Kasdi Merbah University-Ouargla

Faculty of New Technology for Information and Communication

Department of Electronics and Communications

DISSERTATION:

ACADEMIC MASTER

DOMAIN: SCIENCES AND TECHNICS

Field: Electronic

Specialty: Automatic

Presented by:

BAHLOUL Omar REGUIBI Noureddine

Title

Type-2 Fuzzy Control for Twin Rotor (TRMS)

Publically defended on: 07/06/2015

Before the jury

Mr. KADRI Farid	MAA	President	UKM Ouargla
Mr.BECHKA Larbi	MAA	Examiner	UKM Ouargla
Mr.KAFI Mohammed Redouane	MAA	Supervisor	UKM Ouargla
Mr. BEN HELELL Belkhir	MAA	Co- Supervisor	UKM Ouargla

Academic year: 2014/2015

DEDICATION

I dedicate my dissertation; piece of work, to before anyone else, the heart and soul of my entire existence:

My beautiful mother** for the silent heartfelt prayers… Mum! You are and always will be a symbolism of hope, faith and love in my life. And **My wonderful father** for the endless giving and the unconditional support. I will always cherish what you two have done for me and I… I equally love you forever and ever. I also dedicate this piece of work to my loving brothers and sisters, **Abd Rahim, Med Tayab, and Djamal.

*My sister **Aicha , Nafissa, Amina and Nossiaba.***

Also to my uncles and aunts

*As well as to all the Teacher's staff of department Electrical Engineeing specially teacher **Djamel Sami** .*

I dedicate this piece of work and give thanks to all my friends; especially my lifetime friends Also, I cannot forget my friends in college with you it was such an amazing time.

Finally, I genuinely dedicate my graduation dissertation to anyone has ever believed in me and helped channel my talent here on Earth.

… …

… Omar …

DEDICATION

I dedicate my dissertation; piece of work, to before anyone else, the heart and soul of my entire existence:

My beautiful mother** for the silent heartfelt prayers… Mum! You are and always will be a symbolism of hope, faith and love in my life. And **my wonderful father** for the endless giving and the unconditional support. I will always cherish what you two have done for me and I… I equally love you forever and ever. I also dedicate this piece of work to my loving brothers and sisters, **Abd Alhak, Abd Star, Ouanis, Mohaned, Abd Rahman, Ali, Ahamed.

*My sister **Safia, Laila, Messouda, Karima, Ouarda, Fatena.***

Also to my uncles and aunts

*As well as to all the Teacher's staff of department Electrical Engineeing specially teacher **DjamelSami**. And also the teacher's staff of career center of Ouargla.*

I dedicate this piece of work and give thanks to all my friends; especially my lifetime friends Also, I cannot forget my friends in college with you it was such an amazing time.

Finally, I genuinely dedicate my graduation dissertation to anyone has ever believed in me and helped channel my talent here on Earth.

… …

…Noureddine…

ACKNONLGEMENTS

We would like to fully acknowledge and thank Allah for all the blessings and guidance we are granted. Very special thanks from the bottom of our hearts for our supervisors

Mr. KAFI Med Radouaane , Mr. BEN HELELL Belkhir and , **Mr. CHAOUI Hicham**

Who were more than generous with their expertise and precious time. Also, we give lots of credit to all the personnel and teacher's staff of Kasdi Merbah University for all the facilities they have provided through all the process of making this project.

... Noureddine ...

... Omar ...

List of Figures

- List of Tables

List Of Abbreviation And Symbols

$F_{h/v}$: Function of aerodynamic force from tail / main rotor.

m_t : Mass of tail part of the beam.

m_{tr} : Mass of the tail DC motor.

m_{ts} : Mass of the tail shield.

m_m : Mass of main part of the beam.

m_{mr} : Mass of the main DC motor

m_{ms} : Mass of the main shield

m_b : Mass of the counter-weight beam

m_{cb} : Mass of the counter-weight

l_t : Length of tail part of the beam

l_m: Length of main part of the beam

l_b : Length of counter-weight beam

l_{cb} : Distance between the counterweight and the joint

r_{ms} : Radius of the main shield

r_{ts} : Radius of the tail shield

α_h : Horizontal position of TRMS beam

α_v : Vertical position of TRMS beam

Ω_h : Angular velocity of TRMS beam in horizontal plane

Ω_v : Angular velocity of TRMS beam in vertical plane

J_v : Moment of inertia about horizontal axis

$T_{fric,v}$: Torque of the friction force in vertical plane

$T_{fric,h}$: Torque of the friction force in horizontal plane

$T_{cable}(\alpha_h)$: Torque of the flat cable force

u_v : input voltage of the DC motor in vertical plane

T_{mr} : Time constant of the main rotor

K_{mr} : Static gain of the main DC motor

u_h : The input voltage of the DC motor in horizontal plane

T_{tr} : Time constant of the tail rotor

K_{tr} : The static gain tail DC motor

DOF: degree-of-freedom

MIMO: Multiple-Input Multiple-output

T1FS: type-1 fuzzy set

T2FS: type-2 fuzzy set

FOU: footprint of uncertainty

FLC: fuzzy logic controller

FLS: fuzzy logic system

IT2FLC: interval fuzzy logic controller

IT2 FS: Interval Type-2 Fuzzy set

IT1 FS: Interval Type-1 Fuzzy set

TRMS: Twin Rotor Mimo System

DC: direct current

μ_A (x): Membership degree (type-1)

μ_A (x , u): Membership degree (type-2)

TSK: Takagi–Sugeno

List of Contents

IV- Chapter 4: Control algorithm of Interval type-2 Fuzzy Logic for TRMS

INTRODUCTION

INTRODUCTION

Automatic control has played a vital role in the advance of engineering and science. In addition to its extreme importance in space-vehicle systems, missile-guidance systems, robotic systems, and the like, automatic control has become an important and integral part of modern manufacturing and industrial processes. For example, automatic control is essential in the numerical control of machine tools in the manufacturing industries, in the design of autopilot systems in the aerospace industries, and in the design of cars and trucks in the automobile industries And so on. Since advances in the theory and practice of automatic control provide the means for attaining optimal performance of dynamic systems, improving productivity, relieving the drudgery of many routine repetitive manual operations, and more, most engineers and scientists must now have a good understanding of this field.[15] So, what is control system? We use the word control to referrer to the act of producing a desired result [1.].

From that definition, the main purpose of control system is to develop a control law that provides a physical process of desired properties. To check the performance of developed control law, a first approach is to test the validity of the latter on the process itself. This technique can be dangerous and sometimes even impossible to implement, such as in the case of spatial structures, nuclear, etc. so the solution of that is to design an accurate mathematical model that describes a system completely, in order to analyze a dynamic system or the simulation of performance obtained in closed loop. The derivation of this model base upon the fact that the dynamic system can be completely described by known differential equations or by experimental test data. The ability to analyze the system and determine its performance depends on how well the characteristics can be expressed mathematically [4.].

In general, real systems are essentially nonlinear uncertain and are subject to external disturbances, and structured and unstructured dynamical uncertainties, and external disturbances, are among the typical challenges to be faced. The modeling of these systems is very often. Thus the controller must be so robust in the sense that it will provide a low sensitivity to uncertainties on the parameters, their variations and disturbances. One of the well-known nonlinear control methods using differential geometry, is exact linearization by the control such that the feedback control state. The latter is sensitive to parametric variations, and also several studies have shown that failing to compensate for modeling uncertainties in controlling flexible structures can have negative consequences, such as severe tracking errors, limit cycles, chattering, and excessive noise [14]. Like the helicopters are governed by complex dynamics and hence are inevitably subject to the ubiquitous presence of high, particularly unstructured, modeling nonlinearities, so that simulation model (TRMS), such as friction force and external disturbances, for instance. Thus, modeling the system's dynamics based on presumably accurate mathematical models might lead to undesirable consequences in this case. This raises the urgency to consider alternative approaches for the control of this type of systems (e.g TRMS) to keep up with their increasingly demanding design requirements.[14]

On another aspect, tools of computational intelligence, such as artificial neural networks and fuzzy logic controllers, have been credited in various applications as powerful tools capable of providing robust controllers for mathematically ill-defined systems that may be subjected to structured and unstructured uncertainties.[14] Despite the success witnessed by neural network-based control systems, they remain incapable of incorporating any human-like expertise already acquired about the dynamics of the system in hand, which is considered one of the main weaknesses of such soft computing methodologies. [4]

Type-1 fuzzy logic systems (FLSs) are known for their ability to compensate for structured and unstructured uncertainties, to a certain degree. However, type-2 fuzzy engines have been credited to be more powerful in compensating for even higher degrees of uncertainties. [4] They are particularly suitable for time-variant systems with unknown time-varying dynamics. They also allow for more flexibility to alleviate the problems associated to the uncertainties pertaining to the choice of the system's fuzzy rules and fuzzy membership functions.

The present work capitalizes on the merits and the latest developments of type-2 fuzzy logic theory to implement a type-2 FLC for the control of a helicopter flight simulator (Twin Rotor MIMO system), with uncertain dynamics.

Since conventional type-1 fuzzy FLSs can be used to identify the behavior of this highly nonlinear system with various types of uncertainties but cannot fully capture the uncertainties in the system due to membership functions and knowledge base imprecision [4].and the computational complexity of operations on fuzzy sets increases with the increasing type of the fuzzy set. Therefore, we will use in the main of this work the interval type-2 fuzzy sets for their simplicity and efficiency to capture the severe uncertainties and nonlinearities of TRMS.

The rest of the papers is organized as follows: Chapter-1 outlines the dynamical model of TRMS simulator "Twin Rotor MIMO System". Chapter-2 since we cannot jump directly to T2FLC, Firstly, We have to illustrate conventional type-1 fuzzy logic controller and its theory. Chapter-3 T2FLC and IT2FLC with its basic theory. Chapter-4 dedicated to use Matlab to design IT2FLC to control TRMS and simulation results are reported and discussed and compare with T1FLC result. Finally, we conclude with a general conclusion.

CHAPTER ONE

Dynamic model of TRMS

I.1- Introduction:

Similar to most flight vehicles, the helicopter consists of several elastic parts such as rotor, engine and control surfaces. The nonlinear aerodynamic forces and gravity act on the vehicle, and flexible structures increase complexity and make a realistic analysis difficult. For control purpose, it is necessary to find a representative model that shows the same dynamic characteristics as the real aircraft. The behavior of a nonlinear TRMS (Figure I.1), in certain aspects resembles that of a helicopter. It can be well perceived as a static test rig for an air vehicle with formidable control challenges. This TRMS consists of a beam pivoted on its base in such a way that it can rotate freely in both its horizontal and vertical planes. There are two rotors (the main and tail rotors), driven by DC motors, at each end of the beam. If necessary, either or both axes of rotation can be locked by means of two locking screws provided for physically restricting the horizontal or vertical plane rotation. Thus, the system permits both 1 and 2 degree-of-freedom (DOF) experiments. The two rotors are controlled by variable speed electric motors enabling the helicopter to rotate in a vertical and horizontal plane (pitch and yaw). The tail rotor could be rotated in either direction, allowing the helicopter to yaw right or left. The motion of the helicopter was damped by a pendulum, which hung from a central pivot point. In a typical helicopter, the aerodynamic force is controlled by changing the angle of attack of the blades. The mathematical model of the TRMS is developed under following two parts with its Figure I.1. [3]

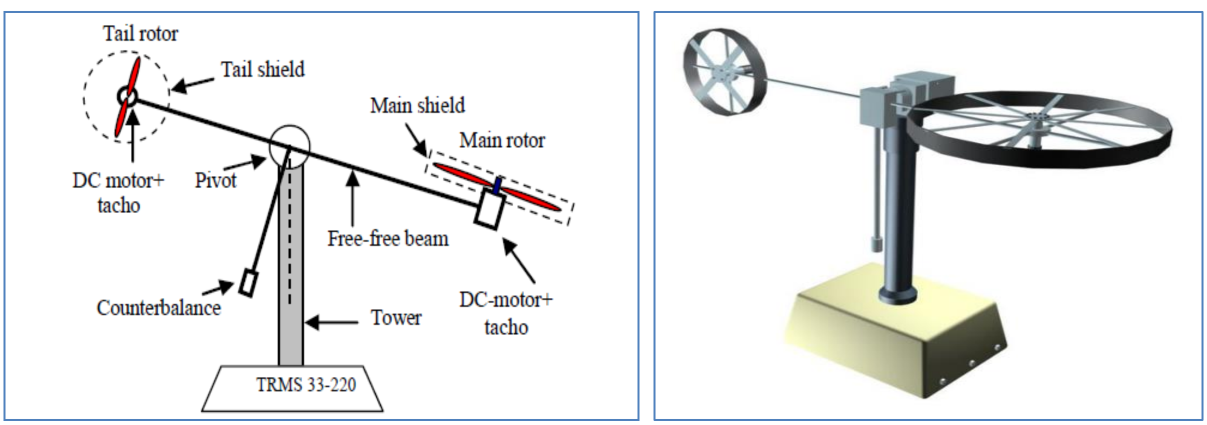


Figure I.1: The twin rotor MIMO system

I.2- 1 DOF TRMS modeling in vertical plane:

The TRMS possesses two permanent magnet DC motors; one for the main and the other for the tail propelling. The motors are identical with different mechanical loads. [4] So the mathematical model of the system in vertical plane is described in (I.1) to (I.4) (see Figure I.2).

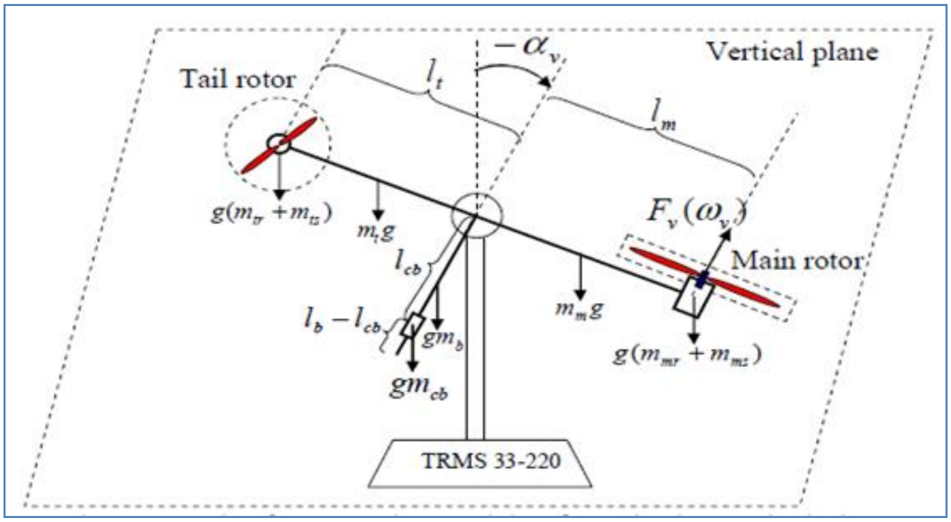


Figure I.2: Gravity forces and propulsive force in the vertical plane

In (I.1) the first term denotes the torque of the propulsive force due to the main rotor, the second term refers to the torque of the friction force, and the torque of gravity force is shown in the third term. [4]

$$J_v \frac{d\Omega_v}{dt} = l_m F_v(w_v) - T_{fric,y} + g[(A - B)\cos\alpha_v - C\sin\alpha_v] \quad (I.1)$$

Where:

$$A = \left(\frac{m_t}{2} + m_{tr} + m_{ts}\right) l_t$$

$$B = \left(\frac{m_m}{2} + m_{mr} + m_{ms}\right) l_m$$

$$C = \left(\frac{m_b}{2} l_b + m_{cb} l_{cb}\right)$$

$$\frac{d\alpha_v}{dt} = \Omega_v$$

The propulsive force F_v moving the joined beam in the vertical direction is describing by a nonlinear function of the angular velocityw_v. [5]

$$F_v(w_v) = -3.48 \times 10^{-12} w_v^5 + 1.09 \times 10^{-9} w_v^4 + 4.123 \times 10^{-6} w_v^3 - 1.632 \times 10^{-4} w_v^2 + 9.54410^{-2} w_v \quad (I.2)$$

The angular velocity w_v of main propeller is a nonlinear function of a rotation angle of the DC motor describing by: [5].

$$w_v(u_{vv}) = 90.90u_{vv}^6 + 599.73u_{vv}^5 - 129.26u_{vv}^4 - 1238.64u_{vv}^3 + 63.45u_{vv}^2 + 1238.41u_{vv} \quad (I.3)$$

The model of the motor-propeller dynamics is obtained by substituting the nonlinear system by a serial connection of a linear dynamics system. This can be expressed as: [5]

$$\frac{du_{vv}}{dt} = \frac{1}{T_{mr}}(-u_{vv} + u_v) \quad (I.4)$$

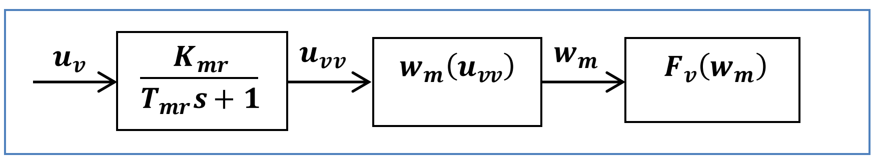


Figure I.3: The relationship between the input voltage and the propulsive force for the main rotor

I.3- 1 DOF TRMS modeling in horizontal plane:

As it has been mentioned before, the motors for horizontal and vertical movement are identical with different mechanical loads. So, all the related equations are same and, therefore, are not repeated here. The mathematical model of the remaining parts of the system in horizontal plane is described in (I.3) to (I.5) (see Figure I.4). In (I.3) the first term is the torque of propulsive force due to the tail rotor, the second term implies the torque of the friction force, and the third term refers to the torque of the flat cable force that is completely nonlinear and can be obtained by point by point measurement. [4]

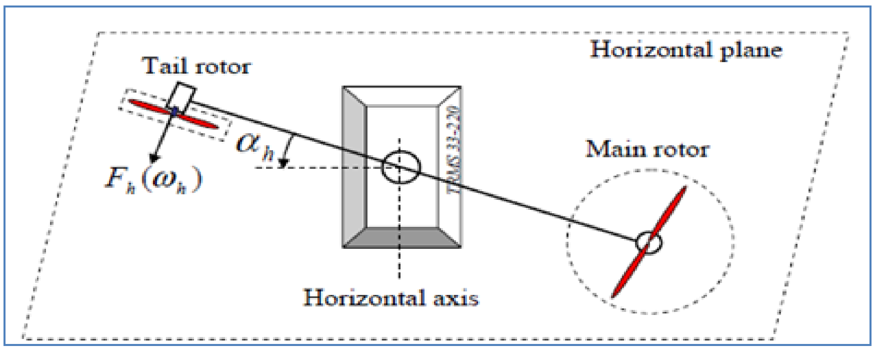


Figure I.4: Propulsive force in the horizontal plane

$$\frac{d\Omega_h}{dt} = \frac{l_t F_v(w_h)\cos\alpha_v - T_{fric,h} - T_{cable}(\alpha_h)}{D\cos^2\alpha_v + E\sin^2\alpha_v + F} \qquad (I.5)$$

Where:

$$\alpha_v = cte$$

$$D = \left(\frac{m_m}{3} + m_{mr} + m_{ms}\right){l^2}_m + \left(\frac{m_t}{3} + m_{tr} + m_{ts}\right){l^2}_t$$

$$E = \frac{m_b}{3}{l^2}_b + m_{cb}{l^2}_{cb}$$

$$F = \frac{m_{ts}}{2}{r^2}_{ts} + m_{ms}{r^2}_{ms}$$

$$\frac{d\alpha_h}{dt} = \Omega_h$$

Also, the propulsive force F_h moving the joined beam in the Horizontal direction is describing by a nonlinear function of the angular velocity w_h.[5]

$$F_h(w_h) = -3\times10^{-14}w_h^5 + 1.595\times10^{-11}w_h^4 + 2.511\times10^{-7}w_h^3 - 1.808\times10^{-4}w_h^2 + 0.8080\,w_h \qquad (I.6)$$

Also The angular velocity w_v of tail propeller is a nonlinear function of a rotation angle of the DC motor describing by: [5]

$$w_h(u_{hh}) = 2020u_{hh}^5 + 194.69u_{hh}^4 - 4283.15u_{hh}^3 - 262.87u_{hh}^2 + 3796.83u_{hh} \tag{I.7}$$

The model of the motor-propeller dynamics is obtained by substituting the nonlinear system by a serial connection of a linear dynamics system. This can be expressed as: [3]

$$\frac{du_{hh}}{dt} = \frac{1}{T_{tr}}(-u_{hh} + u_h) \tag{I.8}$$

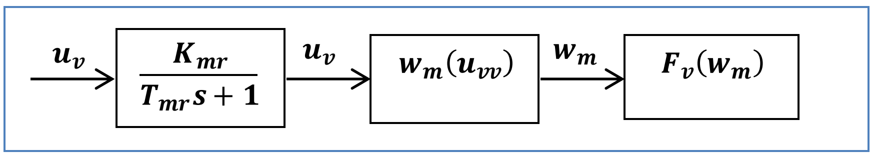


Figure I.5: The relationship between the input voltage and the propulsive force for the tail rotor

Where :

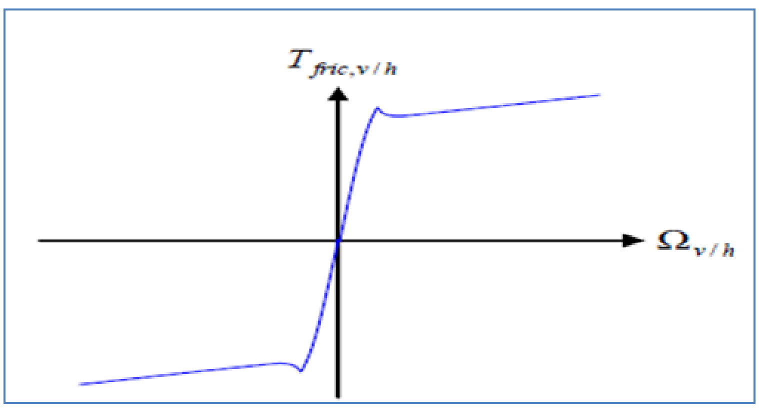


Figure I.6: The torque of the friction force

Figure I.6 shows the torque of the friction force that covers viscous, coulomb and static frictions. It must be noted that there are differences between the input voltage levels in the MATLAB/Simulink environment and the motor terminal voltages, and the relationship between these two sets of values is nonlinear.[4]

The plant constants can be seen in Table I.1

Parameter	Numerical Value
m_{tr}	**0.206 [kg]**
m_{mr}	**0.228[kg]**
m_{cb}	**0.068 [kg]**
m_t	**0.0155 [kg]**
m_m	**0.0145 [kg]**
m_b	**0.022 [kg]**
m_{ts}	**0.165 [kg]**
m_{ms}	**0.225[kg]**
l_t	**0.25 [m]**
l_m	**0.25 [m]**
l_b	**0.26 [m]**
l_{cb}	**0.13 [m]**
r_{ms}	**0.155 [m]**
r_{ts}	**0.10 [m]**
T_{mr}	**1.432 sec**
T_{tr}	**0.3842 sec**
K_{mr}	**1**
K_{tr}	**1**
g	**9.81 $[m/s^2]$**

Table I.1: Parameter Definitions of the TRMS

I.4- Conclusion:

In this chapter, we have illustrated the general overview of real bi-rotor helicopter, and we provided simulation model «TRMS», also through the analysis of that model, we showed that is multivariable, couple, has two degree-of-freedom (DOF), and nonlinear equation in the both vertical and horizontal planes, so that TRMS has uncertain dynamic, in addition, to the effect of The torque of the friction force which cause an addition uncertainty. So by that chapter the analytic model of simulation model is provided.

With all that complexity its «TRMS» control by ordinary controller come more difficult, so for that raison in the next chapter we will use another intelligent controller based on fuzzy logic which can really cope with all that unstructured uncertainties

CHAPTER TWO

Type 1 fuzzy logic

II.1- Introduction:

Fuzzy sets originated in the year 1965 and this concept was proposed by Lofti A.Zadeh. Since then it has grown and is found in several application areas. According to Zadeh, The notion of a fuzzy set provides a convenient point of departure for the construction of a conceptual framework which parallels in many respects of the framework used in the case of ordinary sets, but is more general than the latter and potentially, may prove to have a much wider scope of applicability, specifically in the fields of pattern classification and information processing." Fuzzy logics are multi-valued logics that form a suitable basis for logical systems reasoning under uncertainty or vagueness that allows intermediate values to be defined between conventional evaluations like true/false, yes/no, high/low, etc. These evaluations can be formulated mathematically and processed by computers, in order to apply a more human-like way of thinking in the programming of computers. Fuzzy logic provides an inference morphology that enables approximate human reasoning capabilities to be applied to knowledge-based systems. The theory of fuzzy logic provides a mathematical strength to capture the uncertainties associated with human cognitive processes, such as thinking and reasoning. Fuzzy systems are suitable for uncertain or approximate reasoning, especially for the system with a mathematical model that is difficult to derive. Fuzzy logic allows decision making with estimated values under incomplete or uncertain information. [6]

II.2- Fuzzy Sets:

In the classical set, its characteristic function assigns a value of either 1 or 0 to each individual in the universal set, there by discriminating between members and nonmembers of the crisp set under consideration. The values assigned to. [7]

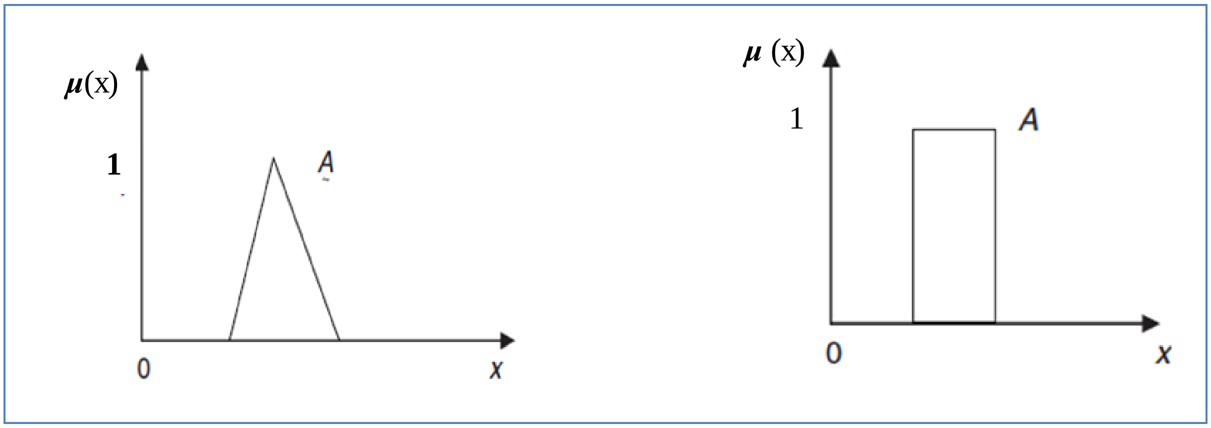


Figure II.1: Membership function of fuzzy set and crisp set

The elements of the universal set fall within a specified range and indicate the membership grade of these elements in the set. Larger values denote higher degrees of set membership such a function is called a membership function and the set is defined by it is a fuzzy set. [7]

A fuzzy set is thus a set containing elements that have varying degrees of membership in the set. This idea is in contrast with classical or crisp, set because members of a crisp set would not be members unless their membership was full or complete, in that set (i.e., their membership is assigned a value of 1). [7] Elements in a fuzzy set, because their membership need not be complete, can also be members of other fuzzy set on the same universe. Fuzzy set are denoted by a set symbol with a tilde under strike. Fuzzy set is mapped to a real numbered value in the interval 0 to 1. If an element of universe, say x, is a member of fuzzy set A, then the mapping is given by $\mu_A(x) \in [0, 1]$. So fuzzy set A fuzzy set on

U is defined as

$$\mu_A(x) : U \rightarrow [0, 1] \qquad \text{(II.1)}$$

Here μ_A is known as the membership function, and $\mu_A(x)$ is known as the membership grade of x. Membership function is the degree of truth or degree of compatibility. The membership function is the crucial component of a fuzzy set. Therefore all the operations on fuzzy sets are defined based on their membership functions. [6]

II.3- Fuzzy Set Operations:

Fuzzy sets are used for the systematic manipulation of vague and imprecise concepts using fuzzy set operations performed by manipulating the membership functions. Let A and B be two point-valued fuzzy sets in universe of discourse U with membership functions $\mu_A(u)$ and $\mu_B(u)$respectively.

Equality $\mu_A(u)= \mu_B(u)$for all $u \in U$.

Sets A and B are equal if they are defined on the same universe and the membership function is the same for both.

in the TableII.1 below illustrate fuzzy set operations

Operation	Fuzzy logic form
Complement	$\mu_{\overline{A}}(u) = 1 - \mu_A(u)$
Union	$\mu_{A\cup B}(u) = max(\mu_A(u), \mu_B(u)) \qquad u \in U$
Intersection	$\mu_{A\cap B}(u) = \min(\mu_A(u), \mu_B(u)) \qquad \mu \in U$
Produit cartesien	$\mu_{A_1\times \ldots\ldots \times A_n}(\mu_1, \ldots, \mu_n) = \min(\mu_{A_1}(u_1), \ldots, \mu_{A_n}(u_n))$ ou $\mu_{A_1\times \ldots \times A_n}(u_1, \ldots, u_n) = \mu_{A_1}(u_1) \times \ldots \times \mu_{A_n}(u_n)$
Fuzzy Relation	$R_{U1,\ldots\ldots,Un} = [((U1, \ldots\ldots Un), \mu_R(\mu_1 \ldots . \mu_n))(\mu_1, .., \mu_n) \epsilon U1 \times \ldots \times Un]$

TableII.1: summarize of Fuzzy Set Operations

Note: If A and $\overline{A}$ are complements, their intersection need not be empty set. Likewise, their union is not necessarily equal to the universe

II.4- Linguistic Variables:

Just like an algebraic variable takes numbers as values, a linguistic variable takes words or sentences as values. The set of values that it can take is called its term set. Each value in the term set is a fuzzy variable defined over a base variable. The base variable defines the universe of discourse for all the fuzzy variables in the term set. In short, the hierarchy is as follows: linguistic variable −→ fuzzy variable −→ base variable.[6]

Illustration:

Let x be a linguistic variable with the label "Age". Terms of this linguistic variable, which are fuzzy sets, could be "old", "young", "very young" from the term set.

T = Old, Very Old, Not So Old, More or Less Young, Quite Young, Very Young Each term is a fuzzy variable defined on the base variable, which might be the scale from 0 to 100 years. [4]

A linguistic variable is represented by triplet (x, T(x), U)

x : Is the name of the linguistic variable (position, speed, angle error …….)

T(x) : Is the set of the fuzzy sets, which is used to define x U: is the universe of discourse which related to linguistic variable x

For example, if the angle is linguistic variable and defined in universe of discourse U= [-1, 1], its fuzzy labels can be :

Negative Big, (NB), Negative Small (NS), Zero (ZR), Positive Small (PS), positive Big (PB)

So the fuzzy sets are:

T (error) = {Negative Big, (NB), Negative Small (NS), Zero (ZR), Positive Small (PS), positive Big (PB)}

The Figure II.2: below depicts these conceptions

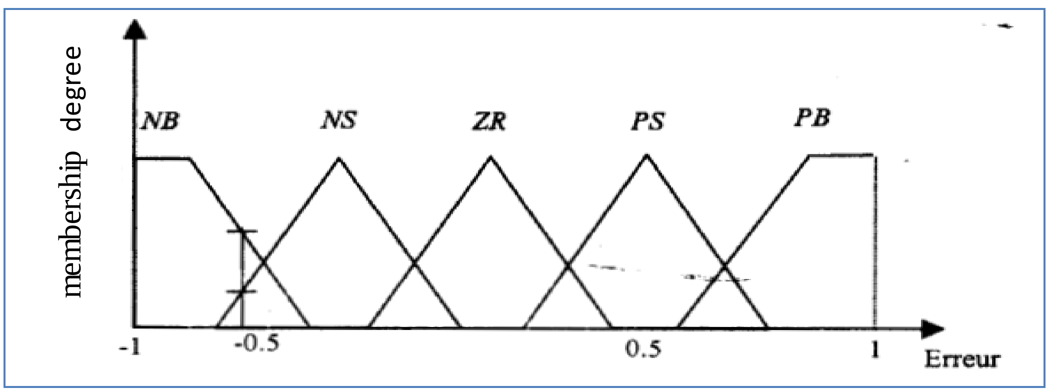


Figure II.2: depicts linguisticts variable 'error' and its fuzzy sets

II.5- Membership Functions:

The membership function $\mu_A(x)$ describes the membership of the element x of the base set X in the fuzzy set A, whereby for $\mu_A(x)$ a large class of functions can be taken. Reasonable functions are often piecewise linear functions, such as triangular or trapezoidal functions. The grade of membership $\mu_A(x_o)$ of a membership function $\mu_A(x)$ describes for the special element x=xo, to which grade it belongs to the fuzzy set A. This value is in the unit interval [0,1]. Of course, xo can simultaneously belong to another fuzzy set B, such that $\mu_B(x_o)$ characterizes the grade of membership of xo to B. This case is shown in Figure II.3. [6]

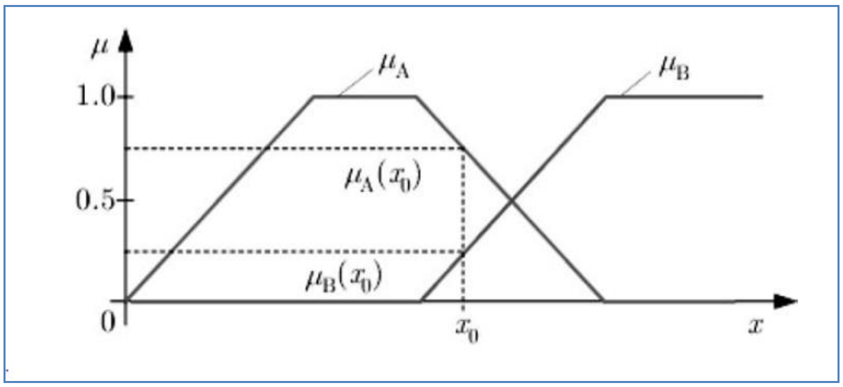


Figure II.3: Membership Grades of x0 in the Sets A and B:

$\mu_{\boldsymbol{A}}(\boldsymbol{x_0}) = \mathbf{0.75}$ and $\mu_{\boldsymbol{B}}(\boldsymbol{x_0}) = \mathbf{0.25}$

The membership for a 50-year old in the set “young” depends on one’s own view. The grade of membership is a precise, but subjective measure that depends on the context. A fuzzy membership function is different from a statistical probability distribution. This is illustrated following egg-eating example. [6]

II.5.1- Types of Membership Functions:

In principle any function of the form A: X→ [0 , 1] describes a membership function associated with a fuzzy set A that depends not only on the concept to be represented, but also on the context in which it is used. The graphs of the functions may have different shapes and may have specific properties. Whether a particular shape is suitable can be determined only in the application context. In certain cases, however, the meaning semantics captured by fuzzy sets is not too sensitive to variations in the shape, and simple functions are convenient. In many practical instances fuzzy sets can be represented explicitly by families of parameterized functions, the most common being the following: [6]

1. Triangular Function
2. Generalized bell Function
4. Trapezoidal Function
5. Gaussian Function

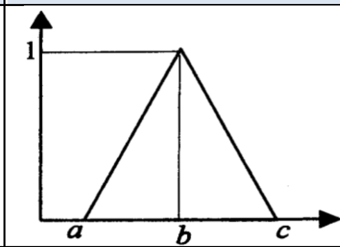
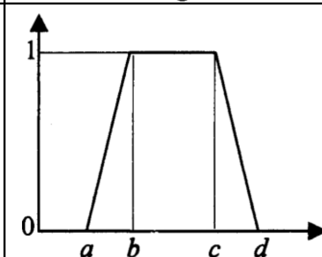
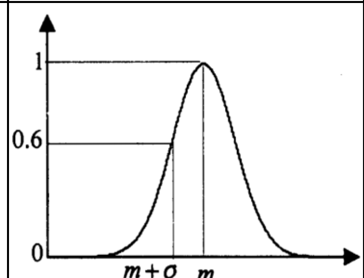
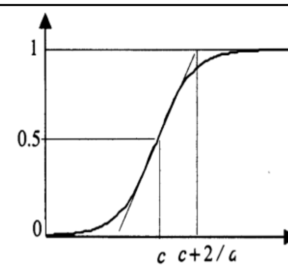
Function	Algebraic form	Graphic form
Triangular Function	it is defined by three parameters $\{a, b, c\}$ $$\mu(x) = \boldsymbol{max}\left(\boldsymbol{min}\left(\frac{\boldsymbol{x-a}}{\boldsymbol{b-a}}, \frac{\boldsymbol{c-x}}{\boldsymbol{c-b}}\right), \boldsymbol{0}\right)$$	
Trapezoidal Function	it is defined by four parameters $\{a, b, c, d\}$ $$\mu(x) = \boldsymbol{max}\left(\boldsymbol{min}\left(\frac{\boldsymbol{x-a}}{\boldsymbol{b-a}}, \boldsymbol{1}, \frac{\boldsymbol{d-x}}{\boldsymbol{d-c}}\right), \boldsymbol{0}\right)$$	
Gaussian Function	It is defined by two parameters $\{m, \sigma\}$ $$\mu(x) = \boldsymbol{exp}\left(-\frac{(\boldsymbol{x-m})^2}{\boldsymbol{2\sigma^2}}\right)$$	
Sigmoïdal Function	It defined by two parameters $\{a, c\}$ $$\mu(x) = \frac{\boldsymbol{1}}{\boldsymbol{1} + \boldsymbol{exp}(-\boldsymbol{a}(\boldsymbol{x-c}))}$$	

Table II.2: Types of Membership Functions

II.6- Fuzzy Controllers

Fuzzy logic controllers are based on the combination of Fuzzy set theory and fuzzy logic. Systems are controlled by fuzzy logic controllers based on rules instead of equations. This collection of rules is known as the rule base usually in the form of IF-THEN-ELSE statements. Here the IF part is known as the antecedent and the THEN part is the consequent. The antecedents are connected with simple Boolean functions like AND, OR, NOT etc., Figure II.4 outlines a simple architecture for a fuzzy logic controller. The outputs from a system are converted into a suitable form by the fuzzification block. Once all the rules have been defined based on the application, the control process starts with the computation of the rule consequences. The computation of the rule consequences takes place within the computational unit. Finally, the fuzzy set is defuzzified into one crisp control action using the defuzzification module. The decision parameters of the fuzzy logic controller are as follows: [6]

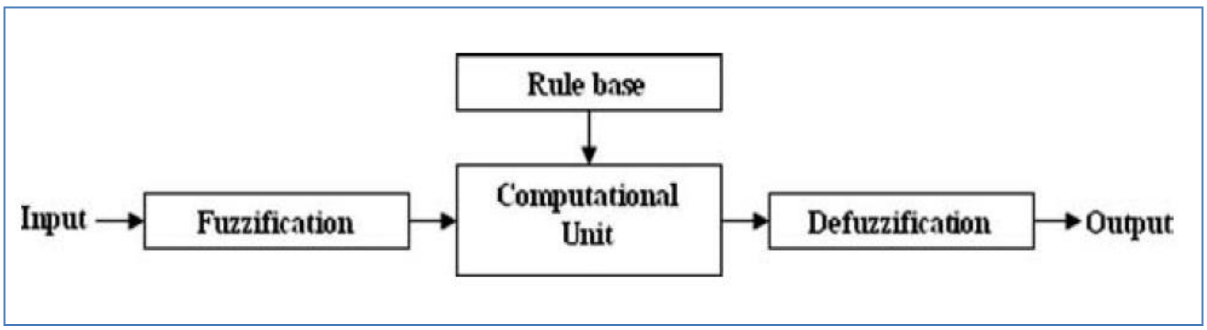


Figure II.4: Fuzzy Controller Architecture

Input Unit: Factors to be considered are the number of input signals and scaling of the input signal.

II.6.1- Fuzzification:

The process by which the input values from sensors are scaled and mapped to the domain of fuzzy variables is known as fuzzification. The fuzzy variables also known as linguistic variables are determined based on intuition (from knowledge) or inference (known facts). These linguistic variables can be either continuous or discrete theoretically, but in practice it should be discrete. Fuzzification is a two step process: Assign fuzzy labels and Assign numerical meaning to each label.[6]

a- Assign fuzzy label:

Each crisp input is assigned a fuzzy label in the universe of discourse. For example for the input parameter height fuzzy labels can be “tall”, “short”, “Normal”, “Very Tall”, and “Very short”. Every crisp input can be assigned multiple labels. As the number of labels increases the resolution of the process is better. In some cases, assigning large number of labels leads to a large computational time and thus making the fuzzy system unstable. Therefore in general the number of labels for a system is limited to an odd number in the range [3 , 9], such that the surface is balanced and symmetric.[6]

b- Assign numerical meaning:

Here membership functions are formed to assign a numerical meaning to each label. The range of the input value that corresponds to a specific label can be identified by the membership function. Though there are different membership function shapes, triangular and trapezoidal membership functions are commonly used to avoid time and space complexity. For each fuzzy set and for each linguistic variable, the grade of membership of a crisp measure in each fuzzy set is ascertained.[6] To illustrate this process assume the input variable e which can be swing angle of inverted pendulum is linguistic variable and its range or in another word universe of discourse is [-a , a]. Assume further that the following seven linguistic values (fuzzy labels) are selected: [6]

NL ---negative large NM ---negative medium NS ---negative small
PL ---positive large PM ---positive medium PS ---positive small
AZ ---approximately zero

Representing, for example, these linguistic variables by triangular-shape fuzzy numbers that are equally spread over each range, we obtain the fuzzy quantization exemplified for variable e in Figure II.5.

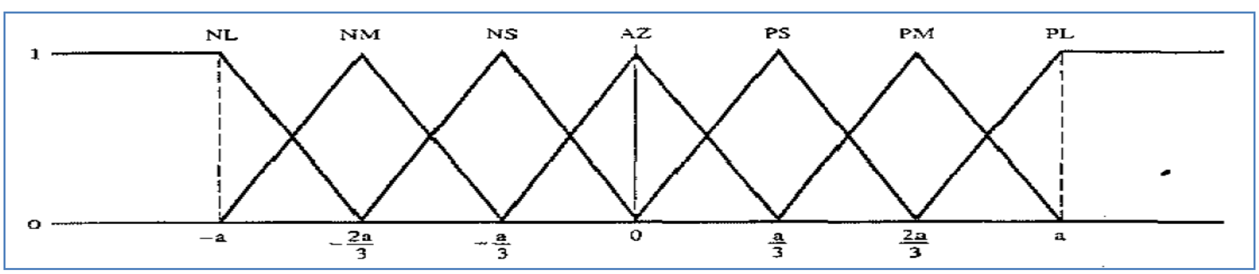


Figure II.5: Possible fuzzy quantization of the range [-a , a] bytriangular-shaped fuzzy numbers.

It is important to realize that the fuzzy quantization defined in figure 12 for range [-a , a] and the seven given linguistic labels are only reasonable example so other shapes of membership functions might be preferable to the triangular shapes. The shapes need not to be symmetric and need not be equally spraed over the given ranges.[8]

II.6.2- Fuzzy Rule Base

For any fuzzy logic operation, the output is obtained from the crisp input by the process of fuzzification and defuzzification. These processes involve the usage of rules, which form the basis to obtain the fuzzy output. A fuzzy if-then rule is also known as fuzzy rule, or fuzzy conditional statement or fuzzy implication. It is generally of the form: [6]

IF antecedent THEN consequent

IF (x is A) THEN (z is Z)

where x, z etc. represent the variables and A, Z are the linguistic values in the universe of discourse. Here the IF part is referred to as the antecedent or premise and the THEN part is referred to as consequent or conclusion. Fuzzy rules are most commonly applied to control systems. The common types of fuzzy rules applied to control systems are the Mamdani fuzzy rules and Takagi–Sugeno (TS) fuzzy rules.[6]

Most of the practical applications do not involve rules like the above mentioned with one antecedent part. These applications involve a compound rule structure. Such rules can be disintegrated into smaller rules and from which simple canonical rules can be formed. These rules have more than one antecedent part connected by conjunction and disjunction connectives. Conjunction connective uses intersection operation involving the "AND" connective as follows.[6]

IF $antecedent_1$ AND $antecedent_2$ AND ......AND $antecedent_n$ THEN consequent

Similarly the disjunction connective uses union operation involving the "OR" connective as follows

IF $antecedent_1$ OR $antecedent_2$ OR ....OR $antecedent_n$ THEN consequent

Likewise complex rules can be broken into simpler forms and connected using the "AND" or "OR" connectives as shown in the following rule. [6]

IF X1 = A1 and X2 = A2 THEN Y = B

IF X1 = A1 and X2 = A3THAN Y = B

Note: (Number of Control Rules), If the number of fuzzy sets or predicates for each input variable is m and the number of system input variables is n, then there are m^n different rules required for a completeness in a conventional system. FLC rule base typically uses a small number of rules to attain completeness. [9]

II.6.3- Aggregation of Rules:

The rule based system involves several rules and each rule provides an output or consequent. The consequent part also known as conclusion is unique for every rule that has been executed based on the input parameters. An overall conclusion has to be obtained from the individual consequents. This method of obtaining the overall conclusion from the set of rules is referred to as aggregation of rules. Fuzzy rules can be aggregated by using the "AND" or "OR" connectives. The process of aggregating the rules using "AND" connective is known as conjunctive aggregation and the process of aggregating the rules using "OR" connective is known as disjunctive aggregation. [6]

Conjunctive aggregation:

Consequent = Consequent_1 AND Consequent_2 AND ... AND Consequent_r

Disjunctive aggregation:

Consequent = Consequent_1 OR Consequent_2 OR ... OR Consequent_r

Using these operators a final decision is made on the output of the fuzzy set.

II.6.4- Fuzzy Inference Methods:

The most commonly used fuzzy inference methods are Mamdani's inference method, Takagi–Sugeno (TS) inference method and the Tsukamoto inference method. All these methods are similar to each other but differ only in their consequents. Mamdani fuzzy inference method uses fuzzy sets as the rule consequent while TS method uses functions of input variables as the rule consequent and the Tsukamoto inference method uses fuzzy set with a monotonically membership function as the rule consequent. [6]

a- Mamdani's Fuzzy Inference Method:

Among the above mentioned inference methods, the Mamdani model is the commonly used method due to its simple min-max structure. The model was proposed by Mamdani (1975), The fuzzy rules are of the form: [7]

IF (input_1 is Linguistic variable_1) AND OR (input_2 is Linguistic variable_2) AND OR ....THEN (output is Linguistic variable_n) [7]

Example :

If x is A1 and y is B1 then z is C1

If x is A2 and y is B2 then z is C2

For this example as a Mamdani inference system is shown in Figure II.6.

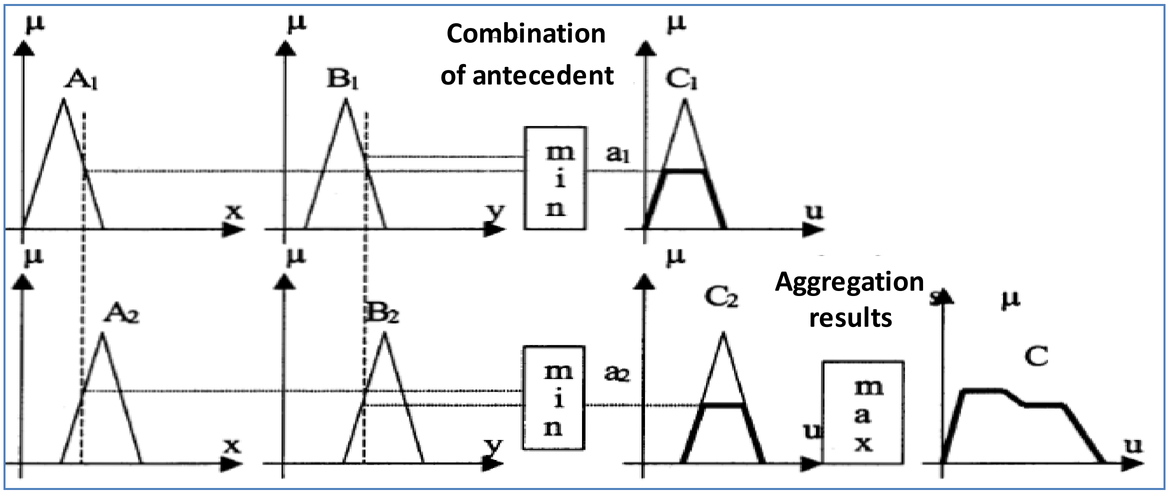


Figure II.6: Example of Mamdani inference system (MAX-MIN)

b- Takagi–Sugeno Fuzzy Inference Method:

The Takagi–Sugeno model also known as TS method was proposed by Takagi and Michio Sugeno in 1985 in order to develop a systematic approach to generate fuzzy rules. The Sugeno type fuzzy inference is similar to the Mamdani inference, they differ from each other in their rule consequent as we mentioned

The general form of a TS rule is [4]

IF antecedent1 AND antecedent2 THEN output = f(x,y)

Here output = f (x,y) is a crisp function in the consequent. This mathematical function can either be linear or nonlinear. Most commonly linear functions are used and adaptive techniques are used for nonlinear equations. The membership function of the rule consequent is a single spike or a singleton in the TS method. [6]

A few examples of the TS method of inference are

IF x is small THEN y = 3x−2

IF x is large THEN y = x + y + 5

A zero order Sugeno fuzzy model uses the rules of the form,

IF antecedent1 AND antecedent2 THEN output z = k

where k is a constant.

In such as case the output of each fuzzy rule will be a constant. The evaluation of fuzzy rules using Sugeno method is shown in Figure II.7.

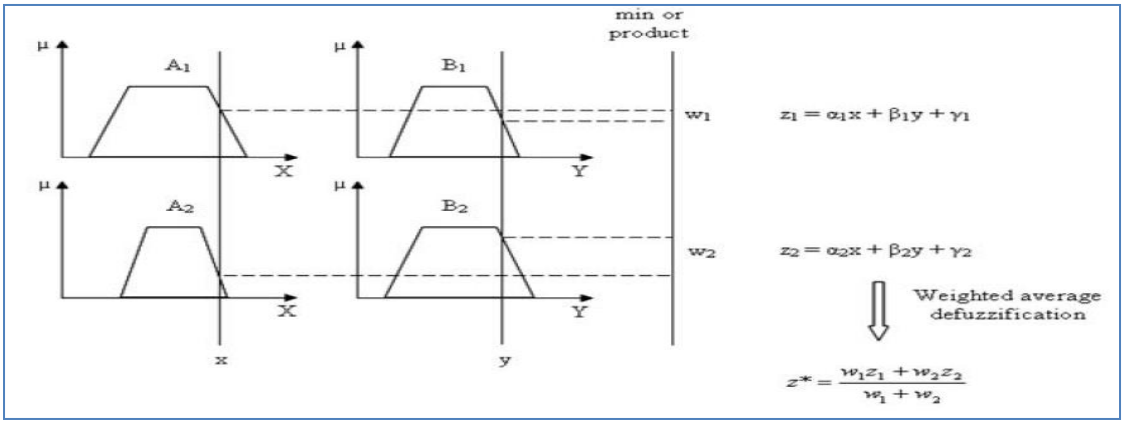


Figure II.7: The Scheme of Sugeno Inference Method

c- Tsukamoto Fuzzy Inference Method

In the Tsukamoto fuzzy model, the rule consequent is represented with a monotonically membership function. The general form of a Tsukamoto [6]

IF antecedent1 AND antecedent2 THEN output = membership function

This method also differs from the Mamdani and Sugeno in terms of its rule consequent. The Tsukamoto method of fuzzy inference in shown in Figure II.8. The output of each rule is defined as a crisp value induced by the firing strength of the rule. The Tsukamoto model also aggregates each of the rule's output using weighted average method of defuzzification thereby reducing the time consumed for the process of defuzzification. [8]

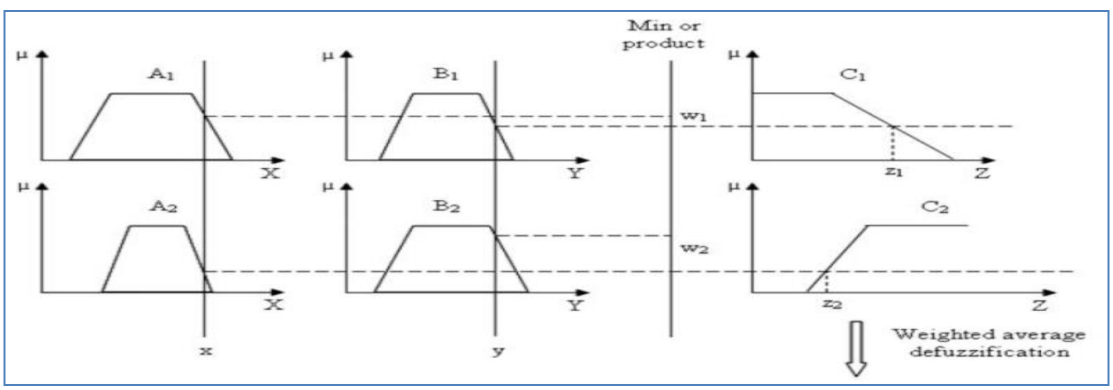


Figure II.8: The Scheme of Tsukamoto Inference Model

II.6.5- Defuzzification

As a result of applying the previous steps, one obtains a fuzzy set from the reasoning process that describes, for each possible value u, how reasonable it is to use this particular value. In other words, for every possible value u, one gets a grade of membership that describes to what extent this value u is reasonable to use. Using a fuzzy system as a controller, one wants to transform this fuzzy information into a single value u' that will actually be applied. This transformation from a fuzzy set to a crisp number is called a defuzzification. The fuzzy results generated cannot be used as such to the applications, hence it is necessary to convert the fuzzy quantities into crisp quantities for further processing. This can be achieved by using defuzzification process by using the methods as follows: [6]

a- Max-Membership principle

- The max membership principle method finds the defuzzified value at which the membership function is a Maximum [6]

- This method of defuzzification is also referred to as the height method.

- The defuzzified value can be determined from the following expression

$\mu_A(Z^*) \geq \mu_A(Z)$

- Computes the defuzzified value at a very fast rate

- Very accurate only for peaked output membership functions

- Graphical representation of max-membership defuzzification shown in Figure II.9

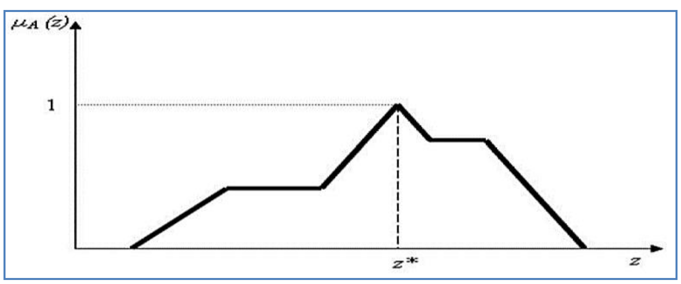


Figure II.9: Max membership method

b- Center of gravity method (COG)

- The COG method of defuzzification was developed by Sugeno in 1985

- This method is also known as center of area or centroid method

Most commonly used method

- Defined as z* = where z* $\frac{\int \mu A(z) z dz}{\int \mu A(z) z dz}$ is the defuzzified output, **μ**A(z) is the aggregated membership function and z is the output variable

- Capable of producing very accurate results
- Major disadvantage - computationally difficult for complex membership functions
- Graphical representation of COG method is shown in Figure II.10

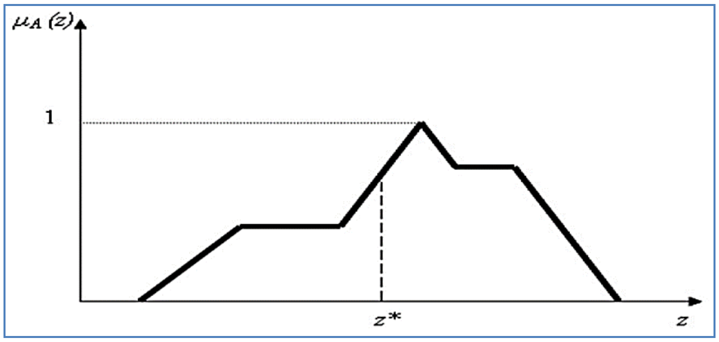


Figure II.10: Centroid defuzzification method

c- Weighted Average method

- In the weighted average method, the output is obtained by the weighted average of the each membership function output of the system. [6]
- This method can be applied only for symmetrical output membership functions
- Each membership function is weighted by its largest membership function- Defined as

where z* $\frac{\sum \mu A(z) z}{\sum \mu A(z)}$ is the defuzzified output, A(z) is the aggregated membership function z and is the weight associated with the membership function.

- The defuzzified value obtained in this method is very close to that obtained by COG method
- Overcomes the disadvantage of COG method - Less computationally intensive
- Graphical representation of Weighted Average method is shown in Figure II.11

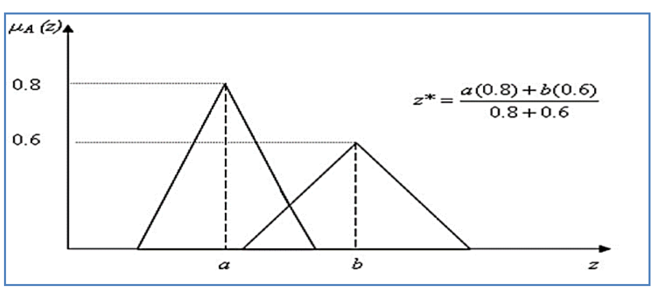


Figure II.11: Weighted Average method

II.8- Conclusions

In this chapter. we have illustrated the basic theory of fuzzy logic controller that is fuzzy sets and fuzzy logic which can be understood as an extension of ordinary sets, and we also explained different method of rule base, inference engine, and deffuzzification to obtain numerical output of the controller, as well as notions of linguistic variables and fuzzy values the inherent of membership function and its types for the antecedent and consequent, and all of that are used in the framework of tye-1 fuzzy logic controller.

T1FLC is powerful to handle the uncertainties more then the classical controller because of the uncertainty in the rule based due to linguistic variable which is based just on intuition of the experts, but it must has well-defined membership function so that causes some of difficulty, for that raison in the next chapter we will use another intelligent controller more powerful than type-1 fuzzy logic controller so-called type-2 fuzzy logic controller.

CHAPTER THREE

Type 2 fuzzy logic

III.1- Introduction to Type-2 Fuzzy Logic Control:

The Uncertainty is an inherent part of intelligent systems used in real-world applications [10]. The use of new methods for handling incomplete information is of fundamental importance [10]. And as Type-1 fuzzy controllers, like the ones mentioned above, whose membership functions are type-1 fuzzy sets, are unable to directly handle such uncertainties [10]. we will describe new method for building intelligent controller "type-2 fuzzy logic controller" using type-2 fuzzy logic in which the antecedent or consequent membership functions are type-2 fuzzy sets. Such sets are fuzzy sets whose membership grades themselves are type-1 fuzzy sets; they are very useful in circumstances where it is difficult to determine an exact membership function for a fuzzy set [10]. As well as Type-2 fuzzy sets that are used in type-2 fuzzy systems can handle such uncertainties in a better way because they provide us with more parameters [10].so we will describe in this chapter the design of intelligent controller using interval type-2 fuzzy logic for minimizing the effects of uncertainty produced by the instrumentation elements, environmental noise, etc.

III.2- Type-2 Fuzzy logic:

The basics of fuzzy logic do not change from type-1 to type-2 fuzzy sets [10], so type-2 fuzzy set, the concept of a type-2 fuzzy set, was introduced by Zadeh [10] as an extension of the concept of an ordinary fuzzy set (henceforth called a “type-1 fuzzy set”). A type-2 fuzzy set is characterized by a fuzzy membership function, i.e., the membership grade for each element of this set is a fuzzy set in [0,1], unlike a type-1 set where the membership grade is a crisp number in [0,1]. Such sets can be used in situations where there is uncertainty about the membership grades themselves, e.g., an uncertainty in the shape of the membership function or in some of its parameters. Consider the transition from ordinary sets to fuzzy sets. When we cannot determine the membership of an element in a set as 0 or 1, we use fuzzy sets of type-1. Similarly, when the situation is so fuzzy that we have trouble determining the membership grade even as a crisp number in [0, 1], we use fuzzy sets of type-2. This does not mean that we need to have extremely fuzzy situations to use type- 2 fuzzy sets [8]. There are many real-world problems where we can not determine the exact form of the membership functions, e.g., in time series prediction because of noise in the data. Another way of viewing this is to consider type-1 fuzzy sets as a first order approximation to the uncertainty in the real-world. Then type-2 fuzzy sets can be considered as a second order approximation. Of course, it is possible to consider fuzzy sets of higher types but the complexity of the fuzzy system increases very rapidly [10].

III.3- general Type-2 Fuzzy Set:

A general type-2 fuzzy set, $\tilde{A}$, may be represented as [9] [10]:

$$\mu_{\tilde{A}}(x,u) : X \rightarrow [0, 1] \qquad \text{(III.1)}$$

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) | \quad \forall x \in X \quad \forall u \in J_x \subseteq [0,1] \} \qquad \text{(III.2)}$$

Where $\mu_{\tilde{A}}(x,u)$ is the type-2 fuzzy membership function in which $0 < \mu_{\tilde{A}}(x,u) > 1$.

$\tilde{A}$ can also be defined as [2]:

$$\tilde{A} = \int_{x\in X}\int_{u\in J_x} \mu_{\tilde{A}}(x,u)/(x,u) \qquad J_x \subseteq [0,1] \qquad \text{(III.3)}$$

Where $\int\ \int$ denotes union over all admissible x and u [11], For discrete universes of discourse $\int$ is replaced by Σ.

J_x is called primary membership of x [11]. Additionally, there is a secondary membership value corresponding to each primary membership value that defines the possibility for primary memberships [11]. Whereas the secondary membership functions can take values in the interval of [0,1] in generalized T2FLSs [11], so, type-2 fuzzy set is one in which the membership grade of every domain point is a Gaussian type-1 set contained in [0,1].as depicted in Figure III.1.[10]

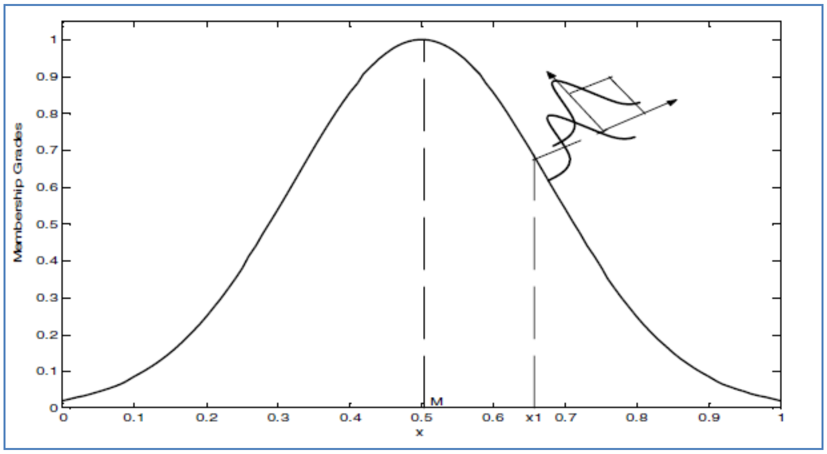


Figure III.1: A type-2 fuzzy set in which the membership grade of every domain point is a Gaussian type-1 set.

Uncertainty in the primary memberships of a type-2 fuzzy set, Ã, consists of a bounded region that we call the "footprint of uncertainty" (FOU), (black region in Figure III.1). Mathematically, it is the union of all primary membership functions. [10]

$$FOU\ (\tilde{A}) = \cup_{x \in X} u \ \in J_X \qquad \text{(III.1)}$$

An "upper membership function" and a "lower membership functions" (Figure III.2) are two type-1 membership functions that are bounds for the FOU of a type-2 fuzzy set Ã. The upper membership function is associated with the upper bound of FOU (Ã) [10] is denoted $\overline{u}_{\tilde{A}}(x)$.[11]. and The lower membership function is associated with the lower bound of FOU (Ã) [10] and is denoted $\underline{u}_{\tilde{A}}(x)$ [13]:

$$\overline{u}_{\tilde{A}}(x) = \overline{FOU(\tilde{A}\,)}\ \ \forall x \in X$$
$$\underline{u}_{\tilde{A}}(x) = \underline{FOU(\tilde{A}\,)}\ \ \forall x \in X \qquad \text{(III.5)}$$

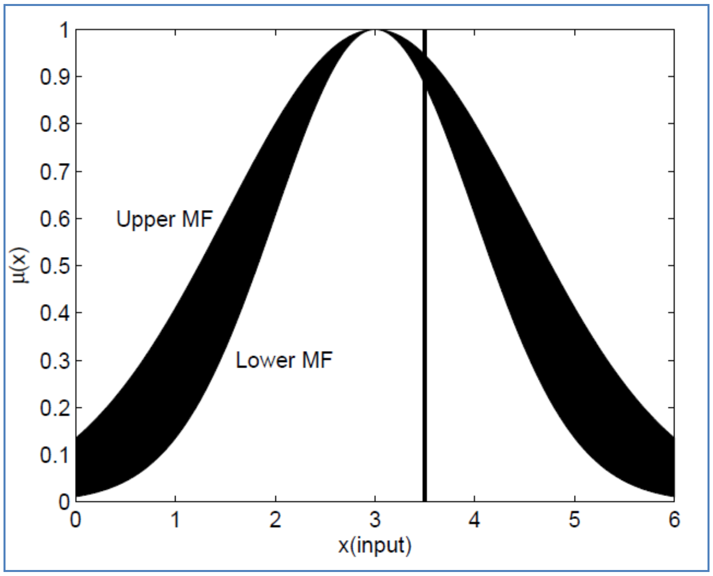


Figure III.2: A Gaussian type-2 fuzzy membership function (FOU), upper and lower membership function

Another way of viewing type-2 membership functions is in a three-dimensional fashion, in which we can better appreciate the idea of type-2 fuzziness. In Figure III.3 we have a three-dimensional view of a type-2 Gaussian membership function [10].

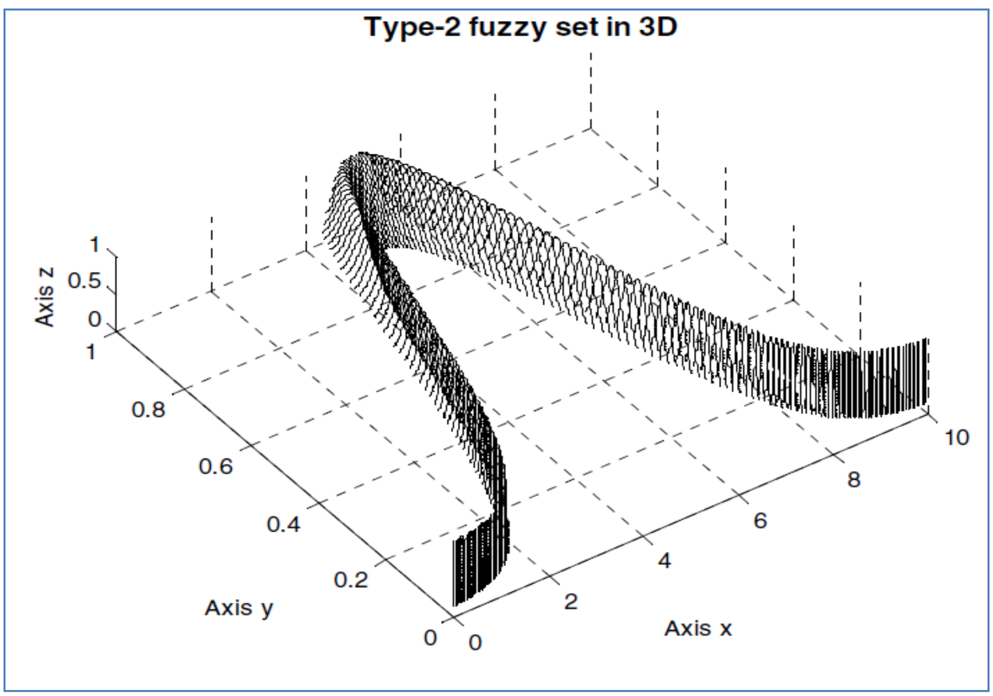


Figure III.3: Three-dimensional view of a type-2 membership function

III.4- Interval Type-2 Fuzzy Sets:

The operations of type-2 fuzzy systems are typically more computationally involved than type-1 systems. This has urged researchers to search for ways to alleviate this high computational burden if type-2 FLCs are to find their way to real-world applications. For this purpose, interval fuzzy sets were introduced [13]. This type of fuzzy sets provides a simplified and efficient alternative to easily compute the input and antecedent operations for FLSs and offers a balanced trade off between performance and complexity [15].

when all secondary membership $\mu_{\tilde{A}}(x,u)$ of $\tilde{A}$ are equal to 1, then $\tilde{A}$ is Interval Type-2 Fuzzy Set, the special case of Equation (III.6) , might be defined for the Interval Type-2 Fuzzy Set as [11]:

$$\tilde{A} = \int_{x\in X}\int_{u\in J_x} 1/(x,u) \qquad J_x \subseteq [0,1] \qquad \text{(III.6)}$$

Both, the general and interval type-2 fuzzy membership functions are three-dimensional (Figure III.4), in this dissertation, the research activities are focused on the interval type-2 membership function.

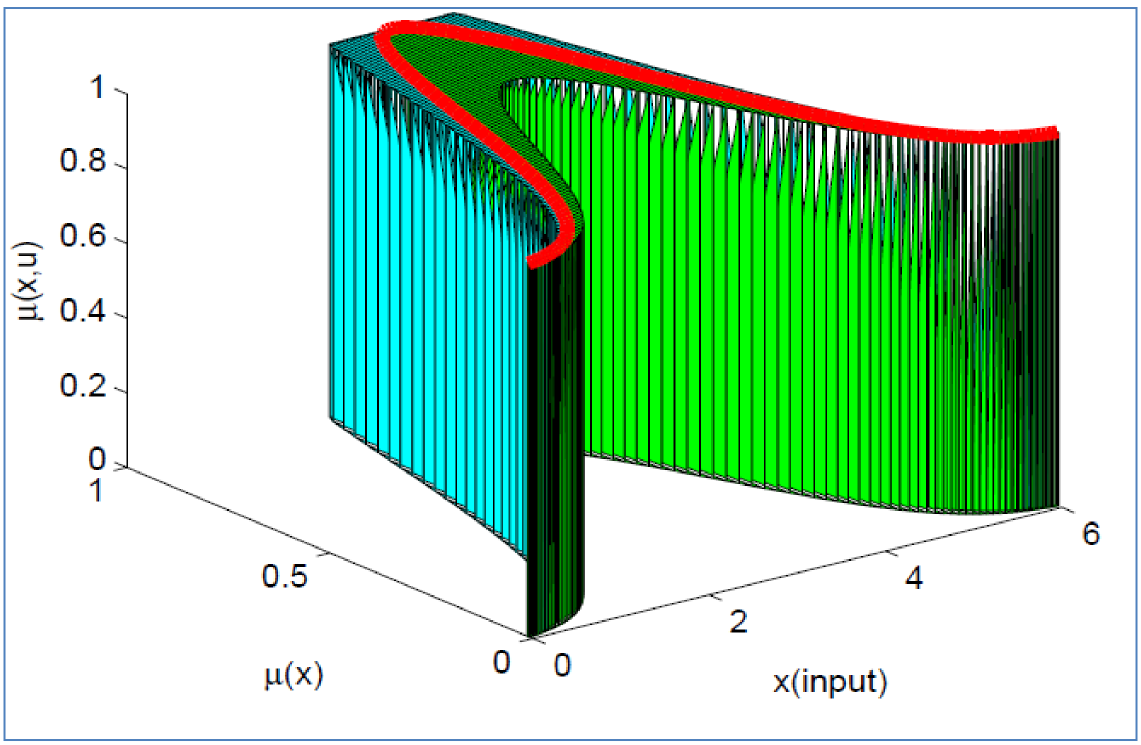


Figure III.4: Three-dimensional representation of interval type-2 fuzzy membership functions.

III.5- Operations of Type-2 Fuzzy Sets:

Our goal now is describing the set theoretic operations of type-2 fuzzy sets. We are interested in the case of type-2 fuzzy sets, $\tilde{A}_i$ (i = 1,…, r), whose secondary membership functions are type-1 fuzzy sets. To compute the union, intersection, and complement of type-2 fuzzy sets, we need to extend the binary operations of minimum (or product) and maximum, and the unary operation of negation, from crisp numbers to type-1 fuzzy sets, because at each x, $u_{\tilde{A}_i}(x,u)$ is a function (unlike the type-1 case, where $u_{\tilde{A}_i}(x)$ is a crisp number). The tool for computing the union, intersection, and complement of type-2 fuzzy sets is Zadeh's extension principle [11].

$$\tilde{A}_1 = \int_x u_{\tilde{A}_1}(x) \quad \text{(III.7)}$$

$$\tilde{A}_2 = \int_x u_{\tilde{A}_2}(x) \quad \text{(III.8)}$$

we focus our attention on set theoretic operations for such general type-2 fuzzy sets.

III.5.1- Union of type-2 fuzzy sets :

The union of $\tilde{A}_1$ and $\tilde{A}_2$ is another type-2 fuzzy set, just as the union of type-1 fuzzy sets A_1 and A_2 is another type-1 fuzzy set. More formally, we have the following expression [11]:

$$\tilde{A}_1 \cup \tilde{A}_2 = \int_{x \in X} u_{\tilde{A}_1 \cup \tilde{A}_2}(x)/x \quad \text{(III.9)}$$

We can explain Equation (III.9) by the "join" operation. Basically, the join between two secondary membership functions must be performed between every possible pair of primary memberships. If more than one combination of pairs gives the same point, then in the join we keep the one with maximum membership grade. We will consider a simple example to illustrate the union operation. In Figure 5 we plot two type-2 Gaussian membership functions, and the union is shown in Figure III. 6. [11]

III.5.2- Intersection of type-2 fuzzy sets

The intersection of $\tilde{A}_1$ and $\tilde{A}_2$ is another type-2 fuzzy set, just as the intersection of type-1 fuzzy sets A_1 and A_2 is another type-1 fuzzy set. More formally, we have the following expression [11]:

$$\tilde{A}_1 \cap \tilde{A}_2 = \int_{x \in X} u_{\tilde{A}_1 \cap \tilde{A}_2}(x)/x \quad \text{(III.10)}$$

We illustrate the intersection of two type-2 Gaussian membership functions in Figure III.5, We can explain Equation (III.10) by the "meet" operation. Basically, the meet between two secondary membership functions must be performed between every possible pair of primary memberships. If more than one combination of pairs gives the same point, then in the meet we keep the one with maximum membership grade [11].

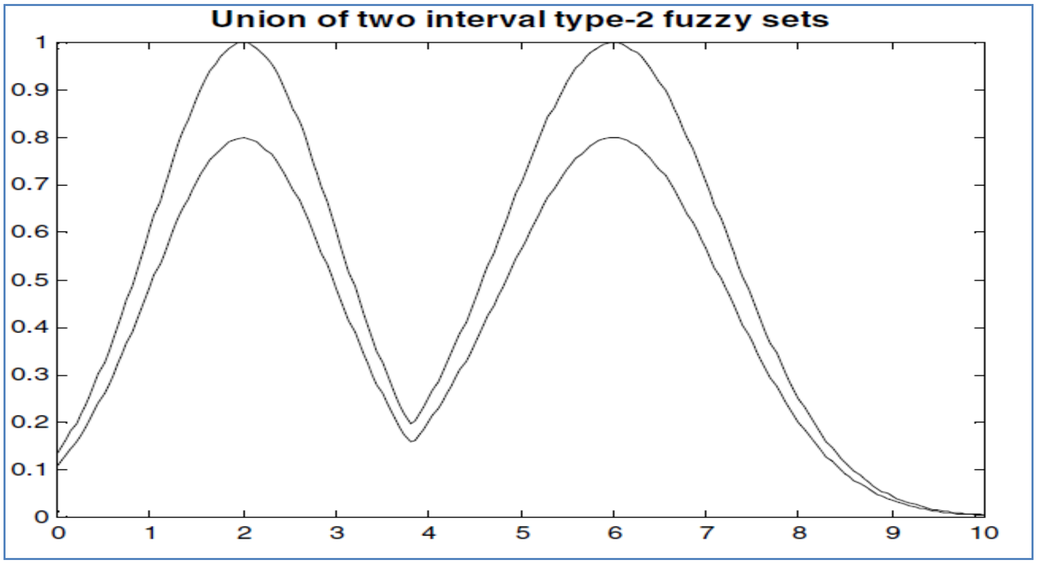


Figure.III.5: Two sample type-2 Gaussian membership functions

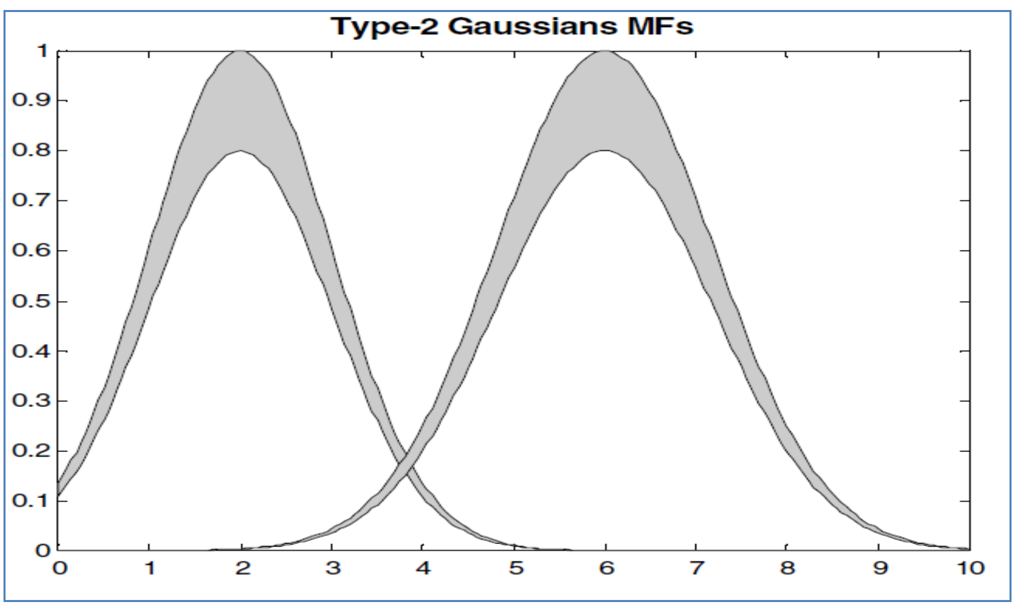


Figure III.6: Union of the two Gaussian membership functions.

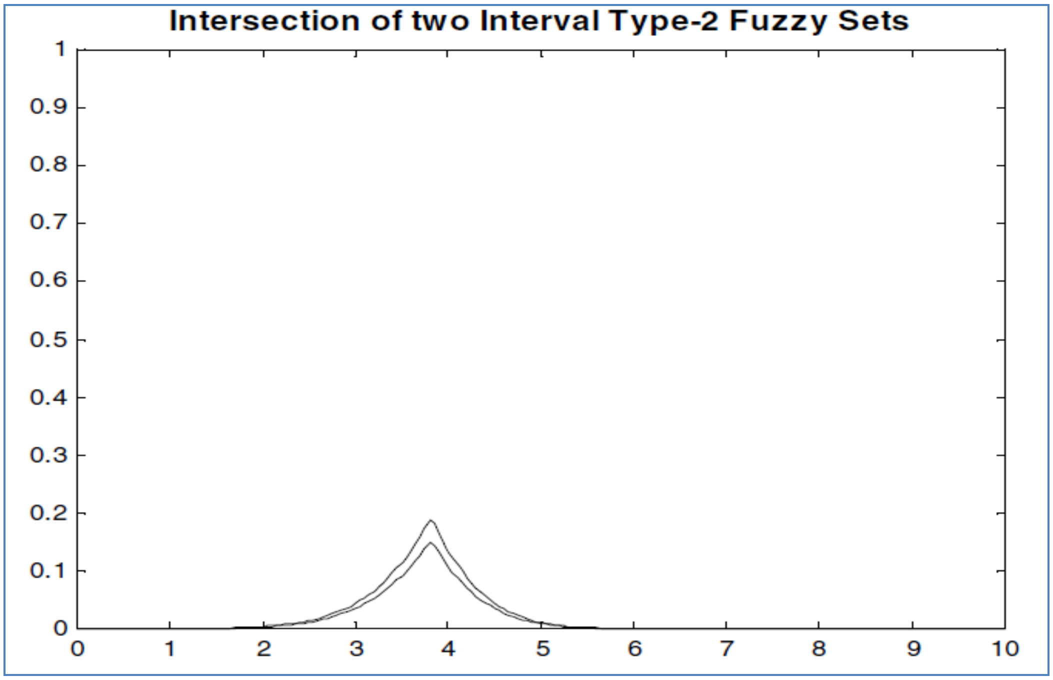


Figure III.7: Intersection of two type-2 Gaussian membership functions

III.5.3- Complement of a type-2 fuzzy set

The complement of set $\tilde{A}$ is another type-2 fuzzy set, just as the complement of type-1 fuzzy set A is another type-1 fuzzy set. More formally we have [11]

$$\overline{\tilde{A}} = \int_{x} u_{\tilde{A}_1}(x) \qquad \text{(III.11)}$$

To-date, because of the computational complexity of using a general T2 FS, most people only use interval T2 FSs in a T2 FLS, the result being an interval T2 FLS (IT2 FLS)[14]. The computations associated with interval T2 FSs are very manageable, which makes an IT2 FLS quite practical [14].

$$\tilde{A}_1 \cup \tilde{A}_2 = 1/[\,\overline{u}_{\tilde{A}_1}(x)\ v\ \overline{u}_{\tilde{A}_2}(x), \underline{u}_{\tilde{A}_1}(x)\ v\ \underline{u}_{\tilde{A}_2}(x)\,] \qquad \forall x \in X \qquad \text{(III.12)}$$

$$\tilde{A}_1 \cap \tilde{A}_2 = 1/[\,\overline{u}_{\tilde{A}_1}(x)\ \Lambda\ \overline{u}_{\tilde{A}_2}(x), \underline{u}_{\tilde{A}_1}(x)\ \Lambda\ \underline{u}_{\tilde{A}_2}(x)] \qquad \forall x \in X \qquad \text{(III.13)}$$

$$\overline{\tilde{A}} = 1/[1- \overline{u}_{\tilde{A}_1}(x)\,, 1-\underline{u}_{\tilde{A}_1}(x)\,] \qquad \forall x \in X \qquad \text{(III.14)}$$

III.6- Design of Interval Type-2 Fuzzy Logic Controllers

In fact, a type-2 fuzzy logic system or controller uses the same familiar notions as used in a type-1 fuzzy logic controller as membership functions, rules, t-norms operations, fuzzification, inference, defuzzification [13], but Since a higher type changes the nature of the membership functions, the operations that depend on the membership functions used in inference change; however, the basic principles of fuzzy logic are independent of the nature of membership functions and hence, do not change [10]. A type-2 fuzzy logic system is very similar to type-1, where it follows the same methodology, but the only difference is in the third block where we no longer speak of only defuzzification but we speak about a type reducer and defuzzification parts that constitute both the output processing block. This difference is mainly associated with the nature of the membership functions, where type-reducer is needed due to the added degree in the kind of fuzzy sets. Figure III.8 presents a type-2 fuzzy logic system [11].

Today, the two most popular fuzzy logic systems used by engineers in control are the Mamdani and TSK(Takagi–Sugeno) systems [11].

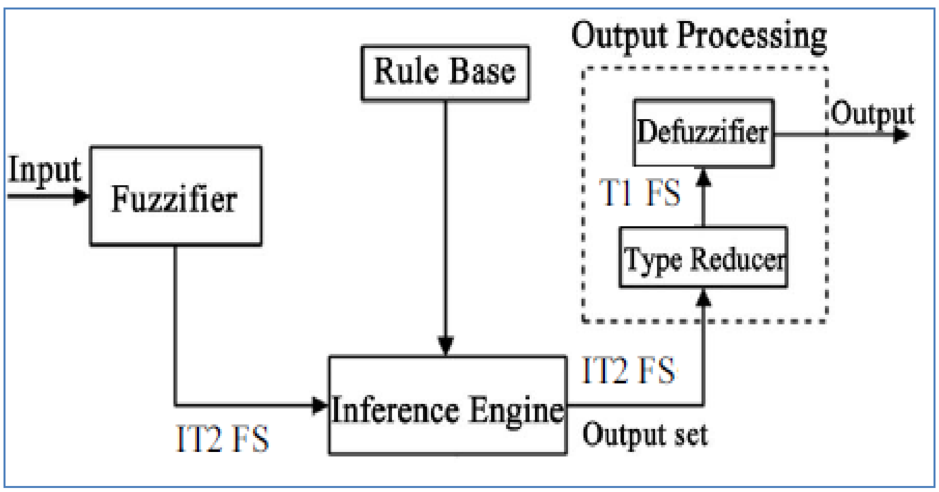


Figure III.8: Type-2 fuzzy logic controller

III.6.1- Mamdani Type-2 Fuzzy Logic Controller:

a- Fuzzification

In this part, we must first define the fuzzy sets of crisp input vector with p elements x = $(x_1,\ldots, x_p)^T$ in the universe of discourse $X_1 \times X_2 \ldots\ldots \times X_p$. those memberships can contain one or several type-2 fuzzy sets. Second, the fuzzifier maps those inputs into the associated fuzzy sets to determine the degree of membership of each input variable. The type-2 fuzzification process is schematically depicted in Figure III.31. For each point of the universe of discourse, the upper and lower membership functions are computed. We consider only singleton fuzzification for which the inputs are crisp values [13], [15].

b- 3.1.2 Type-2 Rule Base

The structure of the rules of a type-2 FLC is similar to that of type-1[15]. A type-2 fuzzy logic with p inputs and $x_1 \in X_1,\ldots x_p \in X_p$ and one output $y \in Y$ and with M rules. The l^{th} rule has the following form [13]:

$$R^l : \text{IF } x_1 \text{ is } \tilde{F}_1^l \text{ and } \ldots\ldots \text{ and } x_p \text{ is } \tilde{F}_p^l \text{ THEN } y \text{ is } \tilde{G}^l \quad l = 1\ldots\ldots M$$

Where $\tilde{F}_p^l$ and $\tilde{G}^l$ are input and output fuzzy labels, respectively. And X_p are universe of discourse

c- Inference

his block expresses the relationship that exists between the input variables (expressed as linguistic variables) and the output variables (also expressed as linguistic variables)[11]. It aggregates the if-then rules stored in the knowledge base with the fuzzy sets generated by the fuzzifier to form an overall output fuzzy set. Similarly, a type-2 fuzzy inference engine provides a mapping from the input type-2 fuzzy sets to the output ones. The intersection of multiple rule antecedents is computed using a t-norm operator while the union of multiple rules is computed through a t-conorm operation [15].

1 .The firing strength of the i^{th} rule is as in (III.15). The result of the input and antecedent operations is an interval type-1 set. [13] [15].

$$F^i(x') = \left[\underline{f^i}\ (x') ; \overline{f^i}\ (x')\right] \equiv \left[\underline{f^i}\ ;\ \overline{f^i}\right] = \left[\underline{u}_{\tilde{F}_1^i}(x_1') * \ldots .* \underline{u}_{\tilde{F}_p^i}(x_p')\ ;\ \overline{u}_{\tilde{F}_1^i}(x_1') * \ldots .* \overline{u}_{\tilde{F}_p^i}(x_p')\right] \quad \text{(III.15)}$$

Where $\underline{u}_{\tilde{F}_p^i}(x_p')$ and $\overline{u}_{\tilde{F}_p^i}(x_p')$ designed respectively upper and lower membership grades of $u_{\tilde{F}^i}(x)$ and

$$\underline{f^i} = \underline{u}_{\tilde{F}_1^i}(x_1') * \ldots .* \underline{u}_{\tilde{F}_p^i}(x_p') \quad , \quad \overline{f^i} = \overline{u}_{\tilde{F}_1^i}(x_1') * \ldots .* \overline{u}_{\tilde{F}_p^i}(x_p') \quad \text{(III.16)}$$

The fired output consequent set of the l rule is a type-1 fuzzy set characterized by a membership function

$$u_{\tilde{B}^l}(y) = \int_{b \in \left[\underline{f}^l * \underline{u}_{\tilde{G}^l}(y) \; \overline{f}^1 * \overline{u}_{\tilde{G}^1}(y) \right]} 1/b^l \qquad \text{(III.17)}$$

If N out of a total of L fuzzy rules in the FLS fire, where N ≤ L, then the overall aggregated output fuzzy set is defined by a type-1 membership function $\mu_{\tilde{B}}(y)$ obtained by combining the fired output consequent sets into one. In other words $\mu_{\tilde{B}}(y) = \cup_{l=1}^{N} u_{\tilde{B}^l}(y)$ where [15] :

$$u_{\tilde{B}^l}(y) = \int_{b \in \left[\left[\underline{f}^1 * \underline{u}_{\tilde{G}^1}(y) \right] V \dots V \left[\underline{f}^N * \underline{u}_{\tilde{G}^N}(y) \right], \left[\overline{f}^1 * \overline{u}_{\tilde{G}^1}(y) V \dots V \left[\overline{f}^N * \overline{u}_{\tilde{G}^N}(y) \right] \right]} 1/b^l \quad \forall\, y \in Y \qquad \text{(III.17)}$$

with the t-norm operator denoted by "*", and t-conorm operator denoted by "V"

Since generally we use the meet operation under product or minimum t-norm. So, at each value of x the intersection and union operations are referred to as the meet and join operations, respectively [13].

Example of firing strength of one rule with two antecedent and one consequent (two input and one output) using t-norm operator is depicted in FigureIII.9.

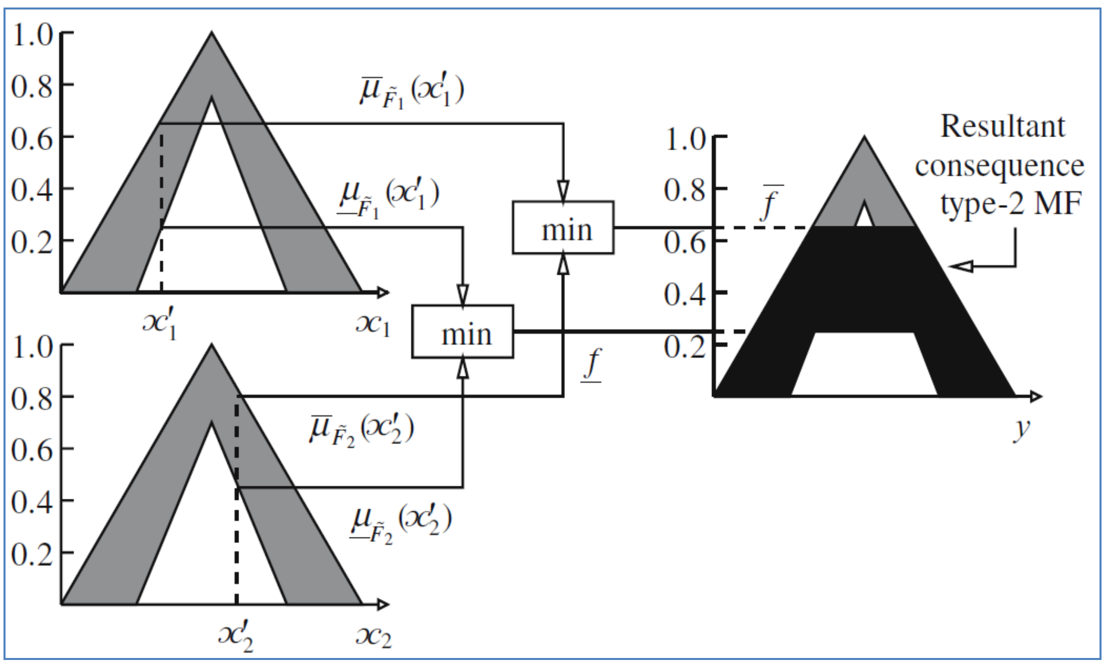


Figure III.9: Interval type-2 inference process

d- Type Reducer and Defuzzification:

In a type-2 fuzzy logic, since this kind of system deal with type-2 sets, then it is necessary to have a type reducer block to map a T2 FS into a T1 FS, and then defuzzification, as usual, maps that T1 FS into a crisp number. We can consider that the defuzzification block of a type-1 fuzzy logic is replaced by the output processing block in a type-2 fuzzy logic. That block consists of type-reducer followed by defuzzification. In fact, type Reducer was proposed by Karnik and Mendel [13]. For now, there are five different type-reduction methods (such as the center-of-sums, the height, the modified height and the center-of-sets, for example). In this dissertation, we will consider the center-of-sets type reduction technique thanks to its computational [15]. Karnik and Mendel defined the centroid of an IT2 FS which is an IT1 FS that is ensured using the Extension Principle.

This IT1 FS is characterized by its left and right end points y_l and y_r which can be written in the following equation (III.18) [13]:

$$[\, y_l, y_r] = \int_{y^1\in[y_l^1,\, y_r^1\,]} \cdots \int_{y^M\in[y_l^M,\, y_r^M\,]} \int_{f^1\in[\underline{f^1},\ \overline{f^1}]} \cdots \int_{f^M\in\left[\underline{f^M},\ \overline{f^M}\right]} 1/\ \frac{\sum_{i=1}^{M} f^i y^i}{\sum_i^M f^i} \qquad \text{(III.18)}$$

this interval set is determined by its two end points, y_l and y_r which corresponds to the centroid of the type-2 interval consequent set $\tilde{G}^i$ [10].

III.6.2- Interval Type-2 TSK Fuzzy Logic Controller :

The differences between a type-2 TSK fuzzy logic controller and a Mamdani T2 fuzzy logic consist essentially of the definition of outputs and then on the consequent part of rules. Consider we have a type-2 TSK fuzzy logic with p inputs $x \in X_1, \ldots x_p \in X_p$ and one output $y \in Y$, and with M rules. The l^{th} rule can be expressed as [10]:

R^l : IF x_1 *is* $\tilde{F}_1^l$ and ….. and x_p *is* $\tilde{F}_p^l$ THEN $y^l = c_0^l + c_1^l x_1 + \ldots\ldots + c_p^l x_p$ (III.19)

A type-2 TSK fuzzy logic controller or system (T2 TSK fuzzy logic) was firstly introduced by Liang and Mendel [13]. Although TSK type-1 fuzzy systems have received a lot attention, the literature on TSK type-2 fuzzy systems is few. Liang and Mendel applied type-2 TSK systems in channel equalization of channels [13]. Where, according to them, there are three models of T2 TSK fuzzy logics depending on the kind of the antecedent and consequent part of rules, to have: T2 TSK- Model I, T2 TSK-Model II and T2 TSK-Model III. We can see in Table III.4 and 2 the difference between those models. Where c_i^l , C_i^l are the consequent parameters, y^l and Y^l are the outputs of the lth rule, $\tilde{F}_j^l$ (j ,…, p) are type-2 fuzzy sets and F_j^l(j ,…, p) are type-1 fuzzy sets. The firing strength of the ith rule W^i(x) with meet operation under product or minimum t-norm is an interval type-1 set expressed as follows [11]:

$$W^i(x) = \left[\underline{w}^i\,(x)\,;\ \overline{w}^i(x)\right] \qquad \text{(III.20)}$$

$$\underline{w}^i\,(x) = \underline{u}_{\tilde{F}_1^i}(x_1') * \ldots * \underline{u}_{\tilde{F}_p^i}(x_p') \quad , \quad \overline{w}^i(x) = \overline{u}_{\tilde{F}_1^i}(x_1') * \ldots * \overline{u}_{\tilde{F}_p^i}(x_p')$$

Model	Antecedents	Consequents
Model I	Type-2 fuzzy sets	type-1 fuzzy sets
Model II	Type-2 fuzzy sets	crisp numbers
Model III	Type-1 fuzzy sets	type-1 fuzzy sets

Table III.1: Models of T2 TSK FLS

TSK FLS	Rules R^l
Type-1	IF x_1 is F_1^l and ….. and x_p is F_p^l THEN $y^l = c_0^l + c_1^l x_1 + \ldots + c_p^l x_p$
T2 Model I	IF x_1 is $\tilde{F}_1^l$ and ….. and x_p is $\tilde{F}_p^l$ THEN $y^l = c_0^l + c_1^l x_1 + \ldots + c_p^l x_p$
T2 Model II	IF x_1 is $\tilde{F}_1^l$ and ….. and x_p is $\tilde{F}_p^l$ THEN $y^l = c_0^l + c_1^l x_1 + \ldots + c_p^l x_p$
T2 Model III	IF x_1 is F_1^l and ….. and x_p is F_p^l THEN $y^l = c_0^l + c_1^l x_1 + \ldots + c_p^l x_p$

Table III.2: Rules of T2 TSK FLS.

The final output is also an interval type-1 set and is calculated as follows [13]:

$$Y(Y^1,\ldots,Y^M,W^1,\ldots,W^M) = [y_l,\ y_r] = \int_{y^1}\ldots\int_{y^M}\int_{w^1}\ldots\int_{w^M} 1/\ \frac{\sum_{i=1}^M w^i y^i}{\sum_i^M w^i} \qquad \text{(III.21)}$$

Where $y^i \in Y^i$, and $Y^i = [y_l^i,\ y_r^i]$, (i=1,…., M), thus for each rule we will obtain y_l and y_r Since all sets are crisp, the Equation (III.21) results to [13]:

$$y_l = \frac{\sum_{i=1}^M w_l^i\, y_l^i}{\sum_i^M w_l^i} \quad ; \quad y_r = \frac{\sum_{i=1}^M w_r^i\, y_l^i}{\sum_i^M w_r^i} \qquad \text{(III.23)}$$

And the defuzzified output is:

$$y = (y_l,\ y_r)/2 \qquad \text{(III.24)}$$

Conclusion

In this chapter, we have explained another intelligent controller more powerful then type-1 fuzzy logic controller, so-called interval type-2 fuzzy logic controller based on interval type-2 fuzzy set since general type-2 FLC is more computation needed because the secondary membership of first is always equal to 1, as well as we have seen its membership function has uncertainties known as FOU bonded by the upper and lower type-1 membership function means that the uncertainty is not just on the rules but also in MF, so that it (IT2FLC) can cope with all the kind of the uncertainty

Produced by the instrumentation elements, environmental noise, etc. And we have also illustrated the operations that needed in the flowchart of design that IT2FLC of the two most popular fuzzy logic controller used by engineers in control are the Mamdani and TSK (Takagi–Sugeno) systems.

After all that bunch of information about type-n fuzzy logic, the next chapter we will design IT2FLC to control the helicopter flight simulator TRMS (Twin Rotor MIMO system), and its simulation results will be given and compare with simulation results of type-1 fuzzy logic controller.

CHAPTER FOUR

Control algorithm of Interval type-2 Fuzzy Logic for TRMS

IV.1- Introduction:

This part will apply the three following tools in order to control the TRMS system:

- Using Type1 toolbox.
- Using IT2FLC toolbox
- Using MTLAB control algorithm of Type-2

Our objective is reaching a good controller that can control elevation angle and azimuth angle, follow referential signal, and maintain system performance.

This chapter is divided into two parts: simulation and experimental. The simulation part applied the three above tools Type1 toolbox, IT2FLC toolbox and Type-2 program, and the results will be mentioned later on. On the other hand, the experimental part applied Type1 toolbox, IT2FLC toolbox tool, but it couldn't obtain the expected results. As for Type-2 program, it was unable to apply it due to dis-correspondence of equipment, or due to the unsustainability of equipment as a result of the concentration of computation in this type.

IV.2- The Control of TRMS

The implementation of the fuzzy controller in terms of typt-1 and type-2 fuzzy sets, has two input variables, which are the error e(t), the difference between the reference signal and the output of the process, as well as the error variation Δe(t) :

$$e(t) = r(t) - y(t)$$

$$\Delta e(t) = e(t) - e(t-1)$$

so the control system can be represented as in Figure IV.1.

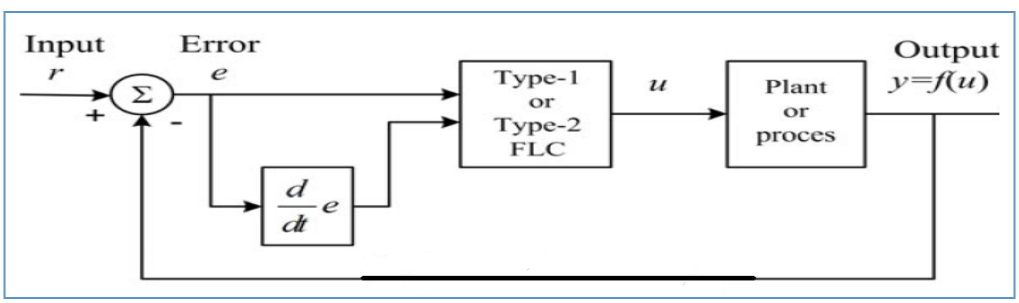


Figure IV.1: System used for obtaining the Simulation and the Experimental results for control

In Figure IV.2 represented the two fuzzy controllers, used to control the TRMS system. These are independent of one another, the first is used to adjust the angle of elevation, and the second to set the azimuth angle. Each of these two controllers with two inputs characterizing the error and its variation and an output which characterizes the variation of the control. The sum of both commands generated by these two regulators, the overall shape control that stabilizes the system.

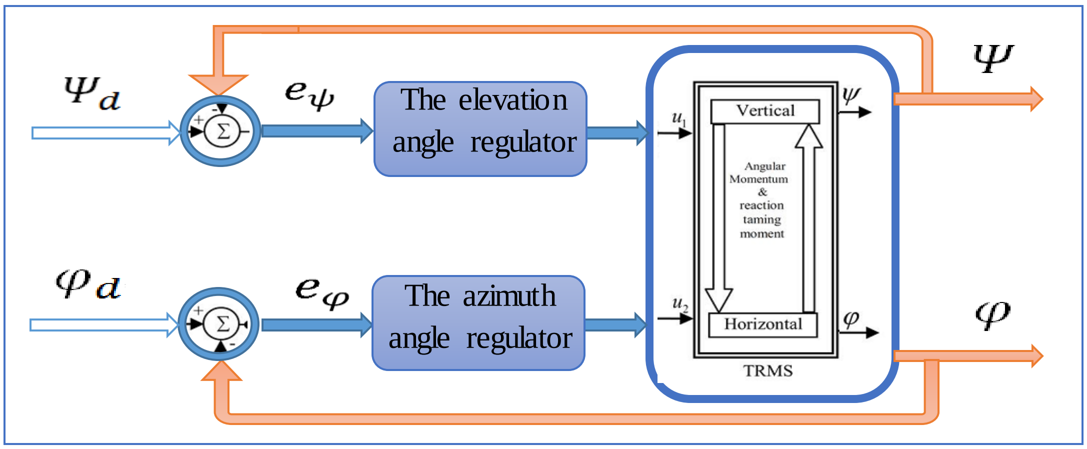


Figure IV.2: Block diagram of the control scheme

The inputs of both fuzzy controllers are beam angle error and the derivative of beam angle error.

Given the two position errors

$$e_{\psi} = \psi - \psi_d \qquad , \qquad e_{\varphi} = \varphi - \varphi_d$$

The controllers have used in this work to obtain our latter results are:

1- Using Type1 toolbox. (MathWork)
2- Using IT2FLC. [18]

- **Introduction to IT2FLC toolbox**:

Fuzzy Logic Toolbox Version 2.0.1 (R11) 16-Sep-1998

Generally the toolbox (MathWorks) of IT2FLC is used for understand the flow chart to design fuzzy logic controller, so they never uses to design practical controller. The IT2FLS Toolbox includes a series of program files (some of them in FigureIV.3) to do its role for example (fuzzy2.m) for display editor of IT2FLC (Figure IV.4) and its function is fuzzy2, so fuzzy2 is both a directory and a function, (defuzz2.m) for defuzzification, and so on. And also the toolkit supports the implementation of several types of fuzzy logic inference systems and has several aspects of its capabilities to allow the straightforward implementation of type-1 and interval type-2 fuzzy systems are:

The Mamdani and Takagi-Sugeno-Kang (TSK) Interval Type-2 Fuzzy Inference Models , and the design of Interval Type-2 membership functions and operators are implemented in the IT2FLS Toolbox (Interval Type-2 Fuzzy Logic Systems) reused from the Matlab® commercial Fuzzy Logic Toolbox.

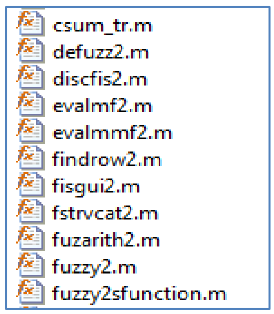


Figure IV.3: example of program files of IT2FLC toolbox

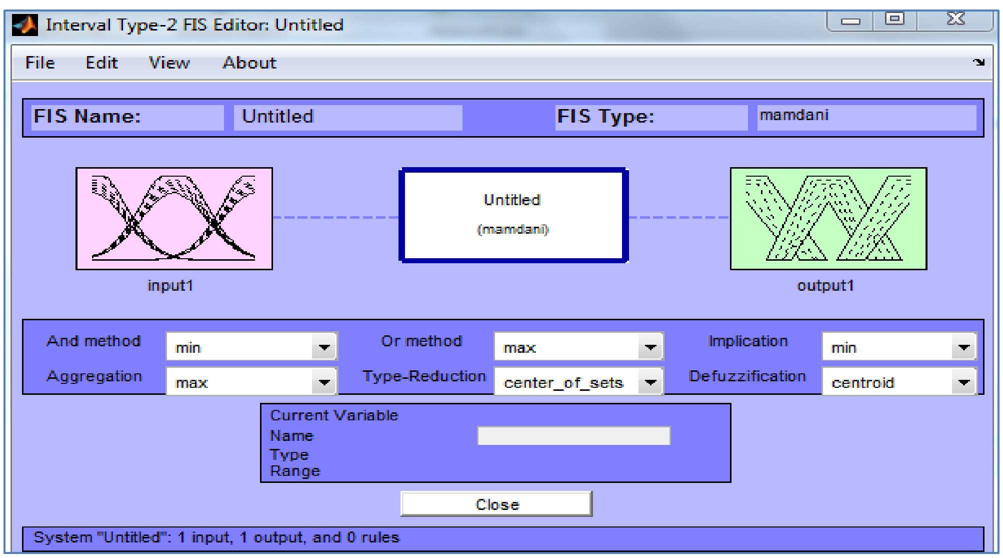


Figure IV.4: IT2FIS editor of its toolbox

3- Using MTLAB control algorithm of fuzzy_type2. [19]

According to Ph.D. Hicham Chaoui. University of Ottawa, Ottawa, Ontario, Canada.

IV.3- Type-1 toolbox:

In this work type-1 Mamdani method has been opted, and the AND method, OR method, implication, aggregation and defuzzifier are chosen to be min, max, min, max and centroid, respectively. also we have used a triangular membership functions.

IV.3.1- Controller 1 for The elevation angle :

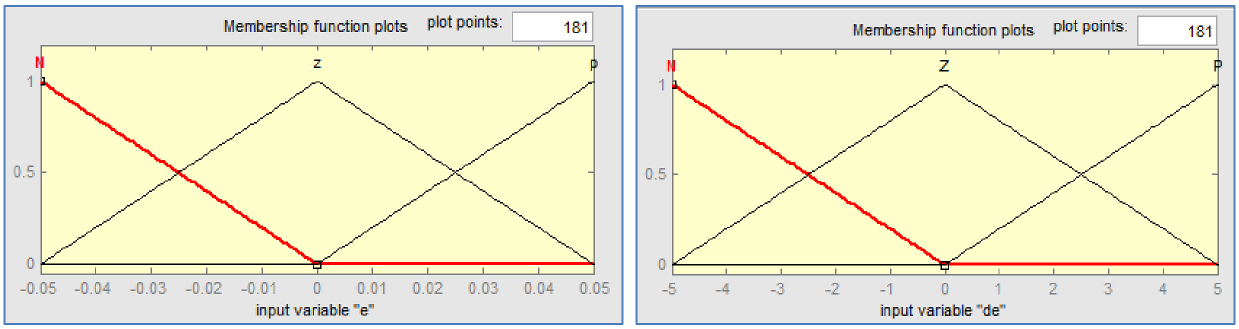


Figure IV.5: Input **e** or **$\dot{e}$** membership functions for Type 1.

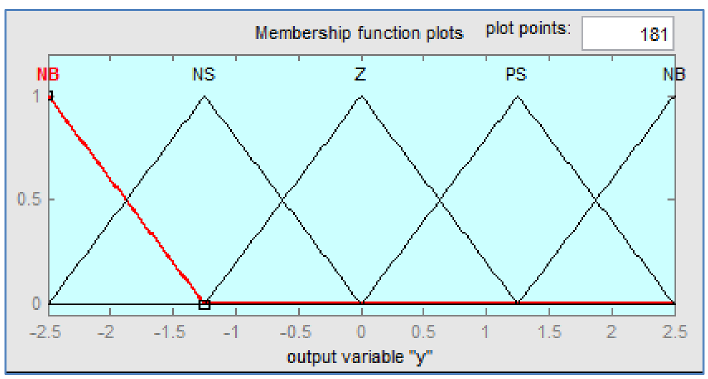


Figure IV.6: Output y membership functions for Type 1.

IV.3.2- Controller 2 for The azimuth angle:

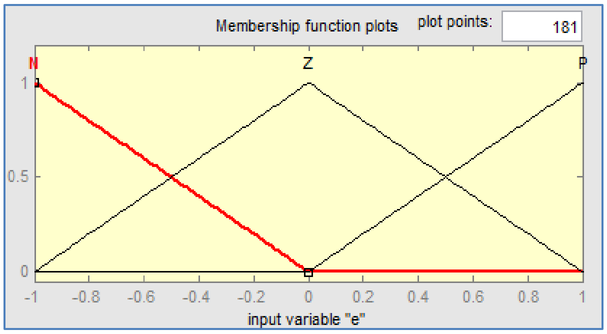


Figure IV.7: Input **e** or **ė** membership functions for Type 1.

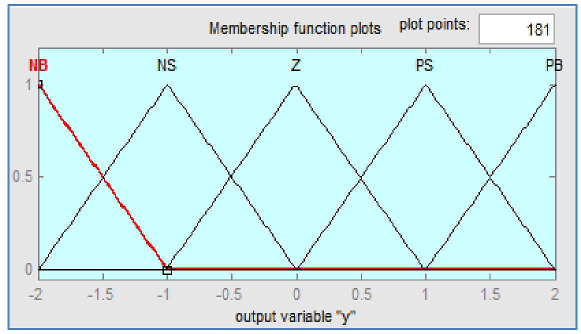


Figure IV.8: Output y membership functions for Type 1.

IV.3.3- Rule Base:

These rules are chosen is such a way as to accomplish the following controller's behavior:

1- When the signals errors are far from zero, then the fuzzy logic controller output assumes a high value.

2- When the inputs are approaching to the zero, the output is adjusted to a smaller value for a smoother approach.

3- Once the inputs are zero, then the output is set to zero.

The controller fuzzy rules are gathered in Table IV.7.

ė \ e	N	Z	P
N	NB	NS	Z
Z	NS	Z	PS
P	Z	PS	PB

Table IV.1: Fuzzy rules for our Type1

IV.3.4- block diagram of Type1(toolbox):

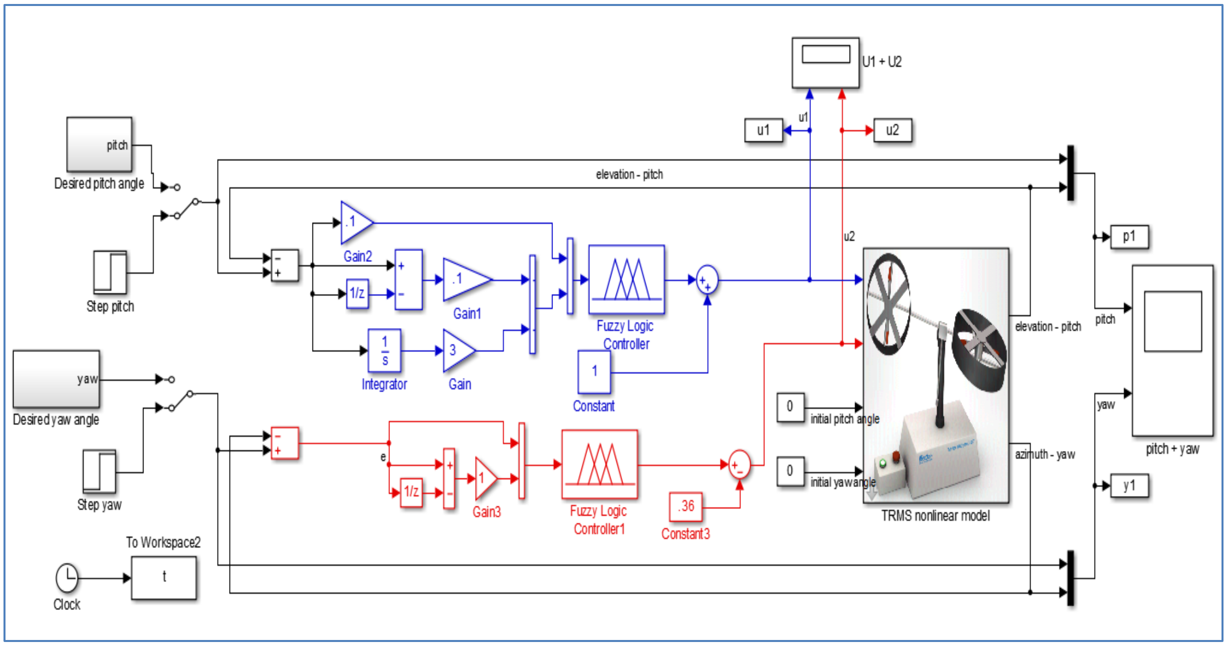


Figure IV.9: block diagram of Type1(toolbox)

IV.4- IT2FLC (toolbox):

In this work singleton type-2 Mamdani method has been opted, and the AND method, OR method, implication, aggregation, Type-Reduction and defuzzifier are chosen to be min, max, min, max, center-of-sets and centroid, respectively. also we have used a Gaussian membership functions.

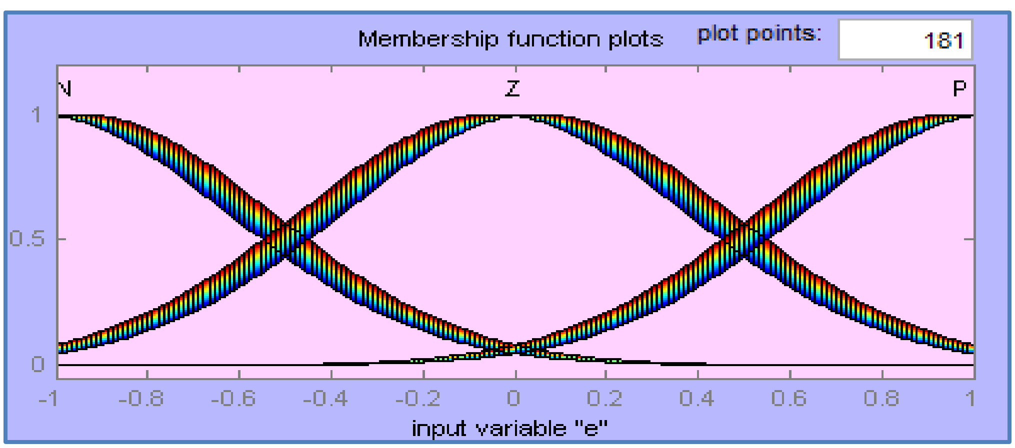


Figure IV.10: Input **e** or **ė** membership functions for IT2FLC.

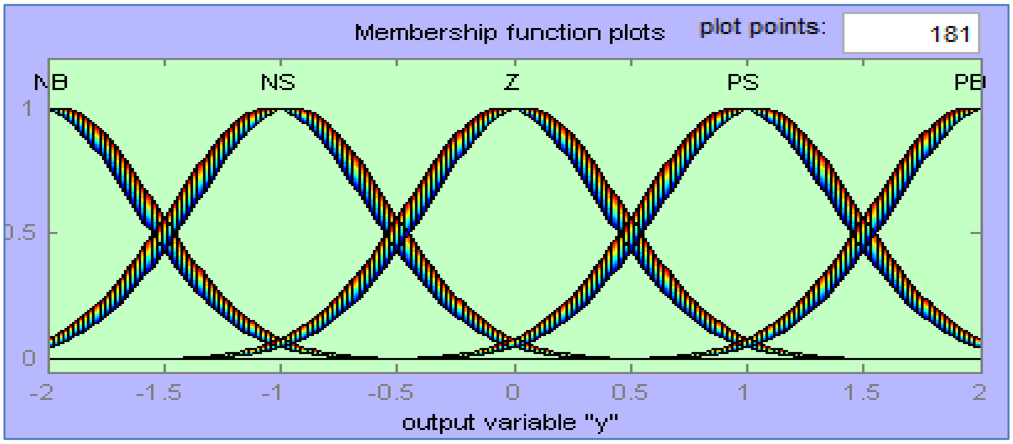


Figure IV.11: Output y membership functions for IT2FLC.

IV.4.1- Rules base:

We use same Rules base as like as we had shown in Type 1 (toolbox)

IV.4.2- block diagram of IT2FLC(toolbox):

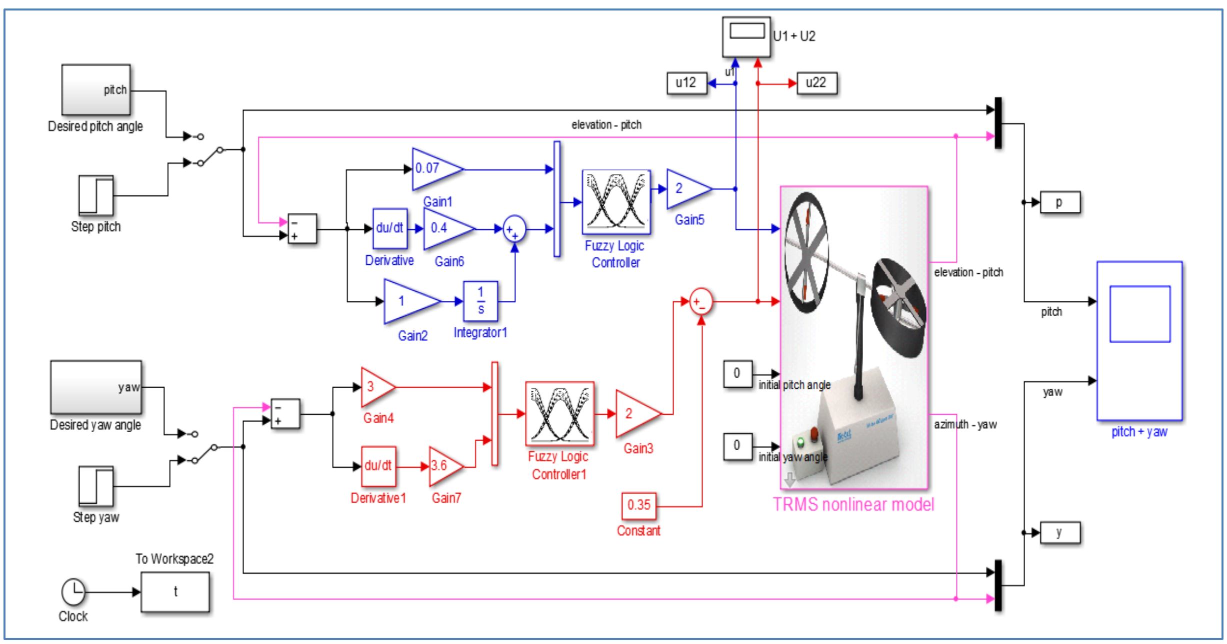


Figure IV.12: block diagram of IT2FLC(toolbox)

IV.5- control algorithm of fuzzy_type2:

IV.5.1- block diagram of Type2 (program):

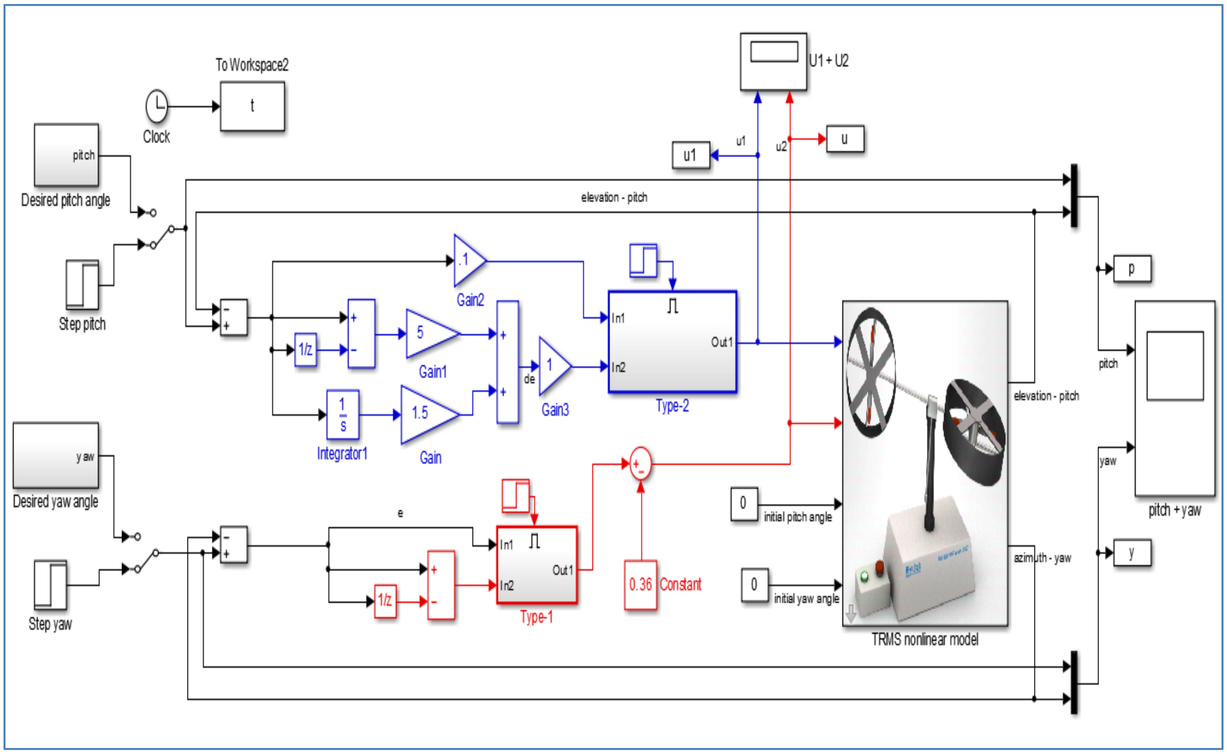


Figure IV.13: block diagram of Type 2 (program)

The parameter's algorithm at first and after our change:

	At first	Our change
Controller1	Gain =1	Gain = 1.5
Controller2	Out1	Out1 - 0.36

Table IV.2: The parameter's algorithm at first and after our change

IV.6- Part 1: Simulation

- **2 Degree of freedom (DOF) rotor control:**

a- For : Step pitch = 0.4 and Step yaw = 0.5

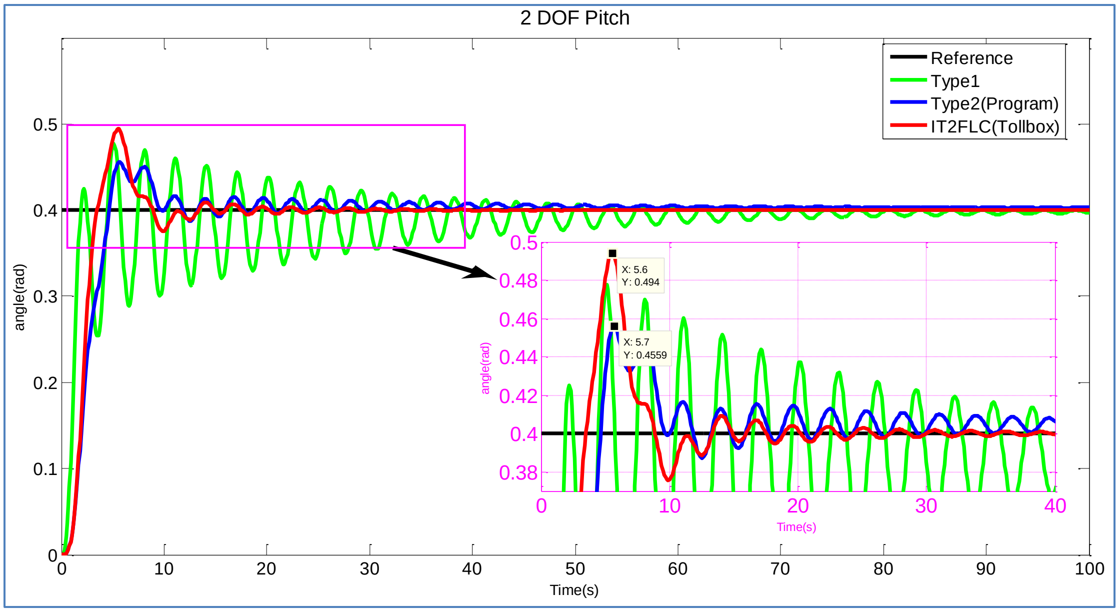


Figure IV.14: pitch stabilization.

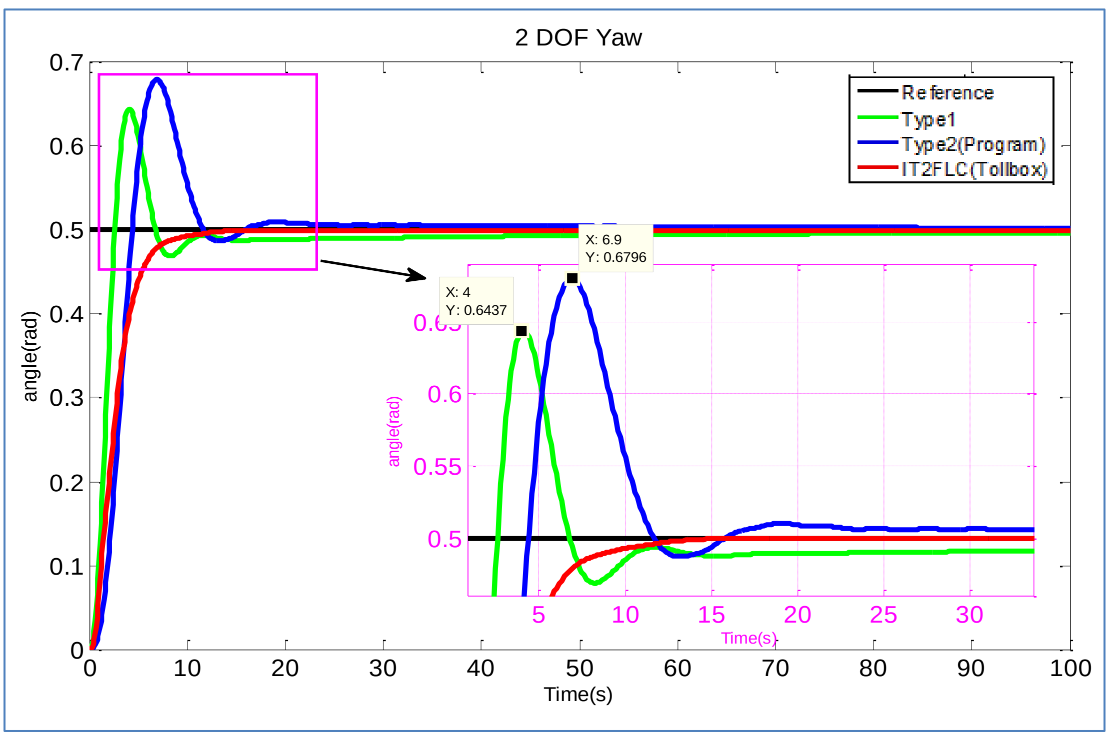


Figure IV.15: yaw stabilization

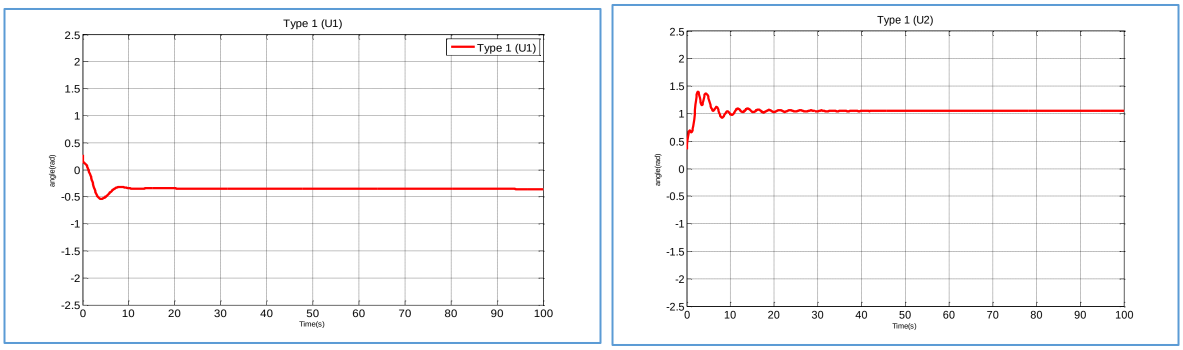


Figure IV.16: stapilistion signal U1 and U2 of T1FLC for pitch and yaw respectively

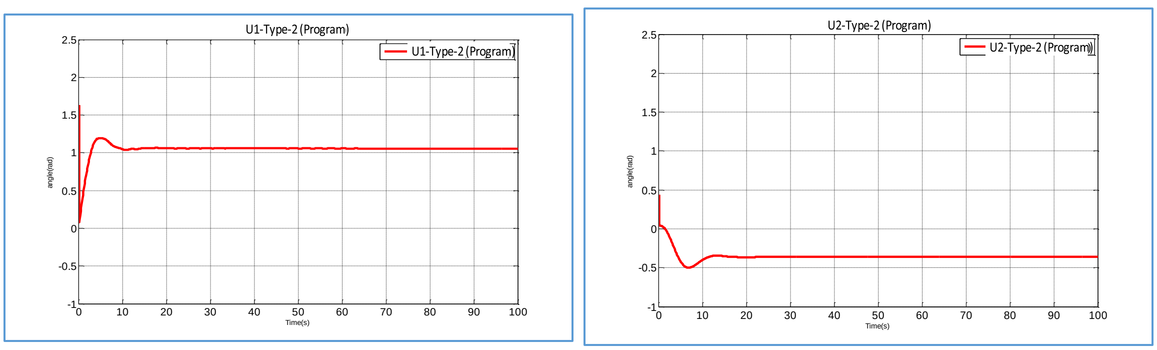


Figure IV.17: stapilistion signal U1 and U2 of Type-2 (program) for pitch and pitch respectively

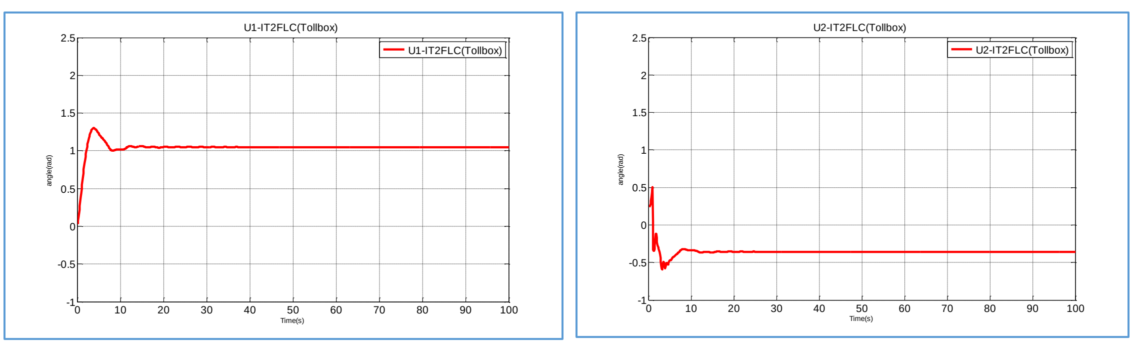


Figure IV.18: stapilistion signal U1 and U2 of IT2FLC(toolbox) for pitch and yaw respectively

b- For : Desird pitch = 0.4 and Desird yaw = 0.5

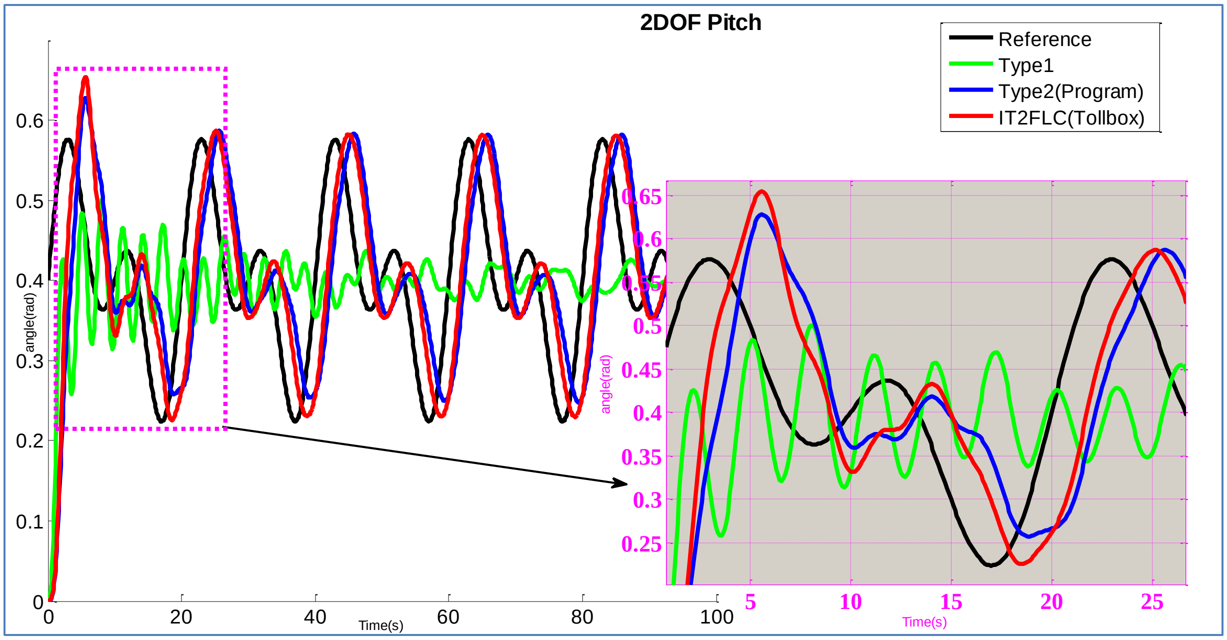


Figure IV.19: desired wave responses of pitch

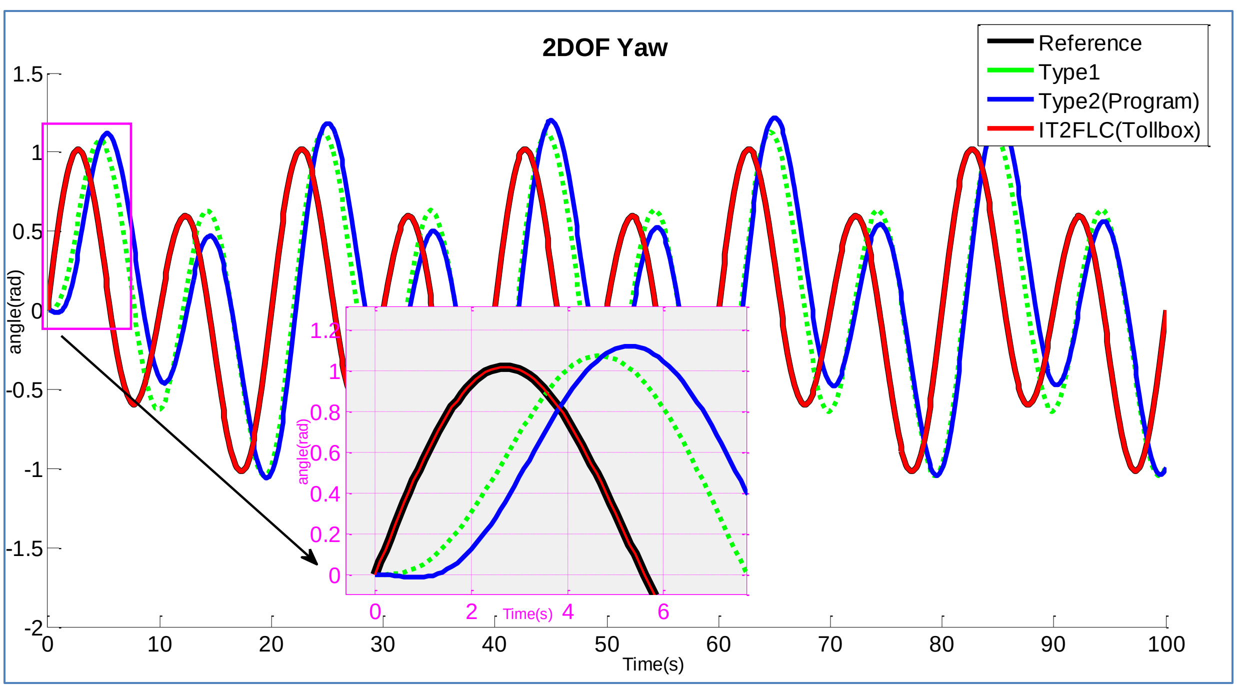


Figure IV.20: desired wave responses of yaw

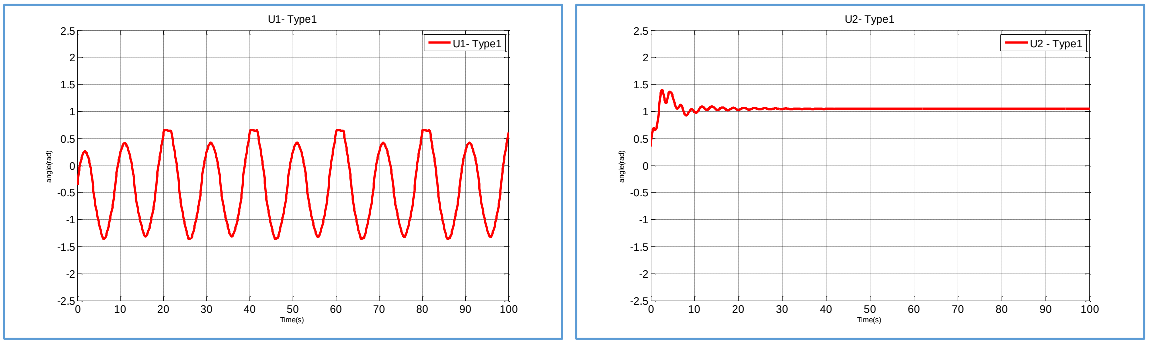


Figure IV.21: desired trejectior signal U1 and U2 of T1FLC for pitch and yaw respectively

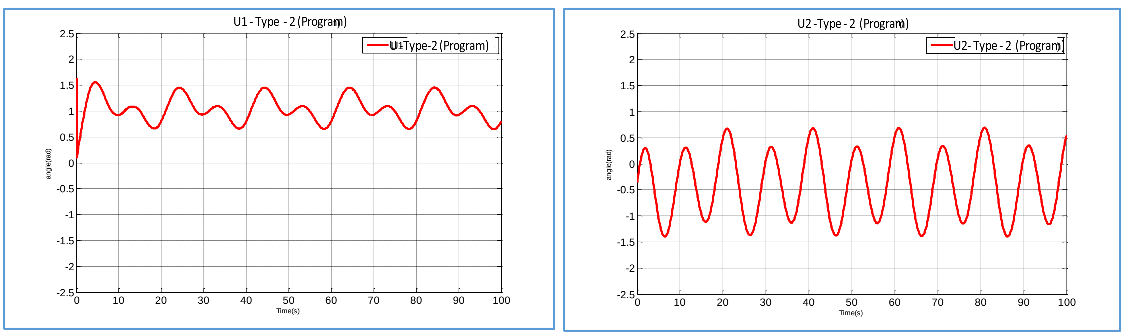


Figure IV.22: desired trejectior signal U1 and U2 of Type-2 (program) for pitch and yaw respectively

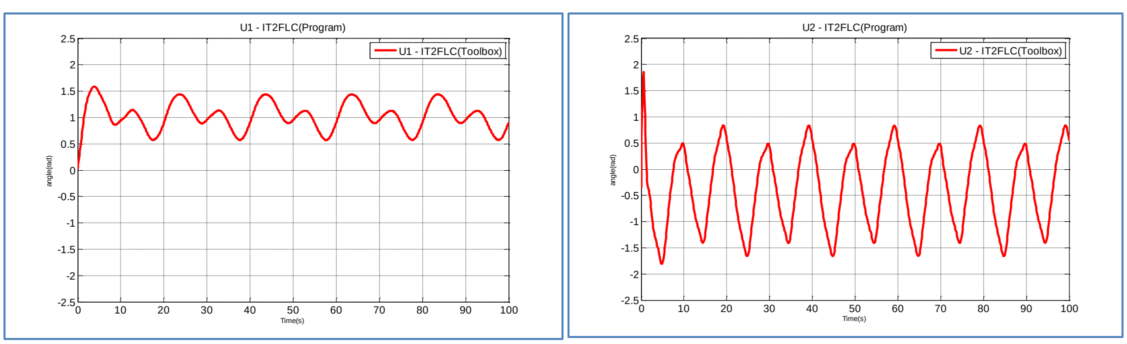


Figure IV.23: desired trejectior signal U1 and U2 of IT2FLC (toolbox) for pitch and yaw respectively

IV.5- Discussion the simulations results

The simulation results of the TRMS attitude dynamics control are presented. The both of initial values of the pitch and yaw angles are taken 0 radians. The results obtained for the attitude stabilization of the TRMS are given in the Figure IV.44, IV.45 for the pitch and yaw angles respectively. It can be seen that, the Type-2 fuzzy logic controller ensures a good convergence, but for IT2FC(toolbox) is better then Type-2 (program). and the yaw angle time response is relatively quick compared to the pitch angle response for all the controllers . Also, as illustrated in Figure IV.44, the type-2 fuzzy logic controller provides a better performance than the type-1 fuzzy logic controller. Especially, the type-2 fuzzy logic controller presents fast step responses with small oscillations, as opposed to IT2FLC (toolbox).

FigureIV.49 and Figure IV.50 show the trajectory tracking accuracy of the proposed control low. In the case of the type-2 fuzzy controller, the actual angles pitch and yaw of the helicopter converge, without oscillation, to their desired values, specially when IT2FLC(toolbox) is used, While in the case of the type-1 fuzzy controller, oscillations with big amplitude are observed.

From all the obtained results, it can clearly be seen that, in the case of the type 2 fuzzy controller, all outputs converge accurately to their desired values. A poor performance is obtained in the case of the type-1 fuzzy logic controller. Thus, the type1- fuzzy logic controller cannot be used in mathematically ill-defined systems that may be subjected to structured and unstructured uncertainties

IV.6- Conclusion:

This chapter is the main part of our work; we have used different tools (toolboxes and program) to illustrate the powerful of IT2FLC than T1FLC that by simulation results of application with different tests (stabilization and desired wave) for controlling concrete system (TRMS). From that we can say it is better to use IT2FLC (not T1 or classical methods) to control like that kind of systems with mathematically ill-defined and that may be subjected to structured and unstructured uncertainties.

CONCLUSION

CONCLUSION

The goal of this work is to design control algorithm to control mathematically ill-defined (uncertain dynamic) system (TRMS) that may be subjected to structured and unstructured uncertainties and noisy environment. A new approach for the attitude stabilization for that kind of system “two degrees of freedom TRMS is presented. This approach is based on the type-2 fuzzy logic controller. The main strength of the proposed control algorithm is more robust than previous control algorithm (type-1FLC) with respect to parametric uncertainties and noise measurement. The proposed approach has been successfully applied, in simulation and practical results, to the control of two degrees of freedom helicopter in the presence of parametric uncertainties and noise measurement.

REFERENCES :

- [1]. Ashish Tewari, 2002, Modern Control Design with MATLAB and SIMULINK, 518p, ISBN 0 471 496790.
- [2]. Abderrahmen BOUGUERRA, Djamel SAIGAA, Kamel KARA, Samir ZEGHLACHE, and Keltoum LOUKAL, Fault-Tolerant Control of a 2 DOF Helicopter (TRMS System) Based on H_{∞}.
- [3]. Samir Zeghlache, Kamel Kara, and Djamel Saigaa. On Type-2 Fuzzy Logic Control of a 2-DOF Helicopter (TRMS system). In Central European Journal Engineering.• 4(3) • 2014 • 303-315
- [4]. A.Rahideh and M.H.Shaheed. iEEE on Hybrid Fuzzy-PID-based Control of a Twin Rotor MIMO System
- [5]. Samir ZEGHLACHE, Abderrahmen BOUGUERRA, and Mohamed CHEMACHEMA. iEEE on Feedback Linearization Design Applied to the Position Control of 2-DOF Helicopter, Volume 54, Number 3-4, 2013
- [6] S. sumathi and surekha Paneerselvam: Computational Intelligence Paradigms Theory and Applications using MATLAB®, CRC Press, © 2010 by Taylor and Francis Group, LLC
- [7] S. N. Sivanandam, S. Sumathi and S. N. Deepa Introduction to Fuzzy Logic using MATLAB, © Springer-Verlag Berlin Heidelberg 2007.
- [8] Fuzzy Sets & Fuzzy Logic Theory and Applications George j. Klir and Bo Yuan, © 1995.
- [9] Topic 5 - Fuzzy Logic Control, NgeeAnn the Polytechnic of School of Engineering CCA yck (Jan 2003)
- [10]. Oscar Castillo, 2012, Type-2 Fuzzy Logic in Intelligent Control Applications, 199p, ISBN 978-3-642-24662-3
- [11]. Erdal Kayacan, Graduate Program in Electrical and Electronic Engineering Boğaziçi University 2011, INTERVAL TYPE-2 FUZZY LOGIC SYSTEMS: THEORY AND DESIGN.
- [12]. Suman Lata and Mohammad Ayyub, Interval Type-2 Fuzzy Logic System, ISSN 0974- 2174, Volume 7, Number 3 (2014), pp. 219-224

- [13].Nesrine Baklouti, Robert John and Adel M. Alimi, Interval Type-2 Fuzzy Logic Control of Mobile Robots, Journal of Intelligent Learning Systems and Applications, 2012, 4, 291-302
- [14].Jerry M. Mendel, Robert I. John and Feilong Liu, Interval Type-2 Fuzzy Logic Systems Made Simple, IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 14, NO. 6, DECEMBER 2006
- [15]. Hicham Chaoui and Wail Gueaieb, Type-2 Fuzzy Logic Control of a Flexible-Joint Manipulator, © Springer Science + Business Media B.V. 2007
- [16]. John J. D'Azzo, Constantine H. Houpis and Stuart N. Sheldon, 2003, LINEAR CONTROL SYSTEM ANALYSISAND DESIGN WITH MATLAE, Fifth Edition, Revised and Expanded.
- [17]. Katsuhiko Ogata, 2002, Madern Control Enginring, 976 p.
- [18]- Kelly Liu 7-10-96 Ned Gulley, 4-30-94, Copyright (c) 1994-98 by The MathWorks, Inc. $Revision: 1.32 $ $Date: 1998/09/03 14:17:14 $
- [19]- Hicham Chaoui, Ph.D. May 2015. fuzzy_type2.

Abstract:

The TRMS is versatile platform, its dynamic includes nonlinearities, parametric uncertainties and is subject to unknown external disturbances. Such complicated dynamics involve designing sophisticated control algorithms that can deal with these difficulties. So In this dissertation, the type fuzzy logic controller, as opposed to interval type-2 fuzzy logic controller is proposed for physical system TRMS (twin rotor MIMO system) control problem. That is firstly after an overview of type-1 fuzzy logic controller. And for interval type-2 fuzzy logic controller, Using Gaussian membership functions and based on a human operator experience, two controllers are designed to control the position of the pitch and yaw the angles of the TRMS. Simulation (MATLAB) results are given to illustrate the effectiveness of the proposed control scheme interval type-2 fuzzy logic controller in comparison with type-1 fuzzy logic controller.

Key words: • TRMS system • Nonlinear system

Résume

Le TRMS est plate-forme polyvalente, sa dynamique inclut les non-linéarités, des incertitudes paramétriques et est soumis à des perturbations externes inconnus. Ces dynamiques complexes impliquent la conception des algorithmes de contrôle sophistiqués qui peuvent faire face à ces difficultés. Donc Dans cette thèse, le contrôleur de logique floue type-2, par contre le contrôleur de logique floue intervalle type-2 est proposé pour TRMS de système physique (double système rotor MIMO) de problème de contrôle. Cela est tout d'abord après un aperçu de type-1 contrôleur de logique floue. Et pour le contrôleur de logique floue intervalle type-2, Utilisation de fonctions d'appartenance gaussiennes et basée sur une expérience de l'opérateur humain, deux contrôleurs sont conçus pour contrôler la position du tangage et lacet les angles de la TRMS. Simulation (MATLAB) les résultats sont donnés pour illustrer l'efficacité de la commande proposée régime le contrôleur de logique floue intervalle type-2 en comparaison avec le contrôleur de logique floue type-1.

Les mot clé • système TRMS • système non linéaire

ملخص :

TRMS له قاعدة بيانات متعددة الاستعمالات، تتضمن ديناميكيته نظام غير خطي، وخصائص غير محددة، وهو عرضة لاضطرابات خارجية غير معروفة. مثل هذه الديناميكيات المعقدة تشمل تصميم خوارزمية تحكم معقدة بإمكانها التعامل مع هذه الصعوبات. لذلك، فإن هذه المذكرة تقترح حاكم المنطق الغامض النوع-2 أو بالاحرى حاكم المنطق الغامض مجال النوع-2 للتحكم بالنظام فيزيائي TRMS وهذا بعد إلقاء نظرة عامة على النوع الأول حاكم المنطق الغامض النوع-1 ، و حاكم المنطق الغامض مجال النوع-2 باستخدام دالة الإنتماء نوع Gussian وهذا إعتمادا على الخبير . تم تصميم متحكمان للتحكم في وضعية زاوية إرتفاع وزاوية إنعراج لي TRMS. تم عرض نتائج المحاكاة في برنامج المتلاب لتوضيح فعالية نظام التحكم المقترح حاكم المنطق الغامض مجال النوع-2 مقارنة مع حاكم المنطق الغامض النوع-1.

الكلمات الداله : TRMSنظام ، نظام غير خطي