g. one for each cell).
- Generate cases. For each input vector generate the appropriate output action (i.e. the control actions). Register into which cells the output actions fall, given a particular input vector.
- Perform a sensitivity analysis. This may enable you to remove some of the rules, thus making the system smaller and more efficient.
- Deal with the cases for which you do not have data on the basis of intuition or prior knowledge.

In practical applications a combination of methods will typically be used to generate the FAM rule-bank, for example adaptive clustering together with theoretical analysis (using prior knowledge and intuition).

## The truck backer-upper

Let us look at another example, the truck backer-upper. Like the inverted pendulum it is one of the universal benchmarks in the optimization literature. It will be used to illustrate a number of points. Membership functions can vary depending on the degree of precision required. Sensitivity analysis can be used to test the rule set for various properties. The truck backer-upper with a trailer requires a three-dimensional FAM rule bank.

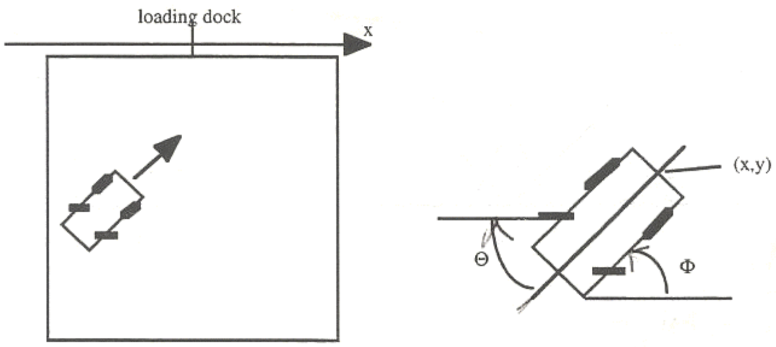


Figure 7.17: The truck backer-upper.

Let us simplify the problem a bit by only specifying the x-coordinate rather than the x-y coordinate. The input variables are then the x coordinate, and the angle of the truck Φ . The output variable is the steering angle Θ. The ranges of the variables are:

$0 \le x \le 100$

$-90 \le \Phi \le 270$

$-30 \le \Theta \le 30$

Let us, once again, use our commonsense to determine the fuzzy variables and the membership functions. The following table shows the labels for the variables:

| Angle Φ | x-position x | steering angle Θ |
|---|---|---|
| RB: Right Below | LE: Left | NB: Negative Big |
| RU: Right Upper | LC: Left Center | NM: Negative Medium |
| RV: Right Vertical | CE: Center | NS: Negative Small |
| VE: Vertical | RC: Right Center | ZE: Zero |
| LV: Left Vertical | RI: Right | PS: Positive Small |
| LU: Left Upper | | PM: Positive Medium |
| LB: Left Below | | PB: Positive Big |

The membership functions are shown in figure 7.18.

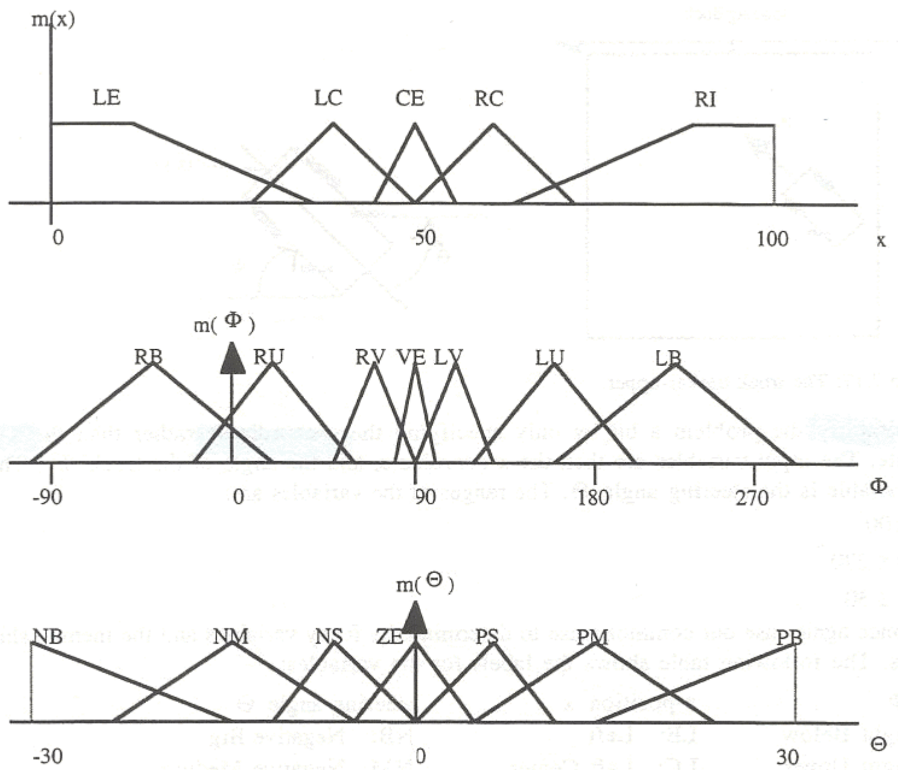


Figure 7.18: The membership functions for the truck backer-upper.

| Φ \ X | LE | LC | CE | RC | RI |
|---|---|---|---|---|---|
| RB | PS | PM | PM | PB | PB |
| RU | NS | PS | PM | PB | PB |
| RV | NM | NS | PS | PM | PB |
| VE | NM | NM | ZE | PM | PM |
| LV | NB | NM | NS | PS | PM |
| LU | NB | NB | NM | NS | PS |
| LB | NB | NB | NM | NM | NS |

Figure 7.19: Rule-bank for the truck backer-upper.

## Sensitivity analysis

Generally speaking the goal of sensitivity analysis is to get a better feel for the system's behavior. In particular we are interested in its robustness. For examples, we would like to know what happens if a rule is changed or removed, or if additional rules are added. Remember that one of the goals is to have as few rules as possible in the system—not all the cells in a rule-bank need to be filled. We have to find out which ones are essential. As one might expect, Kosko (1992) demonstrates that the fuzzy controller for the truck backer-upper is very robust (in additional to being better than the neural controller).

## Fuzzy truck-and-trailer controller

The problem of the truck backer-upper can be made increasingly more complex. For example, a trailer can be added, giving the system additional degrees of freedom. Instead of having only the angle $\Phi$ of the truck we have two angles, one for the truck and one for the trailer. The consequence is that there are now three antecedents. The FAM rule-bank is no longer a two-dimensional table, but a three-dimensional cube. If we want to represent the rule-bank two-dimensionally we have to list the 2-d rule-bank for all the values of one variable, say x. For each value of x there is a table.

This should suffice to provide a good idea of what fuzzy logic is all about, how the basic mechanisms work, and how it can be applied. We will now discuss a number of important related topics.

# 7.5 Hardware and software

As mentioned initially, fuzzy is considered a "quality seal" for many consumer products. In order to bring it into these products, fuzzy logic has to be implemented in cheap ways. In order to develop fuzzy logic controllers, tools are required. There are many software tools as well as specialized hardware available for fuzzy processing. While for consumer electronics speed is often not truly critical there are many applications in which high precision and high speed is required (e.g. control of a car engine). This is why specialized hardware has been developed.

Recent implementations of fuzzy logic in analog and digital hardware are described in IEEE Micro, August 1995, special issue on fuzzy hardware. For example, Donald L. Hung (1995) reports on an implementation of dedicated digital hardware (a field-programmable gate array) capable of performing 3.3Mio FLIPS (Fuzzy Logic Inferences Per Second). While currently most fuzzy logic processors are digital, there have been attempts to develop fuzzy inferencing systems based on analog fuzzy chips (e.g. Miki and Yamakawa, 1995). Recently, hardware for combining neural network and fuzzy logic technology has become increasingly popular: they take account of the fact that the real world is intrinsically fuzzy and they exploit the learning and generalization capabilities of neural networks (see next chapter).

On the software side there are a number of software packages available which typically include libraries for standard membership functions (triangular shaped, or trapezoid), as well as interpreters for fuzzy inference. An example is demonstrated in class. The software enables the user to quickly come up with a system and perform a sensitivity analysis: rules and membership functions can be changed rapidly so that the engineer quickly gets a good feel for the system. This kind of prototyping can form the basis for hardware implementations for a particular product.

We conclude by providing a summary of different types of implementations of fuzzy systems.

Table 7.2: Implementation of fuzzy systems(from Miki and Yamakawa, 1995)

| Implementation | Advantages | Disadvantages |
|---|---|---|
| Software implementation | Easy to implement; low cost | Low-speed operation |
| Implementing fuzzy logic operations in microcode | Easy programming | — |
| CPU supported by fuzzy coprocessor | High-speed fuzzy operation | Data-handling time limits the operation speed |
| CPU including fuzzy operation at hardware level (fuzzy CPU, fuzzy CPU core) | Additional high-speed fuzzy operation | — |
| Exclusive fuzzy inference engine (fuzzy chip) | Fastest fuzzy operation; suitable structure for fuzzy logic architecture | High cost |

# 7.6 Fuzzy logic for real-world computing

## The appeal of fuzzy logic

As pointed out initially, fuzzy logic introduces a number of highly interesting concepts in dealing with the real world. In the real world, there is intrinsic uncertainty. If we are to automate any sort of activity in the real world, we must take this intrinsic uncertainty into account—one way or other. Fuzzy thinking contrasts with probabilistic thinking. The latter deals with uncertainty because of lack of information (e.g. lack of knowledge of future events), but once the information is available, the uncertainty has been removed. By contrast, fuzzy thinking tries to capture those uncertainties that are simply inherent in anything in the real world. When we say that it will rain tomorrow with 20% chance, we have two types of uncertainty, a probabilistic one (lack of information—we don't know whether it will rain or not), and one having to do with fuzzy categories (even if we talk about the current weather conditions, we still may not be sure whether it is raining or not). Note that the way we have been using the term "fuzzy thinking" is in the sense of "applying concepts from fuzzy logic", rather than unclear and nebulous thinking.

As pointed out earlier, fuzzy logic is especially popular in Japan. There are various reasons for it. First, it seems that this kind of thinking, a kind of non-Cartesian thinking, comes more natural to Eastern peoples than to Western ones. For Westerners, the principle of the excluded middle is a very natural one. It is hard for us to imagine how things might be different. Fuzzy thinking, in some very important sense, reflects a different way of being in the world.

Second, though originally developed in the US, it has been brought to its current level of popularity and success by the Japanese, not by the Americans or European. In this sense, it is something genuinely Japanese.

## The success of fuzzy logic

There is no doubt that fuzzy logic has been very successful. All the products which are now available in consumer and industrial products testify that this is indeed the case. In this section we would like to look at some of the success factors for fuzzy logic applications: Why has fuzzy logic been so successful? What do we have to take into account when defining a project?

### *The right level of application*

The careful choice of application is, of course, crucial for any technology. If we look at where fuzzy logic has been used, it is one the one hand consumer products, and on the other industrial applications. Most of it has been at the level of *fuzzy control*. Even though the technology of fuzzy logic permits, in principle, the development of high-level reasoning systems, control problems are typically much easier to handle (a) in terms of unforeseen situations, and (b) in terms of interpretation of real-world situations.

Concerning (a), the real world is defined by the designer's low-level ontology (sensors, actuators). In consumer applications, it is relatively simple: concentrations of dirt and calcium in a washing machine, time-of-flight signals in ultrasound-based auto-focus cameras, or temperatures and temperature differences in refrigerators. Note also that in these types of applications, the damage entailed if for some reason the device is not function correctly, is relatively minor. In industrial applications, or in transportation systems, a kind of "over-engineering" will have to take place: redundant sensors and monitoring systems will have to be introduced in order to minimize the occurrence of unanticipated situations. Remember that the big problem are not the situations in which the systems encounters an unforeseen combination of facts, but rather one in which the system does not "realize" that it is confronted with a novel situation, because the appropriate sensors are not available. In a subway system, for example, unforeseen combinations of sensor data can be dealt with by simply stopping the train (to reduce risk of accident). Redundant measure of speed might include wheel turns per second, rpms of the electrical motors, time between stations, etc.

The more difficult aspect concerns (b), interpretation of real-world situations. High-level reasoning in humans is grounded in a person's experience with the physical and social environment. This is the basis for all reasoning and judgment. Since a machine cannot make such experiences, it will not be able to interpret these situations in similar ways. This is, as we mentioned in earlier chapters, one of the main reason why expert systems have not been as successful as originally expected: they tried to automate processes at the level of decision making requiring a "social individual". Fuzzy logic might seem more appropriate, since it takes the fuzziness of natural language and everyday concepts into account. But just like traditional rule-based systems, the—high-level—concepts are not grounded. This is why, one might speculate, natural language understanding systems based on fuzzy logic will have much the same problems as traditional ones.

Choosing the right level of application is, of course, a prerequisite for success: if this choice has been inappropriate, there is no hope. But there is another factor that is equally important: where is there a chance of success for the technology? Let us illustrate this point with the case study on kiln control systems that we outlined in chapter 3.

### *"Side effects" of introducing fuzzy technology: a case study*

Introducing fuzzy technology implies that the data required is of sufficient quality. Moreover, what is the value of the best control software, if the hardware cannot exploit it? For example, if the controller "suggests" to increase the dosage of fuel by a very small amount, and the physical system only works on the basis of tons, the controller will be pretty useless. Thus, if fuzzy controllers are to be introduced, the technology of the application domain has to be right. Where can we expect big successes? Typically in those places where the technology is currently outdated and not well-maintained. This is exactly what the proponents of fuzzy control in large companies have been doing. First, they automated factories in developing countries, rather than in countries where the maintenance level and the technology standard is generally high, like Switzerland, Germany, or Japan.

Holderbank, for example, with their concept of "high-level control", which is based on fuzzy logic, have reported tremendous successes. We can safely trust the figures reported. What the reports do not stress, however, is the fact that in order for "high-level control" to work, the factories had to be modernized in terms of equipment (sensor technology, effectors), and in terms of enforcement of maintenance rules. They deliberately chose factories in Brasil and Pakistan first. Had they started in Switzerland (so the expert said explicitly), they could not have demonstrated impressive successes since they are already working at top performance with the current systems.

In summary, the introduction of fuzzy control often has highly beneficial "side-effects" that do not directly relate to fuzzy technology.

### *Quality and efficiency of engineers*

If fuzzy technology is to work, it has to be done seriously. Real-world applications require experienced engineers for tuning the rule-sets, for choosing the variables and for adjusting the membership functions. The high quality of Japanese engineers is certainly an important success factor. Even though fuzzy logic is easy to understand for any computer science student, this should not be interpreted as implying that applications are easy to develop, even though the intuitive plausibility of the technology may act as an enhancer.

### *Other factors for Japanese success*

So far, we have discussed success factors generally. Now we briefly look at some additional factors that might have contributed to the success of fuzzy logic in *Japan.*

We have already mentioned the fit with Eastern thinking. Another one concerns a different attitude towards automation. This can be seen in consumer products, but also in industry. Automation is considered intrinsically positive, at least this has been the case until very recently. There is a slight shift in manufacturing, where the degree of automation is no longer increased, but rather there is a tendency to reduce it somewhat. But still, the main thrust of development is on automation. This relates to the following problem. For Japanese people it is OK to have a device where you push a button "on" and then you let the machine do the work for you. Westerners like to have many buttons to have more control over the device.

American marketing analysts have suggested that one of the success factors might be the determination with which the Japanese have introduced digital technology into consumer products. The stress is on digital technology, rather than fuzzy logic technology. They might have achieved the same level of success, so the analysts, with other, more conventional technology (like classical control theory).

While this may be true, it is missing an important psychological point. Whenever a technology is to be successful, a kind of "romantic" vision is needed. The vision of going to the moon has been such a vision, and has enormously boosted technological development in the sixties. Building "intelligent machines" (whatever that exactly means), has generated an enormous amount of highly creative research in artificial intelligence and related fields. "Fuzzy" might just be such a vision capable of generating enough enthusiasm.

As a last success factor that benefited the Japanese success, was certainly an excellent marketing strategy. This strategy is, in some sense, related to the vision provided by "fuzzy". The Japanese managed to promote "fuzzy" to become a quality standard in a world where everybody is complaining about degradation of quality: quality of service, quality of products, and generally quality of life.

### *Fuzzy logic or other approaches*

There has been a lot of discussion whether fuzzy logic is superior to other approaches like neural networks, or control theory. Reasons given concern the inclusion of expert knowledge, ease of implementation, robustness, and lower computational requirements. As we saw above, good engineers are still required to get an application working. The choice of the variables and the shapes of the membership functions require a lot of expertise and know-how. Fuzzy controllers cannot be designed by beginners only familiar with the principles of fuzzy logic: the designer needs to have experience with controllers. This is sometimes used as an argument against fuzzy logic. But the fact remains that it is easy to incorporate a priori knowledge. On the other hand, because of the complexity of the interpolation process (via the membership functions), it is very difficult to derive proofs for the control laws generated.

Obviously, what can be achieved by fuzzy logic approaches, can, one way or other, also be achieved by other methods. More bluntly speaking: in the end everything maps down to machine code. The question seems less whether it could be done by other methods, too, but rather whether it is done or not. The Japanese have done it with fuzzy logic. And they have been successful. It is clear that success always produces envy and draws criticism.

## 7.7 A note on cognitive science

One of the goals of artificial intelligence is to make models of intelligence. As we discussed in earlier chapter, the symbol processing model (the PSSH) has only had limited success. Many alternative approaches have been proposed such as neural networks, Bayesian approaches, autonomous agents, and, of course, fuzzy logic. Natural language, for example, is vague. We continuously use expressions such as big, small, intelligent, colorful, etc. They do not have a crisp, well-defined meaning. Thus, it seems natural to apply fuzzy logic to natural language processing or to the modeling of psychological concepts in general.

Many theorists of fuzzy logic, including Lotfi Zadeh himself, and H.-J. Zimmermann in Aachen, Germany, have tried to apply fuzzy logic to natural language understanding. Our hypothesis is that this will not work any better than previous approaches. Let us examine this more closely.

Just as in classical approaches, a high-level domain ontology is required. The only thing which changes is that objects only belong to these categories to a certain extent. A fifty year old person belongs to the category of old people only to degree, say 0.6. But the category is there a priori, as defined by the designer. The grounding (see chapter 4) is lacking just as much as it is in expert systems. As long as high-level categories, fuzzy or crisp, are pre-defined, the system will never be able to develop its own categorization of the real world, and will therefore never have natural language capacities as we know them from humans. This is a cognitive science perspective. It may well be the case, that—pragmatically speaking—an acceptable level of performance can be achieve, for example for automatic translation programs.

The same arguments apply, in essence to expert systems work. The minute fuzzy logic is applied to high-level human reasoning and decision making, it is—for the same reasons, i.e. grounding—doomed to failure.

# Points to remember

- Fuzzy logic has been enormously successful in many applications, especially in consumer products. But many industrial applications have also been reported.
- Fuzzy logic takes the intrinsic uncertainty inherent in the real world into account. Even though all the information is available about a circle drawn on paper, it only belongs to the category of circles to a certain degree.
- There is a fundamental difference between probabilistic and fuzzy logic thinking. The fact that I don't know whether it will rain tomorrow, is due to lack of information. The fact that I can't say whether it is currently (really) raining or not (drizzle, heavy fog), is due to the intrinsic uncertainty.
- Fuzzy sets differ from classical crisp sets by their membership functions which vary between 0 and 1, rather than being binary. All the operations like intersection or union can be generalized to fuzzy sets.
- In contrast to crisp set theory, the intersection of a fuzzy set with its compliment is not the empty set. Aristotelian thinking in terms of the excluded middle leads to paradoxes.
- The "Kosko cube" is an excellent instrument to visualize fuzzy sets: they are interpreted as points in an n-dimensional hypercube. The corners in this hypercube correspond to the classical crisp sets.
- Size and "fuzziness" of a fuzzy set have straightforward geometric interpretations in the "Kosko cube" view.
- There are five steps in the development of a fuzzy rule system, namely identification of the fuzzy variables, definition of the membership functions, construction of the FAM rules, combination of the outputs into a geometrical output, and the defuzzification.
- Fuzzy inference procedures apply all rules in parallel. Typical inference methods used are correlation-minimum and correlation-product. An OR-function is used on the

consequent side. The most often used defuzzification method calculates, in essence, the center of gravity.

- Fuzzy systems can be made adaptive. Combinations between fuzzy and neural systems have also become popular.
- A lot of specialized hardware and software is available for implementing and prototyping fuzzy systems.
- The choice of the right level of application is a crucial success factor.
- Fuzzy logic is not a "deus ex machina": its applications requires good experienced engineers.
- Fuzzy logic is not a good tool for cognitive science.