Reprinted from the Journal of the Institution of Electronics and Telecommunication Engineers, Vol. 37, No. 5&6, 1991, pp 548-560

Fuzzy Logic and Approximate Reasoning : An Overview

SANKAR K PAL AND DEBA PRASAD MANDAL

Electronics and Communication Science Unit, Indian Statistical Institute, Calcutta 700 035, India

Approximate Reasoning is the process or processes by which a possible imprecise conclusion is deduced from a collection of imprecise premises. Fuzzy logic plays the major role in approximate reasoning. It has the ability to deal with different types of uncertainty.

An overview of the different aspects of the theory of approximate reasoning has been provided here based on the existing literature. Suitable illustrations are included, whenever necessary, to make the concept clear. Some of the implementation of the theory to real life problems have been mentioned. Finally, a linguistic recognition system based on approximate reasoning has been described along with its implementation in speech recognition problem.

Indexing terms : Fuzzy sets, Approximate reasoning, Management of uncertainty, Recognition system

LOGIC, according to Webster's dictionary, is the science of the formal principles of reasoning. In this sense, fuzzy logic is concerned with the formal principles of approximate reasoning. To be more specific, it aims at modelling the human reasoning system. Most of human reasoning is approximate rather than precise in nature. Based on a store of knowledge, we have the ability to infer an approximate answer to a question. For example:

(*i*) Ram is much younger than Madhu. What is the age of Ram?

(*ii*) Smartness is attractive. Jaga is smart. Is Jaga attractive?

Fuzzy logic addresses these problems in the following ways. First, the meaning of an imprecise proposition is represented as an elastic constraint on a variable; and second, the answer to a query is deduced through a propagation of elastic constraints.

By approximate reasoning, we mean a type of reasoning, which is neither very exact nor very inexact. In other words, it is the process or processes by which a possible imprecise conclusion is deduced from a collection of imprecise premises. In a simplest way, we can say that, fuzzy logic plays a key role in approximate reasoning. The distinguished features of a fuzzy logic [1] are (*i*) fuzzy truth values expressed in linguistic terms; (*ii*) imprecise truth tables, and (*iii*) rules of inference whose validity is approximate rather than exact. In these respects fuzzy logic differs significantly from standard logical systems ranging from the classical Aristotelian to inductive logics and many-valued logics.

Management of uncertainty is an important issue in the design of a knowledge based systems, because much of the information in the knowledge base is imprecise, incomplete or not totally reliable. In the existing knowledge based systems, uncertainty is dealt with through a combination of predicate logic and probability-based methods. A serious short coming of these methods is that they are not capable of handling pervisive fuzziness of information in the knowledge base and, as a result, are mostly ad hoc in nature.

Zadeh has suggested an alternative approach [2] to the management of uncertainty which is based on fuzzy logic. A feature of fuzzy logic which is of particular importance to the management of uncertainty in knowledge based system is that it provides a systematic frame work for dealing with fuzzy quantifiers *eg*, most, many, about, few etc. In this way, fuzzy logic subsumes both predicate logic and probability theory, and makes it possible to deal with different types of uncertainty within a single conceptual frame-work.

During the past few years fuzzy logic has found several applications ranging from process control to medical diagnosis. The basic idea underlying fuzzy logic was suggested by Zadeh [3-5]. Mamdadani and Assilian found its first application [6] in connection with the regulation of a steam engine. One of the first commercial applications of fuzzy logic to process control was the development by F L Smidth & Co of Copenhagen of a micro processor controller for cement klins [7]. Important applications of fuzzy logic to engineering design and systems analysis have been made by Baldwin and his associates in the department of engineering & mathematics at the University of Bristol in England; Prade and Dubois of Paul Sabatier University in Toulouse, France; Sugeno of the Tokyo Institute of Technology in Yokshama; and Mizumoto of the Osaka Electro-Communication University, among others [8].

An overview of the existing theory of approximate reasoning has been provided here. This includes an introduction to the concept linguistic variable; overview of classical logics and fuzzy logic; the concept of transition between different truth spaces, operations in truth space; fuzzy restriction and composition inference. Its various applications are mentioned. Finally, a linguistic recognition system based on approximate reasoning has been presented along with its successful implementation to speech recognition problem.

LINGUISTIC VARIABLES

The basic concept in fuzzy logic, that plays a key role in approximate reasoning is a linguistic variable, which in early seventies was called a variable of higher order rather than a fuzzy variable. A linguistic variable, as its name suggests, is a variable whose values are not numbers but words or sentences in a natural language. For example, height is a linguistic variable if its values are linguistic rather than numerical, *ie*, short, tall, very short, not short, not very tall, quite tall, not very short and not very tall etc, rather than 100, 110, 120, ... (in cm). In general, the values of a linguistic variable can be generated from a primary term (*eg*, "tall"), its antonyms (*eg*, "short"), a collection of modifiers (*eg*, "not", "very", "more or less", "quite", "not very" etc) and the connectors ("and" and "or").

Definition 1 [9]

A linguistic variable is characterized by a quintuple $(X, T(X), U, G, M)$ in which X is the name of the variable; $T(X)$ is the term set of X; U is a universe of discourse; G is a syntactic rule which generates the terms in $T(X)$; and M is a semantic rule which associates with each linguistic value X its meaning where $M(X)$ denotes a fuzzy subset of U. For a particular X, the name generated by G, is called a term.

Example 1

Suppose X is a linguistic variable with the label "Height" with $U = [0,250]$. Terms of this linguistic variable, which are fuzzy sets, could be called "tall", "short", "very tall" and so on. The base variable U is the height in cm of persons. $M(X)$ is the rule that assigns a meaning, that is, a fuzzy set to the term. For example, we can write for the term "tall"

$$M(\text{tall}) = \{x, \mu_{tall}(x)\}, \qquad x \in [0,250]$$

$$\text{where} \quad \mu_{tall}(x) = \begin{cases} 0 & \text{for } x \in [0,150] \\ [1+((x-150)/10^{-2}]^{-1} & \text{for } x \in [150,250] \end{cases} \tag{1}$$

$T(X)$ will define the term set of the variable X for instance in this case

$T(\text{Height})$ = {tall, very tall, not very tall, quite tall, short, more or less short, ...}

where $G(X)$ is a rule which generates the terms in the term set. This is explained in Fig 1.

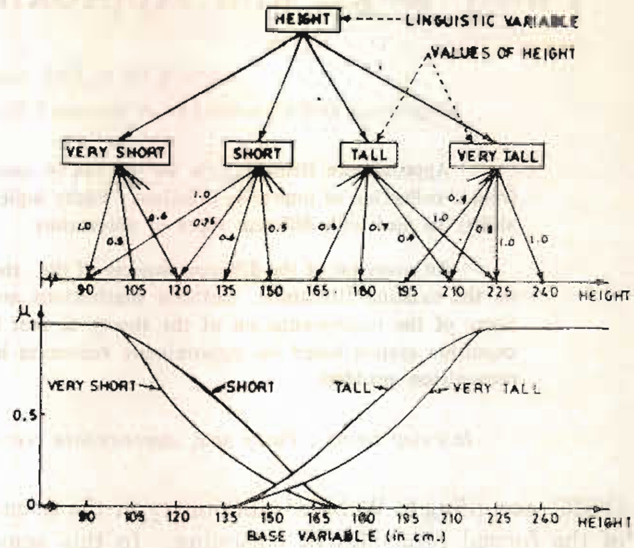

Fig 1 Linguistic variable height with its compatibility

Definition 2 [10]

A linguistic hedge or a modifier is an operator, which modifies the meaning of a term or more generally of a fuzzy set. If A is a fuzzy set then the modifier m generates the composite term $B = m(A)$.

The mathematical models which are used very frequently for modifiers are

$$\text{Concentration :} \quad \mu_{CON(A)} = (\mu_A(x))^2 \tag{2a}$$

$$\text{Dilation :} \quad \mu_{DIL(A)} = (\mu_A(x))^{1/2} \tag{2b}$$

Contrast intensification :

$$\mu_{INT(A)}(x) = \begin{cases} 2(\mu_A(x))^2 & \text{for } \mu_A(x) \in [0,0.5] \\ 1-2(1-\mu_A(x))^2 & \text{otherwise} \end{cases} \tag{2c}$$

Generally, the following linguistic hedges are associated with the above mentioned mathematical operators:

If A is a term (a fuzzy set) then

very A	$\equiv$ CON(A)	(3*a*)
More or less A	$\equiv$ DIL(A)	(3*b*)
plus A	$\equiv A^{1.25}$	(3*c*)
minus A	$\equiv A^{0.75}$	(3*d*)
slightly A	$\equiv$ INT[plus A and not (very A)]	(3*e*)

REASONING WITH LOGIC

Classical logics

Logics, as bases for reasoning, can be distinguished by their topic-neutral (context-independent) items, truth values, vocabulary (operators), and reasoning procedures (tautologics, syllogism). In Boolean logic, truth values can be 0 (false) or 1 (true) and by means of these truth values, the vocabulary (operators) are defined via truth tables.

The reasoning procedures are generally based on tautologies such as

$$\left.\begin{array}{lll} \text{modus ponens} & : (P \wedge (P \Rightarrow Q)) & \Rightarrow Q \\ \text{modus tollens} & : ((P \Rightarrow Q) \wedge 7Q) & \Rightarrow 7P \\ \text{syllogism} & : ((P \Rightarrow Q) \wedge (Q \Rightarrow R)) & \Rightarrow (P \Rightarrow R) \\ \text{contraposition} & : (P \Rightarrow Q) & \Rightarrow (7Q \Rightarrow 7P) \end{array}\right\} \quad (4)$$

As for example, modus ponens could be interpreted as "If P is true and if the statement 'If P is true then Q is true' is also true then Q is true".

The term "true" is used at different places and in two different senses; all but the last "true" are taken as a matter of fact, while the last "true" is a topic independent logical (neeessary) "true". In Boolean logic, however, these "true's" are all treated the same way. A distinction between material and logical truth is made in so called extended logics. Modus logic distinguishes between necessary and possible truth, tense logic between statements which were true in the past and those which will be true in the future. Epistemic logic deals with knowledge and belief, and deontic logic with what ought to be done and which is permitted to be true. Particularly modal logic might be a very good basis for applying different measures and theories of uncertainty. Another extension of Boolean logic is predicate calculus, which uses quantifiers (all, etc) and predicates in addition to the operators of Boolean logic.

Fuzzy logic

A fuzzy logic, FL, may be viewed, in part, as a fuzzy extension of a non-fuzzy multi-valued logic which constitutes, a base logic for FL. So the standard Lukasiewicz logic L_1, will be used as the base logic for FL. Some operations of L_1 are

$$\left.\begin{array}{ll} v(7p) & = 1-v(p) \\ v(p \vee q) & = \max(v(p), v(q)) \\ v(p \wedge q) & = \min(v(p), v(q)) \\ v(p \Rightarrow q) & = \min(1, 1-v(p)+v(q)) \end{array}\right\} \quad (5)$$

where $v(p)$ denotes the truth-value of a proposition p, 7 is the negation, $\wedge$ is the conjunction, $\vee$ is the disjunction and $\Rightarrow$ is the implication.

The truth values here are linguistic variables. So far possibility theory has primarily been used in order to define operators in fuzzy logic, even though other operators have been investigated [11].

Basic principles [2,12]

The main features of fuzzy logic that differentiate it from traditional logical systems are the following:

(*i*) In two-valued logical systems, a proposition p is either true or false. In multivalued logical system, a proposition may be true or false or have an intermediate truth value, which may be an element of a finite or infinite truth value set T. In fuzzy logic, the truth values are allowed to range over the fuzzy subsets of T.

(*ii*) The predicates in two valued logic are constrained to be crisp in the sense that the denotation of a predicate must be a nonfuzzy subset of the universe of discourse. In fuzzy logic, the predicates may be crisp or, more generally fuzzy.

(*iii*) Two-valued as well as multi-valued logics allow only two quantifiers: all and some. In contrast, fuzzy logic allows, in addition, the use of fuzzy quantifiers exemplified by most, many, several, few, much of etc.

(*iv*) Fuzzy logic provides a method for representating the meaning of both non fuzzy and fuzzy modifiers exemplified by not, very, more or less, much, slightly etc.

(*v*) Fuzzy logic has three principle modes of qualification. These are

#truth-qualification as in "(Gavaskar is tall) is not quite true", where (Gavaskar is tall) is the qualified proposition and the "not quite true" is the qualifying truth value.

#probability-qualification, as in "(Gavaskar is tall) is unlikely", where unlikely is the qualifying fuzzy probability.

#possibility-qualification, as in "(Gavaskar is tall) is almost impossible" where "almost impossible" is the qualifying fuzzy possibility.

As we are concerned here with the approximate reasoning, we have restricted our discussion based on truth value set only.

The truth-value set of FL

The truth value set of FL is assumed to be a countable set T of the form

T = {true, false, not true, very true, ...}.

Let μ_y : [0, 1] → [0, 1] denote the compatibility function of γ. Then the meaning of γ is expressed by

$$\gamma = \int_{\nu=0}^{1} \mu_\gamma(\nu)/\nu \quad (6)$$

where the grade of membership or compatibility of ν in the fuzzy set labeled γ is $\mu_\gamma(\nu)$, and the integral sign, $\int$, denotes the union of fuzzy singletones $\mu_\gamma(\nu)/\nu$.

The term set of the linguistic variable "Truth" has been defined in various ways by different authors. Baldwin [13] defined some of the terms as shown in Fig 2. Here

$$\left.\begin{aligned} \mu_{true}(\nu) &= \nu, && \forall \nu \in [0,1] \\ \mu_{false}(\nu) &= 1-\mu_{true}(\nu), && \forall \nu \in [0,1] \\ \mu_{very\ true}(\nu) &= [\mu_{true}(\nu)]^2, && \forall \nu \in [0,1] \\ \mu_{fairly\ true}(\nu) &= [\mu_{true}(\nu)]^{1/2}, && \forall \nu \in [0,1] \end{aligned}\right\} \quad (7)$$

and so on.

Zadeh [1,14] suggests for the term "true" the membership function

$$\mu_{true}(\nu) = \left\{ \begin{array}{ll} 0 & \text{for } 0 \leqslant \nu \leqslant a \\ 2-((\nu-a)/(1-a))^2 & \text{for } a \leqslant \nu \leqslant (a+1)/2 \\ 1-((\nu-1)/(1-a))^2 & \text{for } (a+1)/2 \leqslant \nu \leqslant 1 \end{array} \right\} \quad (8)$$

where $\nu = (a+1)/2$ is the crossover point, and $a \in [0,1]$ is a parameter which indicates the subjective judgement about a minimum value of ν in order to consider a statement as "true" at all.

In terms of the meaning of "true", the truth-value "false" may be defined as

$$\text{false} = \int_{\nu=0}^{1} \mu_{true}(1-\nu)/\nu \quad (9)$$

where "not true" is given by

$$\text{not true} = \int_{\nu=0}^{1} (1-\mu_{true}(\nu))/\nu \quad (10)$$

Figure 3 shows the terms "true" and "false". Thus, as a fuzzy set, "not true" is the complement of "true" whereas, "false" is the truth value of the proposition "not p", if "true" is the truth value of p. This concept can very easily be visualised if we consider the terms tall, not tall and short.

The membership function of "true" can be chosen from the finite universe of truth values. The fuzzy sets (possibility distribution) of other terms can essentially be determined from the term "true" by applying modifiers (hedges) appropriately.

In general, a fuzzy truth-value, Φ, would normally have to be approximated by a linguistic truth value, Φ^*, which is in T. The relation between Φ^* and Φ will be expressed as

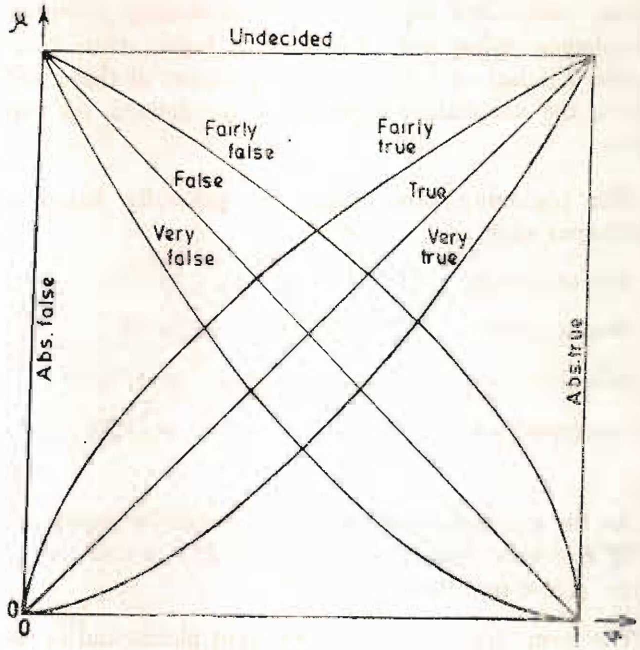

Fig 2 Term set of "truth" [30]

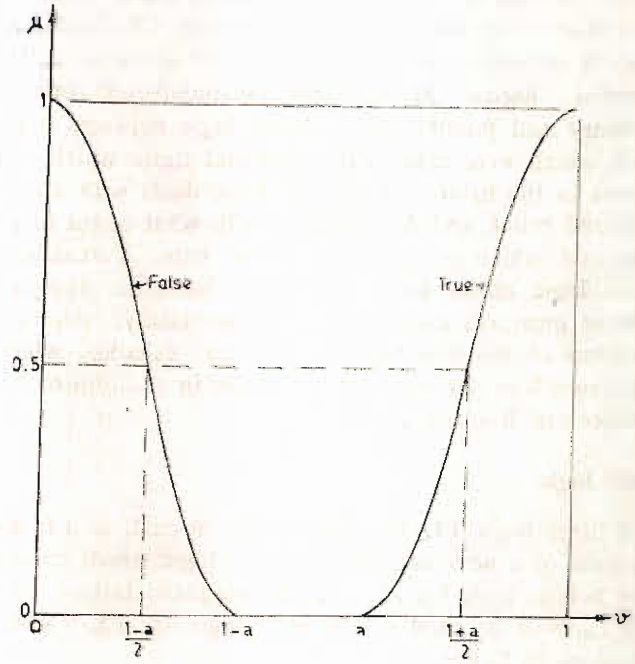

Fig 3 Terms true and false [1]

$$\Phi^* = \text{LA}[\Phi] \quad (11)$$

where LA is an abbreviation for linguistic approximation.

At present, there is no simple or general technique for finding a good linguistic approximation to a given fuzzy subset of V. In most cases, such an approximation is ad hoc in nature, without a precisely defined criterion of the goodness of approximation. This, however, is entirely consistent with the imprecise nature of fuzzy logic and its role in approximate reasoning.

BASIC OPERATIONS IN TRUTH SPACE

For the application of the method of approximate reasoning the information at the start of a problem (the antecedents) and the information required at the end (the consequent) are all presented in terms of fuzzy subsets of various universes of discourse. It is an essential feature of the method that such information is converted to truth restrictions and then evaluated; the result is converted back to a fuzzy subset of the required universe of discourse, called its Truth Functional Modification (TFM).

Truth functional modification

If we have a fuzzy subset of universe of discourse and a truth restriction on that fuzzy subset, then we can find another fuzzy subset of discourse which is an equivalent statement of our knowledge, as follows

$$P \subseteq X,\ Q \subseteq X,\ \gamma \subseteq [0,1]$$

$$(u \text{ is } P) \text{ is } \gamma \equiv (u \text{ is } Q)$$

where
$$\mu_Q(x) = \mu_\gamma(\mu_P(x)), \quad \forall x \in X \tag{12}$$

For example,

(Jayanta is tall) is very true ≡ Jayanta is very tall.

Evaluation of the membership function of (12) is shown in Fig 4. P is shown by its membership as a function of x. γ is shown similarly but with the axes rotated 90° from the conventional orientation so that μ_γ is shown as a function of μ_P. For digital computation of $\mu_Q(x)$, Baldwin [15] derived an algorithm for TFM.

The inverse of truth functional modification is the conversion of the representation of a proposition from the universe of discourse to the truth space. For example, the truth funtional restriction of the statement (Jayanta is tall) given that (Jayanta is fairly tall) is fairly true.

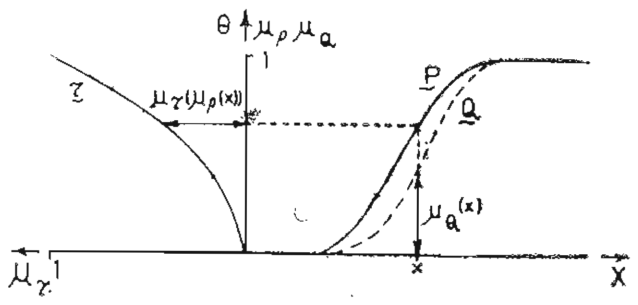

Fig 4 Truth Functional Modification [14]

Once propositions are represented by their truth value restrictions in the truth space, there are some processes that are used in evaluation of any deduction that may be made from them. Conjunction and disjunction are the processes that simply combine informations about propositions that were previously expressed in terms of different universe of discourse.

Conjunction

If we have two truth value restrictions $V(P)$ and $V(Q)$, say, we can combine them with the AND operation as follows

$$\mu_{V(P\ AND\ Q)}(\theta) = \underset{\substack{\alpha,\beta \in [0,1] \\ \theta = \alpha \wedge \beta}}{\text{MAX}} \{\mu_{V(P)}(\alpha) \wedge \mu_{V(Q)}(\beta)\}, \quad \forall \theta \in [0,1] \tag{13}$$

As can be seen from the Fig 5, the conditions of the maximization can be expressed as

$$\mu_{V(P\ AND\ Q)}(\theta) = \underset{\substack{\alpha \in [\theta,1];\beta=\theta \\ \beta \in [\theta,1];\alpha=\theta}}{\text{MAX}} \{\mu_{V(P)}(\alpha) \wedge \mu_{(Q)}(\beta)\}, \quad \forall \theta \in [0,1]$$

$$= [\mu_{V(P)}(\theta) \wedge \underset{\beta \in [\theta,1]}{\text{MAX}} \{\mu_{V(Q)}(\beta)\}]$$

$$\vee [\mu_{V(Q)}(\theta) \wedge \underset{\alpha \in [\theta,1]}{\text{MAX}} \{\mu_{V(P)}(\alpha)\}]$$

$$\forall \theta \in [0,1] \tag{14}$$

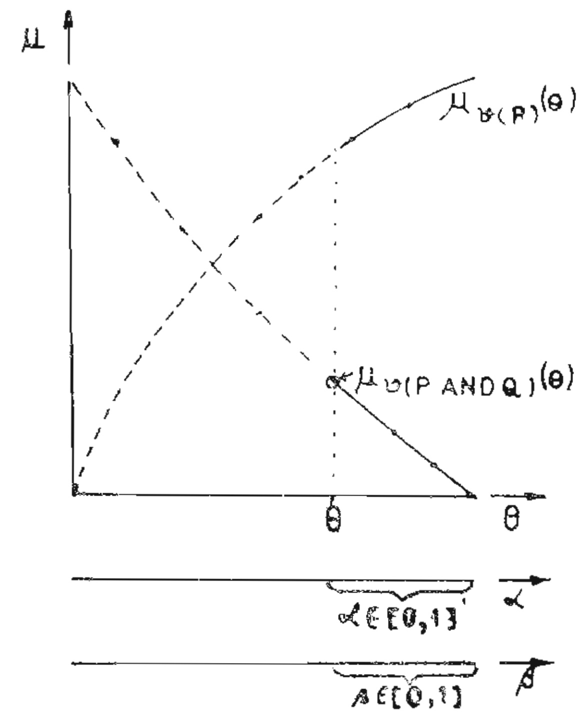

Fig 5 Conjunction [14]

Examble 2: Consider the following two propositions

$p \equiv$ Gautam is fairly tall (*i*)

$q \equiv$ Gautam is very fat (*ii*)

Rephrasing the above two propositions, we get

$p \equiv$ (Gautam is tall) is fairly true (*iii*)

$q \equiv$ (Gautam is fat) is very true (*iv*)

Combining (*iii*) & (*iv*) (*ie*, (*i*) & (*ii*)) by conjunction, we get the proposition r as

$r \equiv$ (Gautam is *tall and fat*) is fairly true

where #fairly true AND very true = fairly true.

Disjunction

Combining the truth value restriction with an OR operation is similar:

$$\mu_{V(P\ OR\ Q)}(\theta) = \underset{\substack{\alpha,\beta \in [0,1] \\ \theta = \alpha \vee \beta}}{\mathrm{MAX}} \{\mu_{V(P)}(\alpha) \wedge \mu_{V(Q)}(\beta)\},\ \forall \theta \in [0,1] \quad (15)$$

As can be seen from the Fig 6, equation (15) is thus changed to

$$\mu_{V(P\ OR\ Q)}(\theta) = [\mu_{V(P)}(\theta) \wedge \underset{\beta \in [0,\theta]}{\mathrm{MAX}} \{\mu_{V(Q)}(\beta)\}] \vee [\mu_{V(Q)}(\theta) \wedge \underset{\alpha \in [0,\theta]}{\mathrm{MAX}} \{\mu_{V(P)}(\alpha)\}] \quad \forall \theta \in [0,1] \quad (16)$$

Examble 3 : Combining (*i*) & (*ii*) of example 2 by disjunction, we get the proposition r as

$r \equiv$ (Gautam is *tall OR fat*) is very true

where #fairly true OR very true $\equiv$ very true.

APPROXIMATE REASONING

Informally, approximate reasoning is the process or processes by which a possible imprecise conclusion is deduced from a collection of imprecise premises. It is the fact that only a small fraction of our thinking could be categorized as precise in either logical or quantitative terms.

Fuzzy restriction [16]

The statements that contain the core of most arguments are expressed in approximate reasoning by fuzzy relations. Consider a fuzzy proposition, p, of the form

$\# p \equiv x$ is Q (17)

where x is a name of an object and Q is a label of a fuzzy

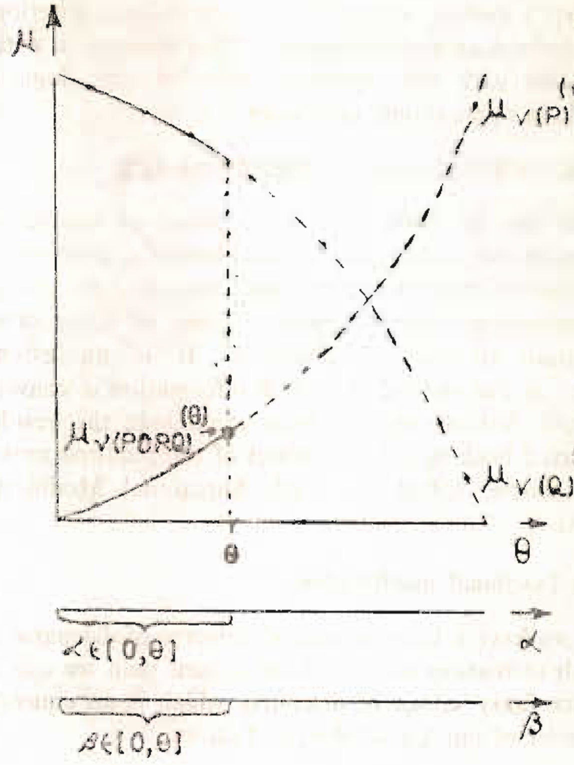

Fig 6 Disjunction [14]

subset of a universe U. Conventionally, p would be interpreted as "x" is a member of Q" or "x belongs to Q".

Definition 3: The meaning of the proposition p is expressed by the relational assignment equation as

$\#\mathbf{R}(A(X)) = Q$ (18)

where A is an implied attribute of x *ie*, an attribute which is implied by x and Q, and **R** denotes a fuzzy restriction on $A(x)$ to which the value Q is assigned by the relational assignment equation.

As an illustration, consider the proposition, "Tarun is young." In this case, the implied attribute is Age and the corresponding relational equation becomes

R (Age (Tarun)) = young

where young being a subset of the interval [0, 100] is defined by a predefined compatibility function. Here Age (Tarun) is the name of a variable and young is its assigned linguistic value.

Thus, approximate reasoning may be viewed as the determination of an approximate solution of a system of relational assignment equations in which the assigned relations are generally, but not necessarily fuzzy rather than non fuzzy subset of a universe of discourse.

The rule of implication is defined as truth function mappings as

$$\mu_I : [0,1] \times [0,1] \rightarrow [0,1] \quad (19)$$

so that $\mu_I(\alpha,\beta) \in [0,1], \forall (\alpha,\beta) \in [0,1] \times [0,1]$.

Composition inference

Zadeh [1] suggested copositional rule of inference, which is defined as follows.

Definition 4: Let A and B denote fuzzy sets in X and $X \times Y$. Then the composition rule of inference asserts that the solution of the relational assignment equations

$$R(x) = A \quad \text{and} \quad R(x,y) = B$$

is given by $$R(y) = A \bigcirc B \equiv C \quad (20)$$

where $A \bigcirc B$ is the max-min composition of A and B. An isomorphic definition is

$$\mu_C(y) = \underset{x}{\text{MAX}} \{\mu_A(x) \wedge \mu_B(x,y)\} \quad (21)$$

Examble 4: Let the universe be $X = \{1,2,3,4\}$.

A = little = {(1,1.0), (2,0.6), (3,0.2), (4,0.0)}

B ≡ "approximately equal" be a fuzzy relation defined by

	1	2	3	4
1	1.0	0.5	0.0	0.0
2	0.5	1.0	0.5	0.0
3	0.0	0.5	1.0	0.5
4	0.0	0.0	0.5	1.0

Applying the max-min-composition for computing

$C(y) = A \bigcirc B$ yields

$$C(y) = \max_x \min \{\mu_A(x), \mu_B(x,y)\}$$

$$= \{(1,1.0), (2,0.6), (3,0.5), (4,0.2)\}$$

≡ more or less little.

The methods will be shown for including truth value restriction on H given one on G (modus ponens) and vice versa (modus tollens). Modus Ponens and Modus Tollens are the two procedures those yield information from the implication statement in the classical syllogism; here these represent approximate operations closely analogous to the deterministic case.

Modus ponens

In traditional logic the main tools of reasoning are tautologies such as for instance the modus ponens, that is

$$[P \wedge (P \Rightarrow Q)] \Rightarrow Q,$$

where P and Q are fuzzy statements or propositions.

The problem to be solved here can be expressed as follows

$$P \Rightarrow Q$$

P has the truth value restriction $V(P)$.

What is the truth value restriction $V(Q)$?

From the composition rule of inference,

$$V(Q) = V(P) \bigcirc I$$

or $$\mu_{V(Q)}(\beta) = \underset{\alpha}{\text{MAX}} \{\mu_{V(P)}(\alpha) \wedge \mu_I(\alpha,\beta)\} \quad (22)$$

where $\mu_I(\alpha,\beta) = 1 \wedge (1-\alpha+\beta)$.

The evaluation of $\mu_{V(Q)}(\beta)$ is shown in Fig 7. An algorithm for this is given by Baldwin [14].

Generalized modus ponens: Let P, P', Q, Q' be fuzzy statements, then the generalized modus ponens reads;

Premise	x is P'	
Implication	if x is P then y is Q	(23)
Conclusion	y is Q'	

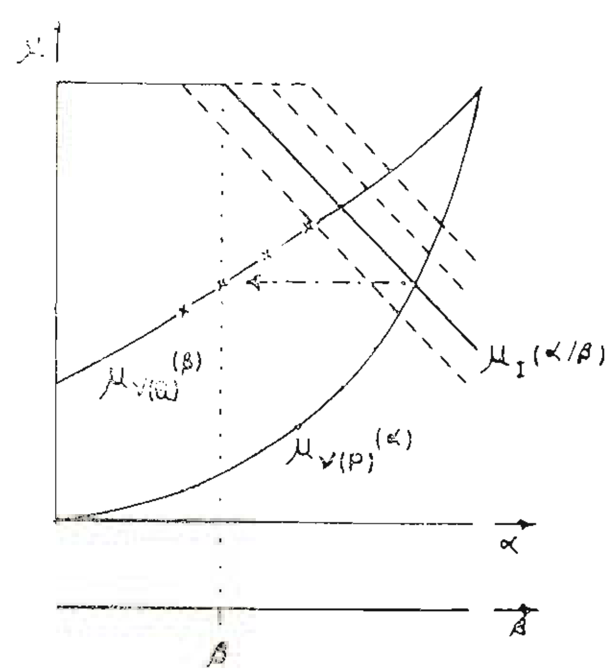

Fig 7 Modus ponens [14]

Example 5:

Ant1 : This tomato is very red

Ant2 : If a tomato is red then the tomato is ripe

Cons : This tomato is very ripe.

Modus tollens

In traditional logic, the tautology for modus tollens is

$$[(P \Rightarrow Q) \wedge Q)] \Rightarrow P$$

The problem to be solved here is the dual of the problem in modus ponens *ie*,

$$P \Rightarrow Q$$

Q has the truth value restriction $V(Q)$.

What is the truth value restriction $V(P)$?

Again the composition rule of inference is used

$$V(P) = V(Q) \bigcirc I$$

or $$\mu_{V(P)}(a) = \underset{\beta}{\mathrm{MAX}} \; \{\mu_{V(Q)}(\beta) \wedge \mu_I(\beta/a)\} \quad (24)$$

The method of computation has shown in Fig 8 and an algorithm to implement this process can be found in Baldwin [14].

Generalized modus tollens: Let P, P', Q, Q' be fuzzy statements, then the generalized modus tollens reads:

Premise	y is Q'	
Implication	If x is P then y is Q	(25)
Conclusion	x is P'	

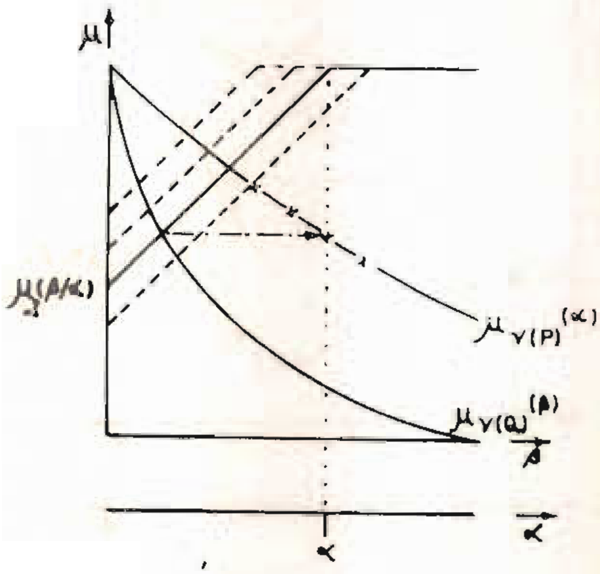

Fig 8 Modus tollens [15]

Example 6

Ant1 : This tomato is very ripe

Ant2 : If a tomato is red then the tomato is ripe

Cons : This tomato is very red.

APPLICATIONS

The easiest way to see how fuzzy logic and approximate reasoning work is to look at their applications. During the past several years, fuzzy logic as well as approximate reasoning has found numerous applications. The first application was done by Mamdani and Assilian [6] in connection with the regulation of a steam engine. Once the basic idea underlying fuzzy logic control became well understood, many applications followed.

Fuzzy controller

Fuzzy control research was started by Mamdani's pioneering work [17], which had been motivated by Zadeh's two papers on fuzzy algorithm [4] and linguistic analysis [9]. The type of a fuzzy variable is to some extend related to the type of fuzzy reasoning method. In fuzzy control, conventional methods of fuzzy reasoning and approximate reasoning are usually used for the sake of simplicity in computation.

The first trial was done by Mamdani on a laboratory steam engine. We can see that a number of applications is now being rapidly increasing. The world's first industrial application of fuzzy control was done by the Danish Cement Klin factory manufacturer, FL Smidth & Co of Copenhagen with the aid of Jonsen and Ostergaard at Technical University of Denmark [7]. Its field test was executed in 1979 and was first marketed in 1980. It is said that the performance of this fuzzy logic controller is slightly better than that of an operator and a reduction in fuel consumption is obtained. Similar process controllers are now being developed and implemented for a variety of other industrial processes.

A particularly interesting type of self-learning fuzzy logic controller was developed at Tokyo Institute of Technology in Japan by Sugeno *et al* [8]. Here the input variables, such as small, big, very small, forward, out, and so on, rather fuzzy than as precise quantity. This controller can drive a model car by employing a sophisticated learning algorithm that automatically reduces the fuzzy control rules through a computer analysis of the operators control actions.

Automatic operation of a train (ATO) is another implementation of fuzzy logic controller, made in Japan by Miyamoto and Yasunoby of the systems Development Laboratory of Hitachi Ltd. [18]. The Fuzzy ATO is based on so-called predictive fuzzy logic controller, in which the weight given to control rules is determined by predicting the results of the execution of the actions given in the rules. Field tests and simulation have shown that the Fuzzy ATO can run a train as skillfully as a experienced human operator.

Fuzzy expert systems

The design of expert systems may be one of the most important applications of fuzzy logic and approximate reasoning. Most of the expert systems existing so far contain an inference engine on the basis of dual logic. The uncertainty is taken care of by Bayesian probability theory. The conclusions are normally associated with a certainty or uncertainty factor expressing stochastic uncertainty, confidence, likelihood, evidence or belief. Only recently the designers of expert systems have become aware of the fact that all types of uncertainty can not be treated in the same way. The most relevant approaches in fuzzy set theory are fuzzy logic and approximate reasoning for the inference engine, the presentation of condition, indicators or symtoms by fuzzy sets, especially linguistic variables to arrive at judgements about secondary phenomena, the use of fuzzy clustering for diagnosis and combination of fuzzy set theory with other approaches; for example Dempster's theory of evidence, to obtain justifiable and interpretable measures of uncertainty. A number of fuzzy expert systems especially in the context of medical decision support systems have been implemented. To name a few, the expert systems CADIAC (internal medicine), EXPERT (rheumatology, ophthalmology), SPERIL (earthquake engineering), SPHINX (medical diagnosis), SYNTEX (management of hospitals), BIMBO (for complex Pattern Recognition problems) have included fuzzy logic and/or approximate reasoning approach in different ways.

A closely related application is the use of fuzzy logic in simulation and models that are intended to aid decision making. In the realm of business decisions, for example, Decision Products Inc of Mountain View, California, has marketed a decision supported system, REVEAL [8], which supports fuzzy logic. Decision makers, in altering their proposed policies, can use the model to assure a policy's effect an profit. Chinese meteorologists in Shanghai used a fuzzy model to determine the best areas for growing rubber trees [18]. The reliability of the method has been tested, based on weather condition during three consecutive harvest around Shanghai, and achieved good agreement with the actual result.

Fuzzy languages

A direct application of approximate reasoning is in the development of various fuzzy algorithms. Fuzzy languages are formulated on the basis of fuzzy logic and approximate reasoning. Several fuzzy languages have been developed by now, such as LPL [19,20], FLIP [21], Fuzzy Planner [22] and Fprolog [23] and others. These are based on LPI, FORTRAN, LISP, PROLOG and other programming languages and they differ in content as well as to their aims. Zadeh [24] developed a fuzzy language PRUF (acronym for Possibilistic Relational Universal Fuzzy) and this is a meaning representation of natural language basing on possibility theory. PRUF may be employed as a language for the presentation of imprecise knowledge and as a means of precisiation of fuzzy propositions expressed in a natural language. In essense, PRUF bears the same relationship to fuzzy logic that predicate calculus does to two valued logic. Thus it serves to translate a set of premises expressed in natural language into expression in PRUF to which the rules of inference of fuzzy logic or approximate reasoning may be applied.

Understanding common sence

This area is still in the early stages of research. Since fuzzy logic and approximate reasoning are based on the idea of linguistic variable, it appears likely that fuzzy logic and approximate reasoning can help with the great difficulties of natural language communication with computers. With fuzzy sets, the implicit quantifiers that always comes up in the situations, such as "Birds can fly" or "When people are in tension, they smoke cigarates", can be easily translated into machine usable form. Actually, fuzzy logic and approximate reasoning enhance the capability of programs of natural language understanding to make approximate logical deductions from incomplete or imprecise knowledge.

Fuzzy hardware

Although, most of the uses of fuzzy logic are on the software area, researchers have been trying to use fuzzy logic in hardware also. The first fuzzy chip was developed by Togai and Watanade [25] at Bell Telephones Laboratories in 1985, and it is now available for commercial use. It is expected that it should find many uses in both fuzzy logic based intelligent controllers and expert systems. Recently, the fuzzy computer developed by Yamakawa [8] at Kumamoto University (the hardware was built by OMRON Tateise Electronics Corporation) has shown great promise as a general purpose tool for performing fuzzy inference at high speed and with remarkable robustness. These may lead to an expanded use of fuzzy logic not only in industrial applications but also in knowledge based systems in which the deduction of answer to a query requires the inference mechanism of fuzzy logic and approximate reasoning. Yamakawa's fuzzy computer may be an important step towards the development of a sixth-generation computer capable of processing common sense knowledge.

While many fuzzy applications are still in an early stage of development, it seems probable that in the next decade fuzzy logic will become routinely applied in many areas of artificial intelligence where communication with people or limitation of their thought process is involved. This may help to bridge the gap between the analogic and flexible thinking of humans and the rigid frame work of present computers.

LINGUISTIC RECOGNITION SYSTEM AND MANAGEMENT OF UNCERTAINTY

Let us now explain, in brief, a recent attempt [26] demonstrating the application of approximate reasoning in designing a general purpose linguistic recognition system. The effectiveness of the algorithm has been demonstrated on speech recognition problem.

The key feature of the recognition system is that it is capable of handling with various imprecise input patterns and of providing decision in natural form along with its degree of certainty. The input feature is considered to be of either linguistic form (*eg*, *F* is small, say) or quantitative form (*eg*, *F* is 500, say) or mixed form (*eg*, *F* is about 500, say) or set form (*eg*, *F* is between 400 and 500, say).

An input pattern in any aforesaid form has been viewed here as consisting of various combinations of the three primary properties SMALL, MEDIUM and HIGH possessed by its different features to some degree. Unlike the existing methods, the compatibility functions of the sets SMALL, MEDIUM and HIGH have all been represented by π functions. The various uncertainties (ambiguities) in the input statement have been managed by providing/modifying membership values heuristically to a great extent.

Representation of the imprecise input X through their primary properties SMALL, MEDIUM and HIGH basically implies that the entire dynamic range of each feature has been divided into three overlapping sub-regions corresponding to these primary properties. So the whole feature space can be divided into 3^N overlapping subspaces for N features, each corresponds to a property combination of the primary properties SMALL, MEDIUM and HIGH. This is explained in Fig 9 for $N = 2$ (*ie*, for two features).

The major operations of such a recognition system is shown in Fig 10. It consists of three blocks/operations, namely Linguistic Feature Extractor (LFE), Fuzzy Classifier and Decision Maker. The LFE gives a characteristic vector $CV(X)$ for an input X. The Fuzzy Classifier uses this $CV(X)$ and a relational matrix to determine the class similarity vector $S(X)$ which denotes the degree of similarity of the input pattern X to the various pattern classes. The Decision Making block gives a natural output regarding the classes from which the pattern X may come along with its degree of certainty based on the similarity vector $S(X)$. It is to mention here that since all the features are not equally important in characterizing a class, a weight matrix has also been incorporated [26].

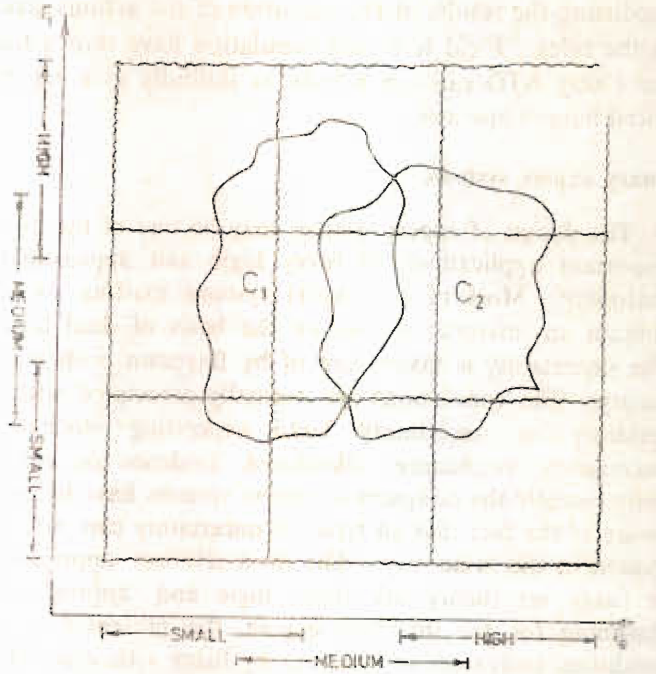

Fig 9 Feature space showing nine overlapping regions in terms of the properties SMALL, MEDIUM and HIGH. Carl lines denote fuzzy boundaries

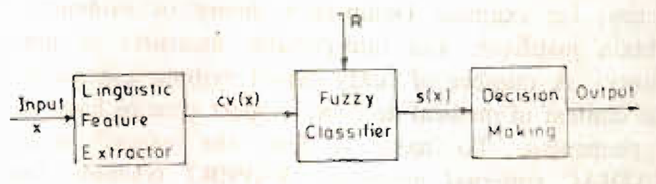

Fig 10 Block diagram of the linguistic recognition system

Linguistic feature extractor (LFE)

First of all, each feature information is considered separately to determine its membership value corresponding to the properties SMALL, MEDIUM and HIGH. The way it has been done for various forms of input is furnished in the next section.

Linguistic form

Here the input pattern information is provided in linguistic form *eg*, the information as "*F* is small" or "*F* is more or less medium" or "*F* is very high" etc. When the input statement contains only the primary terms, its membership values for the sets SMALL, MEDIUM and HIGH, are assigned as

$$\left.\begin{array}{ll} \text{small} & \equiv \{0.8/S,\ 0.2/M,\ 0.0/H\} \\ \text{medium} & \equiv \{0.2/S,\ 0.8/M,\ 0.2/H\} \\ \text{high} & \equiv \{0.0/S,\ 0.2/M,\ 0.8/H\} \end{array}\right\} \quad (26)$$

There may be statements with linguistic hedges *viz*, very, more or less, slightly etc. When the statement contains "very small", its membership value for the property SMALL (as the membership function functions for the primary terms are the π function) will be decreased and the membership value for the property MEDIUM will further be decreased.

The modifications of the membership values may be carried out in a similar manner for other possible linguistic hedges.

Quantitative form

The information in this form are in exact numerical terms, like "F is 500", say. In this case, find the membership value for different linguistic feature properties (*ie*, SMALL, MEDIUM, HIGH) by their corresponding membership functions which are assigned before hand according to the nature of various features.

Mixed form

The information are provided in the form as the mixture of linguistic hedges and quantitative terms such as, "F is about 400" or "F is more or less 400". Since the linguistic term increases the impreciseness in the information, the membership values of the statement as a whole, for different primary properties should be lower than that of the quantitative term alone. The amount of decrease will be determined according to the linguistic hedges.

Set form

Like the mixed form the information here is also a mixture of linguistic hedges and quantitative terms. The only difference lies with the nature of linguistic hedges. The linguistic hedges which are used here in the set form are "less than", "more than", "between" etc, such that the data reflected is a set and atleast one boundary of the data set is known. The examples of this form are "F is less than 400, say" or "F is more than 400, say" or "F is between 400 and 500, say".

First the membership value for the various primary properties with respect to the quantitative terms (*eg*, 400) are calculated. We know that the compatibility functions considered for the primary sets are all standard π functions of the form $\pi(x, \beta, \Gamma)$ where β is the ideal point *ie*, the point where the membership value is 1.0. Modify the membership value $\mu(400)$ to obtain that of "less than 400" as

$$\mu_s\,(\text{less than } h) = \begin{cases} \{\mu_s(h)\} & \text{if } h \geqslant \beta \\ \{\mu_s(h)\}^2 & \text{if } h \leqslant \beta \end{cases} \qquad (27)$$

corresponding to a primary property s.

The modification of the membership values may be made similarly for the hedges "more than" or "greater than" and any other which are used to represent data in the set form.

Determination of characteristic vectors (CV)

After obtaining the membership values of features for the properties SMALL, MEDIUM and HIGH, class membership of a pattern, $CV(X)$, corresponding to all property combinations is then computed. Let us consider the ith property combination

$$(p_1^i, p_2^i, \ldots, p_N^i) \qquad (28)$$

where depending on the value of i, p_m^i denotes one of the primary properties SMALL, MEDIUM, HIGH and represents the set "F_m is p_m^i". So the ith element of the characterized vector, $CV(X)$ is defined as

$$CV_i(X) = \left(\prod_{m=1}^{N} \mu_{F_m p_m^i}\right)^{1/N} \qquad (29)$$

where $\mu_{F_m p_m^i}$ is the membership value of the set "F_m is p_m^i for an input pattern.

Fuzzy classifier

This block uses the characteristic vector $CV(X)$ (corresponding to an input pattern X) and a relational matrix, R, to determine the class similarity vector $S(X)$ which denotes the degree of similarity of X to the various pattern classes. The relational matrix R, is determined from training samples.

The fuzzy classifier incorporates composition rule of inference. The class similarity vector $S(X)$ $(=\{s_j(X)\})$ is determined as

$$\begin{aligned} s_j(X) &= CV(X) \bigcirc R \\ &= \underset{i=1,2,\ldots 3^N}{\text{Max}} \{cv_i(X) * R[i,j]\}^{1/2} \\ & \qquad j = 1,2,\ldots,M \end{aligned} \qquad (30)$$

where $cv_i(X)$ is the ith element of the characteristic vector $CV(X)$ for the unknown pattern X and $R[i,j]$ is the (i,j)th entry of the relational matrix R.

Confidence factor and output

A measure of confidence factor (CF) is defined as [26]

$$CF = \frac{1}{2}\left[\{s_{mod}(X)\}^{fmod} + \frac{1}{(M - f_{mod})} \sum_{i=1}^{N} \{s_{mod}(X) - s_i(X)\}\right]$$
$$0 \leqslant CF \leqslant 1 \qquad (31)$$

where f_{mod} is the frequency of the highest entry in $S(X)$; $s_i(X)$ is ith entry of the $S(X)$; $s_{mod}\,(X)$ is the highest entry in $S(X)$ and M is the number of pattern classes. If the CF value lies between 0.2 and 0.8, we find the confidence factor (CF2) for the second highest entry in the similarity vector $S(X)$ by the same formula as in (31). Depending on the value of CF's, the final output of the recognition system is given in natural form.

Implementation and discussion

The above mentioned algorithm was implemented on a set of Indian Telugu vowel sounds in a consonant-vowel-consonant context uttered by three speakers in the age group 30 to 35 years. Six vowel classes (δ, a:, i, u, e, o) having ill-defined boundaries [27-30] were considered. Their first two formants frequencies F_1 and F_2 (which were extracted through spectrum analysis of Speech data) were considered here to be the features. The testing data set consists of 871 deterministic and a few imprecise data.

The overall recognition score for various sizes of samples is shown in Fig 11 by divided-bar diagram. The recognition scores are grouped in four categories, namely first correct choice, combined correct choice, second correct choice and fully wrong choice. Here the first correct choice set includes those samples for which classifier's first choice agrees with their actual class. Combined correct choice includes those samples where one of the combined choices is correct. Second correct choice includes those samples for which their second choice corresponds to the actual vowel class. Vowels not falling under the above mentioned categories are termed as misclassified or fully wrong choice.

The overall recognition score (Fig 11) corresponding to the first choice is seen to be quite satisfactory considering the fact that it accepts approximate feature information and even that information relates only to F_1 and F_2. Furthermore, since it provides natural output in

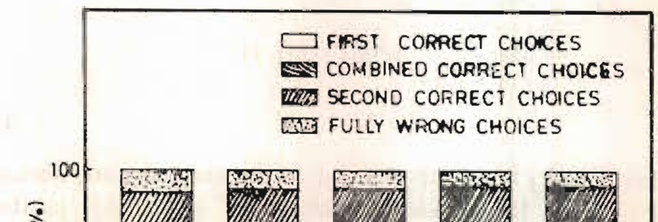

Fig 11 Divided-bar diagram showing the overall recognition score for different sizes of training samples

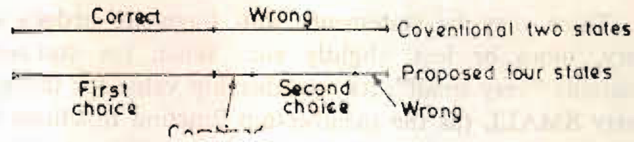

Fig 12 Four state vs conventional two state output

four states, it has very low ($\approx 5\%$) misclassification rate as compared to those ($\approx 20\%$) in [27,28,30] which give two state hard decision like "correct" or "wrong". This is explained in Fig 12 where the category "wrong", in two state conventional system, has been decomposed into three categories in the proposed system. Because of the flexibility, the proposed system has therefore a provision of improving its efficiency significantly by incorporating combined and second choices under the control of a supervisory scheme.

ACKNOWLEDGEMENT

The authors gratefully acknowledge Prof D Datta Majumder for his interest in this work and Mr S Chakraborty for drawing the diagrams

REFERENCES

1. L A Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans Systems Man, Cyberns*, vol SMC-3, pp 28-44, 1973.

2. L A Zadeh, The role of fuzzy logic in the management of uncertainty in expert systems, *Fuzzy Sets Systems*, vol 11, pp 199-223, 1983.

3. R R Yager, Reasoning with conjunctive knowledge, *Fuzzy Sets Systems*, vol 28, pp 69-83, 1988.

4. L A Zadeh, Fuzzy algorithm, *Information Control*, vol 12, pp 94-102, 1968.

5. L A Zadeh, A rationale for fuzzy control, *J Dynamic Systems Measurement Control*, vol 94, Series G, pp 3-4, 1972.

6. E H Mamdani & S Assilian, A case study on the application of fuzzy set theory to automatic control, *Proc IFAC Stochastic Control Symp*, Budapest, 1974.

7. L P Holmblad & J J Ostergaard, Control of a cement Klin by fuzzy logic, in *Fuzzy Information and Decision Process* (Eds M M Gupta & E Sanchez), North-Holland, Amsterdam, pp 387-397, 1982.

8. *Preprints of the Second Congress of the International Fuzzy Systems Association*, Tokyo, Japan, 1987.

9. L A Zadeh, The concept of a linguistic variable and its applications to approximate reasoning, *Information Science*, vol 8, pp 199-249, vol 8, pp 4301-357, vol 9, pp 43-80, 1975.

10. M Mizumoto & H J Zimmermann, Comparison of fuzzy reasoning methods, *Fuzzy Sets Systems*, vol 8, pp 253-283, 1982.

11. L A Zadeh, Fuzzy Logic, *IEEE Trans Computer*, vol C-37, pp 83-92, April, 1988.

12. J F Baldwin, A new approach to approximate reasoning using a fuzzy logic, *Fuzzy Sets Systems*, vol 2, pp 309-325, 1979.

13. L A Zadeh, Fuzzy logic and approximate reasoning, *Systhese*, vol 30, pp 407-428, 1973.

14. J F Baldwin & N C F Guild, Feasible algorithm for approximate reasoning using fuzzy logic, *Fuzzy Sets Systems*, vol 3, pp 225-251, 1980.

15. L A Zadeh, Calculus of fuzzy restriction, in *Fuzzy Sets and Their Application to Cognitive and Decision Processes* (Ed L A Zadeh), Academic Press, pp 1-39, 1975.

16. E H Mamdani, Application of fuzzy logic to approximate reasoning using linguistic systems, *IEEE Trans Computer*, vol C-26, pp 1182-1191, 1977.

17. L A Zaheh, Making Computer think like a people, *IEEE Spectrum*, pp 26-32, August 1984.

18. J M Adamo, L P L—A Fuzzy programming language-1. Syntactic aspects, *Fuzzy Sets Systems*, vol 3, pp 151-180, 1980.

19. J M Adamo, L P L—A Fuzzy programming language-1. Semantic aspects, *Fuzzy Sets Systems*, vol 3, pp 261-290, 1980.

20. R Giles, A computer program for fuzzy reasoning, *Fuzzy Sets Systems*, vol 4, pp 221-234, 1980.

21. R Kling, Fuzzy-planner—reasoning with inexact concepts in a procedural problem-solving language, *J Cyberns*, vol 4, pp 105-122, 1974.

22. T P Martin, J F Baldwin, & B W Pilsworth, The implementation of prolog—A fuzzy prolog interpreter, *Fuzzy Sets Systems*, vol 23, pp 119-129, 1987.

23. L A Zadeh, Test-score semantics as the basic for a computational approach to the representation of meaning, *Literary Linguistic Computing*, vol 1, pp 24-35, 1986.

24. M Toga & H Watanade, Expert systems on a chip: An engine for real-time approximate reasoning, *IEEE Expert*, vol 1, pp 55-62, 1986.

25. S K Pal & D P Mandal, Linguistic recognition system based on approximate reasoning, *Inform Sci*, vol 61, pp 135-161, 1992.

26. S K Pal, Optimum guard zone for self-supervised learning, *IEE Proc Part E*, vol 127, pp 7-14, 1982.

27. S K Pal & D Dutta Majumder, Fuzzy sets and decision making approach in vowel and speaker recongnition, *IEEE Trans System, Man Cyberns*, vol SMC-7, pp 625-629, 1977.

28. S K Pal & D Dutta Majumder, *Fuzzy Mathematical Approach to Pattern Recognition*, Wiley (Halsted Press), New York, 1986.

29. S K Pal, A Pathak, & C Basu, Dynamic guard zone for self-supervised learning, *Pattern Recogination Lett*, vol 7, pp 135-144, 1988.

30. J F Baldwin, Fuzzy logic and its application to Fuzzy reasoning, in *Advances in Fuzzy Set Theory and Applications* (Eds M M Gupta, R K Ragade & R R Yager), North-Holland, Amsterdam, 1979, pp 93-115.