FUZZY LOGIC CONTROLLER

Project Report submitted

To

MAHATMA GANDHI UNIVERSITY

In partial fulfilment of the requirement

For the Award of the degree of

MASTER OF SCIENCE IN MATHEMATICS

By

ROSHNI SURESH ARAKAL

Reg No. 180011015188

DEPARTMENT OF MATHEMATICS

ST PAUL'S COLLEGE, KALAMASSERY

2018 - 2020

CERTIFICATE

This is to certify that the project entitled **"FUZZY LOGIC CONTROLLER"** is a bonafide record of studies undertaken by **ROSHNI SURESH ARAKAL** (Reg no. 180011015188), in partial fulfillment of the requirements for the award of M.Sc. Degree in Mathematics at Department of Mathematics, St. Paul's College, Kalamassery, during 2018 – 20.

Ms. NISHA VM
Department of Mathematics

Dr. SAVITHA K S
Head of Department
Department of Mathematics

Examiner:

DECLARATION

I, **ROSHNI SURESH ARAKAL** declare that, this project titled **“FUZZY LOGIC CONTROLLER”** has been prepared by me under the under the supervision of **Ms. Nisha V M**, Department of Mathematics, St. Pauls’s College , Kalamassery.

I also declare that this project has not been submitted by me fully or partially for the awards of any degree, diploma, title or recognition earlier.

Date:

Place: **Kalamassery** **ROSHNI SURESH ARAKAL**

ACKNOWLEDGEMENT

I am highly indebted to my guide **Ms. NISHA V M**, Department of Mathematics, St. Paul's College, Kalamassery for providing me necessary stimulus for preparation of this project.

I would like to acknowledge my deep sense of gratitude to **Dr. SAVITHA K.S**, Head of the Department of Mathematics, St. Paul's College Kalamassery for the support and inspiration rendered to me in this project.

I express my sincere gratitude to **Prof. VALENTINE D'CRUZ**, Principal, St. Paul's College Kalamassery for the wholehearted support and help.

I also wish to express my sincere thanks to all the faculty members of Department of Mathematics for the help and encouragement to bring this project a successful one.

ROSHNI SURESH ARAKAL

FUZZY LOGIC CONTROLLER

CONTENTS

CHAPTER 1

1.1 INTRODUCTION

The term fuzzy means things which are not clear or vague. The concept of a fuzzy set was published in 1965 by Lotfi A. Zadeh. Since that seminal publication, the fuzzy set theory is widely studied and extended. Its application to the control theory became successful and revolutionary especially in the seventies and eighties; the applications to data analysis, artificial intelligence, and computational intelligence are intensively developed, especially since the nineties.

The classical set theory is built on the fundamental concept of a set of which an individual is either a member or not a member; it is not allowed that an element is in a set and not in the set at the same time. In real life, we may come across a situation where we can't decide whether the statement is true or false. Thus, many real-world application problems cannot be described and handled by the classical set theory, including all those involving elements with only partial membership of a set. On the contrary, fuzzy set theory accepts partial memberships and therefore in a sense generalizes the classical set theory to some extent. We can also consider the uncertainties of any situation. New computing methods based on fuzzy logic can be used in the development of intelligent systems for decision making, identification, pattern recognition, optimization, and control.

Fuzzy logic refers to the study of methods and principles of human reasoning. Classical logic, as common practice, deals with propositions that are either true or false. Because one, and only one truth value (either true or false) is assumed by a logical function of a (finite) number of logical variables (hypothetical propositions), the classical logic is also called a two-valued logic. It is now well understood and well accepted that many propositions are both partially true and partially false. To describe such partial truth values by some new rules, in a way to extend and generalize the two-valued logic, multivalued logics were proposed and developed. As the first attempt, several three-valued logics have now been well established, with their own rationale. It is common in these logics to introduce a “neither” in-between “true” and “false.” It has turned out that three-valued logics are successful both logically and mathematically. Motivated by the useful three-valued logics, n-valued logics were developed in the 1930s. In particular, the n-valued logic of Lukasiewicz even allows $n = \infty$. It has lately been understood that there exists an isomorphism between the two-valued logic and the crisp set theory, and, similarly, there is an isomorphism between the Lukasiewicz logic and the fuzzy set theory.

Fuzzy logic has rapidly become one of the most successful of today technologies for developing sophisticated control systems. The reason for which is very simple. Fuzzy logic addresses such applications perfectly as it resembles human decision making with an ability to generate precise solutions from certain or approximate information. It fills an important gap in engineering design methods left vacant by purely mathematical approaches (e.g. linear control design), and purely logic-based approaches (e.g. expert systems) in system design. While other approaches require accurate equations to model real-world behaviors, fuzzy design can accommodate the ambiguities of real-world human language and logic. It provides both an intuitive method for describing systems in human terms and automates the conversion of those system specifications into effective models.

Although fuzzy logic has applications in a number of different areas, it is not yet known to people unfamiliar with intelligent systems how it can be applied in different products that are currently available in the market. For many people, the engineering and scientific meaning of the word fuzzy is still fuzzy. It is important that these people understand where and how fuzzy logic can be used.

1.2 PRELIMINARIES

Fuzzy set

Let X be a space of points or objects with a generic element of X denoted by x. A **fuzzy set** A in X is characterized by a membership function A(x) which associates with each point in X a real number in the interval [0,1], with the value of A(x) at x representing the grade of membership of x in A. When A is a set in ordinary sense (crisp set) its membership function can take only two values 0 and 1 with A (x) =1or 0 according as x does or does not belong to A. In this case A(x) reduces to the characteristic function of the set A.

A fuzzy set is **empty** if and only if its membership function is identically zero on X.

Two fuzzy sets A and B are **equal** if and only if A(x) =B(x) for all x in X.

The **complement** of a fuzzy set A with respect to a universal set X is denoted by A' and is defined by

$$A'(x) = 1 - A(x).$$

Fuzzy subset

Given two fuzzy sets A and B, A is a subset of B if and only if $A(x) \leq B(x)$ for all $x \in X$.

Union

The union of two fuzzy sets A and B with respective membership functions A(x) and B(x) is a fuzzy set C written as $C = A \cup B$, whose membership function is related to those of A and B by

$C(x) = \max[A(x), B(x)], x \in X$

Or in abbreviated form $C = A \vee B$.

Intersection

The intersection of two fuzzy sets A and B with respective membership functions A(x) and B(x) is a fuzzy set C written as $C = A \cap B$ whose membership function is related to those of A and B by

$C(x) = \min[A(x), B(x)], x \in X$

Or in abbreviated form $C = A \wedge B$.

Alpha cut

Given a fuzzy set $A: X \rightarrow [0, 1]$ the α-cut of A is denoted by $^{\alpha}A$ is the collection of all points of X whose membership grade is at least α.

ie; $^{\alpha}A = \{x \in X: A(x) \geq \alpha\}$

The strong α-cut of A denoted by $^{\alpha+}A$ is the collection of all points of X whose membership grade is strictly greater than α.

ie; $^{\alpha+}A = \{x \in X: A(x) > \alpha\}$

Level set

Given a fuzzy set $A: X \rightarrow [0, 1]$ the level set of A denoted by $\lambda(A)$ is the set of all membership grades of points of A.

ie; $\lambda(A) = \{\alpha \in [0,1]: A(x) = \alpha \text{ for some } x \in X\}$

Support

The support of A is the collection of all points of X whose membership grade is non-zero.

ie; support of A is ${}^{0+}A$.

Core

The core of A is the collection of all points of X whose membership grade is 1.

ie; core of A = ${}^{1}A$

Height

The height of A is the largest membership grade obtained.

ie; h (A) = sup A (x), x ∈X.

Normal

A fuzzy set is called normal if h (A)=1.

Convexity (convex fuzzy set)

A fuzzy set A is convex if and only if

$A [\lambda x_1 + (1 - \lambda) x_2] \geq \min [A(x_1), A(x_2)]$; for all x_1 and x_2 in X and all λ in [0,1].

Cardinality

Cardinality of A is the sum of membership grades of elements of X with respect to the fuzzy set A.

Fuzzy Expression

In the fuzzy expression, a fuzzy proposition can have its truth value in the interval [0,1].

The fuzzy expression function is a mapping function from [0,1] to [0,1]

$f: [0,1] \rightarrow [0,1]$

In the fuzzy logic, the operation such as negation (~ or ¬), conjunction (∧), disjunction (∨) are used as in the classical logic.

Definition (Fuzzy Logic)

The fuzzy logic is a logic represented by the fuzzy expression which satisfies the followings

1. Truth values ,0 and 1, and variable x_i (∈ [0,1], i=1, 2, ... n) are fuzzy expressions
2. If f is a fuzzy expression, ~f is also a fuzzy expression
3. If f and g are fuzzy expression, f ∧ g and f ∨ g are also fuzzy expressions

Operations in Fuzzy Expressions

There are some operators in the fuzzy expression such as ¬ (negation), ∧ (conjunction), ∨ (disjunction) and → (implication). However, the meaning of operators may be different according to literature. If we follow Lukasiewiz's definition, the operators are defined as follows for a, b ∈ [0, 1].

1. Negation $\bar{a} = 1-a$
2. Conjunction a ∧ b= min (a, b)
3. Disjunction a ∨ b= max (a, b)
4. Implication a → b = min (1,1+b-a)

TABLE: The properties of fuzzy logic operations

1	Involution	$\bar{\bar{a}} = a$
2	Commutativity	a ∧ b = b ∧ a a ∨ b = b ∨ a
3	Associativity	(a ∧ b) ∧ c = a ∧ (b ∧ c) (a ∨ b) ∨ c = a ∨ (b ∨ c)
4	Distributivity	a ∨ (b ∧ c) = (a ∨ b) ∧ (a ∨ c) a ∧ b ∨ c) = (a ∧ b) ∨ (a ∧ c)
5	Idempotency	a ∧ a =a a ∨ a =a

6	Absorption	a ∨ (a ∧ b) =a a ∧ (a ∨ b) =a
7	Absorption by 0 and 1	a ∧ 0 =0 a ∨ 1 =1
8	Identity	a ∧ 1 =a a ∨ 0 =a
9	De Morgan's law	$\overline{a \wedge b} = \bar{a} \vee \bar{b}$ $\overline{a \vee b} = \bar{a} \wedge \bar{b}$

But we have to notice that the law of contradiction and law of excluded middle are not verified in the fuzzy logic

Example

We can see that the two properties are not satisfied in the following examples

(1) Law of contradiction

Assume a in [0,1]

a ∧ ā = min [a, ā]

= min [a, 1-a]

$$= \begin{cases} a, \ if\ 0 \le a \le 0.5 \\ 1-a, \ if 0.5 \le a < 1 \end{cases}$$

Therefore $0 < a \wedge \bar{a} \le 0.5$

Then $a \wedge \bar{a} \neq 0$

(2) Law of excluded middle

Suppose a is in [0, 1]

a ∨ ā = max [a, ā]

= max [a,1-a]

$$=\begin{cases} a, if 0.5 \le a < 1 \\ 1-a, if 0 < a \le 0.5 \end{cases}$$

Therefore $0.5 \le a \vee \bar{a} < 1$

Then $a \vee \bar{a} = 1$ if a=0 or 1, $a \vee \bar{a} < 1$ otherwise.

Linguistic Variables

A fuzzy linguistic variable (V) is an attribute whose domain contains linguistic values (also called fuzzy terms). The linguistic term is used to express the concept and knowledge in human communication, whereas membership function is useful for processing numeric input data. Associating a fuzzy set to a linguistic term, it offers two important benefits :

1. The association makes it easier for human experts to express their knowledge using the linguistic terms
2. The knowledge expressed to linguistic terms is easily comprehensible.

1.3 FUZZY RULES

Among all the techniques developed using fuzzy sets, the fuzzy if-then rule (or, in short, the fuzzy rule) is by far the most visible one due to its wide range of successful applications. Fuzzy if-then rules have been applied to many disciplines such as control systems decision making, pattern recognition, and system modeling. Fuzzy if-then rules also play a critical role in industrial applications ranging from consumer products, robotics, manufacturing, process control, medical imaging, to financial trading.

1.3.1 Structure of fuzzy rules

A fuzzy rule is the basic unit for capturing knowledge in many fuzzy systems. A fuzzy rule has two components: an if-part (also referred to as the antecedent) and a then-part (also referred to as the consequent):

IF< antecedent > THEN < consequent >

The antecedent describes a condition, and the consequent describes a conclusion that can be drawn when the condition holds.

The structure of a fuzzy rule is identical to that of a conventional rule in artificial intelligence. The main difference lies in the content of the rule antecedent -the antecedent of a fuzzy rule describes an elastic condition (a condition that can be satisfied to a degree) while the antecedent of a conventional rule describes a rigid condition (a condition that is either satisfied or dissatisfied).For instance, consider the two rules below:

R1: IF the annual income of a person is greater than 120K. THEN the person is rich.

R2: IF the annual income of a person is *High*, THEN the person is Rich.

Where *High* is a fuzzy set. The rule R1 is a conventional one, because its condition is rigid. In contrast, R2 is a fuzzy rule because its condition can be satisfied to a degree for those people whose income lies in the boundary of the fuzzy set *High* representing high annual income. Like conventional rules, the antecedent of a fuzzy rule may combine multiple simple conditions into a complex one using three logic connectives: AND (conjunction), OR (disjunction), and NOT (negation).For instance, a loan approval system may contain the following fuzzy rule:

IF the annual income of a person is *High* AND

(the credit report of the person is *Fair* OR the person has a *Valuable* real estate asset)

AND the amount of the loan requested is NOT *Jumbo*

THEN recommend approving the loan.

The consequent of fuzzy rules can be classified into three categories:

1. *Crisp Consequent*: IF... THEN y = a, where a is non-fuzzy numeric value or symbolic value.
2. *Fuzzy Consequent*: IF... THEN y is A where A is a fuzzy set.
3. *Functional Consequent*: IF x_1 is A_1 AND x_2 is A_2 AND... x_n is A_n THEN

$y = a_o + \sum_{i=1}^{n} a_i x_i$ where a_o, a_1, …a_n are constants.

Each type of rule consequent has its merit. Generally speaking, fuzzy rules with a crisp consequent can be processed more efficiently. A rule with a fuzzy consequent is easier to understand and more suitable for capturing imprecise human expertise. Finally, rules with a functional consequent can be used to approximate complex nonlinear models using only a small number of rules.

The fuzzy sets in a rule's antecedent define a fuzzy region of the input space covered by the rule (i.e., the input situations that fit the rule's condition completely or partially), whereas the fuzzy sets in a rule's consequent describe the vagueness of the rule's conclusion. This difference has an important impact on the design of membership functions. Generally speaking, the membership functions of an input variable should cover the entire input space, whereas the membership functions of an output variable are not subject to such a constraint.

1.3.2 Fuzzy implication

A fuzzy rule generally assumes the form

R: if x is A, then y is B

Where A and B are linguistic values defined by fuzzy sets on the universe of discourse X and Y, respectively. The rule is also called a fuzzy implication or fuzzy conditional statement. The part "x is A" is called the "antecedents" or "premise", while "y is B" is called the "consequence" or "conclusion". It is sometimes abbreviated as R: A→B. In essence, the expression described a relation between two variables x and y .This suggests that a fuzzy rule can be defined as a binary relation R on the product space X×Y.

R can be viewed as a fuzzy set with a two dimensional membership function

$$\mu_R(x, y) = f(\mu_A(x), \mu_B(y))$$

where the function f, called the "fuzzy implication function", performs the task of transforming the membership degrees of x in A and y is B into those of (x, y) in A × B .We introduce here two well-known fuzzy implication functions.

1. Min operation rule of fuzzy implication [Mamdani]: It interprets the fuzzy implication as the minimum operation.

$$R_C = A \times B$$
$$= \int_{X \times Y} \mu_A(x) \wedge \mu_B(y) / (x, y)$$,where ∧ is the min operator

2. Product operation rule of fuzzy implication [Larsen] : It implements the implication by the product operation

$$R_P = A \times B$$
$$= \int_{X \times Y} \mu_A(x) \cdot \mu_B(y) / (x, y)$$, where · is the algebraic product operator.

Example

Consider the following,

If temperature is high, then humidity is fairly high

It is a fuzzy rule and a fuzzy relation. We want to determine the membership function of the rule. Let T and H be the universe of discourse of temperature and humidity, respectively, and let's define variables $t \in T$ and $h \in H$. We represent the fuzzy terms "high" and "fairly high" by A and B respectively:

A= "high", $A \subseteq T$

B= "fairly high", $B \subseteq H$

TABLE: membership of A in T (temperature)

t	20	30	40
$\mu_A(t)$	0.1	0.5	0.9

TABLE: membership of B in H (humidity)

h	20	50	70	90
$\mu_B(h)$	0.2	0.6	0.7	1

Then the above rule can be rewritten as:

R (t, h): If t is A, then h is B

In the rule (relation), we can find two predicate propositions:

R (t): t is A

R (h): h is B

the rule becomes

R (t, h): R (t) $\rightarrow$R (h)

If we know membership function of A and B, we can determine R (t, h) =A×B by using the fuzzy implication function where R (t, h) $\subseteq$ T×H. In order to get the relation for the implication in the above fuzzy rule, we have to select an implication function between A and B. For simplicity, let's take the min operation of Mamdani

R_C (t, h) =A×B

$$= \int \mu_A(t) \wedge \mu_B(h) / (t, h)$$

when we apply the min operation on the Cartesian product A×B , we obtain the relation R_C as shown in the table below

TABLE: membership of rule R_C =A × B

t h	20	50	70	90
20	0.1	0.1	0.1	0.1
30	0.2	0.5	0.5	0.5
40	0.2	0.6	0.7	0.9

This membership of R_C represents the fuzzy rule. Note that μ_{Rc} (20, 50) =0.1 is obtained by the min between $\mu_A(20)=0.1$ and $\mu_B(50)=0.6$. Similarly,μ_{Rc} (30, 20) =0.2 from $\mu_A(30)=0.5$ and $\mu_B(20)=0.2$.

1.3.3 Compositional rule of inference

Let's consider a single fuzzy rule and its inference.

R_1: if v is A then w is C

Input: v is A′

Result: C′

A⊂U, C⊂W, v ∈ U and w ∈C.

The fuzzy rule is interpreted as an implication (A→C) and which is defined in the product space U×W.

R_1:A→C or R_1=A×C

R_1 ⊂ U×W

When the input A′ is given to the inference system, the output C′ is obtained through the inference operation denoted by the composition operator "o".

C = A′ o R_1.

This inference procedure is called the "compositional rule of inference". Therefore in real systems, the inference is determined by two factors: the "implication operator" and "composition operator". For the implication in the Cartesian product space R=A×C , the two operators are often used :

- Mamdani implication(R_C) : min operator
- Larsen implication (R_P) : algebraic product operator

For the composition, we have introduced also two operations

- ❖ Mamdani composition : max – min
- ❖ Larsen composition : max –product

The following lemmas will show the inference mechanism when a simple fuzzy rule is given.

Lemma 1: (For 1 singleton input, result C′ is obtained from C and matching degree α_1)

When a fuzzy rule R_1 and singleton input u_0 are given

R_1: If u is A then w is C,

Or R_1:A→C

The inference result C′ is defined by the membership function $\mu_{c'}(w)$

$\mu_{c'}(w) = \alpha_1 \wedge \mu_c(w)$ for R_C (Mamdani implication); where $\mu_A(u_0) = \alpha_1$

$\mu_{c'}(w) = \alpha_1 \cdot \mu_c(w)$ for R_P (Larsen implication); where $\mu_A(u_0) = \alpha_1$

Lemma 2: (For 1 fuzzy input, result C′ is obtained from C and matching degree α_1)

When a fuzzy rule R_1:A→C and input A′ are given, the inference result C′ is defined by the membership function $\mu_{c'}$

$\mu_{c'}(w) = \alpha_1 \wedge \mu_c(w)$ for R_C

$\mu_{c'}(w) = \alpha_1 \cdot \mu_c(w)$ for R_P; where $\alpha_1 = \max_u[\mu_{A'}(u) \wedge \mu_A(u)]$

Lemma 3: (Total result C′ is an aggregation of individual results C_i')

The result of inference C is an aggregation of result C_i' derived from individual rules.

$$C' = A' \circ \bigcup_{i=1}^{n} R_i = \bigcup_{i=1}^{n} A' \circ R_i = \bigcup_{i=1}^{n} C_i'$$

Corollary of Lemma 3 :(Lemma 3 in the case of multiple inputs)

R: $\bigcup_{i=1}^{n} R_i$, R_i : A_i and $B_i \rightarrow C_i$

The result of inference C is an aggregation of result C_i' derived from individual rules.

$$C' = (A', B') \circ \bigcup_{i=1}^{n} R_i = \bigcup_{i=1}^{n} (A', B') \circ R_i = \bigcup_{i=1}^{n} C_i'$$

Lemma 4: For singleton input, C_i' is determined by the minimum matching degree of A_i and B_i)

$\mu_{C_i'}(w) = \alpha_i \wedge \mu_{C_i}(w)$ for R_C

$\mu_{C_i'}(w) = \alpha_i \cdot \mu_{C_i}(w)$ for R_P

Where $\alpha_i = \mu_{A_i}(u_0) \wedge \mu_{B_i}(v_0) = \min[\mu_{A_i}(u_0), \mu_{B_i}(v_0)]$

Lemma 5: (For fuzzy input, C_i' is determined by the minimum matching degree of (A' and A_i) and (B' and B_i))

$\mu_{C_i'}(w) = \alpha_i \wedge \mu_{C_i}(w)$ for R_C

$\mu_{C_i'}(w) = \alpha_i \cdot \mu_{C_i}(w)$ for R_P

Where $\alpha_i = \min [\ \max_u (\mu_{A'}(u) \wedge \mu_{A_i}(u)),\ \max_v (\mu_{B'}(v) \wedge \mu_{B_i}(v))]$.

CHAPTER 2

2.1 FUZZY LOGIC CONTROLLER (FLC)

A Fuzzy Controller is a controller that contains an often non-linear mapping that has been defined using fuzzy logic-based rules.

Fuzzy control systems are used to control consumer products, such as washing machines, video cameras, and rice cookers, as well as industrial processes, such as cement kilns, underground trains, and robots. Fuzzy control is a control method based on fuzzy logic. Just as fuzzy logic can be described simply as "computing with words rather than numbers"; fuzzy control can be described simply as "control with sentences rather than equations". A fuzzy controller can include empirical rules, and that is especially useful in operator controlled plants.

Increasing demands for flexibility and fast reactions in modern process operation and production methods result in nonlinear system behavior of partly unknown systems, and this necessitates application of alternative control methods to meet the demands. Fuzzy logic control can play an important role because knowledge based design rules can easily be implemented in systems with unknown structure, and it is going to be a conventional control method since the control design strategy is simple and practical and is based on linguistic information. Computational complexity is not a limitation anymore because the computing power of computers has been significantly improved even for high speed industrial applications.

Applying fuzzy logic to a control problem permits the designer to make precise decisions about imprecise data. As variables are not forcefully fitted into crisp sets, the consequences of wrong classification are minimized, so it alleviates problems due to noise, disturbance, and overall uncertainty.

The FLC is considered as the good methodology because it yields result superior to those obtained by conventional control algorithms .In particular FLC is useful in two cases,

1. The control process is too complex to analyze by conventional quantitative techniques.
2. The available sources of information are interpreted quantitatively, inexactly or uncertainly.

The basic configuration of FLC consists of four main components: Fuzzification interface, Knowledge base, decision- making logic (inference) and defuzzification interface.

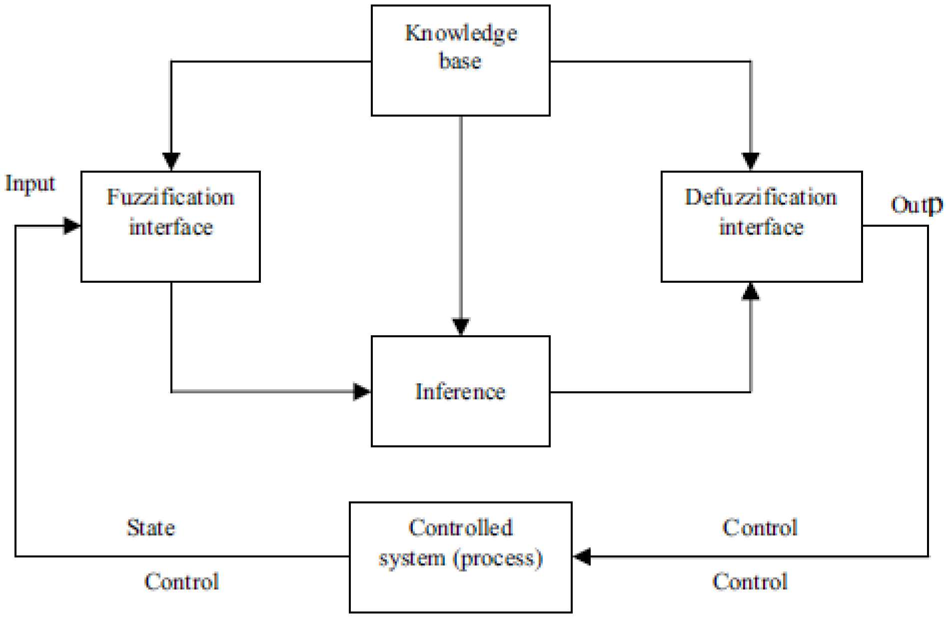


Fig.2.1. Configuration of FLC

(1) The fuzzification interface transforms input crisp values into fuzzy values and it involves the following functions

- Receives the input values
- Transforms the range of values of input variables into corresponding universe of discourse.
- Converts input data into suitable linguistic values (fuzzy sets).

This component is necessary when input data are fuzzy sets in the fuzzy inference.

(2) The knowledge base contains knowledge of the application domain and the control goals. It consists of a database and a linguistic rule base.

- The data base contains necessary definitions which are used in control rules and data manipulation.
- The linguistic rule base defines the control strategy and goal by means of linguistic control rules.

(3) The decision- making logic performs the following functions

- Simulates the human decision making procedure based on fuzzy concepts.
- Infers fuzzy control actions employing fuzzy implication and linguistic rules.

(4) The defuzzification interface performs the following functions

- A scale mapping which converts the range of output values into corresponding universe of discourse.
- Defuzzification which yields a non-fuzzy control action from an inferred fuzzy control action.

Advantages of fuzzy logic controller

- Cheaper − Developing a FLC is comparatively cheaper than developing model based or other controller in terms of performance.
- Customizable − FLCs are customizable.
- Emulate human deductive thinking − Basically FLC is designed to emulate human deductive thinking, the process people use to infer conclusion from what they know.
- Reliability − FLC is more reliable than conventional control system.
- Efficiency − Fuzzy logic provides more efficiency when applied in control system.

Disadvantages of fuzzy logic controller

- Requires lots of data − FLC needs lots of data to be applied.
- Needs high human expertise − This is one drawback as the accuracy of the system depends on the knowledge and expertise of human beings.
- Needs regular updating of rules − The rules must be updated with time.

2.2. FUZZIFICATION INTERFACE COMPONENT

In the fuzzification component, there are three main issues to be considered: scale mapping of input data, strategy for noise and selection of fuzzification function.

1. Scale mapping of input data: We have to decide a strategy to convert the range of values of input variables into corresponding universe of discourse. When an input value is come through a measuring system, the values must be located in the range of input variables. For example, if the range of input variables was normalized between -1 and +1, a procedure is need which maps the observed input value into the normalized range.

2. Strategy for noise: When observed data are measured, we may often think that the data were disturbed by random noise. In this case, a fuzzification operator should convert the probabilistic data into fuzzy numbers. In this way computational efficiency is enhanced since fuzzy numbers are much easier to manipulate than random variables. Otherwise, we assume that the observed data do not contain vagueness and then we consider the observed data as a fuzzy singleton. A fuzzy singleton is a precise value and hence no fuzziness is introduced by fuzzification in this case. In control applications, the observed data are usually crisp and used as fuzzy singleton input in the fuzzy reasoning.

3. Selection of fuzzification function: A fuzzification operator has the effect of transforming crisp data into fuzzy set.

$x = \text{fuzzifier}(x_0)$

Where x_0 is an observed crisp value and x is a fuzzy set, and fuzzifier represents a fuzzification operator.

2.3 KNOWLEDGE BASE COMPONENTS

The knowledge base of an FLC consists of two parts, a database and a fuzzy control rule base.

DATABASE

In the database part, there are four principal design parameters for an FLC: discretization and normalization of the universe of discourse, fuzzy partition of input and output spaces, and membership function of primary fuzzy set.

- Discretization and normalization of the universe of discourse: The modeling of uncertain information with fuzzy set raises the problem of quantifying such information for digital computers. A universe of discourse in an FLC is either discrete or continuous .If the universe is continuous; a discrete universe may be formed by a discretization procedure. A data set may be also normalized into a certain range of data.
- Fuzzy partition of input and output spaces: A linguistic variable in the antecedent of a rule forms a fuzzy input space, while that in the consequent of the rule forms a fuzzy output space. In general, a linguistic variable is associated with a term set .A fuzzy partition of the space determines how many terms should exist in a term set. This is the same problem to find the number of primary fuzzy sets (linguistic terms).

There are seven linguistic terms often used in the fuzzy inference:

NB: negative big

NM: negative medium

NS: negative small

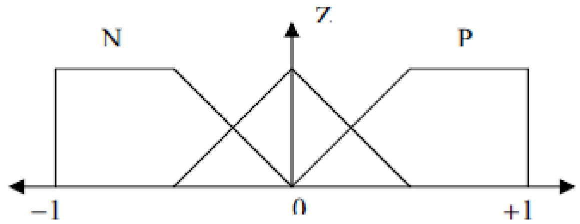
ZE: zero

PS: positive small

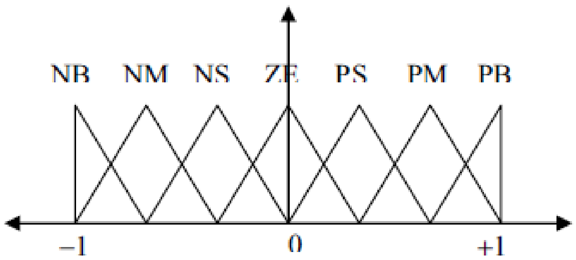
PM: positive medium

PB: positive big

A typical example is given in Fig.3.2 representing two fuzzy partitions in the same normalized universe [-1, +1].



(a) N: negative, Z: zero, P: positive



(b) NB, NM, NS, ZE, PS, PM, PB

Fig.2.2. Example of fuzzy partition with linguistic terms

The number of fuzzy terms in an input space determines the maximum number of fuzzy control rules. Suppose a fuzzy control system with two input and one output variables. If the input variables have 5 and 4 terms, the maximum number of control rules that we can construct is 20 (5 ×4) as shown in (Fig 2.3). (Fig 2.4) shows an example of system having 3 fuzzy rules.

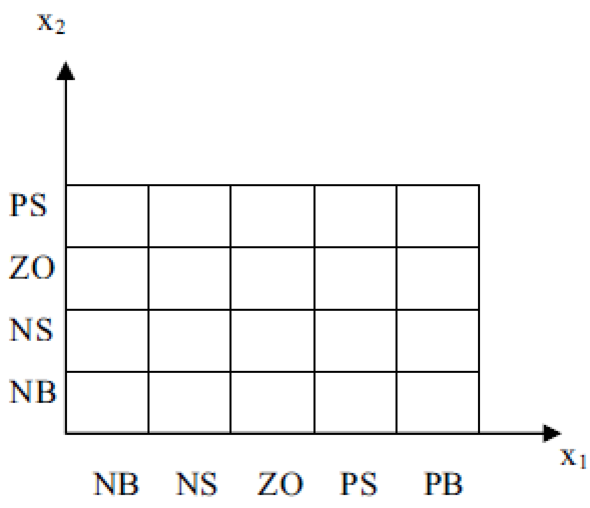


Fig.2.3. A fuzzy partition in 2-dimensional input space

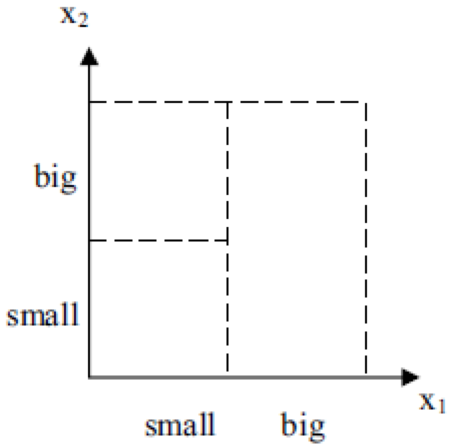


Fig2.4.A fuzzy partition having three rules

- Membership function of primary fuzzy set: There are various types of membership functions such as triangle, trapezoid and bell shapes.

RULE BASE

A fuzzy system is characterized by a set of linguistic statements usually represented in the form of if-then rules. The rule base consists of the if-then rules.

1. Source of fuzzy control rules: There are two principal approaches to the derivation of fuzzy control rules. The first method in which rules are formed by analyzing the behavior of a controlled process. The second approach is basically a deterministic method which can systematically determine the linguistic structure of rules.
   We can use four modes of derivation of fuzzy control rules. These four modes are not mutually exclusive, and it is necessary to combine them to obtain an effective system.

   - Expert experience and control engineering knowledge: operating manual and questionnaire.
   - Based on operators' control actions: observation of human controller's actions in terms of input-output operating data.
   - Based on the fuzzy model of a process: linguistic description of the dynamic characteristics of a process.

2. Types of fuzzy control rules: There are two types of control rules; state evaluation control rules and object evaluation fuzzy control rules.

- State evaluation fuzzy control rule:_State variables are in the antecedent part of rules and control variables are in the consequent part. In case of multiple input single outputs, they are characterized as a collection of rules of the form

  R_i: if x is A_i and y is B_i then z is C_i, i=1, 2… n.

  where *x*,… *y* and *z* are linguistic variables representing the process state variable and the control variable. A_i… B_i and C_i are linguistic values of the variables *x*,… *y* and *z* in the universe of discourse U, …V and W, respectively i = 1, 2, … , n. That is,

$$x \in U, A_i \subset U$$

$$y \in V, B_i \subset V$$

$$z \in W, C_i \subset W$$

  The state evaluation rules evaluate the process state (eg: state, state error, change of error) at time t and compute a fuzzy control action at time t.

- Object evaluation fuzzy control rules: It is also called predictive fuzzy control .They predict present and future control actions and evaluate control objectives .A typical rule is described as

  R_n: if (z is C_n→(x is A_n and y is B_n)) then z is C_n;

  n = 1, 2….n. *x* and *y* are performance indices for the evaluation and *z* is control command. In linguistic terms, the rule is interpreted as: if the performance index x is A_i and index y is B_i when a control command z_i is C_i, then this rule is selected, and the control command C_i is taken to be the output of the controller.

CHAPTER 3

3.1 INFERENCE (DECISION MAKING LOGIC)

In general, in the decision making logic part, we use three inference methods

1) Mamdani Method

2) Tsukamoto Method

3) TSK Method.

1) MAMDANI METHOD

This method uses the minimum operation R_C as a fuzzy implication and the max-min operator for the composition. Let's suppose the rule base is given in the following form.

R_i: if u is A_i and v is B_i, then w is C_i , i=1,2,3,…n

for u∈ U , v∈ V ,and w ∈W.

then R_i= (A_i and B_i) →C_i is defined by

$$\mu_{R_i} = \mu_{(A_i and B_i \to C_i)}(u, v, w)$$

i. When input data are singleton $u=u_0$,$v=v_0$

$$\mu_{C_i'}(w) = [\mu_{A_i}(u_0)\ and\ \mu_{B_i}(v_0)] \to \mu_{C_i}(w)$$

The Mamdani Method uses the minimum operator (∧) for the fuzzy implication (→). From lemma 4,

$$\mu_{C_i'}(w) = \alpha_i \wedge \mu_{C_i}(w)$$

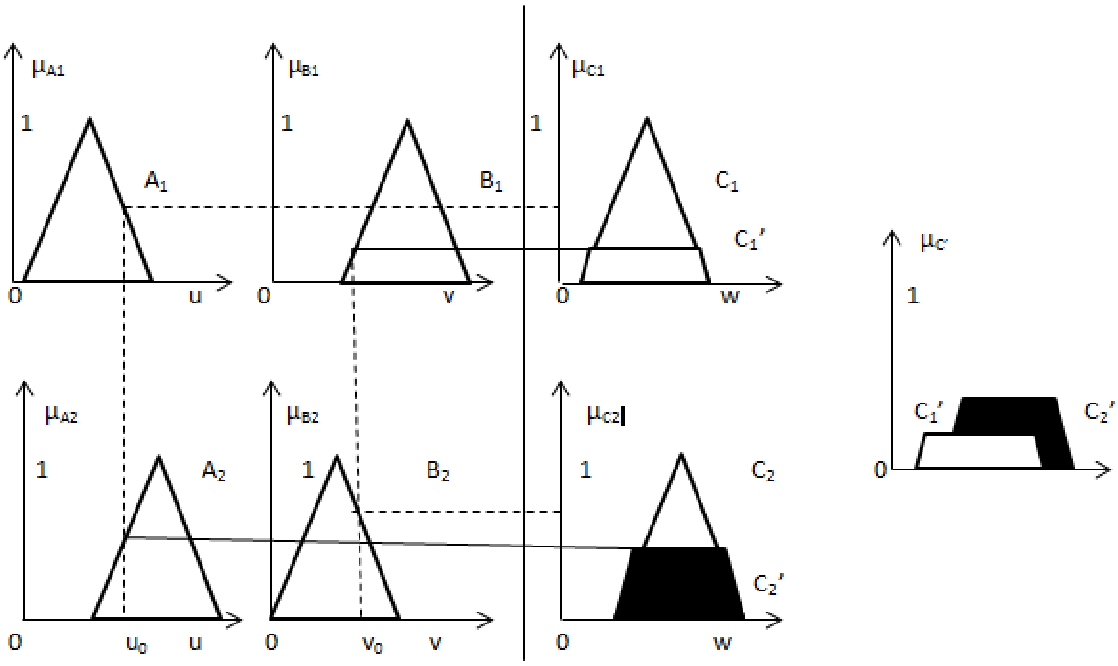
where $\alpha_i = \mu_{A_i}(u_0) \wedge \mu_{B_i}(v_0)$

From lemma 3, we know the membership function μ_C of the inferred consequence C is given by the aggregated result derived from individual control rules. Thus, when there are two rules R_1 and R_2,

$$\mu_{C'}(w) = \mu_{C_1'} \vee \mu_{C_2'}$$

$$= [\, \alpha_1 \wedge \mu_{C_1}(w)] \vee [\alpha_2 \wedge \mu_{C_2}(w)]$$

The procedure of Mamdani fuzzy inference when the inputs are given as singletons is represented in the figure below



Therefore in general, from lemma 3,

$$\mu_{C'}(w) = \bigvee_{i=1}^{n} [\, \alpha_i \wedge \mu_{C_i}(w)] = \bigvee_{i=1}^{n} \mu_{C_i'}(w)$$

$$C' = \bigcup_{i=1}^{n} C_i'$$

ii. When input data are fuzzy sets, A′ and B′

From lemma 5,

$$\mu_{C_i'}(w) = \alpha_i \wedge \mu_{C_i}(w)$$

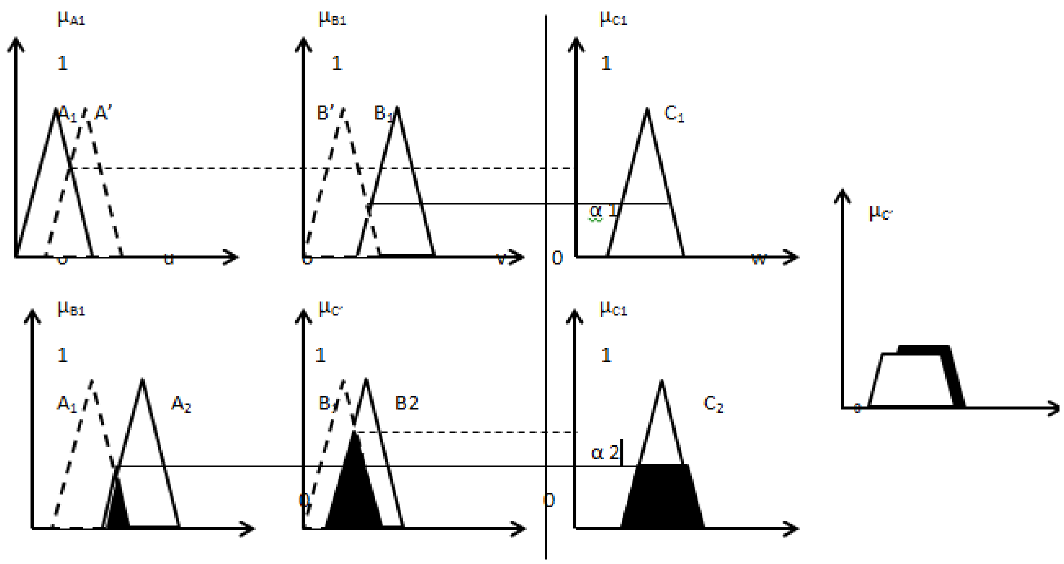
Where $\alpha_i = \min [\, \max_u(\mu_{A'}(u) \wedge \mu_{A_i}(u)),\ \max_v(\mu_{B'}(v) \wedge \mu_{B_i}(v))]$.

From lemma 3, we have the aggregated result

$\mu_{C'}(w) = \vee_{i=1}^{n}[\, \alpha_i \wedge \mu_{C_i}(w)] = \vee_{i=1}^{n} \mu_{C_i'}\,(w)$

$C' = \cup_{i=1}^{n} C_i'$

The graphical interpretation of this inference is given in figure.



2) TSUKAMOTO METHOD

In this method, the consequence of each fuzzy rule is represented by a fuzzy set with a monotonic membership function.

The rule base has the form as:

R_i: if u is A_i and v is B_i, then w is C_i ,i=1,2,…,n

where $\mu_{C_i}(w)$ is a monotonic function.

As a result, the inferred output of each rule is defined as a crisp value induced by the rule's matching degree. The overall output is taken as the weighted average of each rule's output.

We suppose that the set C_i has a monotonic membership function $\mu_{C_i}(w)$ and that α_i is the matching degree of i th rule.

(1) For the singleton input (u_0,v_0)

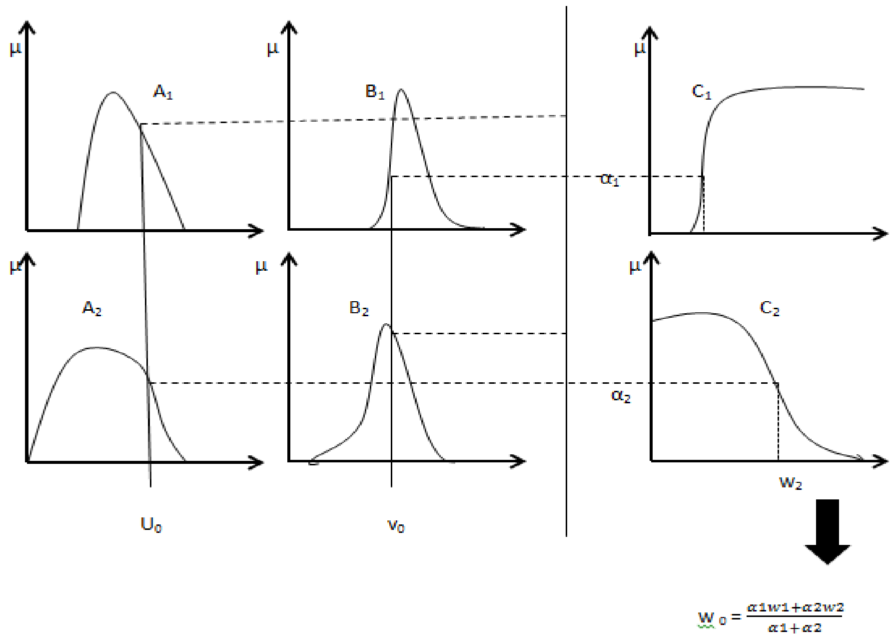
$\alpha_i = \mu_{A_i}(u_0) \wedge \mu_{B_i}(v_0)$

(2) For the fuzzy set (A′, B′)

α_i= min [$\max\limits_{u}$($\mu_{A'}(u) \wedge \mu_{A_i}(u)$), $\max\limits_{v}$($\mu_{B'}(v) \wedge \mu_{B_i}(v)$)].

Then the result of i^{th} rule is obtained by

$$w_i = \mu_{C_i}^{-1}(\alpha_i)$$



The final result is derived from the weighted average like in the following where there are two rules.

$$w_0 = \frac{\alpha_1 w_1 + \alpha_2 w_2}{\alpha_1 + \alpha_2}$$

Since each rule infers a crisp result, the Tsukamoto fuzzy model aggregates each rule's output by the weighted average method. Therefore it avoids the time-consuming process of defuzzification.

3) TSK METHOD

This method was proposed by Takagi, Sugeno and Kang. A typical fuzzy rule in this model has the form

If u is A and v is B then w = f(u, v)

A and B are fuzzy sets in the antecedent while w=f (u,v) is a crisp function in the consequent. Usually f (u, v) is a polynomial in the input variable u and v, and thus this method works when inputs are given as singleton values.

For simplicity, assume we have two fuzzy rules as follows.

R_1: if u is A_1 and v is B_1 then $w = f_1(u, v) = p_1u + q_1v + r_1$

R2: if u is A_1 and v is B_1 then $w = f_2(u, v) = p_2u + q_2v + r_2$

where p_1, p_2, q_1 and q_2 are constant.

The inferred value of the control action from the first rule is $f_1(u_0, v_0)$ where u_0 and v_0 are singleton inputs and α_1 is the matching degree. The inferred value from the second is $f_2(u_0, v_0)$ with the matching degree α_2. The matching degrees are obtained like in the previous methods.

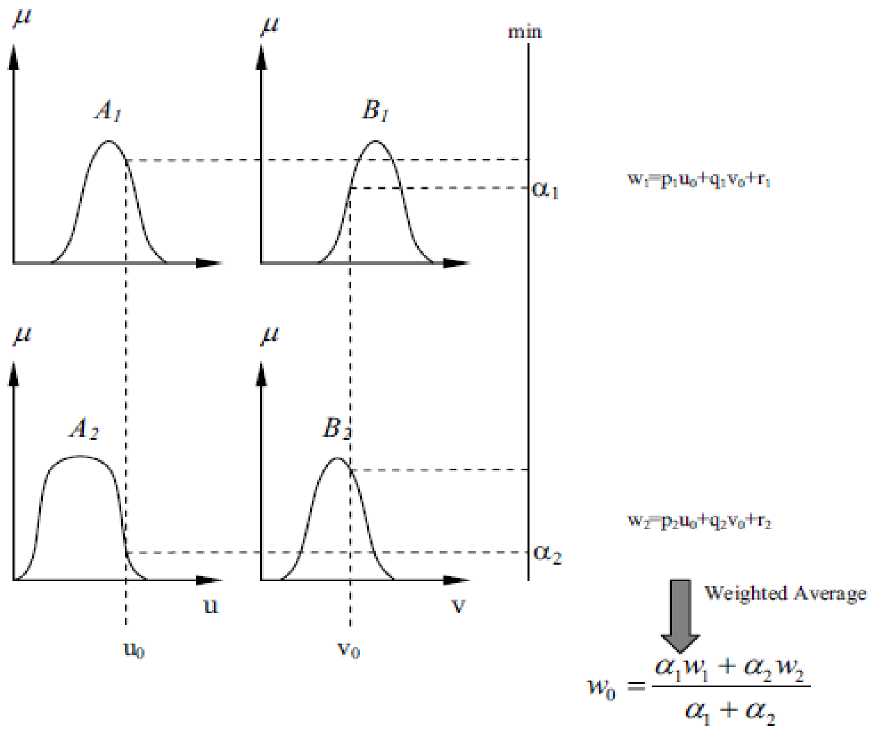
$\alpha_i = \mu_{A_i}(u_0) \wedge \mu_{B_i}(v_0)$

They are all crisp values. The aggregated result is given by the weighted average.

$$w_0 = \frac{\alpha_1 f_1(u_0, v_0) + \alpha_2 f_2(u_0, v_0)}{\alpha_1 + \alpha_2}$$

$$= \frac{\alpha_1 w_1 + \alpha_2 w_2}{\alpha_1 + \alpha_2}$$

This method also saves the defuzzification time because the final result w_0 is a crisp value.



3.2. DEFUZZIFICATION

In many practical applications, a control command is given as a crisp value. Therefore it is needed to defuzzify the result of the fuzzy inference. A defuzzification is a process to get a non-fuzzy control action that best represents the possibility distribution of an inferred fuzzy control action. Unfortunately, we have no systematic procedure for choosing a good defuzzification strategy, and thus we have to select one in considering the properties of application case. The three commonly used strategies are described below.

1. MEAN OF MAXIMUM METHOD (MOM)

The MOM strategy generates a control action which represents the mean value of all control actions whose membership functions reach the maximum. In the case of discrete universe the control action may be expressed as

$$z_0=\sum_{j=1}^{k}\frac{z_j}{k}$$

z_j: control action whose membership functions reach the maximum.

k: number of such control actions

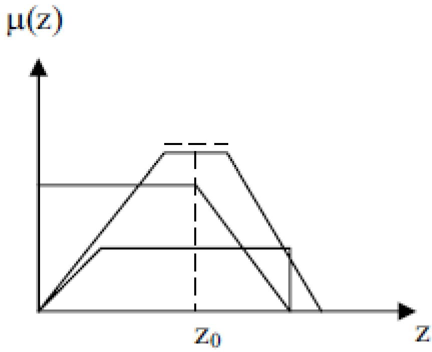


Fig. Mean Of Maximum (MOM)

2. CENTER OF AREA METHOD (COA)

The widely used COA strategy generates the center of gravity of the possibility distribution of a fuzzy set C. In the case of a discrete universe, thus method gives

$$z_0 = \frac{\sum_{j=1}^{n}\mu_C(z_j).z_j}{\sum_{j=1}^{n}\mu_C(z_j)}$$

where n is the number of quantization levels of the output, C is a fuzzy set defined on the output dimension (z).

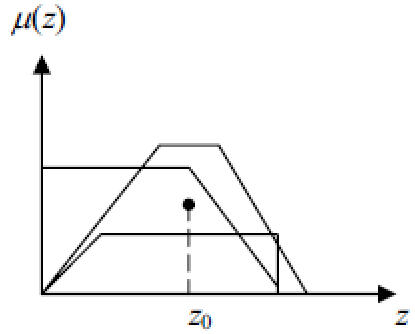


Fig. Center Of Area (COA)

3. BISECTION OF AREA (BOA)

The BOA generates the action z_0 which partitions the area into two regions with the same area.

$$\int_{\alpha}^{Zo} \mu_C(z)dz = \int_{z_0}^{\beta} \mu_C\,(z)\mathrm{dz}$$

Where α = min {z | zϵW}, β = max {z | zϵW}.

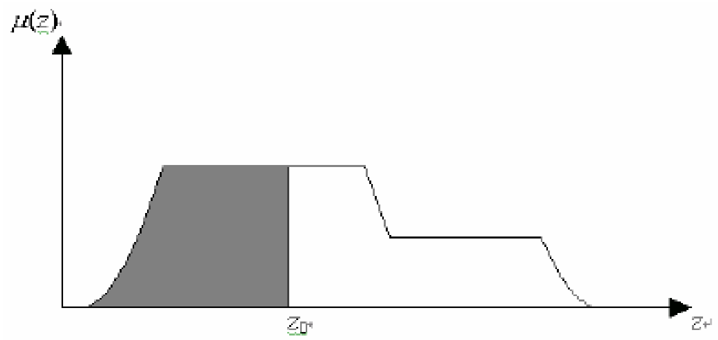


Fig. Bisector Of Area (BOA)

3.3 DESIGN PROCEDURE OF FLC

When we decided to design a fuzzy logic controller, we can follow the following design procedure

(1) *Determination of state variables and control variables*

In general, the control variable is determined depending on the property of process to be controlled. But we have to select the state variables. In general, state, state error and error difference are often used. The state variables are input variables, and the control variables are output of our controller to be developed.

(2) *Determination of inference method*

We select one method among four inference methods described in the previous section. The decision is dependent upon the properties of process to be studied.

(3) *Determination of fuzzification method*

It is necessary to study the property of measured data of state variables. If there is uncertainty in the data, the fuzzification is necessary, and we have to select a fuzzification method and membership functions of fuzzy sets. If there is no uncertainty, we can use singleton state variables.

(4) *Discretization and normalization of state variable space*

In general, it is useful to use discretized and normalized universe of discourse. We have to decide whether it is necessary and how we can do.

(5) *Partition of variable space*.

The state variables are input variables of our controller and thus the partition is important for the structure of fuzzy rules. At this step, partition of control space (output space of the controller) is also necessary.

(6) *Determination of the shapes of fuzzy sets*

It is necessary to determine the shapes of fuzzy sets and their membership functions for the partitioned input spaces and output spaces.

(7) Construction of fuzzy rule base

Now, we can build control rules. We determined the variables and corresponding linguistic terms in antecedent part and consequent part of each rule. The architecture of rules is dependent upon the inference method determined in step 2).

(8) *Determination of defuzzification strategy*

In general, we use singleton control values and thus we have to determine the method.

(9) *Test and tuning*

It is almost impossible to obtain a satisfactory fuzzy controller without tuning. In general it is necessary to verify the controller and tune it until when we get satisfactory results.

3.4 FUZZY EXPERT SYSTEMS

An expert system is a program which contains human expert's knowledge and gives answers to the user's query by using an inference method. The knowledge is often stored in the form of rule base, and the most popular form is that of "if-then".

A fuzzy expert system is an expert system which can deal with uncertain and fuzzy information. In our real world, a human expert has his knowledge in the form of linguistic terms. Therefore it is natural to represent the knowledge by fuzzy rules and thus to use fuzzy inference methods.

The structure of a fuzzy expert system is similar to that of the fuzzy logic controller. Its configuration is shown in figure below. As in the fuzzy logic controller, there can be fuzzification interface, knowledge base, and inference engine (decision making logic). Instead of the defuzzification module, there is the linguistic approximation module.

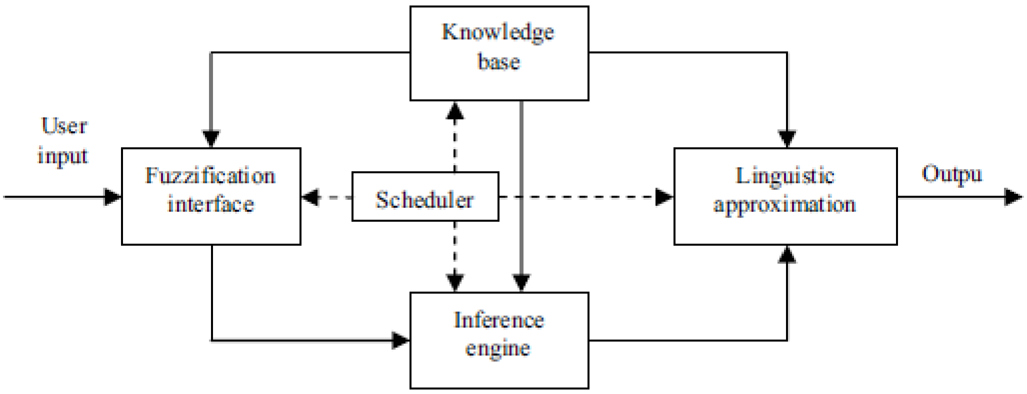


Fig. Configuration of fuzzy expert system

3.4.1 FUZZIFICATION INTERFACE

This module deals with the user's request, and thus we have to determine the fuzzification strategy. If we want to make the fuzzy expert system receive linguistic terms, this module has to have an ability to handle such fuzzy information. The fuzzification strategy, if necessary, is similar to that of the fuzzy logic controller.

Contrary to the fuzzy logic controller, it is not needed to consider the discretization or normalization. But the fuzzy partition and assigning fuzzy linguistic terms to each sub region are necessary.

The expert's knowledge may be represented in the form of "if-then" by using fuzzy linguistic terms. Each rule can have its certainty factor which represents the certainty level of the rule. This certainty factor is used in the aggregation of the results from each rule.

3.4.2 INFERENCE ENGINE (DECISION MAKING LOGIC)

The fuzzy expert systems can use the inference methods of the fuzzy logic controller. The system does not deal with a machine or process, and thus it is difficult to have a fuzzy set with monotonic membership function in the consequent part of a rule. Therefore Mamdani method is often used.

3.4.3 LINGUISTIC APPROXIMATION

A fuzzy expert system does not control a machine nor a process, and thus, in general, the defuzzification is not necessary. Instead of the defuzzification module, sometimes we need a linguistic approximation module. This module finds a linguistic term which is closest to the obtained fuzzy set. To do it, we may use a measuring technique of distance between fuzzy sets.

3.4.4 SCHEDULER

This module controls all the processes in the fuzzy expert system. It determines the rules to be executed and sequence of their executions. It may also provide an explanation function for the result. For example, it can show the reason how the result was obtained.

CHAPTER 4

APPLICATIONS

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 an 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

4.1 FUZZY LOGIC WASHING MACHINE

Washing machine is of great domestic necessity as it frees us from the burden of washing our clothes and saves ample of our time. We will discuss the aspect of designing and developing of Fuzzy Logic based, Smart Washing Machine. The regular washing machine (timer based) makes use of multi-turned timer based start-stop mechanism which is mechanical as is prone to breakage. In addition to its starting and stopping issues, the mechanical timers are not efficient with respect of maintenance and electricity usage. A number of international renowned companies have developed the machine with the introduction of smart artificial intelligence. Such a machine makes use of sensors and smartly calculates the amount of run-time (washing time) for the main machine motor. Real-time calculations and processes are also catered in optimizing the run-time of the machine. Here we deal with FLC (Fuzzy Logic Controller) based Washing Machine, which is capable of automating the inputs and getting the desired output (wash-time).Consider an individual doing the laundry for the first time and is not familiar with the type of fabric or detergent to be used. In order to solve the laundry problem, he would seek input from professionals to save the effort. Consider a solution combining the inputs from detergent maker, fabric maker and professional laundry servicemen into an implementable model that would facilitate the unaware person doing laundry for the first time. This is an effort to propose the same by suggesting a model that can effectively reduce the problem of laundry.

The primary concept in designing a Fuzzy Logic Controller (FLC) lies behind the information gathered from various sources of experience or experts (Laundry-Load Characteristics). For instance, when considering the design of washing machine, we have to keep the following information summarized up: -

4.1.1. Type of Clothes (Fabric or Textile)

The nature of clothes that require a wash in the washing machine. The input from textile engineer or a person who deals in fabrics will be desirable.

4.1.2. Type of Detergent (Chemical Property)

Different detergent reacts differently on varied nature of clothes. A chemical engineer who is involved in the manufacture of type of detergent will provide the requisite information.

4.1.3. Type of Stain (Chemical Property)

The chemical property of stain on the piece of cloth will vary. Therefore, in order to remove the stains one has to use different methods.

4.1.4. Temperature of Washing Water

The information regarding the upper and lower threshold of temperature that the fabric of clothes, detergent and stain require.

4.1.5. Other Factors

Among the miscellaneous factors, weight of clothes and water, amount of water soaked after wash-water drainage, spin time required to remove the soaked water and heat required to dry the wash-load.

4.2.1. Parameters

All of the parameters could be gathered in a smart way as well. Obviously, automating the (laundry load) inputs and making them decide the desired and required (washing parameters) output would be a smart way of doing laundry. Controlling these parameters could lead to a cleaner laundry; conserve water, save detergent, electricity, time, and money.

Inputs

The characteristics of the laundry load (inputs) include: the actual weight, fabric types, and amount of dirt.

Outputs

The washing parameters (outputs) include: amount of detergent, washing time agitation, water level, and temperature.

4.2.2. Practical Model

In the practical model implementation, consider, for simplicity, as shown in Figure 1, a washing machine with two inputs (Saturation Time and Dirtiness) and one output (Wash Time). For practical purposes type and amount of detergent has been kept manual.

Fig.1 Fuzzy Logic Controller (FLC) of Smart Washing Machine (SWM)

Saturation Time

The saturation time of dirt or stain in water or conversely the amount of time water takes to saturate with the dirt. It is measured by the opaqueness or opacity of wash water with the use of optical sensor. The output of optical sensor is scaled on fuzzy rule base.

Dirtiness

It is the type of stain or type of dirtiness of cloth. In order to measure the dirtiness, pressure sensor system is incorporated. It allows the cloth to run a wash-cycle under idle phase and measure the pressure induced by the clothes with the dirtiness. The pressure sensor system is scaled as per fuzzy rule base.

Wash Time

The output desired or the wash-time required to clean the dirt. It also includes the re-run cycle, stained /saturated wash-water drainage and fresh wash-water refill.

4.2.3. Fuzzy Subsets

In practical observation, simple mathematics cannot relate or extract the inputs (Saturation Time and Dirtiness) into meaningful output (Wash-Time). Only a well-informed and knowledgeable user can do the task manually having all the requisite information of clothes material, detergents and nature of wash. With the use of Fuzzy Logic, a rule-base is created based on the knowledge of the user to control or automate the process.

Dirtiness

The dirtiness is defined in the range from 0 to 30, by defined fuzzy subsets: Low, Medium and High as shown in Figure 2. For ideal demonstration, the subsets are kept triangular; however they can be adoptive in shapes and ranges in reality. At any given instance, all the three subsets will trigger a certain amount of scale. For example, at range 15(exactly middle), High is 0.0, Low is also 0.0 but Medium subset is 0.99, which means all three subsets will execute in results extraction to 0.0, 0.99 and 0.0 levels.

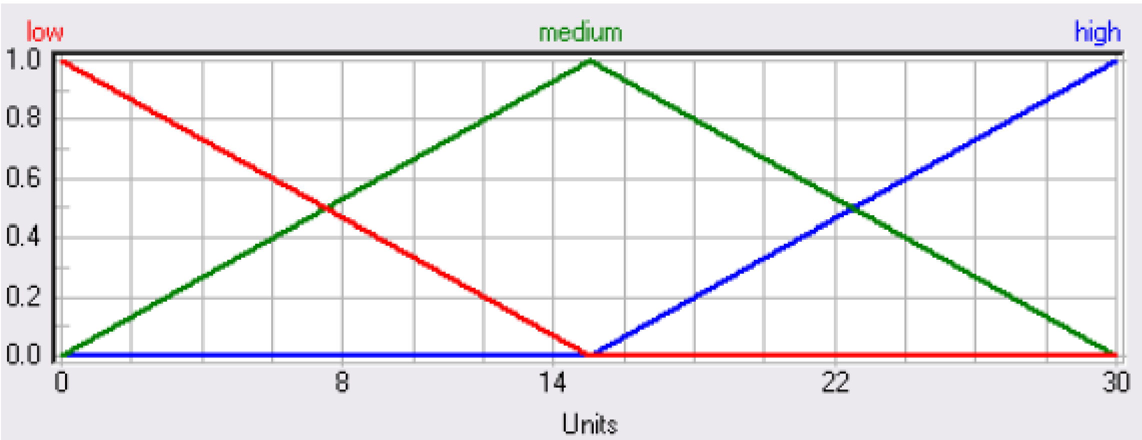


Fig.2. Variable Subset (Dirtiness)

Saturation Time

The Saturation time is defined in the range of 0 to 10 minutes and has been classified into three sub levels (Low, Medium and High). The same is shown in Figure 3.

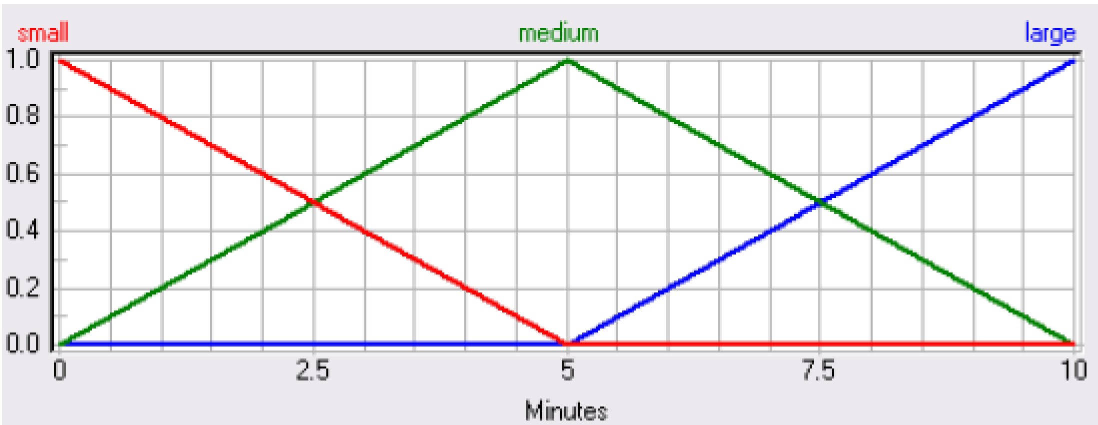


Fig. 3. Variable Subset (Saturation Time)

Wash Time

The wash time is defined in the range from 0 to 15 minutes, based on the fuzzification of above mentioned inputs (saturation time and dirtiness). Further, scaling has been done into five sub levels. The output is defined by the Fuzzy subsets as shown in Figure 4.

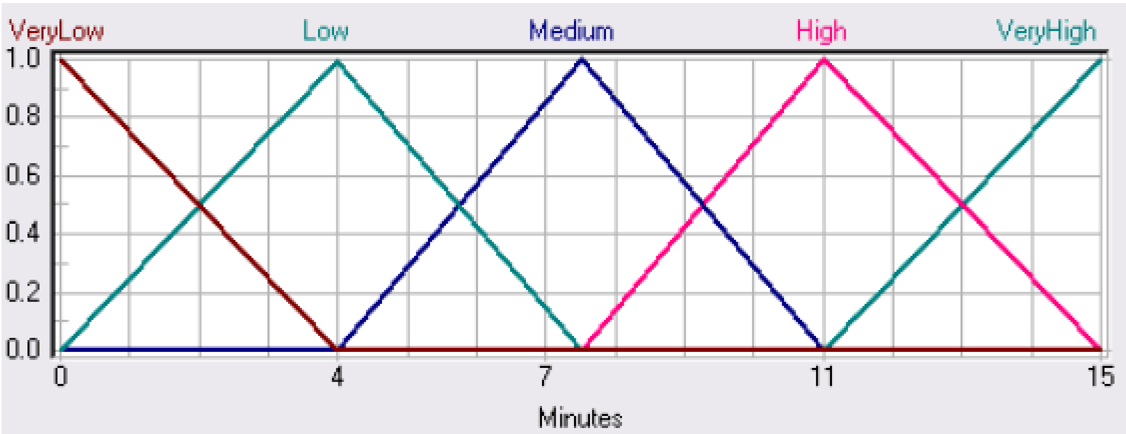


Fig. 4. Output Subset (Wash-Time)

4.2.4. Output Evaluation

Rule Base

After fuzzification, the fuzzified input parameters are evaluated using IF/THEN rules. Two control inputs each with 3 fuzzy regions are used in this case. This gives a possible of 3 x 3 = 9 possible input conditions, but for practical purposes in this washing machine only 5 sub levels have been used. However, the algorithm is adaptive and maybe tailored based on more precise input sets. For instance, if DIRTINESS is LOW and SATURATION_TIME is MEDIUM then WASH_TIME is LOW. The complete input and output parameters are given in the table below.

Table 1. Fuzzy Logic Ranges for Output (Wash-Time)

		Dirtiness		
		LOW	MEDIUM	HIGH
Saturation Time	SMALL	VERY_LOW	LOW	MEDIUM
	MEDIUM	LOW	MEDIUM	HIGH
	LARGE	MEDIUM	HIGH	VERY_HIGH

Output Sub Levels

The wash time sub levels have been defined in the range given in table 2. For example, if sub level, medium is triggered in output wash-time, the washing time will be from 4 – 11 minutes.

Table 2. Physical Output Ranges (Wash-Time)

WASH_TIME	
LINGUISTIC VARIABLE	RANGE (MINUTES)
VERY_LOW	0-4
LOW	0-8
MEDIUM	4-11
HIGH	7-15
VERY_HIGH	11-15

Rule Evaluation

Since, we have established that both inputs will trigger a particular sub level of output subset. However, the same is quantified when executed through software. Figure 5 shows that input of dirtiness of medium level is 0.48 of the scale 1 and saturation time of medium level is 0.57. The output is evaluated by ANDing the inputs and resultantly 0.48 of High level of wash time is triggered. And based on the time scales given in table 2, the physical time will be 11 minutes which is approximately half of High sublevel of wash time.

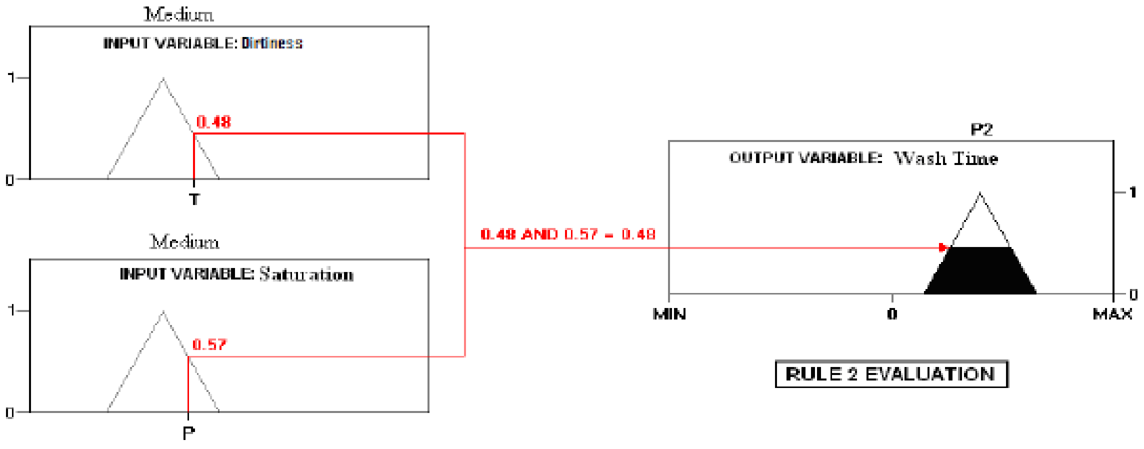


Fig.5. Rule Evaluation

Conclusion

The commercial application of fuzzy has made its mark over the past few decades because of the theoretically infinite range of control over a particular application. The precision in wash time will not only economize energy resources (including electricity & water) but also benefit the user to save finances in commercial laundry solutions offered in the market.

4.2 FUZZY LOGIC RICE COOKER

4.2.1. Introduction

We aim at presenting the idea of controlling the cooking time of rice based on the type of rice and quantity of water using fuzzy logic control. We describe the procedure that can be used to get a suitable cooking time for different types of rice. The process is based entirely on the principle of taking non precise inputs from the sensors, subjecting them to fuzzy arithmetic and obtaining a crisp value of the cooking time. It is quite clear that this method can be used in practice to further automate the rice cookers. Nevertheless, this method, though with much larger number of input parameters and further complex situations, is being used by the giants like LG and Samsung. The rice cooker features with advanced logic technology, which allows it to think for itself and make adjustments to the temperature and timing of batch of rice totally on the cooking. A spherical inner cooking pan and heating system distributes heat evenly so the rice at the bottom is the same consistency.

4.2.2. Architecture

The problem has been simplified by using only two variables. The two inputs are:

1. Type of rice

2. Water quantity

Figure 1 shows the basic block diagram of the fuzzy controller. The fuzzy controller takes two inputs (as stated for simplification), processes the information and outputs a cooking status. How to get these two inputs can be left to the sensors (optical, electrical or any type). The working of the sensors is not a matter of concern in this model. We assume that we have these inputs at our hand. Anyway the two stated points need a bit of introduction which follows. The type of rice is determined by the user who applies it to the cooker for cooking. On the other hand, water quantity is determined by the rice cooker automatically is determined by the time of saturation, the time it takes to reach saturation. Saturation is a point, at which there is no more appreciable change in the color of the water. Type of rice determines which type of rice is to be cooked whereas water quantity determines the how much water needed to cook a particular type of rice. Thus a fairly straight forward sensor system can provide us the necessary input for our fuzzy controller.

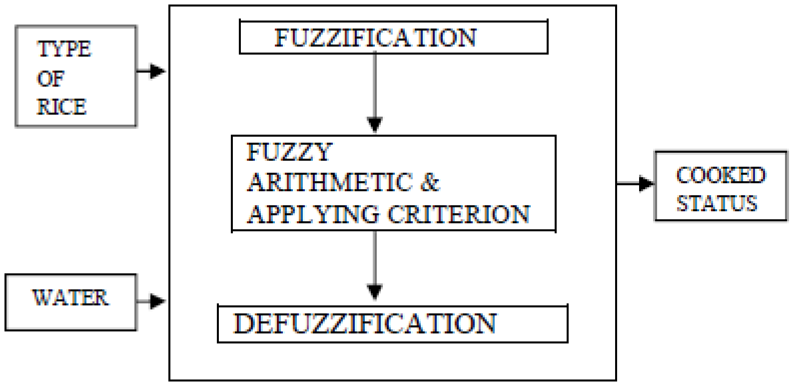


Fig. 1 Basic block diagram of fuzzy controller process.

4.2.3 Fuzzification, Fuzzy Based Rules

Fuzzification

Before the details of the fuzzy controller are dealt with, the range of possible values for the input and output variables are determined. These (in language of Fuzzy Set theory) are the membership functions used to map the real world measurement values to the fuzzy values, so that the operations can be applied on them. Figure 2 shows the labels of input and output variables and their associated membership functions. Values of the input variables type_of_rice and water quantity are normalized over the domain of optical sensor.

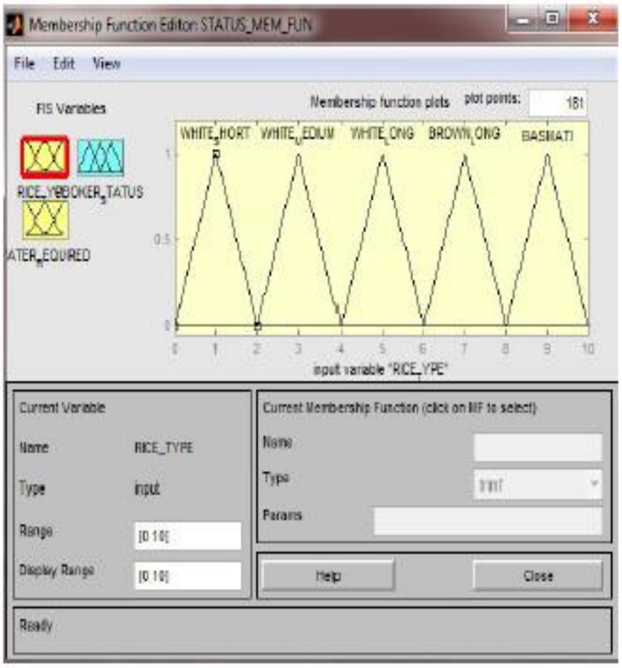


Fig 2. Type of rice

The different types of rice applied as input for cooking are basmati, white short, white medium, white long and brown long. These are taken as linguistic variables for the input type of rice. The corresponding membership function plot for type of rice is shown in fig.2.The second input is the quantity of water required. The value of water is taken in ml (milliliter).The membership function plot for water quantity is in given in fig.3

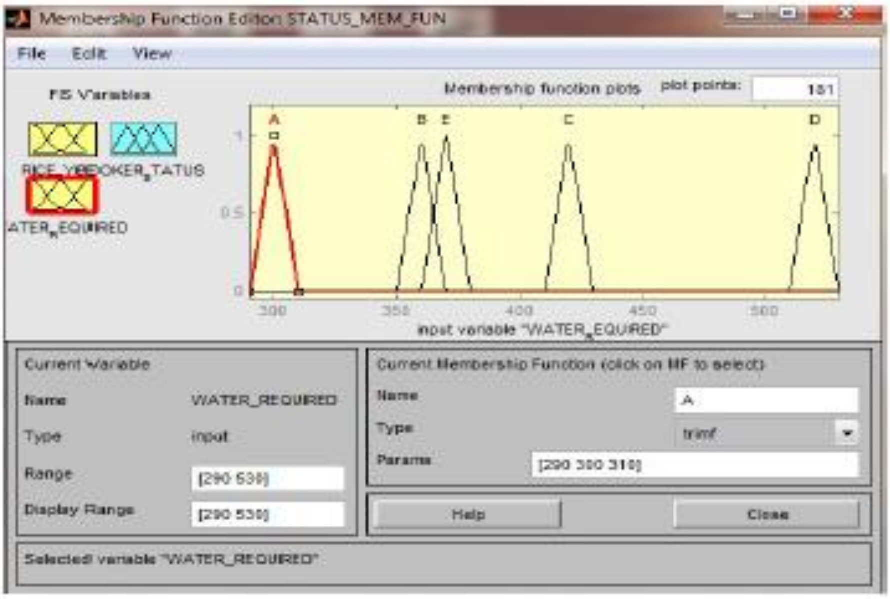


Fig.3 Water quantity

Fuzzy based rules

The decision which the fuzzy controller makes is derived from the rules which are stored in the database. These are stored in a set of rules. Basically the rules are if-then statements that are intuitive and easy to understand, since they are nothing but common English statements. Rules used in this paper are derived from common sense, data taken from typical home use, and experimentation in a controlled environment.

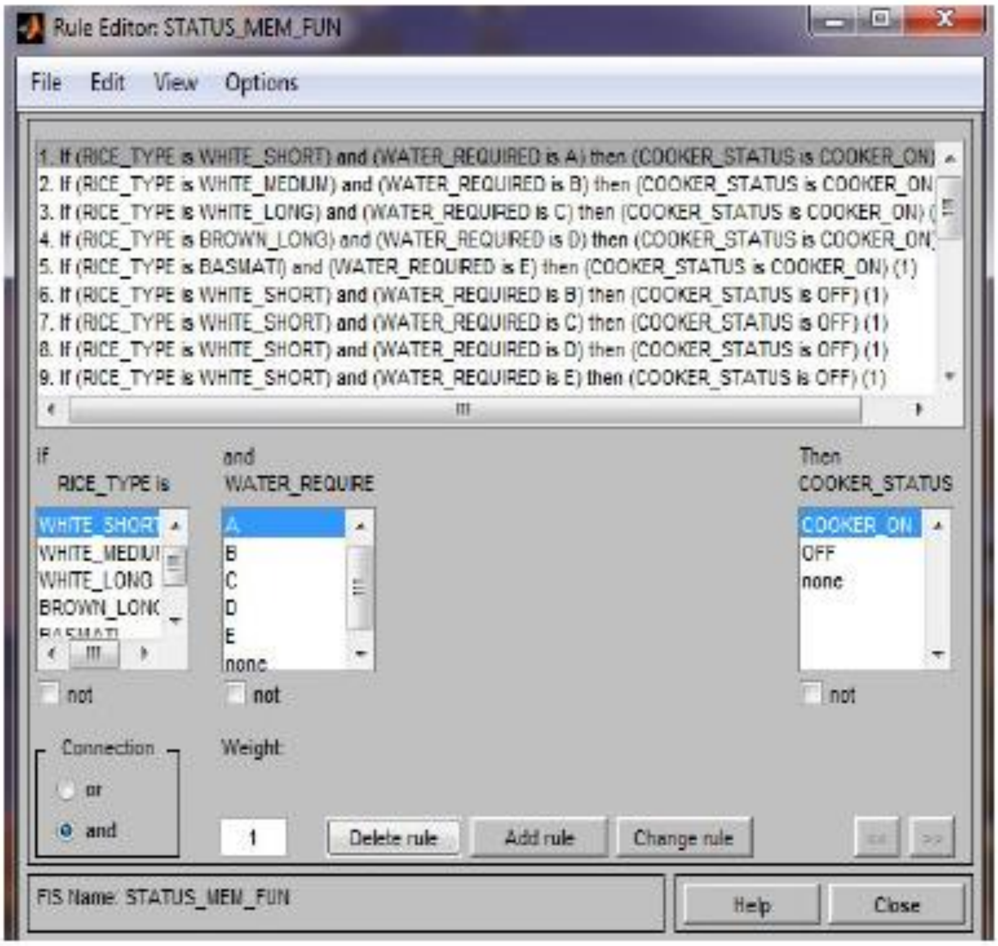


Fig 4. Fuzzy based rules

4.2.4. Experiment

Different types of rice and water quantity are taken for experimental purpose. The respective water quantity for different types of rice and rice quantity taken is as given below.

TABLE I

TYPE OF RICE	WATER QUANTITY(in ml)	RICE QUANTITY(in grams)
WHITE SHORT	360	240
WHITE MEDIUM	360	240
WHITE LONG	420	240
BROWN LONG	520	240
BASMATI	360	240

From the table it should be understood that for rice type white short water needed is 360 ml and rice quantity is 240 gms and so on. It means that if the water quantity for a particular type of rice falls below the quantity mentioned in the above table then the cooker will not start. For example for 240 gms cooking of basmati rice, minimum water required is 360 ml but if the water falls below this quantity then the cooker will not get on. The membership function plot for cooker output is shown in figure 5.

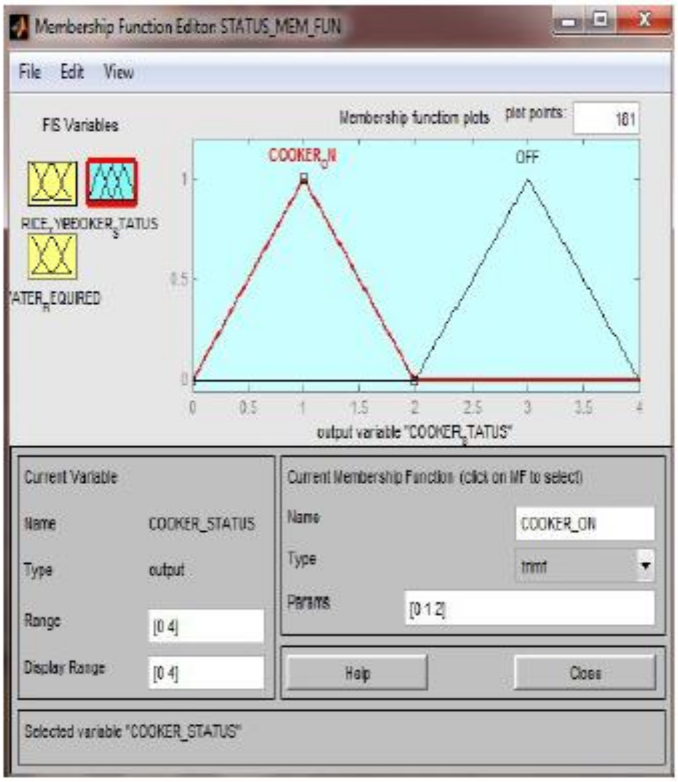


Fig.5 Cooker status

By the use of fuzzy logic control we have been able to obtain a cooking time for different type of rice and for different quantity of water. The conventional method required the human interruption to decide upon what should be the cook time for different rice types. In other words this situation analysis ability has been incorporated in the cooker which makes the cooker much more automatic and intelligent which represents the decision taking power of the new arrangement.

4.3 FUZZY LOGIC MODEL FOR PREVENTION OF ROAD ACCIDENTS

Traffic accidents are rare and random. However, many people died or were 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 to these results, which are mainly driver fault, lack of infrastructure, environment, literacy, weather conditions etc. Cost of traffic accidents is roughly 3% of gross national product. However, I 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.

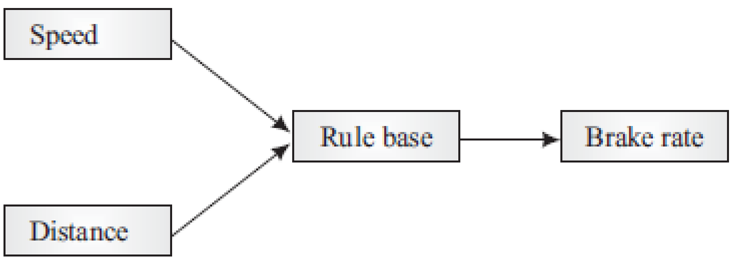
4.3.1. Traffic accidents and traffic safety

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 behavior in traffic.

4.3.2. Application of fuzzy logic

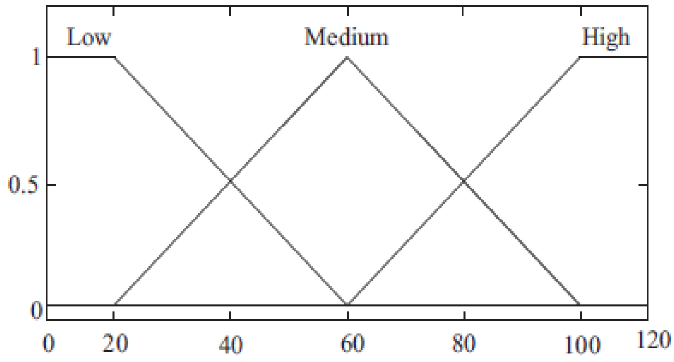
In the study, a model was established which estimates brake rate using fuzzy logic. The general structure of the model is shown in figure below



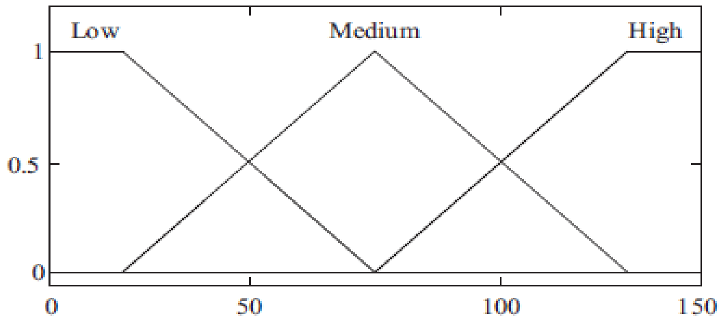
General structure of fuzzy logic model

4.3.3. Membership functions

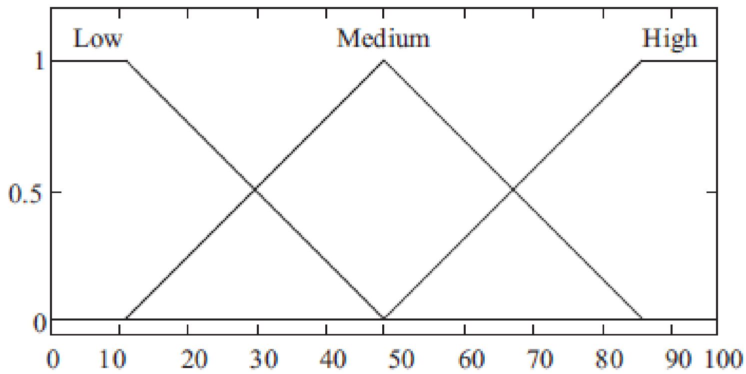
In the established model, different membership functions were formed for speed, distance and brake rate. Membership functions are given in figures below. For maximum allowable car speed (in motorways) in India, speed scale selected as 0-120 km/h on its membership function. Because of the fact that current 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.



Membership function of speed



Membership function of distance



Membership function of brake rate

4.3.4. Rule base

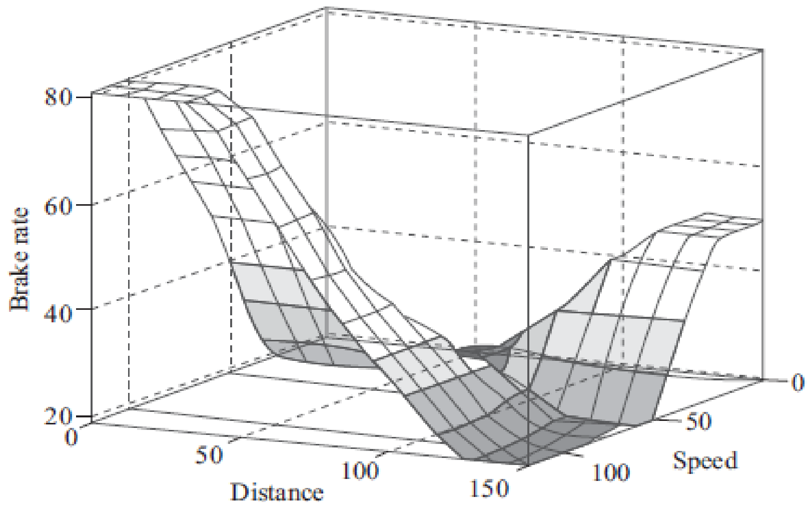
We need a rule base to run the fuzzy model. Fuzzy Allocation Map (rules) of the model was constituted for membership functions whose figures are given on the table below. It is important that the rules were not completely written for all probability. Figure 6 shows the relationship between inputs, speed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

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

4.3.5. Output

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



Relationship between inputs and brake rate

4.3.6. Conclusions

Many people die or injure because of traffic accidents in India. Many reasons can contribute to 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 a 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 to reduce the traffic accident rate. This model can be adapted to vehicles.

CONCLUSION

Fuzzy logic is an extension of two-valued or Boolean logic. It is applied to control systems and other fields like facial pattern recognition, medical diagnosis and treatment plans, transmission systems,vacuum cleaners etc. This project presents a brief overview of fuzzy logic controller. We have discussed about the four components of a fuzzy logic controller(fuzzification, knowledge base, inference engine,defuzzification), the fuzzy expert systems and a general methodology for constructing fuzzy logic controller. Fuzzy logic is applied with great success in various control applications such as control of traffic lights, air conditioners, refrigerators etc. Here we have discussed in detail the design of fuzzy logic rice cooker, washing machine and a model for prevention of road accidents. Although fuzzy logic has applications in a number of different areas and wide range of researches are ongoing ,it is not yet known to people other than researchers and those familiar with intelligent systems. For many people the engineering and scientific meaning of the word fuzzy is still fuzzy. Actually fuzzy is not fuzzy once we get to know it.

REFERENCES

- First Course on Fuzzy Theory and Applications; Kwang H. Lee
- Fuzzy Logic: Intelligence, Control and Information; John Yen, Reza Langari
- https://www.researchgate.net/publication/276353926_An_Intelligent_Fuzzy_Based_Technique_Of_Making_Food_Using_Rice_Cooker.
- https://arxiv.org/ftp/arxiv/papers/1701/1701.01654.pdf
- https://www.tutorialspoint.com/fuzzy_logic/fuzzy_logic_control_system.htm
- Fuzzy logic and neural networks(basic concepts and applications); Chennakesava R**.** Alava
- An Egg-Boiling Fuzzy Logic Robot – CISIEEE: https://youtu.be/J_Q5X0nTmrA