CHAPTER 9

Fuzzy Logic Applications

9.1 WHY USE FUZZY LOGIC?

Here is a list of general observations about fuzzy logic:

1. Fuzzy logic is conceptually easy to understand.

   The mathematical concepts behind fuzzy reasoning are very simple. What makes fuzzy nice is the "naturalness" of its approach and not its far-reaching complexity.

2. Fuzzy logic is flexible.

   With any given system, it's easy to massage it or layer more functionality on top of it without starting again from scratch.

3. Fuzzy logic is tolerant of imprecise data.

   Everything is imprecise if you look closely enough, but more than that, most things are imprecise even on careful inspection. Fuzzy reasoning builds this understanding into the process rather than tacking it onto the end.

4. Fuzzy logic can model nonlinear functions of arbitrary complexity.

   You can create a fuzzy system to match any set of input-output data. This process is made particularly easy by adaptive techniques like ANFIS (Adaptive Neuro-Fuzzy Inference Systems), which are available in the Fuzzy Logic Toolbox.

5. Fuzzy logic can be built on top of the experience of experts.

   In direct contrast to neural networks, which take training data and generate opaque, impenetrable models, fuzzy logic lets you rely on the experience of people who already understand your system.

6. Fuzzy logic can be blended with conventional control techniques.

   Fuzzy systems don't necessarily replace conventional control methods. In many cases fuzzy systems augment themand simplify their implementation.

7. Fuzzy logic is based on natural language.

   The basis for fuzzy logic is the basis for human communication. This observation underpins many of the other statements about fuzzy logic.

The last statement is perhaps the most important one and deserves more discussion. Natural language, that which is used by ordinary people on a daily basis, has been shaped by thousands of years of human history to be convenient and efficient. Sentences written in ordinary language represent a triumph of efficient communication. We are generally unaware of this because ordinary language is, of course, something we use every day. Since fuzzy logic is built.

9.2 APPLICATIONS OF FUZZY LOGIC

Fuzzy logic deals with uncertainty in engineering by attaching degrees of certainty to the answer to a logical question. Why should this be useful? The answer is commercial and practical. Commercially, fuzzy logic has been used with great success to control machines and consumer products. In the right application fuzzy logic systems are simple to design, and can be understood and implemented by non-specialists in control theory.

In most cases someone with a intermediate technical background can design a fuzzy logic controller. The control system will not be optimal but it can be acceptable. Control engineers also use it in applications where the on-board computing is very limited and adequate control is enough. Fuzzy logic is not the answer to all technical problems, but for control problems where simplicity and speed of implementation is important then fuzzy logic is a strong candidate. A cross section of applications that have successfully used fuzzy control includes:

1. Environmental
    - Air Conditioners
    - Humidifiers
2. Domestic Goods
    - Washing Machines/Dryers
    - Vacuum Cleaners
    - Toasters
    - Microwave Ovens
    - Refrigerators
3. Consumer Electronics
    - Television
    - Photocopiers
    - Still and Video Cameras – Auto-focus, Exposure and Anti-shake
    - Hi-Fi Systems
4. Automotive Systems
    - Vehicle Climate Control
    - Automatic Gearboxes
    - Four-wheel Steering
    - Seat/Mirror Control Systems

9.3 WHEN NOT TO USE FUZZY LOGIC?

Fuzzy logic is not a cure-all. When should you not use fuzzy logic? Fuzzy logic is a convenient way to map an input space to an output space. If you find it is not convenient, try something else. If a simpler solution already exists, use it. Fuzzy logic is the codification of common sense-use common sense when you implement it and you will probably make the right decision. Many controllers, for example, do a fine job without using fuzzy logic. However, if you take the time to become familiar with fuzzy logic, you will see it can be a very powerful tool for dealing quickly and efficiently with imprecision and non-linearity.

9.4 FUZZY LOGIC MODEL FOR PREVENTION OF ROAD ACCIDENTS

Traffic accidents are rare and random. However, many people died or injured because of traffic accidents all over the world. When statistics are investigated India is the most dangerous country in terms of number of traffic accidents among Asian countries. Many reasons can contribute these results, which are mainly driver fault, lack of infrastructure, environment, literacy, weather conditions etc. Cost of traffic accident is roughly 3% of gross national product. However, agree that this rate is higher in India since many traffic accidents are not recorded, for example single vehicle accidents or some accidents without injury or fatality.

In this study, using fuzzy logic method, which has increasing usage area in Intelligent Transportation Systems (*ITS)*, a model was developed which would obtain to prevent the vehicle pursuit distance automatically. Using velocity of vehicle and pursuit distance that can be measured with a sensor on vehicle a model has been established to brake pedal (slowing down) by fuzzy logic.

9.4.1 Traffic Accidents And Traffic Safety

The general goal of traffic safety policy is to eliminate the number of deaths and casualties in traffic. This goal forms the background for the present traffic safety program. The program is partly based on the assumption that high speed contributes to accidents. Many researchers support the idea of a positive correlation between speed and traffic accidents. One way to reduce the number of accidents is to reduce average speeds. Speed reduction can be accomplished by police surveillance, but also through physical obstacles on the roads. Obstacles such as flower pots, road humps, small circulation points and elevated pedestrian crossings are frequently found in many residential areas around India. However, physical measures are not always appreciated by drivers. These obstacles can cause damages to cars, they can cause difficulties for emergency vehicles, and in winter these obstacles can reduce access for snow clearing vehicles. An alternative to these physical measures is different applications of Intelligent Transportation Systems (*ITS*). The major objectives with *ITS* are to achieve traffic efficiency, by for instance redirecting traffic, and to increase safety for drivers, pedestrians, cyclists and other traffic groups.

One important aspect when planning and implementing traffic safety programs is therefore drivers' acceptance of different safety measures aimed at speed reduction. Another aspect is whether the individual's acceptance, when there is a certain degree of freedom of choice, might also be reflected in a higher acceptance of other measures, and whether acceptance of safety measures is also reflected in their perception of road traffic, and might reduce dangerous behaviour in traffic.

9.4.2 Fuzzy Logic Approach

The basic elements of each fuzzy logic system are, as shown in Figure 9.1, rules, fuzzifier, inference engine, and defuzzifier. Input data are most often crisp values. The task of the fuzzifier is to map crisp numbers into fuzzy sets (cases are also encountered where inputs are fuzzy variables described by fuzzy membership functions). Models based on fuzzy logic consist of "If-Then" rules. A typical "If-Then" rule would be:

If the ratio between the flow intensity and capacity of an arterial road is SMALL
Then vehicle speed in the flow is BIG

The fact following "If" is called a premise or hypothesis or antecedent. Based on this fact we can infer another fact that is called a conclusion or consequent (the fact following "Then"). A set of a large number of rules of the type:

If premise
Then conclusion is called a fuzzy rule base.

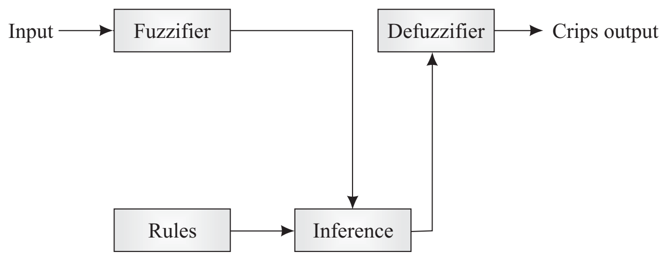

Fig. 9.1 Basic elements of a fuzzy logic.

In fuzzy rule-based systems, the rule base is formed with the assistance of human experts; recently, numerical data has been used as well as through a combination of numerical data-human experts. An interesting case appears when a combination of numerical information obtained from measurements and linguistic information obtained from human experts is used to form the fuzzy rule base. In this case, rules are extracted from numerical data in the first step. In the next step this fuzzy rule base can (but need not) be supplemented with the rules collected from human experts. The inference engine of the fuzzy logic maps fuzzy sets onto fuzzy sets. A large number of different inferential procedures are found in the literature. In most papers and practical engineering applications, minimum inference or product inference is used. During defuzzification, one value is chosen for the output variable. The literature also contains a large number of different defuzzification procedures. The final value chosen is most often either the value corresponding to the highest grade of membership or the coordinate of the center of gravity.

9.4.3 Application

In the study, a model was established which estimates brake rate using fuzzy logic. The general structure of the model is shown in Fig. 9.2.

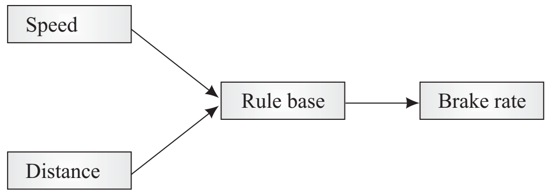

Fig. 9.2 General structure of fuzzy logic model.

9.4.4 Membership Functions

In the established model, different membership functions were formed for speed, distance and brake rate. Membership functions are given in Figures 9.3, 9.4, and 9.5. For maximum allowable car speed (in motorways) in India, speed scale selected as 0-120 km/h on its membership function. Because of the fact that current distance sensors perceive approximately 100-150 m distance, distance membership function is used 0-150 m scale. Brake rate membership function is used 0-100 scale for expressing percent type.

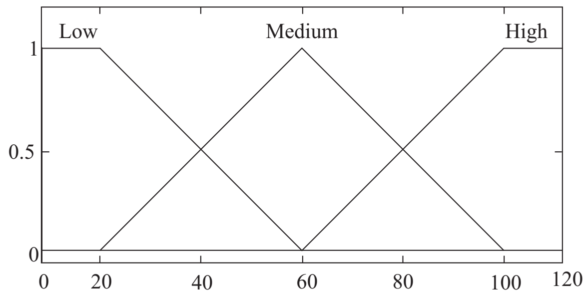

Fig. 9.3 Membership function of speed.

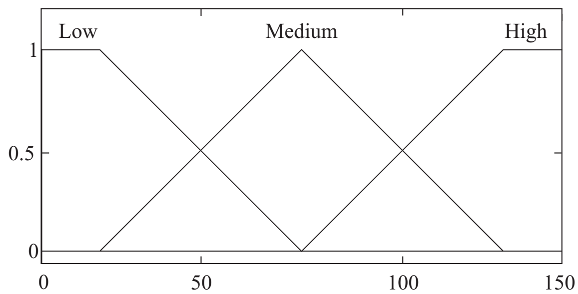

Fig. 9.4 Membership function of distance.

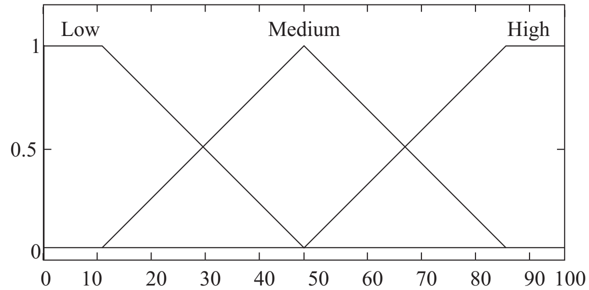

Fig. 9.5 Membership function of brake rate.

9.4.5 Rule Base

We need a rule base to run the fuzzy model. Fuzzy Allocation Map (rules) of the model was constituted for membership functions whose figures are given on Table-9.1. It is important that the rules were not completely written for all probability. Figure 6 shows that the relationship between inputs, speed and distance, and brake rate.

Table 9.1: Fuzzy allocation map of the model

Speed	*Distance*	*Brake rate*
LOW	LOW	LOW
LOW	MEDIUM	LOW
LOW	HIGH	MEDIUM
MEDIUM	LOW	MEDIUM
MEDIUM	MEDIUM	LOW
MEDIUM	HIGH	LOW
HIGH	LOW	HIGH
HIGH	MEDIUM	MEDIUM
HIGH	HIGH	LOW

9.4.6 Output

Fuzzy logic is also an estimation algorithm. For this model, various alternatives are able to cross-examine using the developed model. Fig. 9.6 is an example for such the case.

9.4.7 Conclusions

Many people die or injure because of traffic accidents in India. Many reasons can contribute these results for example mainly driver fault, lack of infrastructure, environment, weather conditions etc. In this study, a model was established for estimation of brake rate using fuzzy logic approach. Car brake rate is estimated using the developed model from speed and distance data. So, it can be said that this fuzzy logic approach can be effectively used for reduce to traffic accident rate. This model can be adapted to vehicles.

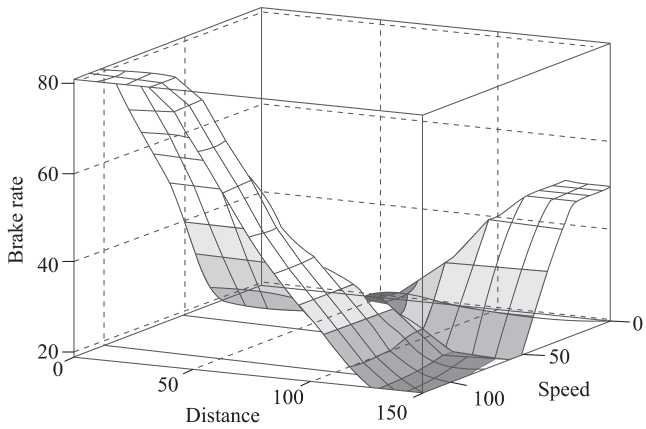

Fig. 9.6 Relationship between inputs and brake rate.

9.5 FUZZY LOGIC MODEL TO CONTROL ROOM TEMPERATURE

Although the behaviour of complex or nonlinear systems is difficult or impossible to describe using numerical models, quantitative observations are often required to make quantitative control decisions. These decisions could be the determination of a flow rate for a chemical process or a drug dosage in medical practice. The form of the control model also determines the appropriate level of precision in the result obtained. Numerical models provide high precision, but the complexity or non-linearity of a process may make a numerical model unfeasible. In these cases, linguistic models provide an alternative. Here the process is described in common language.

The linguistic model is built from a set of ***if-then*** rules, which describe the control model. Although Zadeh was attempting to model human activities, Mamdani showed that fuzzy logic could be used to develop operational automatic control systems.

9.5.1 The Mechanics of Fuzzy Logic

The mechanics of fuzzy mathematics involve the manipulation of fuzzy variables through a set of linguistic equations, which can take the form of ***if–then*** rules. Much of the fuzzy literature uses set theory notation, which obscures the ease of the formulation of a fuzzy controller. Although the controllers are simple to construct, the proof of stability and other validations remain important topics. The outline of fuzzy operations will be shown here through the design of a familiar room thermostat.

A fuzzy variable is one of the parameters of a fuzzy model, which can take one or more fuzzy values, each represented by a fuzzy set and a word descriptor. The room temperature is the variable shown in Fig. 9.7. Three fuzzy sets: ‘hot’, ‘cold’ and ‘comfortable’ have been defined by membership distributions over a range of actual temperatures.

The power of a fuzzy model is the overlap between the fuzzy values. A single temperature value at an instant in time can be a member of both of the overlapping sets. In conventional set theory, an object (in this case a temperature value) is either a member of a set or it is not a member. This implies a crisp

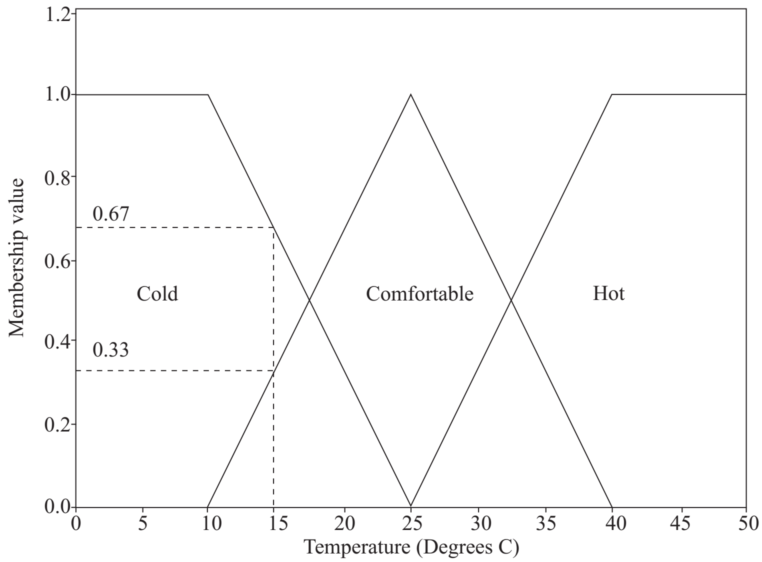

Fig. 9.7 Room temperature.

boundary between the sets. In fuzzy logic, the boundaries between sets are blurred. In the overlap region, an object can be a partial member of each of the overlapping sets. The blurred set boundaries give fuzzy logic its name. By admitting multiple possibilities in the model, the linguistic imprecision is taken into account.

The membership functions defining the three fuzzy sets shown in Fig. 9.7 are triangular. There are no constraints on the specification of the form of the membership distribution. The Gaussian form from statistics has been used, but the triangular form is commonly chosen, as its computation is simple. The number of values and the range of actual values covered by each one are also arbitrary. Finer resolution is possible with additional sets, but the computation cost increases.

Guidance for these choices is provided by Zadeh's ***Principle of Incompatibility:*** *As the complexity of a system increases, our ability to make precise and yet significant statements about its behaviour diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics.*

The operation of a fuzzy controller proceeds in three steps. The first is fuzzification, where measurements are converted into memberships in the fuzzy sets. The second step is the application of the linguistic model, usually in the form of ***if-then*** rules. Finally the resulting fuzzy output is converted back into physical values through a defuzzfication process.

9.5.2 Fuzzification

For a single measured value, the fuzzification process is simple, as shown in Fig. 9.7. The membership functions are used to calculate the memberships in all of the fuzzy sets. Thus, a temperature of 15°C becomes three fuzzy values, 0.66 'cold', 0.33 'comfortable' and 0.00 'hot'.

A series of measurements are collected in the form of a histogram and use this as the fuzzy input as shown in Fig. 9.8. The fuzzy inference is extended to include the uncertainty due to measurement error as well as the vagueness in the linguistic descriptions. In Fig. 9.8 the measurement data histogram is normalized so that its peak is a membership value of 1.0 and it can be used as a fuzzy set. The membership of the histogram in 'cold' is given by: max $\{\min [\mu_{cold}(T), \mu_{histogram}(T)]\}$ where the maximum and minimum operations are taken using the membership values at each point T over the temperature range of the two distributions.

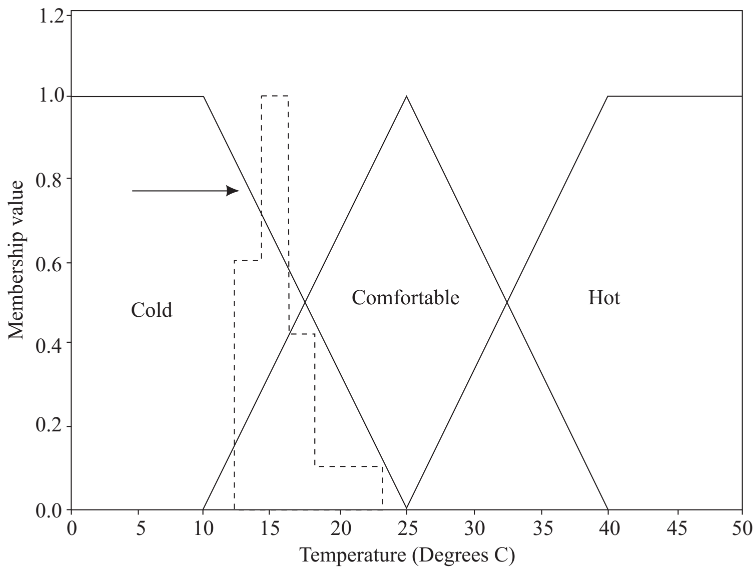

Fig. 9.8 Fuzzification with measurement noise.

The minimum operation yields the overlap region of the two sets and the maximum operation gives the highest membership in the overlap. The membership of the histogram in 'cold', indicated by the arrow in Fig. 9.8, is 0.73. By similar operations, the membership of the histogram in 'comfortable' and 'hot' are 0.40 and 0.00. It is interesting to note that there is no requirement that the sum of all memberships be 1.00.

9.5.3 Rule Application

The linguistic model of a process is commonly made of a series of ***if - then*** rules. These use the measured state of the process, the rule antecedents, to estimate the extent of control action, the rule consequents. Although each rule is simple, there must be a rule to cover every possible combination of fuzzy input values. Thus, the simplicity of the rules trades off against the number of rules. For complex systems the number of rules required may be very large.

The rules needed to describe a process are often obtained through consultation with workers who have expert knowledge of the process operation. These experts include the process designers, but more importantly, the process operators. The rules can include both the normal operation of the process as well as the experience obtained through upsets and other abnormal conditions. Exception handling is a particular strength of fuzzy control systems.

For very complex systems, the experts may not be able to identify their thought processes in sufficient detail for rule creation. Rules may also be generated from operating data by searching for clusters in the input data space. A simple temperature control model can be constructed from the example of Fig. 9.7:

Rule 1 : IF (Temperature is Cold) THEN (Heater is On)

Rule 2 : IF (Temperature is Comfortable) THEN (Heater is Off)

Rule 3 : IF (Temperature is Hot) THEN (Heater is Off)

In Rule 1, (Temperature is Cold) is the membership value of the actual temperature in the 'cold' set. Rule 1 transfers the 0.66 membership in 'cold' to become 0.66 membership in the heater setting 'on'. Similar values from rules 2 and 3 are 0.33 and 0.00 in the 'off' setting for the heater. When several rules give membership values for the same output set, Mamdani used the maximum of the membership values. The result for the three rules is then 0.66 membership in 'on' and 0.33 membership in 'off'.

The rules presented in the above example are simple yet effective. To extend these to more complex control models, compound rules may be formulated. For example, if humidity was to be included in the room temperature control example, rules of the form: IF (Temperature is Cold) AND (Humidity is High) THEN (Heater is ON) might be used. Zadeh defined the logical operators as AND = Min (μ_A, μ_B) and OR = Max (μ_A, μ_B), where μ_A and μ_B are membership values in sets A and B respectively. In the above rule, the membership in 'on' will be the minimum of the two antecedent membership values. Zadeh also defined the NOT operator by assuming that complete membership in the set A is given by $\mu_A = 1$. The membership in NOT (A) is then given by μNOT (A) = $1 - \mu_A$. This gives the interesting result that A AND NOT (A) does not vanish, but gives a distribution corresponding to the overlap between A and its adjacent sets.

9.5.4 Defuzzification

The results of rule application are membership values in each of the consequent or output sets. These can be used directly where the membership values are viewed as the strength of the recommendations provided by the rules. It is possible that several outputs are recommended and some may be contradictory (e.g. heater on and heater off). In automatic control, one physical value of a controller output must be chosen from multiple recommendations. In decision support systems, there must be a consistent method to resolve conflict and define an appropriate compromise. Defuzzification is the process for converting fuzzy output values to a single value or final decision. Two methods are commonly used.

The first is the maximum membership method. All of the output membership functions are combined using the OR operator and the position of the highest membership value in the range of the output variable is used as the controller output. This method fails when there are two or more equal maximum membership values for different recommendations. Here the method becomes indecisive and does not produce a satisfactory result.

The second method uses the center of gravity of the combined output distribution to resolve this potential conflict and to consider all recommendations based on the strengths of their membership values. The center of gravity is given by $X_F = \dfrac{\int x\ \ (x)dx}{\int\ \ (x)dx}$ where x is a point in the output range and X_F is the final control value. These integrals are taken over the entire range of the output. By taking the center of gravity, conflicting rules essentially cancel and a fair weighting is obtained.

The output values used in the thermostat example are singletons. Singletons are fuzzy values with a membership of 1.00 at a single value rather than a membership function between 0 and 1 defined over an interval of values. In the example there were two, 'off' at 0% power and 'on' at 100% power. With singletons, the center of gravity equation integrals become a simple weighted average. Applying the rules gave $\mu_{ON} = 0.67$ and $\mu_{OFF} = 0.33$. Defuzzifying these gives a control output of 67% power. Although only two singleton output functions were used, with center of gravity defuzzification, the heater power decreases smoothly between fully on and fully off as the temperature increases between 10°C and 25°C.

In the histogram input case, applying the same rules gave $\mu_{ON} = 0.73$ and $\mu_{OFF} = 0.40$. Center of gravity defuzzification gave, in this case, a heater power of 65%. The sum of the membership functions was normalized by the denominator of the center of gravity calculation.

9.5.5 Conclusions

Linguistic descriptions in the form of membership functions and rules make up the model. The rules are generated *a priori* from expert knowledge or from data through system identification methods. Input membership functions are based on estimates of the vagueness of the descriptors used. Output membership functions can be initially set, but can be revised for controller tuning.

Once these are defined, the operating procedures for the calculations are well set out. Measurement data are converted to memberships through fuzzification procedures. The rules are applied using formalized operations to yield memberships in output sets. Finally, these are combined through defuzzification to give a final control output.

9.6 FUZZY LOGIC MODEL FOR GRADING OF APPLES

Agricultural produce is subject to quality inspection for optimum evaluation in the consumption cycle. Efforts to develop automated fruit classification systems have been increasing recently due to the drawbacks of manual grading such as subjectivity, tediousness, labor requirements, availability, cost and inconsistency.

However, applying automation in agriculture is not as simple as automating the industrial operations. There are two main differences. First, the agricultural environment is highly variable, in terms of weather, soil, etc. Second, biological materials, such as plants and commodities, display high variation due to their inherent morphological diversity. Techniques used in industrial applications, such as template matching and fixed object modeling are unlikely to produce satisfactory results in the classification or control of input from agricultural products. Therefore, self-learning techniques such as neural networks (*NN*) and fuzzy logic (*FL*) seem to represent a good approach.

Fuzzy logic can handle uncertainty, ambiguity and vagueness. It provides a means of translating qualitative and imprecise information into quantitative (linguistic) terms. Fuzzy logic is a non-parametric classification procedure, which can infer with nonlinear relations between input and output categories, maintaining flexibility in making decisions even on complex biological systems.

Fuzzy logic was successfully used to determine field trafficability, to decide the transfer of dairy cows between feeding groups, to predict the yield for precision farming, to control the start-up and shut-down of food extrusion processes, to steer a sprayer automatically, to predict corn breakage, to manage crop production, to reduce grain losses from a combine, to manage a food supply and to predict peanut maturity.

The main purpose of this study was to investigate the applicability of fuzzy logic to constructing and tuning fuzzy membership functions and to compare the accuracies of predictions of apple quality by a human expert and the proposed fuzzy logic model. Grading of apples was performed in terms of characteristics such as color, external defects, shape, weight and size. Readings of these properties were obtained from different measurement apparatuses, assuming that the same measurements can be done using a sensor fusion system in which measurements of features are collected and controlled automatically. The following objectives were included in this study:

1. To design a *FL* technique to classify apples according to their external features developing effective fuzzy membership functions and fuzzy rules for input and output variables based on quality standards and expert expectations.
2. To compare the classification results from the *FL* approach and from sensory evaluation by a human expert.
3. To establish a multi-sensor measuring system for quality features in the long term.

9.6.1 Apple Defects Used in the Study

No defect formation practices by applying forces on apples were performed. Only defects occurring naturally or forcedly on apple surfaces during the growing season and handling operations were accounted for in terms of number and size, ignoring their age. Scars, bitter pit, leaf roller, russeting, punctures and bruises were among the defects encountered on the surfaces of Golden Delicious apples. In addition to these defects, a size defect (lopsidedness) was also measured by taking the ratio of maximum height of the apple to the minimum height.

9.6.2 Materials and Methods

Five quality features, color, defect, shape, weight and size, were measured. Color was measured using a *CR*-200 Minolta colorimeter in the domain of L, a and b, where L is the lightness factor and a and b are the chromaticity coordinates. Sizes of surface defects (natural and bruises) on apples were determined using a special figure template, which consisted of a number of holes of different diameters. Size defects were determined measuring the maximum and minimum heights of apples using a Mitutoya electronic caliper. Maximum circumference measurement was performed using a Cranton circumference measuring device. Weight was measured using an electronic scale. Programming for fuzzy membership functions, fuzzification and defuzzification was done in Matlab.

The number of apples used was determined based on the availability of apples with quality features of the 3 quality groups (bad, medium and good). A total of 181 golden delicious apples were graded first by a human expert and then by the proposed fuzzy logic approach. The expert was trained on the external quality criteria for good, medium and bad apple groups defined by USDA standards (USDA, 1976). The USDA standards for apple quality explicitly define the quality criteria so that it is quite straightforward for an expert to follow up and apply them. Extremely large or small apples were already excluded by the handling personnel. Eighty of the apples were kept at room temperature for 4 days while another 80 were kept in a cooler (at about 3°C) for the same period to create color variation on the surfaces of apples. In addition, 21 of the apples were harvested before the others and kept for 15 days at room temperature for the same purpose of creating a variation in the appearance of the apples to be tested.

The Hue angle ($\tan^{-1}(b/a)$), which was used to represent the color of apples, was shown to be the best representation of human recognition of color. To simplify the problem, defects were collected under a single numerical value, "defect" after normalizing each defect component such as bruises, natural defects, russetting and size defects (lopsidedness).

$$\text{Defect} = 10 \times B + 5 \times ND + 3 \times R + 0.3 \times SD \qquad \text{...(9.1)}$$

where B is the amount of bruising, ND is the amount of natural defects, such as scars and leaf roller, as total area (normalized), R is the total area of russeting defect (normalized) and SD is the normalized size defect. Similarly, circumference, blush (reddish spots on the cheek of an apple) percentage and weight were combined under "Size" using the same procedure as with "Defect"

$$\text{Size} = 5 \times C + 3 \times W + 5 \times BL \qquad \text{...(9.2)}$$

where C is the circumference of the apple (normalized), W is weight (normalized) and BL is the normalized blush percentage. Coefficients used in the above equations were subjectively selected, based on the expert's expectations and USDA standards (USDA, 1976).

Although it was measured at the beginning, firmness was excluded from the evaluation, as it was difficult for the human expert to quantify it nondestructively. After the combinations of features given in the above equations, input variables were reduced to 3 defect, size and color. Along with the measurements of features, the apples were graded by the human expert into three quality groups, bad, medium and good, depending on the expert's experience, expectations and USDA standards (USDA, 1976). Fuzzy logic techniques were applied to classify apples after measuring the quality features. The grading performance of fuzzy logic proposed was determined by comparing the classification results from *FL* and the expert.

9.6.3 Application of Fuzzy Logic

Three main operations were applied in the fuzzy logic decision making process: selection of fuzzy inputs and outputs, formation of fuzzy rules, and fuzzy inference. A trial and error approach was used to develop membership functions. Although triangular and trapezoidal functions were used in establishing membership functions for defects and color (Fig. 9.9 and 9.10), an exponential function with the base of the irrational number *e* was used to simulate the inclination of the human expert in grading apples in terms of size (Fig. 9.11).

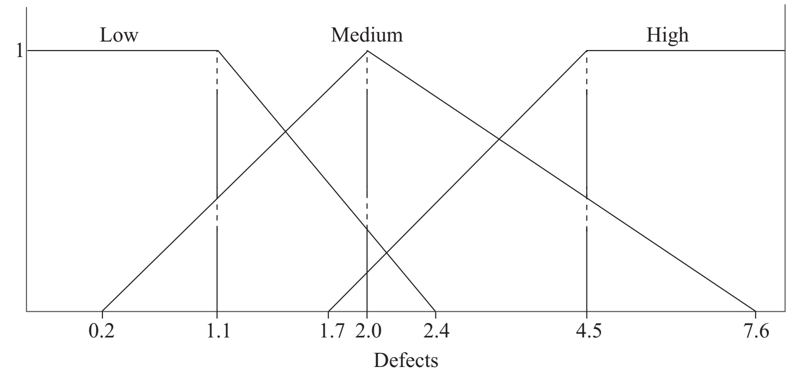

Fig. 9.9 Membership functions for the defect feature.

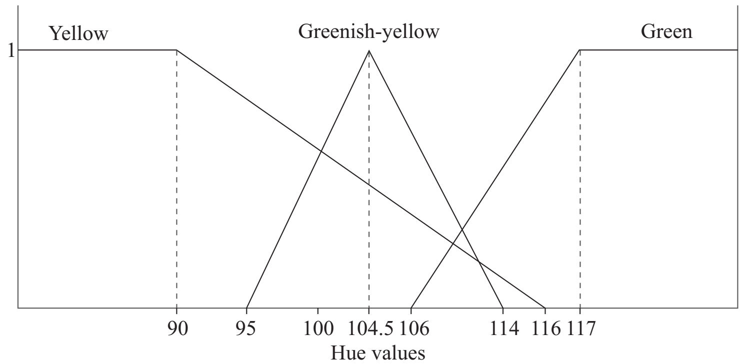

Fig. 9.10 Membership functions for the color feature.

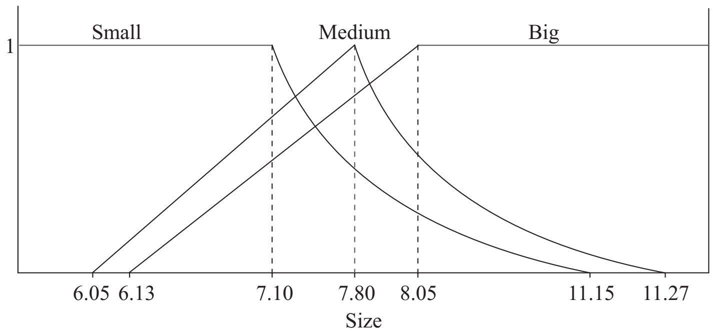

Fig. 9.11 Membership functions for the size feature.

$$\text{Size} = e^x \qquad \text{...(9.3)}$$

where e is approximately 2.71828 and x is the value of size feature.

9.6.4 Fuzzy Rules

At this stage, human linguistic expressions were involved in fuzzy rules. The rules used in the evaluations of apple quality are given in Table 9.2. Two of the rules used to evaluate the quality of Golden Delicious apples are given below:

If the color is greenish, there is no defect, and it is a well formed large apple, then quality is very good (rule $Q_{1,1}$ in Table 9.2).

Table 9.2: Fuzzy rule tabulation

	$C_1 + S_1$	$C_1 + S_2$	$C_1 + S_3$	$C_2 + S_1$	$C_2 + S_2$	$C_2 + S_3$	$C_3 + S_1$	$C_2 + S_2$	$C_3 + S_3$
D_1	$Q_{1,1}$	$Q_{1,2}$	$Q_{2,3}$	$Q_{1,3}$	$Q_{2,5}$	$Q_{3,8}$	$Q_{2,6}$	$Q_{2,7}$	$Q_{3,15}$
D_2	$Q_{2,1}$	$Q_{2,2}$	$Q_{3,3}$	$Q_{2,4}$	$Q_{3,6}$	$Q_{3,9}$	$Q_{3,11}$	$Q_{3,13}$	$Q_{3,16}$
D_3	$Q_{3,1}$	$Q_{3,2}$	$Q_{3,4}$	$Q_{3,5}$	$Q_{3,7}$	$Q_{3,10}$	$Q_{3,12}$	$Q_{3,14}$	$Q_{3,17}$

Where, C_1 is the greenish color quality (desired), C_2 is greenish-yellow color quality medium), and C_3 is yellow color quality (bad); S_1, on the other hand, is well formed size (desired), S_2 is moderately formed size (medium), S_3 is badly formed size (bad). Finally, D_1 represents a low amount of defects (desired), while D_2 and D_3 represent moderate (medium) and high (bad) amounts of defects, respectively. For quality groups represented with "Q" in Table 1, the first subscript 1 stands for the best quality group, while 2 and 3 stand for the moderate and bad quality groups, respectively. The second subscript of Q shows the number of rules for the particular quality group, which ranges from 1 to 17 for the bad quality group.

If the color is pure yellow (overripe), there are a lot of defects, and it is a badly formed (small) apple, then quality is very bad (rule $Q_{3,17}$ in Table 9.2).

A fuzzy set is defined by the expression below: $^{-}{}_{C}$

$$D = \{X.\ \mu_0(x))|\ x \in X\} \qquad ...(9.4)$$

$$\mu_0(x): \to [0, 1]$$

where X represents the universal set, D is a fuzzy subset in X and $\mu D(x)$ is the membership function of fuzzy set D. Degree of membership for any set ranges from 0 to1. A value of 1.0 represents a 100% membership while a value of 0 means 0% membership. If there are three subgroups of size, then three memberships are required to express the size values in a fuzzy rule.

Three primary set operations in fuzzy logic are AND, OR, and the Complement, which are given as follows

AND: $$\mu_C \wedge \mu_D = \min\ \{\mu_C, \mu_D\} \qquad ...(9.5)$$

OR: $$\mu_C \cup \mu_D = (\mu_C \vee \mu_D) = \max\ \{\mu_C, \mu_D\} \qquad ...(9.6)$$

$$\text{complement} = \quad = 1 - \mu_D \qquad ...(9.7)$$

The minimum method given by equation (9.5) was used to combine the membership degrees from each rule established. The minimum method chooses the most certain output among all the membership degrees. An example of the fuzzy AND (the minimum method) used in ***if-then*** rules to form the Q_{11} quality group in Table 9.2 is given as follows;

$$Q_{11} = (C_1 \wedge S_1 \wedge D_1) = \min \{C_1, S_1, D_1\} \quad \text{...(9.8)}$$

On the other hand, the fuzzy OR (the maximum method) rule was used in evaluating the results of the fuzzy rules given in Table 9.2; determination of the quality group that an apple would belong to, for instance, was done by calculating the most likely membership degree using equations 9.9 through 9.13. If,

$$k_1 = (Q_{1,1}, Q_{1,2}, Q_{1,3}) \quad \text{...(9.9)}$$

$$k_2 = (Q_{2,1}, Q_{2,2}, Q_{2,3}\ Q_{2,4}, Q_{2,5}, Q_{2,6}) \quad \text{...(9.10)}$$

$$k_3 = (Q_{3,1}, Q_{3,2}, Q_{3,3}\ Q_{3,4}, Q_{3,5},$$
$$Q_{3,6}, Q_{3,7}, Q_{3,8}, Q_{3,9}, Q_{3,10}, Q_{3,11}, Q_{3,12}, Q_{3,13}, Q_{3,14}, Q_{3,15}, Q_{3,16}, Q_{3,17}) \quad \text{...(9.11)}$$

where k is the quality output group that contains different class membership degrees and the output vector y given in equation 10 below determines the probabilities of belonging to a quality group for an input sample before defuzzification:

$$y = [\max (k_1)\ \max (k_2)\ \max (k_3)] \quad \text{...(9.12)}$$

where, for example,

$$\max (k_1) = (Q_{1,1} \vee Q_{1,2} \vee Q_{1,3}) = \max \{Q_{1,1}, Q_{1,2}, Q_{1,3}\} \quad \text{...(9.13)}$$

then, equation 11 produces the membership degree for the best class (Lee, 1990).

9.6.5 Determination of Membership Functions

Membership functions are in general developed by using intuition and qualitative assessment of the relations between the input variable(s) and output classes. In the existence of more than one membership function that is actually in the nature of the fuzzy logic approach, the challenge is to assign input data into one or more of the overlapping membership functions. These functions can be defined either by linguistic terms or numerical ranges, or both. The membership function used in this study for defect quality in general is given in equation 9.4. The membership function for high amounts of defects, for instance, was formed as given below:

$$\frac{(x(1) - 0.24)}{1.76}$$

If the input vector x is given as x = [defects, size, color], then the membership function for the class of a high amount of defects (D_3) is

$$\mu(D_3) = 0, \text{ when } x(1) < 1.75$$

$$\mu(D_3) = \frac{(x(1) - 1.75)}{2.77}, \text{ when } 1.75 \le x(1) \le 4.52 \text{ or} \quad \text{...(9.14)}$$

$$\mu(D_3) = 1, \text{ when } x(1) \ge 4.52$$

For a medium amount of defects (D_2), the membership function is

$$\mu(D_2) = 0, \text{ when defect innput } x(1) < 0.24 \text{ or } x(1) > 7.6$$

$$\mu(D_2) = \qquad , \text{ when } 0.24 \le x(1) \le 2 \quad \text{...(9.15)}$$

$$\mu(D_2) = \frac{(7.6 - x(1))}{5.6}, \text{ when } 2 \le x(1) \le 7.6$$

For a low amount of defects (D_1), the membership function is

$$\mu(D_1) = 0, \text{ when defect input } x(1) > 0.24$$

$$\mu(D_1) = \frac{(0.24 - x(1))}{1.3}, \text{ when } 1.1 \le x(1) \le 2.4 \quad \text{...(9.16)}$$

$$\mu(D_1) = 1, \text{ when } x(1) \le 1.1.$$

Calculations for the quality groups of color and size were performed using the same approach as defect. Three membership functions for the quality classes of defect, color and size are schematically shown in Fig. 9.9, 9.10 and 9.11, respectively.

9.6.6 Defuzzification

The centroid method, which is also known as the center of mass, was used for defuzzification. After execution of the rules established and shown in the previous section, the output grades described below are obtained.

For low quality, degree of membership is calculated using equation 9.17.

$$SC = \frac{[(y(3) \times (c + 1.5))]}{2} \quad \text{...(9.17)}$$

where,

$$c = c_2 - c_1$$

$$c_1 = [0.5 \times y(3) + 0.75]$$

$$c_2 = [2.25 - (0.5 \times y(3))]$$

y (3) is the low quality output from the output vector y (equation 9.12), c, c_1, and c_2 are shown in Fig. 9.12 and *sc* is the area of the trapezoid formed. Membership degrees of *sb* and *sa* are calculated using the same approach in equation 15 for the medium and high quality classes.

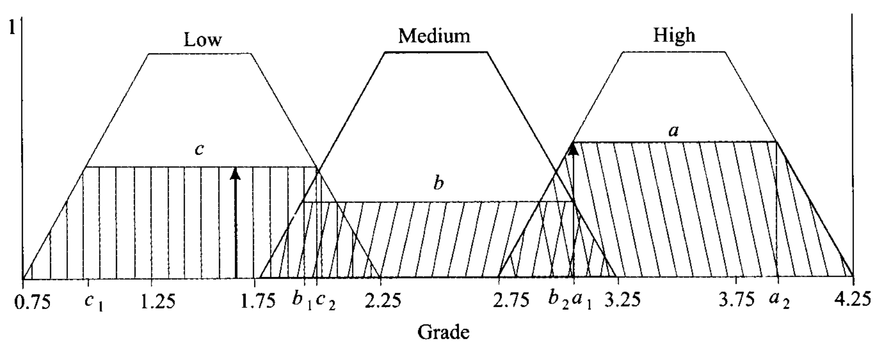

Fig. 9.12 Membership functions for output quality groups and determination of a grade for an apple.

In the defuzzification stage, the overall grade for a particular apple was found by taking the average of the weighted possible outputs using the weighted average method. Equation 9.18 was used for this purpose

$$wa = \frac{sa \times 3.5 + sb \times 2.5 + sc \times 1.5}{sa + sb + sc} \qquad ...(9.18)$$

where wa is the weighted average for the grade of a particular apple. From each of the 3 output categories, trapezoidal areas (sa, sb, and sc) were calculated for the apple being graded. Then, the weighted average of the 3 trapezoidal areas shown in Fig. 9.12 was calculated to find the final grade for the particular apple.

9.6.7 Results and Discussion

In the results of the defuzzification process, grades for all the apples were calculated between 0 and 3.99. Grade (g) ranges for the output quality classes were chosen as follows: $2.3 \leq g \leq 4$ for the best class, $1.4 \leq g < 2.3$ for the moderate class and $0 \leq g < 1.4$ for the bad class. The resulting classification accuracies obtained from fuzzy logic are given in Table 9.3 in comparison with the classification results from the expert. Fuzzy logic predicted around 89% of apples correctly (Table 9.3). Misclassification errors observed were among adjacent groups in general. This kind of error is usually acceptable.

Table 9.3: Comparison if fuzzy logic and human expert in classification of apples.

		Fuzzy logic prediction				
	Class	1	2	3	Total predicted	%
Human Expert	1	121	9	0	130	93.1
	2	6	33	2	41	80.5
	3	1	2	7	10	70.0
Total Observed		128	44	9	161*/181	
%		94.5	75.0	77.8		89.0

*Number of apples correctly classified by fuzzy logic.

Determination of membership functions in terms of shape and boundary has a clear effect on the result of classification performed by fuzzy logic. This situation greatly depends on experience and knowledge. Finding the right shape and the boundaries for the membership function will increase the accuracy of the fuzzy logic application. Statistics of the class populations, such as average, standard deviation and minimum-maximum values, could help the determination of membership functions. Therefore, parameters of fuzzy logic, such as function shape, threshold, which is to determine the overlapping amount and condition among the membership functions, input and output levels, and function rules, must be tested to find the optimum classification result. These application criterions of fuzzy logic that must be investigated are both disadvantageous as it takes time to apply all the alternatives, and powerful as it provides an opportunity to build a system compatible with the standards and expectations.

9.6.8 Conclusion

Fuzzy logic was successfully applied to serve as a decision support technique in grading apples. Grading results obtained from fuzzy logic showed a good general agreement with the results from the human expert, providing good flexibility in reflecting the expert's expectations and grading standards into the results. It was also seen that color, defects and size are three important criteria in apple classification. However, variables such as firmness, internal defects and some other sensory evaluations, in addition to the features mentioned earlier, could increase the efficiency of decisions made regarding apple quality.

9.7 AN INTRODUCTORY EXAMPLE: FUZZY V/S NON-FUZZY

To illustrate the value of fuzzy logic, fuzzy and non-fuzzy approaches are applied to the same problem. First the problem is solved using the conventional (non-fuzzy) method, writing MATLAB commands that spell out linear and piecewise-linear relations. Then, the same system is solved using fuzzy logic.

Consider the tipping problem: what is the "right" amount to tip your waitperson? Given a number between 0 and 10 that represents the quality of service at a restaurant (where 10 is excellent), what should the tip be?

This problem is based on tipping as it is typically practiced in the United States. An average tip for a meal in the U.S. is 15%, though the actual amount may vary depending on the quality of the service provided.

9.7.1 The Non-Fuzzy Approach

Let's start with the simplest possible relationship (Fig. 9.13). Suppose that the tip always equals 15% of the total bill.

tip = 0.15

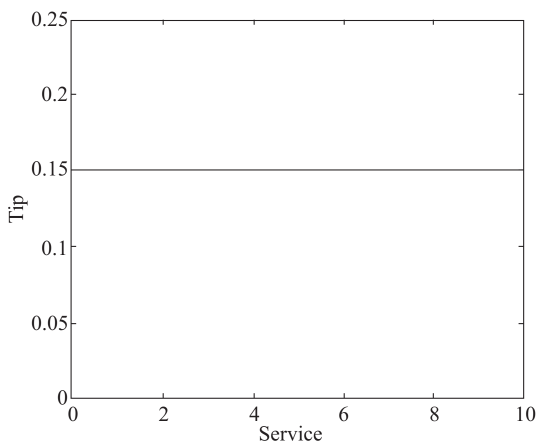

Fig. 9.13 Constant tipping.

This does not really take into account the quality of the service, so we need to add a new term to the equation. Since service is rated on a scale of 0 to 10, we might have the tip go linearly from 5% if the service is bad to 25% if the service is excellent (Fig. 9.14). Now our relation looks like this:

tip = 0.20/10 * service + 0.05

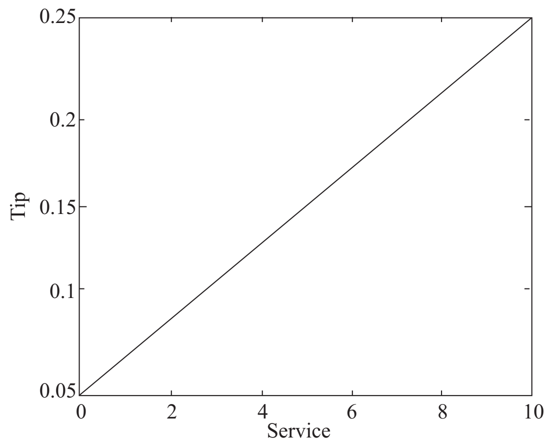

Fig. 9. 14 Linear tipping.

The formula does what we want it to do, and it is pretty straight forward. However, we may want the tip to reflect the quality of the food as well. This extension of the problem is defined as follows:

Given two sets of numbers between 0 and 10 (where 10 is excellent) that respectively represent the quality of the service and the quality of the food at a restaurant, what should the tip be? Let's see how the formula will be affected now that we've added another variable (Fig. 9.15). Suppose we try:

tip = 0.20/20 × (service + food) + 0.05

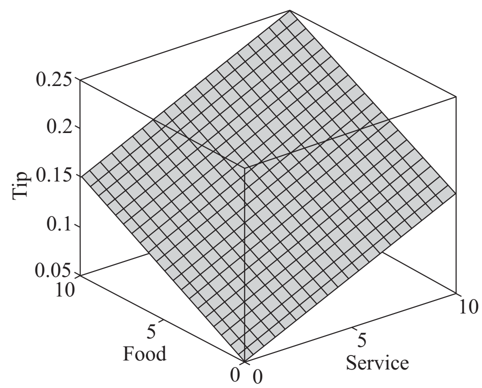

Fig. 9.15 Tipping depend on service and quality of food.

In this case, the results look pretty, but when you look at them closely, they do not seem quite right. Suppose you want the service to be a more important factor than the food quality. Let's say that the service will account for 80% of the overall tipping "grade" and the food will make up the other 20%. Try:

servRatio = 0.8;

tip= servRatio × (0.20/10 × service + 0.05) + (1– servRatio) × (0.20/10 × food + 0.05);

The response is still somehow too uniformly linear. Suppose you want more of a flat response in the middle, *i.e.,* you want to give a 15% tip in general, and will depart from this plateau only if the service is exceptionally good or bad (Fig. 9.16).

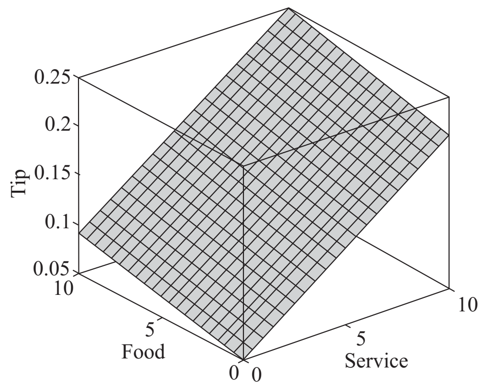

Fig. 9.16 Tipping based on the service to be a more important factor than the food quality.

This, in turn, means that those nice linear mappings no longer apply. We can still salvage things by using a piecewise linear construction (Fig. 9.17). Let's return to the one-dimensional problem of just considering the service. You can string together a simple conditional statement using breakpoints like this:

if service < 3,

tip = (0.10/3) × service + 0.05;

else if service < 7 ,

tip = 0.15;

else if service < =10,

tip = (0.10/3) × (service –7) + 0.15;

end

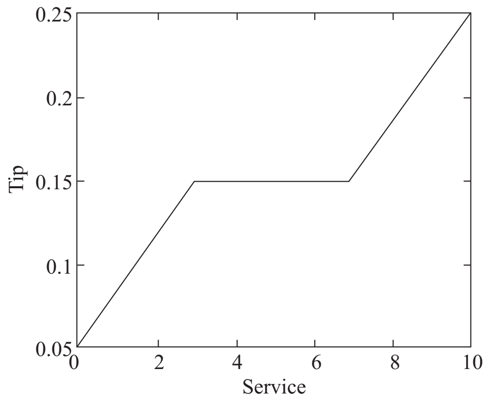

Fig. 9. 17 Tipping using a piecewise linear construction.

If we extend this to two dimensions (Fig. 9.18), where we take food into account again, something like this result:

servRatio = 0.8;

if service < 3,

tip = ((0.10/3) × service + 0.05) × servRatio + (1 – servRatio) × (0.20/10 × food + 0.05);

else if service < 7,

tip = (0.15) × servRatio + (1 – servRatio) × (0.20/10 × food + 0.05);

else,

tip = ((0.10/3) × (service – 7) + 0.15) × servRatio + (1 – servRatio) × (0.20/10 × food + 0.05);

end

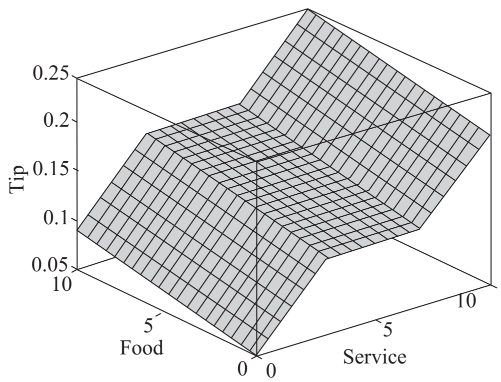

Fig. 9.18 Tipping with two-dimensional variation.

The plot looks good, but the function is surprisingly complicated. It was a little tricky to code this correctly, and it is definitely not easy to modify this code in the future. Moreover, it is even less apparent how the algorithm works to someone who did not witness the original design process.

9.7.2 The Fuzzy Approach

It would be nice if we could just capture the essentials of this problem, leaving aside all the factors that could be arbitrary. If we make a list of what really matters in this problem, we might end up with the following rule descriptions:

1. *If service is poor, then tip is cheap*
2. *If service is good, then tip is average*
3. *If service is excellent, then tip is generous*

The order in which the rules are presented here is arbitrary. It does not matter which rules come first. If we wanted to include the food's effect on the tip, we might add the following two rules:

4. *If food is rancid, then tip is cheap*
5. *If food is delicious, then tip is generous*

In fact, we can combine the two different lists of rules into one tight list of three rules like so:

1. *If service is poor or the food is rancid, then tip is cheap*
2. *If service is good, then tip is average*
3. *If service is excellent or food is delicious, then tip is generous*

These three rules are the core of our solution. And coincidentally, we have just defined the rules for a fuzzy logic system. Now if we give mathematical meaning to the linguistic variables (what is an "average" tip, for example?) we would have a complete fuzzy inference system. Of course, there's a lot left to the methodology of fuzzy logic that we're not mentioning right now, things like:

- How are the rules all combined?
- How do I define mathematically what an "average" tip is?

The details of the method do not really change much from problem to problem - the mechanics of fuzzy logic are not terribly complex. What matters is what we have shown in this preliminary exposition: fuzzy is adaptable, simple, and easily applied.

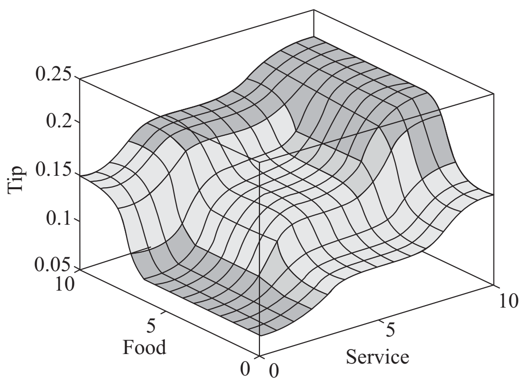

Fig. 9.19 Tipping using fuzzy logic.

Here is the picture associated with the fuzzy system that solves this problem (Fig. 9.19). The picture above was generated by the three rules above.

9.7.3 Some Observations

Here are some observations about the example so far. We found a piecewise linear relation that solved the problem. It worked, but it was something of a nuisance to derive, and once we wrote it down as code, it was not very easy to interpret. On the other hand, the fuzzy system is based on some "common sense" statements. Also, we were able to add two more rules to the bottom of the list that influenced the shape of the overall output without needing to undo what had already been done. In other words, the subsequent modification was pretty easy.

Moreover, by using fuzzy logic rules, the maintenance of the structure of the algorithm decouples along fairly clean lines. The notion of an average tip might change from day to day, city to city, country to country, but the underlying logic the same: if the service is good, the tip should be average. You can recalibrate the method quickly by simply shifting the fuzzy set that defines average without rewriting the fuzzy rules.

You can do this sort of thing with lists of piecewise linear functions, but there is a greater likelihood that recalibration will not be so quick and simple. For example, here is the piecewise linear tipping problem slightly rewritten to make it more generic. It performs the same function as before, only now the constants can be easily changed.

```
% Establish constants
lowTip=0.05; averTip=0.15; highTip=0.25;
tipRange=highTip–lowTip;
badService=0; okayService=3;
goodService=7; greatService=10;
serviceRange=greatService–badService;
badFood=0; greatFood=10;
foodRange=greatFood–badFood;
% If service is poor or food is rancid, tip is cheap
if service<okayService,
tip=(((averTip–lowTip)/(okayService–badService)) ...
*service+lowTip)*servRatio + ...
(1–servRatio)*(tipRange/foodRange*food+lowTip);
% If service is good, tip is average
elseif service<goodService,
tip=averTip*servRatio + (1–servRatio)* ...
(tipRange/foodRange*food+lowTip);
```

```
% If service is excellent or food is delicious, tip is generous
else,
tip=(((highTip–averTip)/ ...
(greatService–goodService))* ...
(service–goodService)+averTip)*servRatio + ...
(1–servRatio)*(tipRange/foodRange*food+lowTip);
end
```

Notice the tendency here, as with all code, for creeping generality to render the algorithm more and more opaque, threatening eventually to obscure it completely. What we are doing here is not that complicated. True, we can fight this tendency to be obscure by adding still more comments, or perhaps by trying to rewrite it in slightly more self-evident ways, but the medium is not on our side.

The truly fascinating thing to notice is that if we remove everything except for three comments, what remain are exactly the fuzzy rules we wrote down before:

```
% If service is poor or food is rancid, tip is cheap
% If service is good, tip is average
% If service is excellent or food is delicious, tip is generous
```

If, as with a fuzzy system, the comment is identical with the code, think how much more likely your code is to have comments! Fuzzy logic lets the language that's clearest to you, high level comments, also have meaning to the machine, which is why it is a very successful technique for bridging the gap between people and machines.