CHAPTER ONE

INTRODUCTION

1.1 General

Aircraft and missiles are usually equipped with a control system to provide stability, disturbances rejection and reference signal tracking. The motion of an aircraft in free flight is extremely complicated. Generally, aircraft contains three translation motions vertical, horizontal and transverse and three rotational motions pitch, yaw and roll by controlling aileron, rudder and elevator. To reduce the complexity of analysis, the aircraft is usually assumed as a rigid body and aircraft motion consists of a small deviation from its equilibrium flight condition. In addition, the control system of aircraft can be divided into two groups, namely longitudinal and lateral control. In longitudinal control, the elevator controls pitch or the longitudinal motion of aircraft system. The pitch of aircraft is controlled by elevator which usually situated at the rear of the airplane running parallel to the wing that houses the ailerons. Pitch control is a longitudinal problem, and this work presents on design an autopilot that controls the pitch of an aircraft. Autopilot is a pilot relief mechanism that assists in maintaining an attitude, heading, altitude or flying to navigation or landing [1].

This work presents investigation into the development of pitch control schemes for pitch angle of an aircraft by using a fuzzy logic control. Fuzzy PID controller is a combination of conventional PID and fuzzy logic control scheme. Performance of control strategy, PID and PD type fuzzy logic controller with respect to the pitch angle of aircraft longitudinal dynamics are investigated. Simulation is developed within MATLAB/SIMULINK for evaluation of the both control strategies.

To demonstrate the effectiveness of the purposed control schemes, the comparative assessment on the system performance for each controller is presented and discussed. Many research works have been done in [1, 11, 14-17] to control the pitch or longitudinal dynamic of an aircraft for flight stability, this research is still remains an open issue in the present and future works.

1.2 Problem Statement

Attitude tracking problem are important examples for non-linear control problems. For example, the response of non-linear plant cannot be tracked into desired pattern linear controller. Thus a changing dynamic controller is important to control thus plant. The targets of an aircraft attitude control are to control or help a pilot to keep pitch attitude constant, stable and make the aircraft return to desired attitude in a reasonable length of time. After disturbance of pitch angle, the conventional PID controllers are used extensively even though they are not very efficient for non-linear dynamic system.

1.3 Objectives

▪ Design of conventional PID controller for an aircraft pitch attitude.

▪ Design of fuzzy logic controller for an aircraft pitch attitude.

▪ Simulation of the proposed control systems developed on MATLAB/SIMULINK.

▪ Comparison of different controllers proposed.

1.4 Methodology

▪ Survey of previous related studies.

▪ Develop of the mathematical model of an aircraft system.

▪ Use of fuzzy logic Mamdani type.

▪ Use of MATLAB Toolbox to model, design and simulate of the proposed system.

1.5 Research Layout

► Chapter one introduced the general overview, problem statement, objective and methodology.

► Chapter two describes briefly the aircraft control systems, the conventional PID and fuzzy logic controller and their tuning parameters.

► Chapter three explains and demonstrates the mathematical model of an aircraft pitch control system and SIMULINK modeling a designing of an aircraft pitch with (conventional and fuzzy) PID controllers.

► Chapter four presents the results of the performance of Mamdani model and the application of designed controllers to the nonlinear model.

► Chapter five concluding remarks on the results obtained and possible future work of the thesis.

CHAPTER TWO

FLIGHT CONTROL SYSTEM

2.1 Introduction

A flight control system is either a primary or secondary system. Primary flight controls provide longitudinal (pitch), directional (yaw), and lateral (roll) control of the aircraft. Secondary flight controls provide additional lift during takeoff and landing, and decrease aircraft speed during flight, as well as assisting primary flight controls in the movement of the aircraft about its axis. Some manufacturers call secondary flight controls auxiliary flight controls. All systems consist of the flight control surfaces, the respective cockpit controls, connecting linkage, and necessary operating mechanisms. Basically, there are three type of flight control systems as discussed below**.**

2.1.1 Mechanical flight control

Mechanical flight control systems are the most basic designs. They are basically unposted flight control systems. They were used in early aircraft and currently in small aero planes where the aerodynamic forces are not excessive. The flight control surfaces (ailerons, elevators, and rudder) are moved manually through a series of push-pull rods, cables, bell cranks, sectors, and idlers. Since an increase in control surface area in bigger and faster aircraft leads to a large increase in the forces needed to move them, complicated mechanical arrangements are used to extract maximum mechanical advantage to make the forces required bearable to the pilots. This arrangement is found on bigger or higher performance propeller aircraft. Some mechanical flight control systems use servo tabs that provide aerodynamic assistance to reduce complexity. Servo tabs are small surfaces hinged to the control surfaces. The mechanisms move these tabs, aerodynamic forces in turn move the control surfaces reducing the amount of mechanical forces needed. This arrangement was used in early piston-engine transport aircraft and early jet transports. The primary flight control system is illustrated in Figure 2.1 [2].

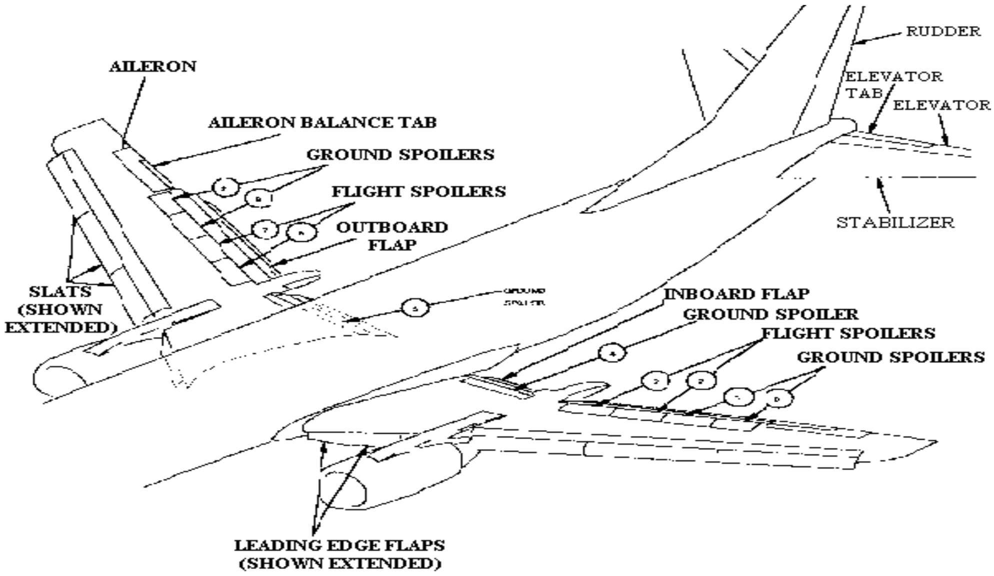


Figure 2.1: The primary flight control system

2.1.2 Hydro mechanical flight control

They are power boosted flight control systems. The complexity and weight of a mechanical flight control systems increases considerably with size and performance of the airplane. Hydraulic power overcomes these limitations. With hydraulic flight control systems aircraft size and performance are limited by economics rather than a pilot's strength. In the power-boosted system, a hydraulic actuating cylinder is built into the control linkage to assist the pilot in moving the control surface. A hydraulic flight control system has two parts mechanical circuit and hydraulic circuit.

The mechanical circuit links the cockpit controls with the hydraulic circuits. Like the mechanical flight control systems, it is made of rods, cables, pulleys, and sometimes chains. The hydraulic circuit has hydraulic pumps, pipes, valves and actuators. The actuators are powered by the hydraulic pressure generated by the pumps in the hydraulic circuit. The actuators convert hydraulic pressure into control surface movements. The servo valves control the movement of the actuators. The pilot’s movement of a control causes the mechanical circuit to open the matching servo valves in the hydraulic circuit. The hydraulic circuit powers the actuators which then move the control surfaces. These powered flight controls are employed in high performance aircraft, and are generally of two types (i) power assisted and (ii) power operated, which are shown in the Figures 2.2 and 2.3 respectively. Both systems are similar in basic forms but to overcome the aerodynamic loads forces are required, which decides the choice of either of the above system [2].

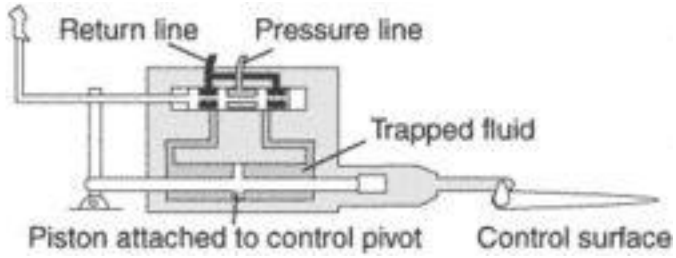


Figure 2.2: Power assisted flight control system

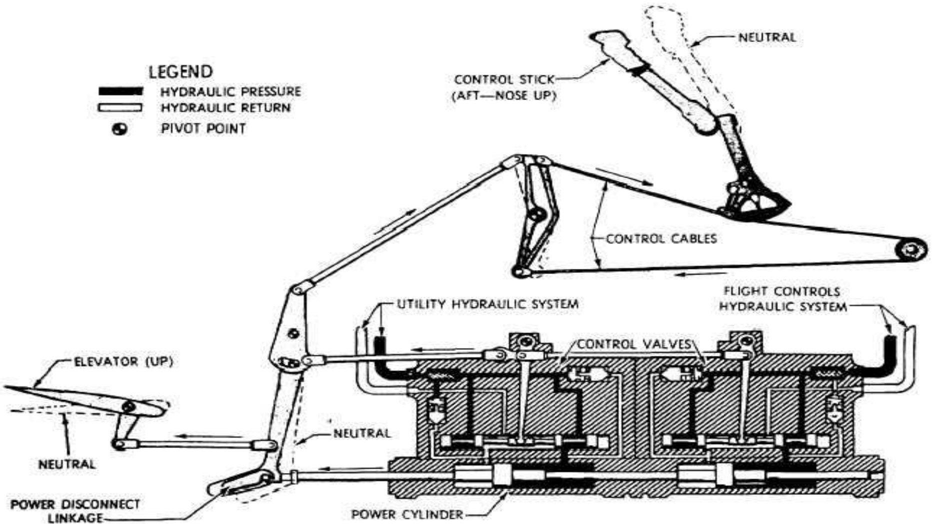


Figure 2.3: Power operated flight control system

2.1.3 Fly by wire

Fly-By-Wire (FBW) is a means of aircraft control that uses electronic circuits to send inputs from the pilot to the motors that move the various flight controls on the aircraft. There are no direct hydraulic or mechanical linkages between the pilot and the flight controls. The total elimination of all the complex mechanical control runs and linkages-all commands and signals are transmitted electrically along wires, hence the name fly-by-wire and there is interposition of a computer between the pilot’s commands and the control surface actuators which is incorporated with air data sensors which supply height and airspeed information to the computer. The fly-by-wire flight control system is shown in Figure 2.4 [2].

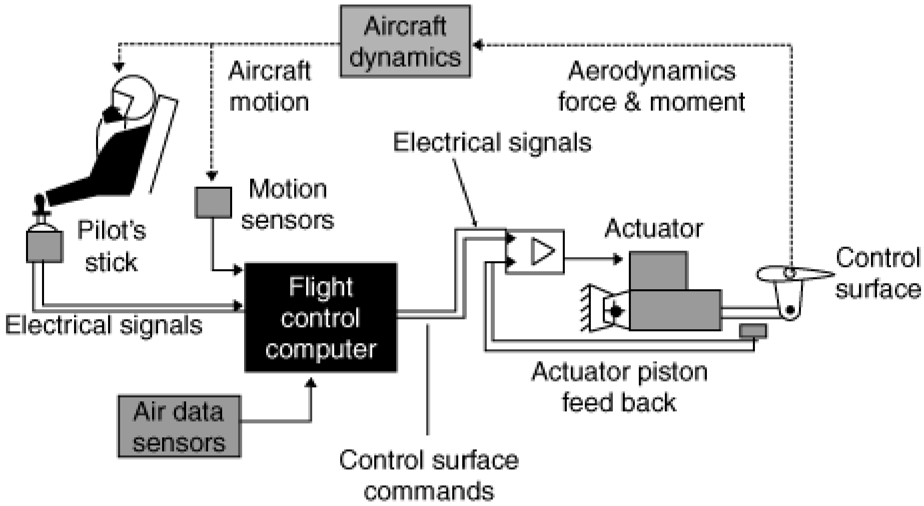


Figure 2.4: The fly-by-wire flight control system

Mechanical and hydraulic flight control systems are heavy and require careful routing of flight control cables through the airplane using systems of pulley and cranks. Both systems often require redundant backup, which further increases weight. The advantages of FBW system are:

- Provides high-integrity automatic stabilization of the aircraft to compensator the loss of natural stability and thus enables a lighter aircraft with a better overall performance.
- Makes the ride much smoother than one controlled by human hands. The capability of FBW systems to maintain constant flight speeds and altitudes over long distances is another way of increasing fuel efficiency. The system acts much like cruise controls on automobiles: less fuel is needed if throttles are untouched over long distances.
- More reliable than a mechanical system because of fewer parts to break or malfunction. FBW is also easier to install, which reduces assembly time and costs. FBW maintenance costs are lower because they are easier to maintain and troubleshoot, and need fewer replacement parts.

- Electrical wires for a flight control system takes up less space inside fuselages, wings, and tail components. This gives designers several options. Wings and tail components can be designed thinner to help increase speed and make them aerodynamically cleaner, and also to reduce weight. Space once used by mechanical linkages can also be used to increase fuel capacities to give the aircraft greater range or payload.

2.2 Principle of Flight Control

The four basic forces acting upon an aircraft during flight are lift, weight, drag and thrust as shown in Figure 2.5 [3].

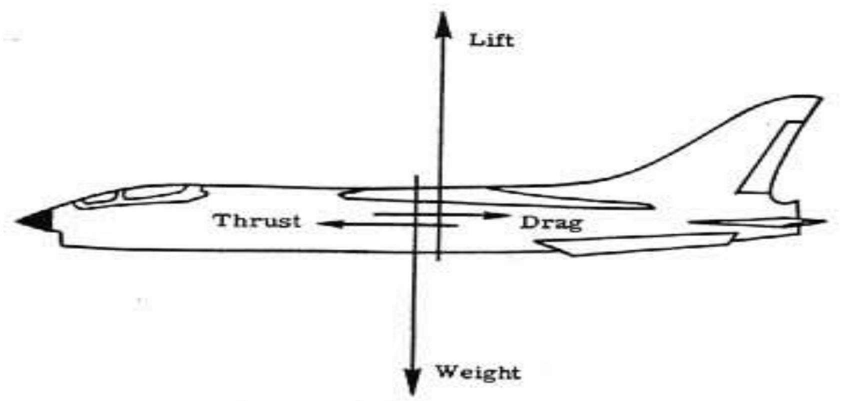


Figure 2.5: Forces acting on an aircraft

▪ **Lift**

Lift is caused by the flow around the aircraft. Lift is the upward force created by the wings, which sustains the airplane in flight. The force required to lift the plane through a stream of air depends upon the wing profile. When the lift is greater than the weight then the plane raises.

▪ **Weight**

Weight is the downward force created by the weight of the airplane and its load; it is directly proportional to lift. If the weight is greater than lift then the plane descends.

▪ **Drag**

The resistance of the airplane to forward motion directly opposed to thrust. The drag of the air makes it hard for the plane to move quickly. Another name for drag is air resistance. It is created or caused by all the parts.

▪Thrust

The force exerted by the engine which pushes air backward with the object of causing a reaction, or thrust, of the airplane in the forward direction [3].

2.3 Flight Control Surfaces

An aircraft requires control surfaces to fly and move in different directions. They make it possible for the aircraft to roll, pitch and yaw. Figure 2.6 shows the three sets of control surfaces and the axes along which they tilt.

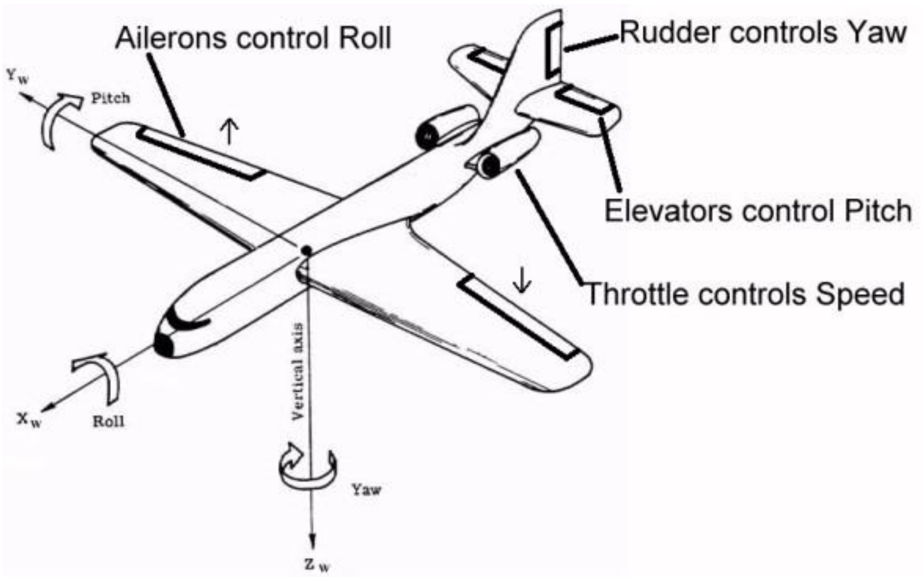


Figure 2.6: The control surfaces and axes

2.3.1 Primary control surfaces

The primary flight controls surfaces are ailerons, elevator and Rudder.

▪ Ailerons

Movement about the longitudinal axis is controlled by the two ailerons, which are movable surfaces at the outer trailing edge of each wing. The movement is roll. If the aileron on one wing is lowered, the aileron on the other will be raised. The wing with the raised aileron goes down because of its decreased lift and the wing with the lowered aileron goes up because of its increased lift. Thus, the effect of moving one of the ailerons is complemented by the simultaneous and opposite movement of the aileron on the other wing.

The ailerons are connected to each other and to the control wheel (or stick) in the cockpit by rods or cables. While applying pressure to the right on the control wheel, the right aileron goes up and the left aileron goes down. Thus, the airplane is rolled to the right as the down movement of the left aileron increases the wing camber (curvature) and the angle of attack. The right aileron moves upward and decreases the camber, what results in a decreased angle of attack. Thus, an increased lift on the left wing and decreased lift on the right wing cause a roll and bank to the right.

- **Elevators**

The movement of the airplane about its lateral axis is controlled by the elevators. This motion is called pitch. The elevators are free to swing up and down and form the rear part of the horizontal tail assembly. They are hinged to a fixed surface; the horizontal stabilizer. A single airfoil is formed by the horizontal stabilizer and the elevators. The chamber of the airfoil can be modified by changing the position of the elevators, which increases or decreases the lift.

Control cables are used to connect the elevators to the control wheel (or stick) as it happens with the ailerons. The elevators move downward when forward pressure is applied on the wheel. Thus, the lift produced by the horizontal tail surfaces is increased, what forces the tail upward, causing the nose to drop. Conversely, the elevators move upward, when back pressure is applied on the wheel, decreasing the lift produced by the horizontal tail surfaces, or maybe even producing a downward force. The nose is forced upward and the tail is forced down.

The angle of attack of the wings is controlled by the elevators. When back pressure is applied on the control wheel, the angle of attack increases as the tail lowers and the nose rises. Conversely, the tail raises and the nose lowers when forward pressure is applied, decreasing the angle of attack.

- **Rudder**

The movement of the airplane about its vertical axis is controlled by the rudder. This motion is called yaw. The rudder is a movable surface hinged to a fixed surface which is the vertical stabilizer, or fin. Its action is like the one of the elevators, except that it swings in a different plane; from side to side instead of up and down. The rudder is connected to the rudder pedals by controlled cables.

2.3.2 Secondary control surfaces

Wing leading and trailing edges are used to increase the aerodynamic performance of the aircraft by reducing stall speed mainly during take-off and landing speed. High lift control is provided by a combination of flaps and leading edge slats. The flap control is affected by several flap sections located on the inboard two-thirds of the wing trailing edges. The flaps are deployed during take-off or the landing approach to increase the wing camber and improve the aerodynamic characteristics of the wing.

▪ **Flaps**

Flaps are mounted on the trailing edge but can also be mounted on the leading edge. They extend the edge by increasing the chord of the wing. They pivot only (simple and split flaps), extend and come down (complex and slotted flaps) or extend and camber (Krueger flaps).

▪ **Slats**

Slats are usually mounted on the leading edge. Slats extend the edge and they sit like a glove on the edge. Slats are an abbreviation for "slotted flaps", which means they have a nozzle like slot between the high-lift device and the wing; on the contrary, flaps do not have this slot [3].

2.4 Aircraft Dynamics

Aerodynamics concerns the motion of air and other gaseous fluids and other forces acting on objects in motion through the air (gases). In effect, aerodynamics is concerned with the object (aircraft), the movement (relative wind), and the air (atmosphere). There are four main forces that act on an aircraft in flight; they are weight, lift, thrust, and drag. It is the interaction between these four forces that result in an airplane's motion as shows in Figure 2.7. An aircraft in flight is free to rotate in three dimensions, the axis that expand lengthwise (nose through tail) is called the longitudinal axis, and rotation regarding this axis is called roll. Pitching moment is the moment acting in the plane containing the lift and the drag, i.e. in the lateral plane when the aircraft is flying horizontally. It is positive when it tends to increase the incidence, or raise the nose of the aircraft upward that goes by vertically through the center of gravity is called the vertical axis, and rotation regarding this axis is called yaw [4].

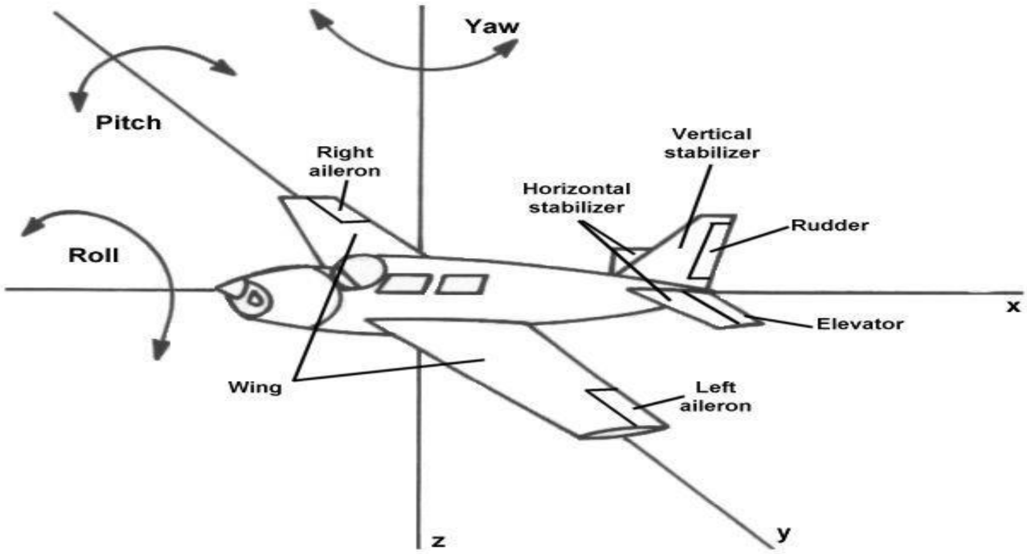


Figure 2.7: The axes of rotation

2.5 Aircraft Longitudinal Control

Control of airplane can be achieved by providing an incremental lift force on one or more of the airplane's lifting surfaces. The incremental lift force can be produced by deflecting the entire lifting surface or by deflecting a flap incorporated in the lifting surface. Because the control flaps of movable lifting surfaces are located at some distance from the center of gravity, the incremental lift force creates a moment about the airplane's center of gravity. If the flap is used, Figure 2.8 shows the three-primary aerodynamic control.

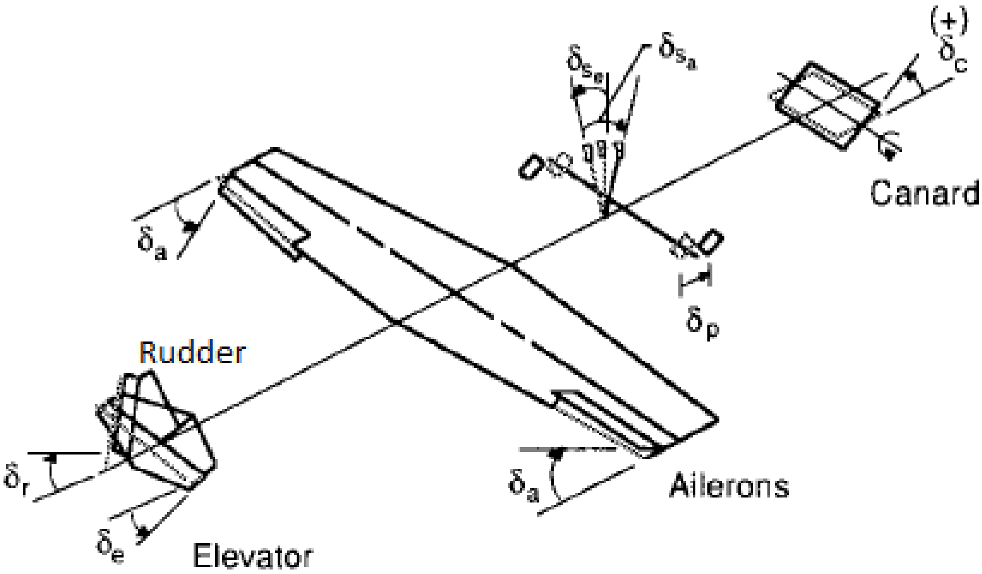


Figure 2.8: The primary aerodynamics control

Pitch control can be achieved by changing the lift on either a forward or aft control surface. If a flap is used, the flapped potion of the vertical trial called elevator. Yow control is achieved by deflecting a flap on the vertical tail called the rudder and roll control can be achieved by deflecting small flaps located outboard toward the wing tips in differential manner. These flaps are called ailerons. A roll moment can also be produced by deflecting a wing spoiler. As the name implies a spoiler disrupts the lift. This is accomplished by deflecting a section of upper wing surface so that flow separate behind the spoiler which causes a reduction in the lifting force. To achieve roll moment, only one spoiler needs to be deflected [4].

2.6 The Displacement Autopilot

One of the earliest autopilots to be used to aircraft control is so-called displacement autopilot. A displacement type autopilot can be used to control the angular orientation of the air plane. Conceptually, the displacement autopilot works in the following manner. In a pitch attitude displacement autopilot, pitch angle is sensed by vertical gyro and compared with desired pitch angle to create an error angle. The difference or error in pitch attitude is used to produce proportional displacement of the elevator so that the error signal is reduced. Figure 2.9 shows the block diagram representation of either a pitch and roll angle displacement autopilot [4].

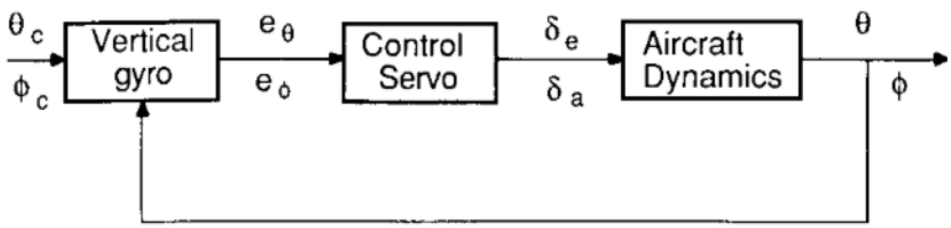


Figure 2.9: The block diagram of pitch or roll displacement autopilot

The basic components of pitch attitude control are shown in Figure 2.9. For this design the reference of pitch angle is compared with actual angle measured by a gyro to produce an error signal to active control servo. In general, the error signal is amplified and sends to the control surface actuator to deflect the control surface. Movement of the control surface causes aircraft achieves a new pitch orientation, which is the fed back to close the loop [4].

2.7 Attitude Control Systems

Attitude control systems find extensive employment on modern aircraft. They form the essential functions of any Automatic Flight Control System (AFCS), in that they allow an aircraft to be placed, and maintained, in any required, specified orientation in space, either in direct response to a pilot's command, or in response to command signals obtained from an aircraft's guidance, or weapons systems. It is through their agency that - unattended operation of an aircraft is possible. In AFCS work, attitude hold, the commonest function, is often referred to, especially in the USA, as a Control Wheel Steering (CWS) mode.

Stability augmentation systems often form the inner loops of attitude control systems; the attitude control systems then form the inner loops for the path control systems. It is often the case that attitude control systems need to use simultaneously several of the aircraft's control surfaces, or they may require the use of feedback signals which depend upon motion variables other than those being controlled directly. Attitude control systems are, consequently, more complex in their operation than stability augmentation systems. Pitch attitude control systems have traditionally involved the use of elevator only as the control in the system. A block diagram of a typical system is shown in Figure 2.10, the representations of the dynamics of both the elevator's actuator and the sensor of pitch attitude are still maintained here.

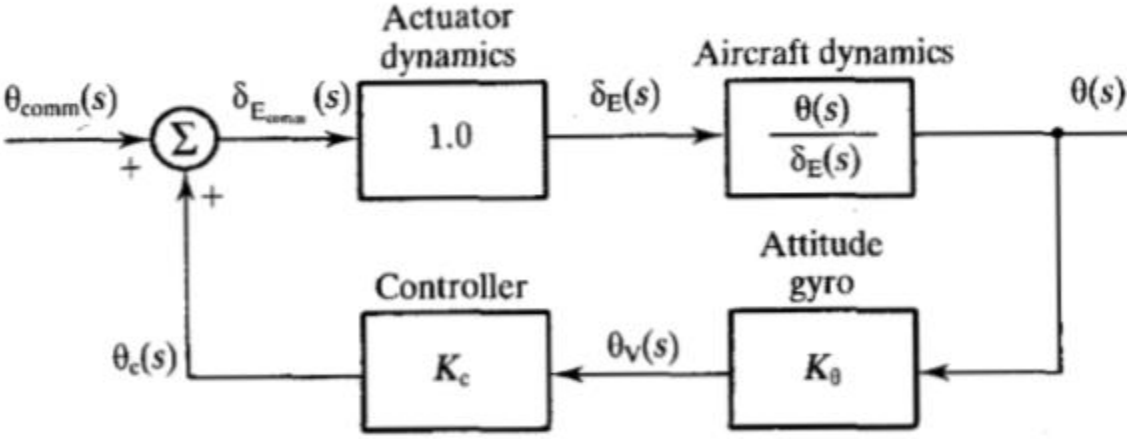


Figure 2.10: The block diagram of pitch attitude control system

Therefore, the feedback control law being considered can be generally expressed in the form:

$$\delta_e = K_c K_\theta \theta \tag{2.1}$$

As the feedback gain, $K_c K_\theta$ is increased, it is found that the aircraft's short period frequency w_{sp}, also increases, although its damping ratio, ζ_{sp}, decreases; however, the damping ratio of the phugoid, ζ_{ph}, increases. The period of the phugoid motion also increases until the mode becomes over-damped and, consequently non-oscillatory [5].

2.8 PID Controller

The PID controller is the most common form of feedback. It was an essential element of early governors and it became the standard tool when process control emerged in the 1940s. In process control, today, more than 95% of the control loops are of PID type, most loops are PI control. PID controllers are today found in all areas where control is used. The controllers come in many different forms. There are stand-alone systems in boxes for one or a few loops, which are manufactured by the hundred thousand yearly. PID control is an important ingredient of a distributed control system. The controllers are also embedded in many special-purpose control systems. PID control is often combined with logic, sequential functions, selectors, and simple function blocks to build the complicated automation systems used for energy production, transportation, and manufacturing. Many sophisticated control strategies, such as model predictive control, are also organized hierarchically. PID control is used at the lowest level; the multivariable controller gives the set points to the controllers at the lower level. The PID controller can thus be said to be the “bread and butter” of control engineering. It is an important’ component in every control engineer’s tool box. PID controllers have survived many changes in technology, from mechanics and pneumatics to microprocessors via electronic tubes, transistors, integrated circuits. The microprocessor has had a dramatic influence on the PID controller. Practically all PID controllers made today are based on microprocessors. This has given opportunities to provide additional features like automatic tuning, gain scheduling, and continuous adaptation [6].

2.8.1 The PID algorithm

By summarizing the key features of the PID controller, the PID algorithm is described by:

$$u(t) = K\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\,\frac{de(t)}{dt}\right) \tag{2.2}$$

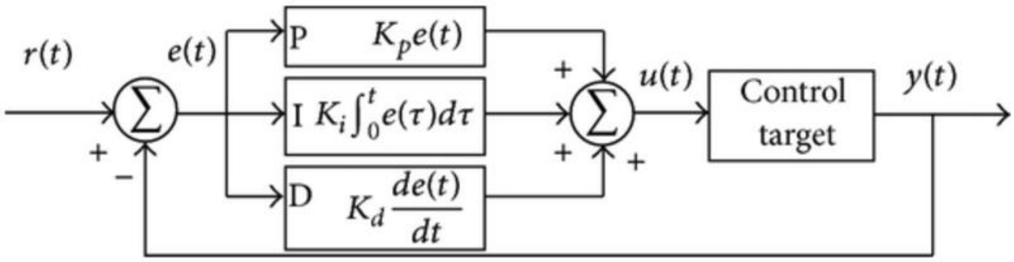


Figure 2.11: The PID algorithm

Where y is the measured process variable, r the reference variable, u is the control signal and e is the control error (e = y_{SP} − y). The reference variable is often called the set point. The control signal is thus a sum of three terms: The P-term (which is proportional to the error), the I-term (which is proportional to the integral of the error), and the D-term (which is proportional to the derivative of the error). The controller parameters are proportional gain K, integral time T_i, and derivative time T_d. The integral, proportional and derivative part can be interpreted as control actions based on the past, the present and the future as is illustrated in Figure 2.12. The derivative part can also be interpreted as prediction by linear extrapolation as is illustrated in Figure 2.12 [6].

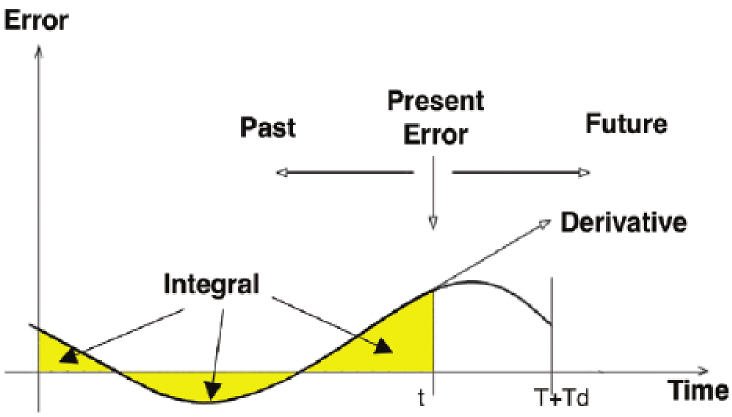


Figure 2.12: A PID controller takes control action

2.8.2 Effects of proportional, integral and derivative action

Proportional control is illustrated in Figure 2.13. The controller is given by Equation (2.2) with $T_i = \infty$ and $T_d = 0$. The process transfer function is:

P(s) = 1/ (s + 1) ^3 (2.3)

The figure shows that there is always a steady state error in proportional control. The error will decrease with increasing gain, but the tendency towards oscillation will also increase [6].

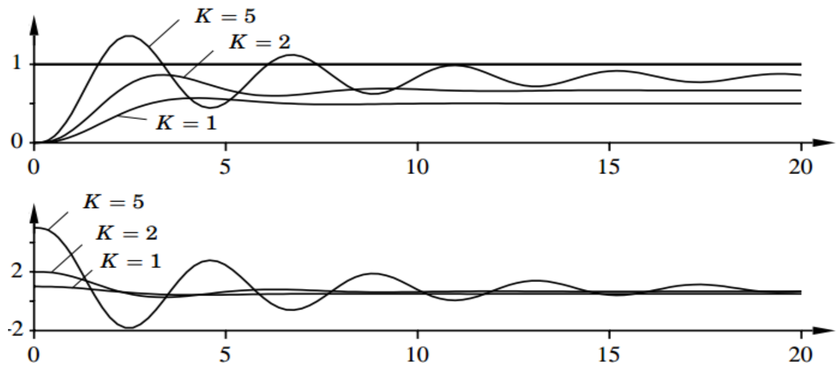


Figure 2.13: Simulation of a closed-loop system with P control

Figure 2.14 illustrates the effects of adding integral. It follows from Figure 2.13 that the strength of integral action increases with decreasing integral time T_i . The Figure shows that the steady state error disappears when integral action is used. The tendency for oscillation also increases with decreasing T_i [6].

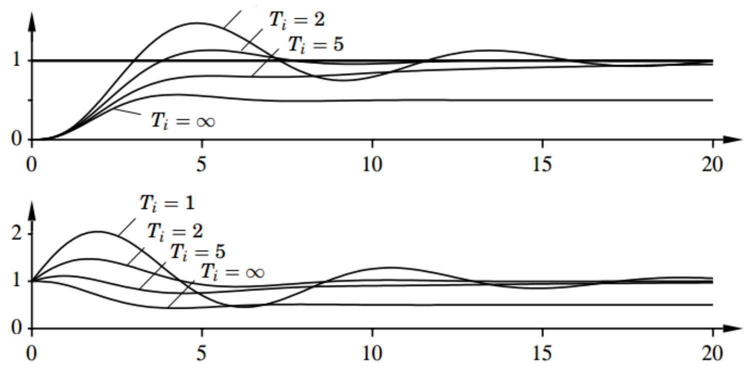


Figure 2.14: Simulation of a closed-loop system with P and I control

The properties of derivative action are illustrated in Figure 2.15. Figure 2.15 illustrates the effects of adding derivative action. The parameters K and T_i are chosen so that the closed-loop system is oscillatory. Damping increases with increasing derivative time, but decreases again when derivative time becomes too large. Recall that derivative action can be interpreted as providing prediction by linear extrapolation over the time T_d. Using this interpretation, it is easy to understand that derivative action does not help if the prediction timeT_d is too large. In Figure 2.15 the period of oscillation is about 6 s for the system without derivative action. The process transfer function in Equation 2.3, the controller gain is K = 3, and the integral time is T_i = 2. Derivative actions cease to be effective when T_d is larger than a1s (one sixth of the period). Also, notice that the period of oscillation increases when derivative time is increased [6].

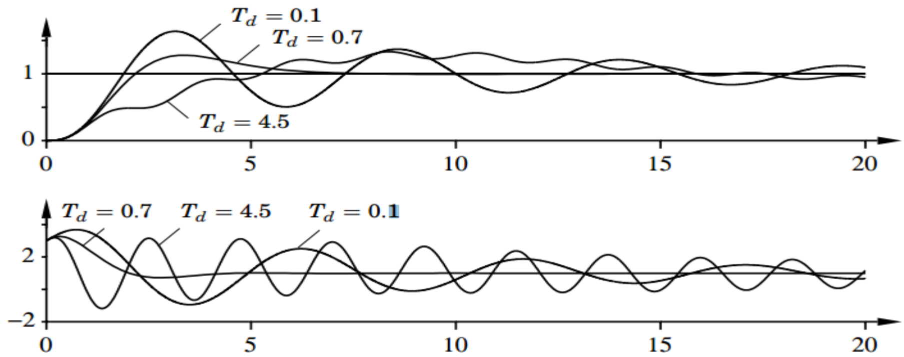


Figure 2.15: Simulation of a closed-loop system with P, I, and D control

2.8.3 PID controller tuning methods

PID controllers are probably the most commonly used controller structures in industry. They do however present some challenges to control the system in the aspect of tuning for the gains required for stability and good transient performance. There are several rules used in PID tuning. One example is that proposed by Ziegler and Nichols in the 1940's [7].

• **Ziegler-Nichols method:**

As a result of empirical tests on a wide variety of process plant, Ziegler and Nichols propose a simple rule of thumb procedure for estimating the values of the controller setting K_P, K_i and K_d for existing operating plant to achieve an optimum transient response. There are two methods, one based on the step response of the open loop system and the other based on information obtained at the stability limit of the process under proportional control. In the first method with the loop open, the plant is subjected to a step change of manipulated variable and the resulting output response curve is characterized by two measured parameters N and L (See Figure 2.16). N is the maximum slope of the curve for a change M of manipulated variable and L is the time at which the line of maximum slope intersects the time axis. The recommendations which Ziegler and Nichols put forward for the controller setting are: [7].

$$\begin{bmatrix} K_c = \frac{M}{L}\ , & & for\ P\ control \\ K_C = 0.9\frac{M}{L}\ ,\ T_i = 3.3L, & & for\ P\ + I\ control \\ K_C = 1.2\ \frac{M}{L}, T_i = 2L,\ T_d = 0.5\ L, & & for\ P + I + D\ control \end{bmatrix} \tag{2.4}$$

Where K_C, T_i and T_d are the parameter values of controller gain, integral action time and derivative action time respectively, as they appear in the control law:

$$G_c(s) = K_C(1 + \frac{1}{T_i S} + T_d S \tag{2.5}$$

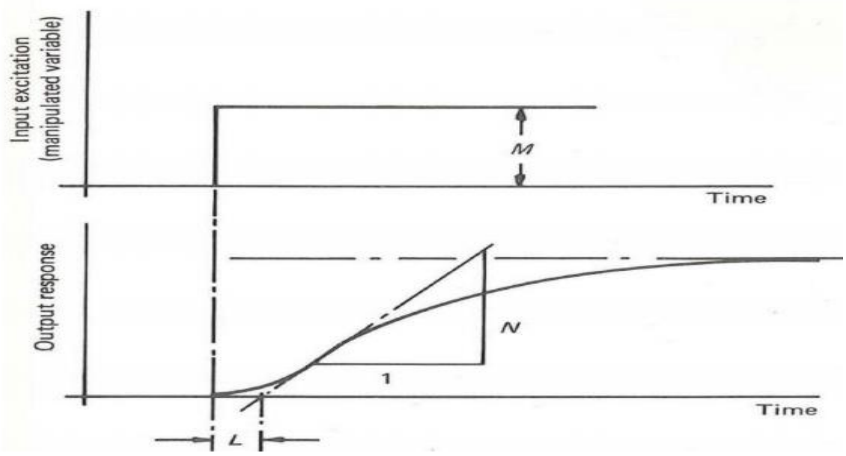


Figure 2.16: Ziegler-Nichols first method

The procedure of the second method is to determine experimentally the limiting condition of stability of the closed loop system under proportional control only, and to use the resulting information to calculate the controller settings. If the limiting value of the gain for stability is K_{crit} and the time period of oscillation is P_{crit}, then the Ziegler-Nichols recommended controller setting are:

$$\begin{bmatrix} K_c = 0.5K_{crit} & for\ I\ control \\ K_C = 0.45K_{crit}\ , T_i = \ 0.83\ P_{crit} & for\ P\ + I\ control \\ K_C = 0.6\ K_{crit}, T_i = \ 0.5\ P_{crit}, T_d = 0.125\ P_{crit} & forP + I + D\ control \end{bmatrix} \tag{2.6}$$

- **Trial and error method:**

PID tuning is the process of finding the proper values of K_P, K_i and K_d gains. The aim of the controller tuning is to obtain both of the following for the control system if possible:

- Fast response.
- Good stability.

Unfortunately, for practical system these two wishes cannot be achieved simultaneously. In other words:

- Faster the response, the worst the stability.
- The better the stability, the slower the response.

So, for the slider control system, some compromises had to be made that will still give the system a real-time response. Real-time does not mean very fast but rather absolute liability. The compromises were to have:

- Acceptable stability.
- Medium fastness of response.

The only way the goal could be achieve was by trial and error; that is, we run the system and set all gains to zero. Little by little the gains are change until the desire state is met. One may ask how we classify acceptable stability more specifically. There is exact definition but a rule of thumb is when the set point assumes a positive step change. Acceptable stability is when the undershoot that follows the first overshoot of the response is small or barely observable.

- **Automatic PID tuning:**

MATLAB provides tools for automatically choosing optimal PID gains which makes the trial and error process described above unnecessary. The tuning algorithm can be accesses directly using PID tune or through a nice Graphical User Interface (GUI) using PID tool. The MATLAB automated tuning algorithm chooses PID gains to balance performance (response time, bandwidth) and robustness (stability margins) [8].

2.9 Fuzzy Logic Control

Fuzzy logic idea is like the human being's feeling and inference process. Unlike classical control strategy, which is a point-to-point control, fuzzy logic control is a range-to-point or range-to-range control. The output of a fuzzy controller is derived from fuzzifications of both inputs and outputs using the associated membership functions. A crisp input will be converted to the different members of the associated membership functions based on its value. From this point of view, the output of a fuzzy logic controller is based on its memberships of the different membership functions, which can be considered as a range of inputs. Fuzzy ideas and fuzzy logic are so often utilized in our routine life that nobody even pays attention to them. For instance, to answer some questions in certain surveys, most time one could answer with 'not very satisfied' or 'quite satisfied', which are also fuzzy or ambiguous answers. Exactly to what degree is one satisfied or dissatisfied with some service or product for those surveys? These vague answers can only be created and implemented by human beings, but not machines. Is it possible for a computer to answer those survey questions directly as a human being did? It is impossible. Computers can only understand either '0' or '1', and 'HIGH' or 'LOW'. Those data are called crisp or classic data and can be processed by all machines. Is it possible to allow computers to handle those ambiguous data with the help of a human being? If so, how can computers and machines handle those vague data? The answer to the first question is yes. But to answer the second question, we need some fuzzy logic techniques and knowledge of fuzzy inference system [9].

The idea of fuzzy logic was invented by professor L. A. Zadeh of the University of California at Berkeley in 1965. This invention was not well recognized until Dr. E. H. Mamdani, who is a professor at London University, applied the fuzzy logic in a practical application to control an automatic steam engine in 1974, which is almost ten years after the fuzzy theory was invented. Then, in 1976, Blue Circle Cement and SIRA in Denmark developed an industrial application to control cement kilns. That system began to operation in 1982. More and more fuzzy implementations have been reported since the 1980s, including those applications in industrial manufacturing, automatic control, automobile production, banks, hospitals, libraries and academic education. Fuzzy logic techniques have been widely applied in all aspects today. To implement fuzzy logic technique to a real application requires the following three steps [9]:

- Fuzzification: convert classical data or crisp data into fuzzy data or membership functions (MFs).

- Fuzzy Inference Process: combine membership functions with the control rules to derive the fuzzy output.
- Defuzzification: use different methods to calculate each associated output and put them into a table: the lookup table. Pick up the output from the lookup table based on the current input during an application.

Figure 2.17 illustrates the component of the fuzzy logic system.

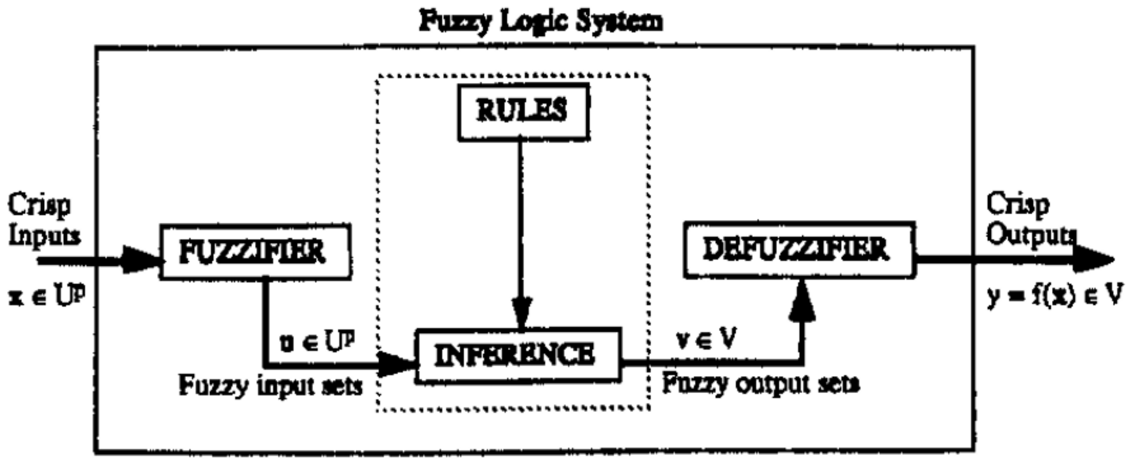


Figure 2.17: The basic component of fuzzy logic system

2.9.1 Fuzzy sets

The concept of the fuzzy set is only an extension of the concept of a classical or crisp set. The fuzzy set is a fundamentally broader set compared with the classical or crisp set. The classical set only considers a limited number of degrees of membership such as ‘0’ or ‘1’, or a range of data with limited degrees of membership. For instance, if a temperature is defined as a crisp high, its range must be between $80F^{\circ}$and higher and it has nothing to do with $70F^{\circ}$or even $60F^{\circ}$. But the fuzzy set will take care of a much broader range for this high temperature. In other words, the fuzzy set will consider a much larger temperature range such as from $0F^{\circ}$to higher degrees as a high temperature. The exact degree to which the $0F^{\circ}$can contribute to that high temperature depends on the membership function. This means that the fuzzy set uses a universe of discourse as its base and it considers an infinite number of degrees of membership in a set. In this way, the classical or crisp set can be considered as a subset of the fuzzy set [9].

2.9.2 Mapping of classical sets to functions

The classical set we discussed in the last section can be represented or mapped into some functions. This means that set-theoretic forms can be related to function theoretic representations, and map elements on one universe of discourse to elements or sets in another universe, very easy and straightforward. Assume that X and Y are two different universes of discourse. If an element x belongs to X and it corresponds to an element y belonging to Y, the mapping between them can be expressed as:

$$\mu_A \text{ (x)} = \begin{Bmatrix} 1, & x \in A \\ 0 & x \notin A \end{Bmatrix} \tag{2.7}$$

Where μ_A represents 'membership' in a set A for the element x in the universe. This membership idea is exactly a mapping from thee element x in the universe X to one of two elements in universe Y, or to elements 0 or 1. Figure 2.18 shows classical set and its operations [9].

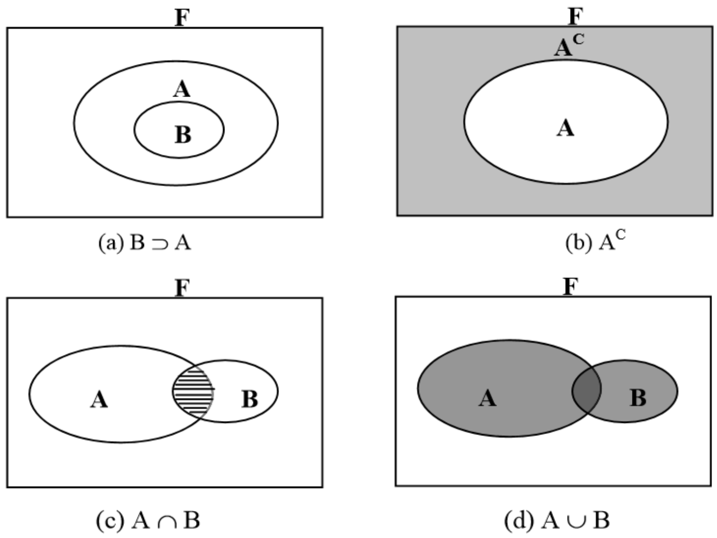
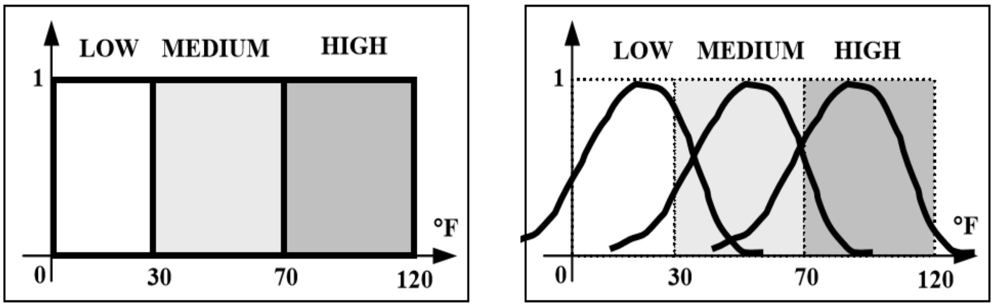


Figure 2.18: The classical set and its operations

2.9.3 Fuzzy sets and operations

As discussed in the previous section, the classical set has a sharp boundary, which means that a member either belongs to that set or does not. Also, this classical set can be mapped to a function with two elements, 0 or 1. For example, in the previous section, the faculty member is defined in the department of computer science as set A. A faculty either fully belongs to this set (μ_A (x)=1) if one is a faculty of the computer science department or has nothing to do with set A (μ_A(x)=0) if he is not a faculty in that department. This mapping is straightforward with sharp boundary without any ambiguity. In other words, this 'fully belonging to' can be mapped as a member of set A with degree of 1, and 'not belong to' can be mapped as a member of set A with degree of 0. This mapping is like a black-and-white binary categorization. Compared with a classical set, a fuzzy set allows members to have a smooth boundary. In other words, a fuzzy set allows a member to belong to a set to some partial degree. For instance, still using the temperature as an example, the temperature can be divided into three categories: LOW ($0F^{\circ}$~$30F^{\circ}$), MEDIUM ($30F^{\circ}$~$70F^{\circ}$) and HIGH ($70F^{\circ}$~$120F^{\circ}$) from the point of view of the classical set, which is shown in Figure 2.19a.



(a) Classical set (b) Fuzzy set

Figure 2.19. Representations of classical and fuzzy sets

A fuzzy set contains elements which have varying degrees of membership in the set, and this is contrasted with the classical or crisp sets because members of a classical set cannot be members unless their membership is full or complete in that set. A fuzzy set allows a member to have a partial degree of membership and this partial degree membership can be mapped into a function or a universe of membership values [9].

2.9.4 Fuzzy rules

In 1973, Lotfi Zadeh published his second most influential paper (Zadeh, 1973). This paper outlined a new approach to analysis of complex systems, in which Zadeh suggested capturing human knowledge in fuzzy rules. A fuzzy rule can be defined as a conditional statement in the form [10]:

IF x is A THEN y is B (2.8)

Where x and y are linguistic variables; A and B are linguistic values determined by fuzzy sets on the universe of discourses X and Y, respectively.

2.9.5 Fuzzy inference

Fuzzy inference can be defined as a process of mapping from a given input to an output using the theory of fuzzy sets.

- **Mamdani-style inference:**

The most commonly used fuzzy inference technique is the so-called Mamdani method. In 1975, professor Ebrahim Mamdani of London University built one of the first fuzzy systems to control a steam engine and boiler combination. He applied a set of fuzzy rules supplied by experienced human operators. The Mamdani-style fuzzy inference process is performed in four steps: Fuzzification of the input variables, rule evaluation, aggregation of the rule outputs, and finally defuzzification. To see how everything fits together, we examine a simple two-input one output problem that includes three rules [10]:

Rule: 1		Rule: 1	
IF	*x* is *A*3	IF	*project_funding* is *adequate*
OR	*y* is *B*1	OR	*project_staffing* is *small*
THEN	*z* is *C*1	THEN	*risk* is *low*
Rule: 2		Rule: 2	
IF	*x* is *A*2	IF	*project_funding* is *marginal*
AND	*y* is *B*2	AND	*project_staffing* is *large*
THEN	*z* is *C*2	THEN	*risk* is *normal*
Rule: 3		Rule: 3	
IF	*x* is *A*1	IF	*project_funding* is *inadequa*
THEN	*z* is *C*3	THEN	*risk* is *high*

where x, y and z (project funding, project staffing and risk) are linguistic variables; A1, A2 and A3 (inadequate, marginal and adequate) are linguistic values determined by fuzzy sets on universe of discourse X (project funding); B1 and B2 (small and large) are linguistic values determined by fuzzy sets on universe of discourse Y (project staffing); C1, C2 and C3 (low, normal and high) are linguistic values determined by fuzzy sets on universe of discourse Z (risk) [10].

- **Takagi-Sagano (T-S) style inference:**

The heuristic technique of Mamdani fuzzy control mentioned in above section lacks the mathematical rigor required to conduct a systematic analysis needed for flight approval although the nonlinear and robust nature of fuzzy control is suited for flight controls. The TS model retains the advantages of the fuzzy control, and it is also constructed in a mathematically rigorous method and thus, stability and control analysis has been developed. In T-S fuzzy model, each rule is represented by a linear time invariant system and the fuzzy inference is constructed such that the model is very close to the aircraft nonlinear dynamics. While in the case of T-S fuzzy model the output is computed with a very simple formula (weighted average, weighted sum), Mamdani fuzzy structure require higher computational effort because of large number of rules to comply with defuzzification of membership functions. This advantage to the T-S approach makes it highly useful despite the more intuitive nature of Mamdani fuzzy reasoning in terms of dealing with uncertainty [11].

2.9.6 Building a fuzzy expert system

To illustrate the design of a fuzzy expert system, a problem of operating a service centre of spare parts is considered. A service center keeps spare parts and repairs failed ones. A customer brings a failed item and receives a spare of the same type. Failed parts are repaired, placed on the shelf, and thus become spares. If the required spare is available on the shelf, the customer takes it and leaves the service centre. However, if there is no spare on the shelf, the customer must wait until the needed item becomes available. The objective here is to advise a manager of the service centre on certain decision policies to keep the customers satisfied. A typical process in developing the fuzzy expert system incorporates the following steps [10]:

- Specify the problem and define linguistic variables.
- Determine fuzzy sets.
- Elicit and construct fuzzy rules.
- Encode the fuzzy sets, fuzzy rules and procedures to perform fuzzy inference into the expert system.
- Evaluate and tune the system.

2.9.7 Structure of a simple open-loop fuzzy controller

The simple open-loop fuzzy controllers, it’s not so simple. The classical design scheme contains the following steps:

- Define the condition interface – fix the way in which observations of the process are expressed as fuzzy sets.
- Design the rule base – determine which rules are to be applied under which conditions.
- Design the computational unit – supply algorithms to perform fuzzy computations. Those will generally lead to fuzzy outputs.
- Determine rules per which fuzzy control statements can be transformed into crisp control actions [12].

The typical structure of a fuzzy controller is given in Figure 2.20.

.

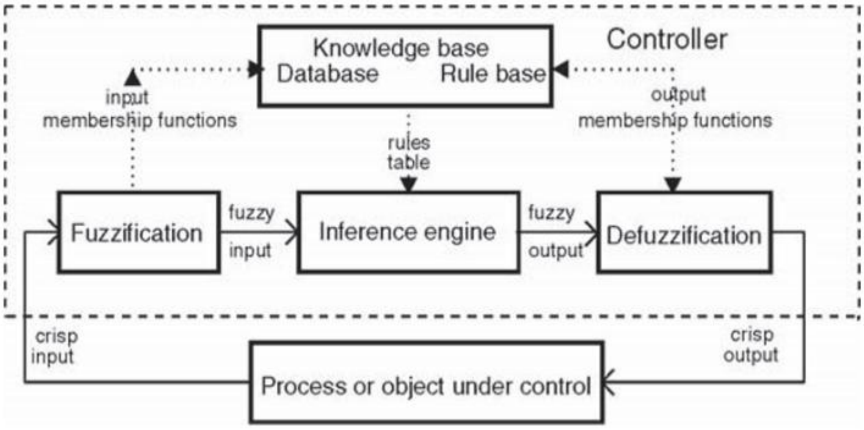


Figure 2.20: The typical structure of a fuzzy controller

The basic Steps for fuzzy controller design are [12]:

- Step 1: Is a usual step in the design of any controller, the variables which can be measured are chosen. They become the inputs of the controller.
- Step 2: Represents the fuzzification process.
- Step 3: Fuzzy inference.
- Step 4: Defuzzification process.

2.9.8 Structure of a feedback PID-like fuzzy controller

In order to build a PID-like FLC, it is required to design a fuzzy inference system with three inputs that represent the proportional, derivative and integral components**.**

- **Fuzzy Controllers as a part of a Feedback System:**

The choice of designing a P-, PD-, PI- or PID-like fuzzy controller, this already implies the choice of process state and control output variables, as well as the content of the rule-antecedent and the rule-consequent parts for each rule. The process state variables representing the contents of the rule-antecedent (if-part of a rule) are selected among:

- Error signal, denoted by e.
- Change-of-error, denoted by Δe.
- Sum-of-errors or an integral error, denoted by Σe.

The control output (process input) variables representing the contents of the rule-consequent (then-part of the rule) are selected among:

- Change-of-control output, denoted by Δu.
- Control output, denoted by u.

The error is the difference between the desired output of the object or process under control or the set-point and the actual output. This is one of the basic milestones in conventional feedback control. Furthermore, by analogy with a conventional controller, we have:

$$\begin{bmatrix} e(t) \; = Y_{SP} - Y(T) \\ \Delta e(t) = e(t) - e(t-1) \\ \Delta u(t) = u(t) - u(t-1) \end{bmatrix} \tag{2.9}$$

In the above expressions, Y_{SP} stands for the desired process output or the set-point, y is the process output variable (control variable), k determines the current time.

- **PD-like fuzzy controller:**

Figure 2.21 shows the block diagram of a PD-like fuzzy controller. The equation giving a conventional PD-controller is:

$$u\,(t) = K_P \times e(t) + K_d \times \Delta e(t), \tag{2.10}$$

Where K_P and K_d are the proportional and the differential gain factors.

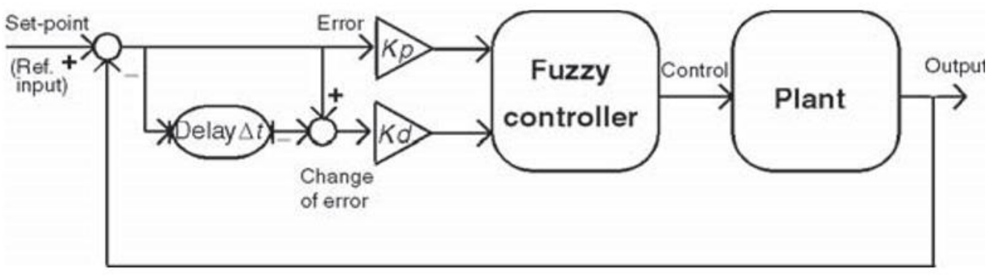


Figure 2.21: A block diagram of a PD-like fuzzy control system

By consideration the Equation (2.10), the PD controller for any pair of the values of error (e) and change-of-error (Δe) calculates the control signal (u). The fuzzy controller should do the same thing. For any pair of error and change-of-error, it should work out the control signal. Then a PD-like fuzzy controller consists of rules, and a symbolic description of each rule is given as:

If e(t) is < property symbol > and Δe(t) is < property symbol > then u(t) is <property symbol >, where <property symbol > is the symbolic name of a linguistic value.

The natural language equivalent of the above symbolic description reads as follows. For each sampling time. t: if the value of error is <linguistic value> and the value of change of-error is <linguistic value> then the value of control output is <linguistic value> [12].

- **Rules table notation:**

This table form is suitable when two inputs and one output are used. On the top side and lift side of the table, the possible linguistic values for the change-of-error (Δe) and the error (e) are written. The cell of the table at the intersection of the row and the column will contain the linguistic value for the output corresponding to the value of the first input written at the beginning of the row and to the value of the second input written on the top of the column [12].

- **PI-like fuzzy controller:**

The equation giving a conventional PI-controller is:

$$u(t) = K_P \times e(t) + K_i \times \int e(t)\, dt \qquad (2.11)$$

Where K_P and K_i are the proportional and the integral gain coefficients. A block diagram for a fuzzy control system looks like Figure 2.22.

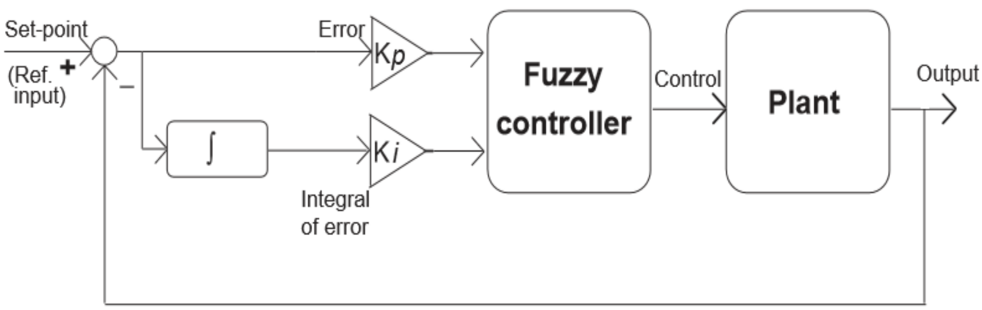


Figure 2.22: A block diagram of a PI fuzzy control system (version 1)

Now the fuzzy controller and the rules table have other inputs. It means that the rules themselves should be reformulated. Sometimes it is difficult to formulate rules depending on an integral error, because it may have the very wide universe of discourse. The integration from the part proceeding can be removed to a fuzzy controller to the part following it. The output of a controller can be integrated, not the input. Then the error and the change of error inputs and still realize the PI-control. When the derivative, with respect to time, of the Equation (2.11) is taken, it is transformed into an equivalent expression:

$$du(t) / dt = K_P \times de(t)/dt + K_i \times e(t) \qquad (2.12)$$

One can see here that one has the error and the change-of-error inputs and one need just to integrate the output of a controller. One may consider the controller output not as a control signal, but as a change in the control signal. The block diagram for this system is given in Figure 2.23. You should remember, that the gain factor K_i is used with the error input and K_P with the change-of-error.

If e is < property symbol > and Δe is < property symbol > then Δu is <property symbol>. In this case, to obtain the value of the control output variable u(t), the change-of-control output Δu(t) is added to u (t – 1). It is necessary to stress here that this takes place outside the PI-like fuzzy controller, and is not reflected in the rules themselves [12].

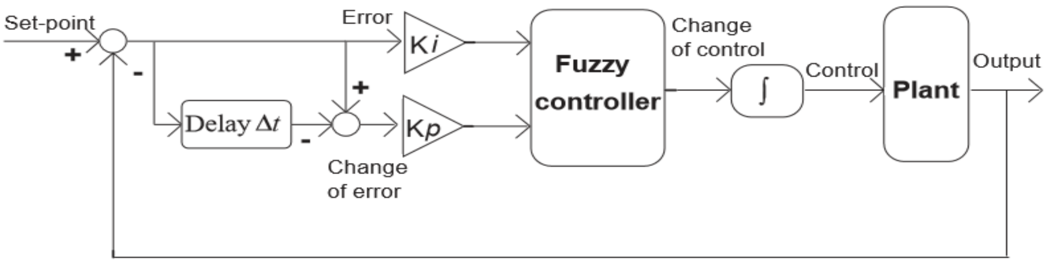


Figure 2.23: A block diagram of a PI fuzzy control system (version 2)

- **PID-like fuzzy controller:**

The equation for a PID-controller is as follows:

$$u = K_P \times e + K_d \times e + K_i \times \int edt. \qquad (2.13)$$

Thus, in the discrete case of a PID-like fuzzy controller one has an additional process state variable, namely sum-of-errors, denoted σe and computed as:

$$\sigma e(t) = \sum_{i=1}^{t} e(i) \qquad (2.14)$$

Have a look at the rules for PD- and PI-like fuzzy controllers and try to write down the rules format for a PID-like controller. The symbolic expression for a rule of a PID-like fuzzy controller is:

If e is < property symbol > and Δe is < property symbol > and σe is < property symbol> then u is < property symbol >. The PID-like fuzzy controller can be constructed as a parallel structure of a PD-like fuzzy controller and a PI-like fuzzy controller as shown in Figure 2.24 with the output approximated as [12]:

$$u = (K_P/2 \times e + K_d \times de/dt) + (K_P/2 \times e + K_i \times \int edt) \qquad (2.15)$$

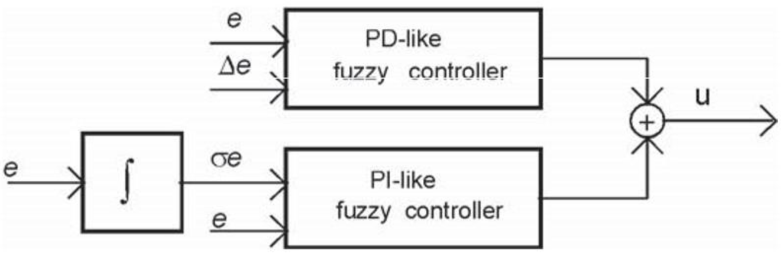


Figure2.24: The structure for a PID-like fuzzy controller

The symbolic expression for a rule in the case of Multiple Inputs and a Single Output (MISO) system is as follows:

If $X_1$is <property symbol> and …………………….... and $X_n$is <property symbol> then u is <property symbol>

Rules of this if – then type are usually derived from a fuzzy process model.

CHAPTER THREE

SYSTEM MODELING AND CONTROL DESIGN

3.1 System Mathematical Model

If the dynamic behavior of a physical system can be represented by an equation, or a set of equations, this is referred to as the mathematical model of the system. Such models can be constructed from knowledge of the physical characteristics of the system, i.e. mass for a mechanical system or resistance for an electrical system. Alternatively, a mathematical model may be determined by experimentation, by measuring how the system output responds to known inputs [13].

This section provides a brief description on the modeling of pitch control longitudinal equation of aircraft, as basis of a simulation environment for development and performance evaluation of the proposed controller techniques. The system of longitudinal dynamics is considered in this investigation and derived in the transfer function and states pace forms. The pitch control system considered in this work is shown in Figure 3.1, where X_b, Y_b and Z_b represent the aerodynamics force components. The θ, ∅ and δ_e represent the orientation of aircraft (pitch angle) in the earth-axis system and elevator deflection angle. The equations governing the motion of an aircraft are a very complicated set of six nonlinear coupled differential equations. Although, under certain assumptions, they can be decoupled and linearized into longitudinal and lateral equations. Aircraft pitch is governed by the longitudinal dynamics. In this example, an autopilot that controls the pitch of an aircraft is designed. The basic coordinate axes and forces acting on an aircraft are shown in the Figure 3.2 [14].

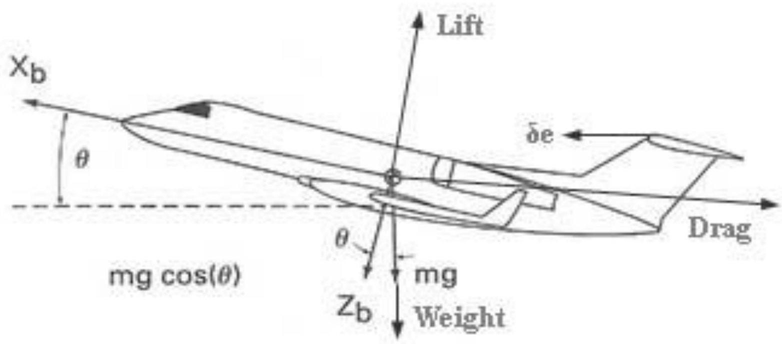


Figure 3.1: Description of pitch control system

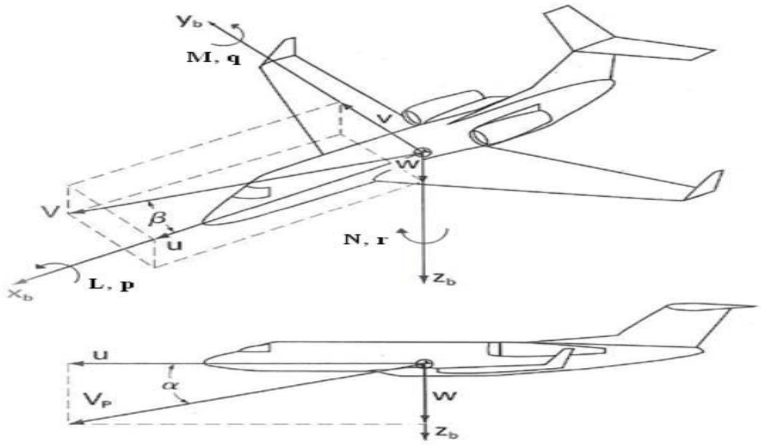


Figure 3.2: Definition of force, moments and velocity in body fixed coordinate

Figure 3.2 shows the forces, moments and velocity components in the body fixed coordinate of aircraft system. The aerodynamics moment components for roll, pitch and yaw axis are representing as L, M and N. The term p, q and r represent the angular rates about roll, pitch and yaw axis while term u, v and w represent the velocity components of roll, pitch and yaw axis. The α and β represent the angle of attack and sideslip. A few assumptions need to be considered before continuing with the modeling process. First, the aircraft is steady state cruise at constant altitude and velocity, thus the thrust and drag are cancel out and the lift and weight balance out each other. Second, the change in pitch angle does not change the speed of an aircraft under any circumstance. The longitudinal stability derivatives parameter used are denoted in Table 3.1. The dynamics pressure and dimensional derivative are Q = 36.8lb/ft^2, QS= 6771 lb, QS c = 38596ft.lb, (c / $2U_0$) = 0.016s

Table 3.1: The longitudinal derivative stability parameters

Parameters	**X-Force (S^{-1})**	**Z-Force (F^{-1})**	**Pitching Moment, (FT^{-1})**
Rolling Velocities	X_u = -0.045	Z_u = -0.369	M_u = -0.369
Yawing Velocities	$X_W = 0.036$ $X_{W^{\cdot}} = 0$	$Z_W = -2.02$ $Z_{W^{\cdot}} = 0$	$M_W = -0.05$ $M_{W^{\cdot}} = 0$
Angle of Attack	$X_\alpha = 0$ $X_{\alpha^{\cdot}} = 0$	Z_α = -355.42 $Z_{\alpha^{\cdot}} = 0$	M_α = -8.8 $M_{\alpha^{\cdot}}$ = -0.8976
Pitching Rate	$X_a = 0$	$Z_a = 0$	M_a = -2.05
Elevator Deflection	$X_{\delta e} = 0$	$Z_{\delta e}$ = *-28.15*	$M_{\delta e}$ = - *11.874*

These values are taken from the data from one of Boeing's commercial aircraft. Referring to the Figure 3.1 and Figure 3.2, the following dynamic equations include force and moment equations are determined as shown in Equations (3.1), (3.2) and (3.3).

$$X - mgS_{\theta} = m(\dot{u} + qv - rv) \quad (3.1)$$

$$Z + mgC_{\theta}C_{\emptyset} = m(\dot{w} + pv - qv) \quad (3.2)$$

$$M = I_Y\dot{q} + rq(I_X - I_Z) + I_{XZ}(P^2 - r^2) \quad (3.3)$$

It is required to completely solve the aircraft problem with considering the following assumption:

- Rolling rate is given as: $p = \dot{\emptyset} - \dot{\psi}S_{\theta}$.
- Pitching rate, $q = \dot{\theta}C_{\emptyset} + \dot{\psi}C_{\theta}S_{\emptyset}$.
- Yawing rate, $r = \dot{\psi}C_{\theta}C_{\Phi} - \dot{\theta}S_{\emptyset}$.
- Pitch angle, $\dot{\theta} = qC_{\emptyset} - rS_{\emptyset}$.
- Roll angle, $\dot{\emptyset} = P + qS_{\emptyset}T_{\theta}$.
- Yaw angle, $\dot{\psi} = (qS_{\emptyset} + r_{\emptyset})\sec\theta$.

Equation (3.1), (3.2) and (3.3) should be linearized using small disturbance theory. The equations are replaced by a variables or reference value plus a perturbation or disturbance as shown below.

$u = u_0 + \Delta u$, $v = v_0 + \Delta v$, $w = w_0 + \Delta w$, $p = p_0 + \Delta p$, $q = q_0 + \Delta q$,

$r = r_0 + \Delta r$ $X = X_0 + \Delta X$, $M = M_0 + \Delta M$, $Z = Z_0 + \Delta Z$, $\delta = \delta_0 + \Delta\delta$

For convenience, the reference flight condition is assumed to be symmetric and the propulsive forces are assumed to remain constant. This implies that, $v_0 = P_0 = q_0 = r_0 = \emptyset_0 = \psi_0 = W_0 = 0$. after linearization the (3.4), (3.5) and (3.6) are obtained.

$$\left(\frac{d}{dt} + X_u\right)\Delta u - X_W\Delta W + (g\cos\theta_0)\Delta\theta = X_{\delta e}\Delta\delta_e \quad (3.4)$$

$$-Z_u\Delta u + \left[(1 - Z_u)\frac{d}{dt} - Z_W\right]\Delta W - \left[(u_0 - Z_0)\frac{d}{dt} - \sin\theta_0\right]\Delta\theta = Z_{\delta e}\Delta\delta_e \quad (3.5)$$

$$-M_u\Delta u - \left(M_W\frac{d}{dt} + M_W\right)\Delta W + \left(\frac{d^2}{dt^2} - M_q\frac{d}{dt}\right)\Delta\theta = M_{\delta e}\Delta\delta_e \quad (3.6)$$

By manipulating the Equations (3.4), (3.5), (3.6) and substituting the parameters values of the longitudinal stability derivatives, the following transfer function for the change in the pitch rate to the change in elevator deflection angle is shown as (3.7) obtained:

$$\frac{\Delta q(s)}{\Delta \delta_e(s)} = \frac{-\left(M_{\delta e} + \frac{M_{\alpha^{\cdot}} Z_{\delta e}}{u_0}\right) S - \left(\frac{M_\alpha Z_{\delta e}}{u_0} - \frac{M_\alpha Z_\alpha}{u_0}\right)}{S^2 - \left(M_q + M_{\alpha^{\cdot}} + \frac{Z_\alpha}{u_0}\right) S + \left(\frac{Z_\alpha M_\alpha}{u_0} - M_\alpha\right)} \tag{3.7}$$

The transfer function of the change in pitch angle to the change in elevator angle can be obtained from the change in pitch rates to the change in elevator angle in the following way:

$$\Delta \mathrm{q} = \Delta \theta^{\cdot} \tag{3.8}$$

$$\Delta \mathrm{q(s)} = \mathrm{S}\Delta\, \theta(\mathrm{S}) \tag{3.9}$$

$$\frac{\Delta\, \theta(\mathrm{S})}{\Delta \delta_e(S)} = \frac{1}{S} \cdot \frac{\Delta q(s)}{\Delta \delta_e(s)} \tag{3.10}$$

Therefore, the transfer function of the pitch control system is obtained in (3.11) and (3.12) respectively [15]:

$$\frac{\Delta \theta(s)}{\Delta \delta_e(s)} = \frac{1}{S} \cdot \frac{-\left(M_{\delta e} + \frac{M_{\alpha^{\cdot}} Z_{\delta e}}{u_0}\right) S - \left(\frac{M_\alpha Z_{\delta e}}{u_0} - \frac{M_\alpha Z_\alpha}{u_0}\right)}{S^2 - \left(M_q + M_{\alpha^{\cdot}} + \frac{Z_\alpha}{u_0}\right) S + \left(\frac{Z_\alpha M_\alpha}{u_0} - M_\alpha\right)} \tag{3.11}$$

$$\frac{\Delta \theta(s)}{\Delta \delta_e(s)} = \frac{1.151S + 0.1774}{S^3 + 0.739\, S^2 + 0.921S} \tag{3.12}$$

3.2 System Controllers Design

The next step is to choose some design criteria. In this chapter, a feedback controller is designed so that in response to a step command of pitch angle the actual pitch angle overshoots less than 10%, has a rise time of less than 2 seconds, a settling time of less than 10 seconds, and a steady-state error of less than 2%. For example, if the reference is 0.2 radians (11 degrees), then the pitch angle will not exceed approximately 0.22 rad, will rise from 0.02 rad to 0.18 rad within 2 seconds, will settle to within 2% of its steady-state value within 10 seconds, and will settle between 0.196 and 0.204 radians in steady-state [15]. In summary, assume that the design requirements for response of pitch autopilot as follows:

- Overshoot less than 10%.
- Rise time less than 2 seconds.
- Settling time less than 7 seconds
- Steady-state error less than 2%.

3.2.1 Conventional PID controller design

In this section, will be designed conventional PID controller for aircraft pitch control are proposed and described in detail. Initially the PID is designed in the closed-loop system for control the pitch of an aircraft. A PID is a generic control loop feedback mechanism widely used in industrial control systems and regarded as the standard control structures of the classical control theory. Figure 3.3 shows the control system of schematic model with general PID controller.

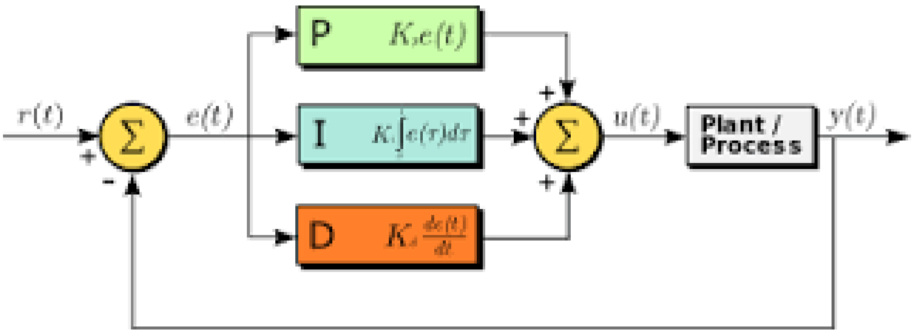


Figure 3.3: PID configuration

But it may make the transient response slower. A derivative control (K_d) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response. The effects of increasing a parameter independently of each of controller parameters, K_P, K_i, and K_d on a closed-loop system are summarized in the Table 3.2.

Table 3.2: The characteristics of P, I, and D controllers

Parameter Increase	Rise Time	Overshoot	Settling Time	Steady-state Error
K_P	↓	↑	Small change	↓
K_i	↓	↑	↑	Great reduce
K_d	Small change	↓	↓	Small change

• **PID controller design with Z – N tuning method:**

In SIMULINK, it is very straightforward to represent and then simulate a mathematical model representing a physical system. The SIMULINK block diagram of the pitch control developed in Equation (3.12) with PID controller is shown in Figure 3.4. PID tuning is the process of finding the proper values of K_P, K_i and K_d gains. Here the PID controller has been designed and implemented using Zeigler-Nichole close loop to tuning the PID parameters until achieved the optimal response of the system with gain values are $K_P = 2.674,\ K_i = 2.549\ and\ K_d = 0.701$.

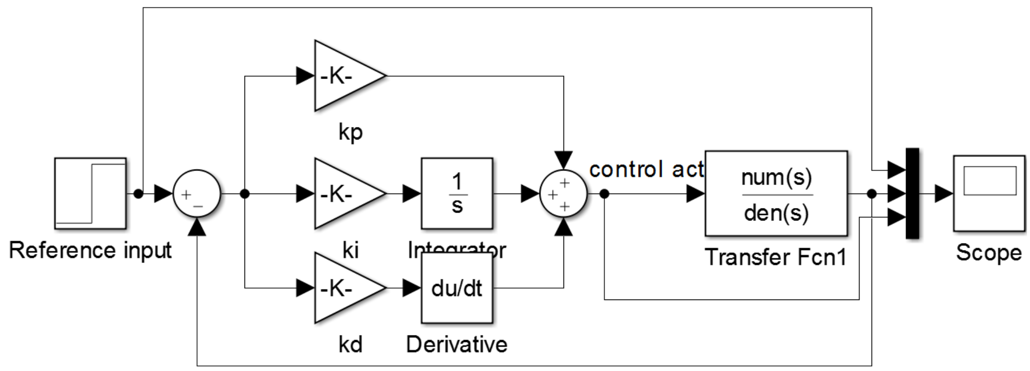


Figure 3.4: Simulink block diagram of the system with PID controller

• **PID controller design with trial and error tuning method:**

Referring to previous chapter in section (2.8.3), the trial and error is the simple method of tuning a PID controller. Once we get the clear understanding the effects of increasing of PID parameters of close loop system, the trial and error method become relatively easy. After applying all the steps have been the optimal response of the system with gain values are $K_P = 7.55, K_i = 1.55\ and\ K_d = 10.76$.

• **PID controller design with automatic tuning method:**

PID tuning is the process of finding the values of proportional, integral, and derivative gains of a PID controller to achieve desired performance and meet design requirements. By using MATLAB toolbox, check step response characteristics provide tools for automatically choosing optimal PID gains, the optimal response of conventional PID controller is obtained as shown in Figure 3.5 with gain values are K_P= 11.4003, K_i= 9.8794 and K_d= 29.0551.

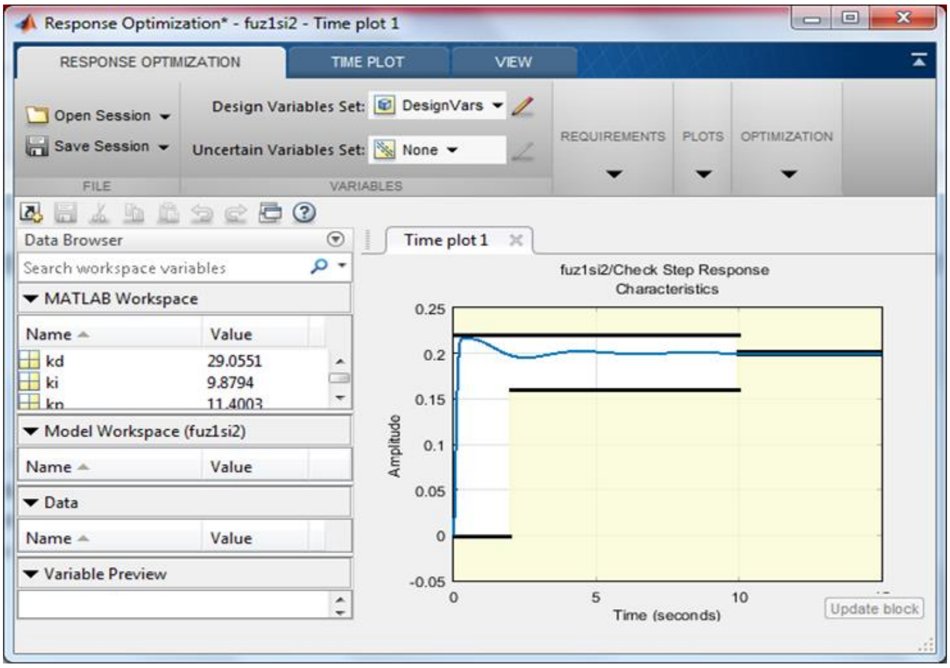


Figure 3.5: Conventional PID parameters tuning by response optimization toolbox

3.2.2 Fuzzy logic controller design

The building of the Fuzzy Inference System (FIS), using the FIS editor of MATLAB, the two inputs to the fuzzy controller are the error (e) represent here the pitch angle which measures the system performance and the pitch rate at which the error changes (de), whereas the output of the control action (u) that is represent elevator deflection, as shown in Figure 3.6.

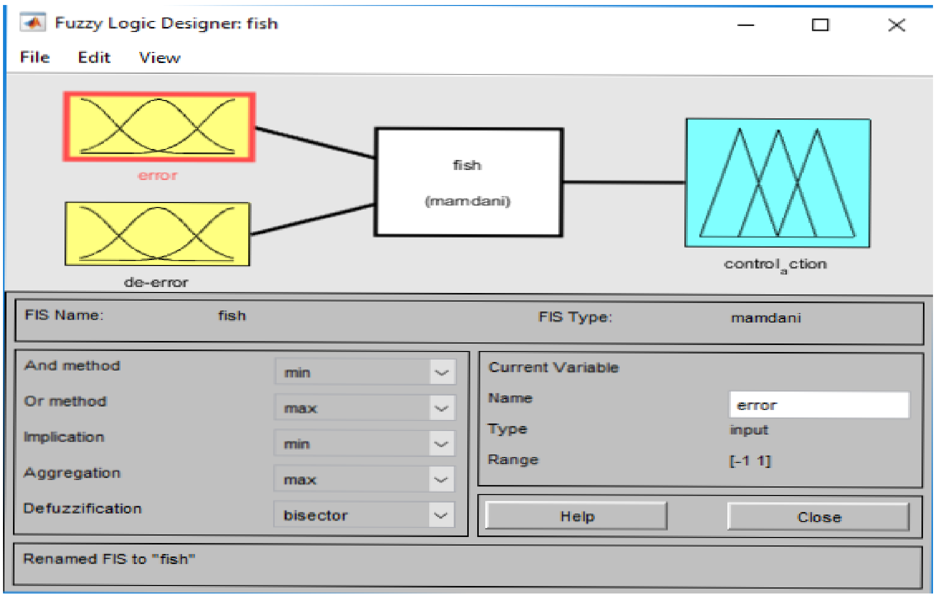


Figure 3.6: The fuzzy inference system

A Mamdani type FLC is used with Gaussian membership functions for the inputs and output. The inputs to the Proportional-Derivative (PD) fuzzy controller are generally generated from the plant output and reference input. Here inputs to the fuzzy PD controller are the pitch angle error e (t) and change in pitch angle error $e_c(t)$ expressed as [18]:

$$e(t) = r(t) - y(t \qquad (3.13)$$

$$e_c(t) = e(t) - e(t-1) \qquad (3.14)$$

The fuzzy sets of each input are represented by seven Gaussian membership functions then there will be 7^2 = 49 rule listed in Table 3.3, which are Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), and Positive Big (PB). While the fuzzy sets of output are represented by three Gaussian membership functions which are; Negative (N), Zero (Z), and Positive (P). But since any input has some contribution in all of the fuzzy sets and will circle around the main diagonal of the fuzzy rule table and settle in the center of this table, recent researches such as [16] and [17] propose to use only the diagonal rules. So, the developed controller uses only the 7 diagonal fuzzy rules as in Table 3.3 to simplify the controller and reduce complex computations.

Table 3.3: Mamdani fuzzy rules with 7 membership functions

	Error							
	Output MF	**NB**	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
Change of Error	**NB**	N	N	N	N	N	N	Z
	NM	N	N	N	N	N	Z	P
	NS	N	N	N	N	Z	P	P
	Z	N	N	N	Z	P	P	P
	PS	N	N	Z	P	P	P	P
	PM	N	Z	P	P	P	P	P
	PB	Z	P	P	P	P	P	P

The range of values of the inputs that can be quantified with the fuzzy sets (universe of discourse) is [-1, 1] and of output is [-10, 10]. The membership functions of all linguistic variables are shown in Figure 3.6. Minimum operator is used to represent the AND in rules premises and the implication and Bisector method for defuzzification. The fuzzy PD membership function for inputs and output are shown in Figures 3.7, 3.8 and 3.9 respectively. Table 3.4 shows the methods used in the fuzzy inference engine.

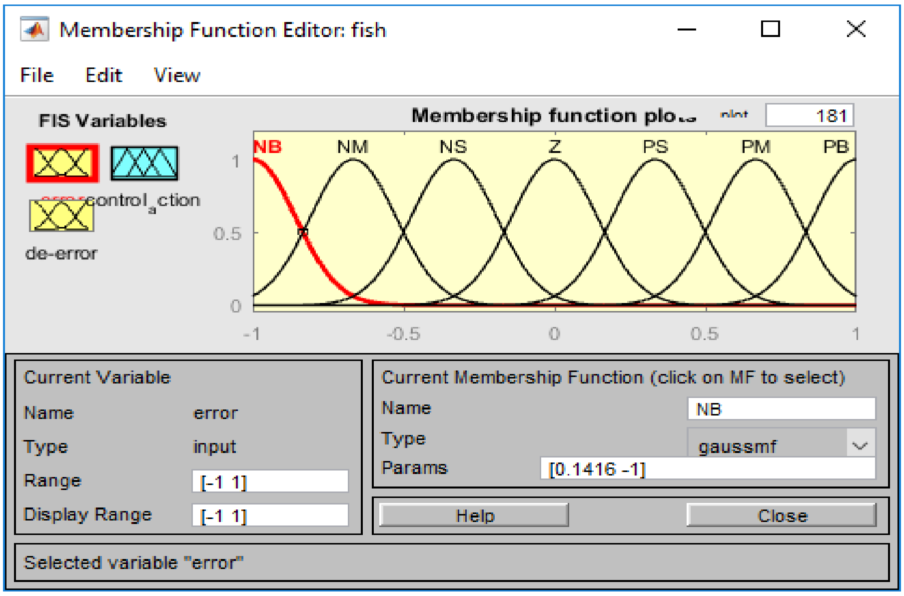


Figure 3.7: The error of pitch angle

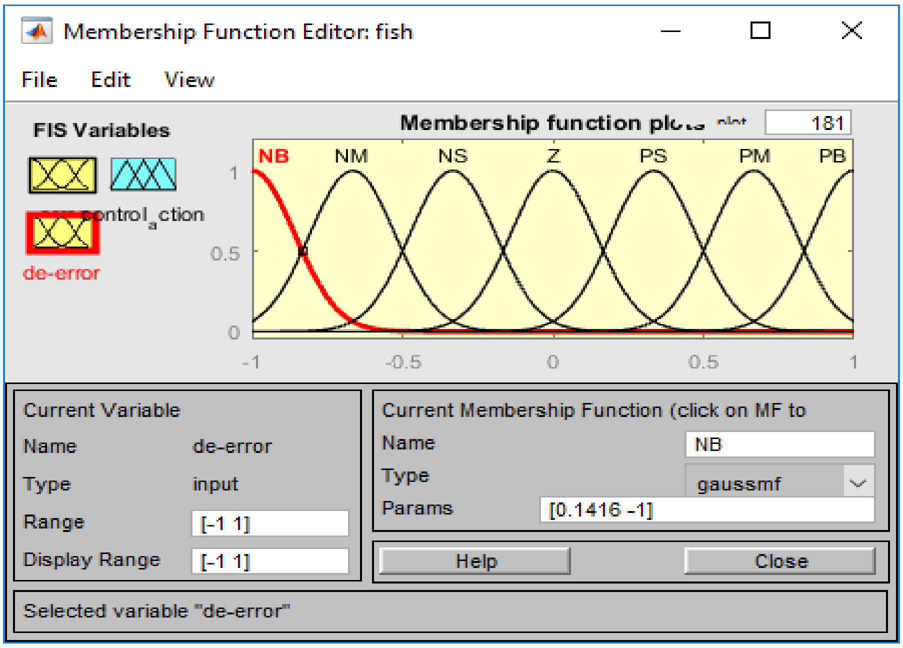


Figure 3.8: The change in pitch angle error

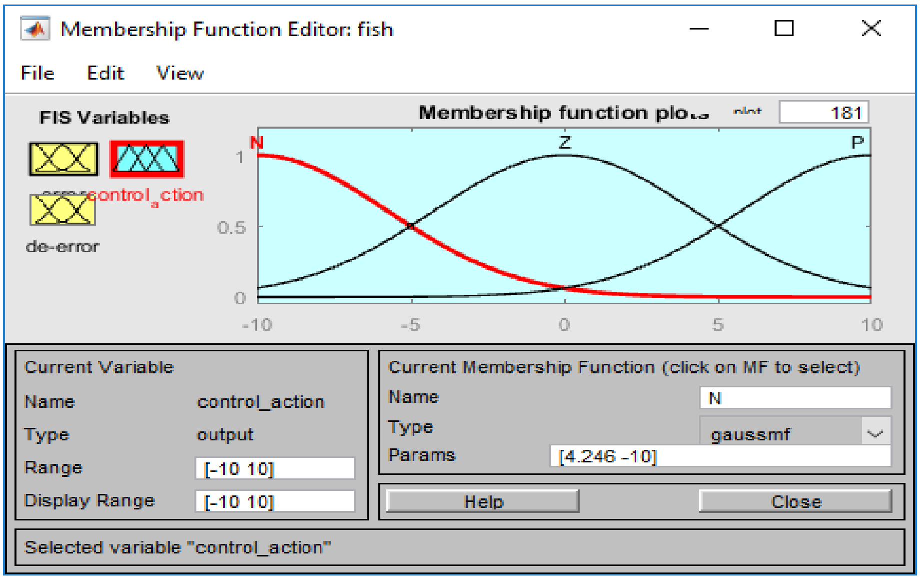


Figure 3.9: The output membership function for control action

Table 3.4: The methods used in the fuzzy inference engine

AND Method	Min
OR Method	Max
Implication	Min
Aggregation	Max
Defuzzification	Bisector

In Figure 3.10 fuzzy if-then rules are shown total 49 rules output variable and in Figure 3.11 shows relation between inputs and output for fuzzy controller.

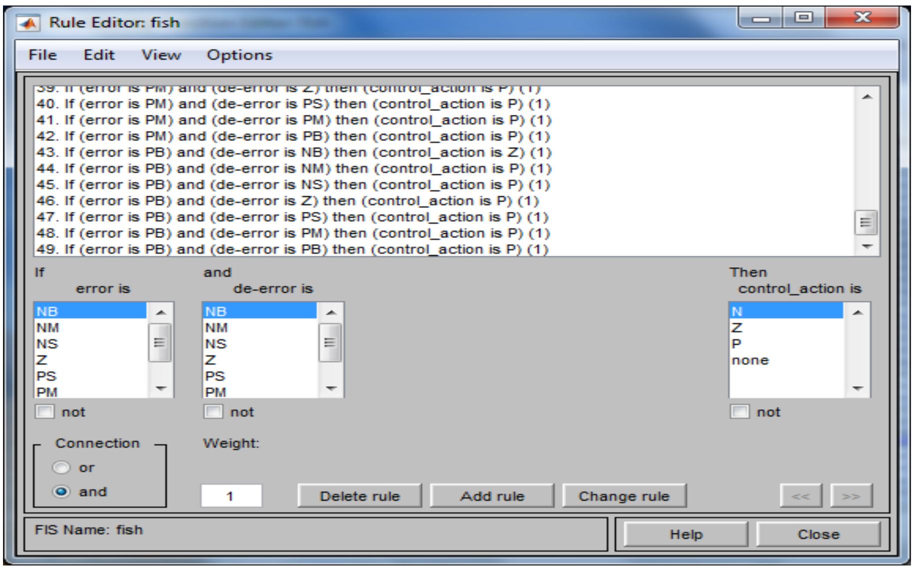


Figure 3.10 fuzzy if-then rules

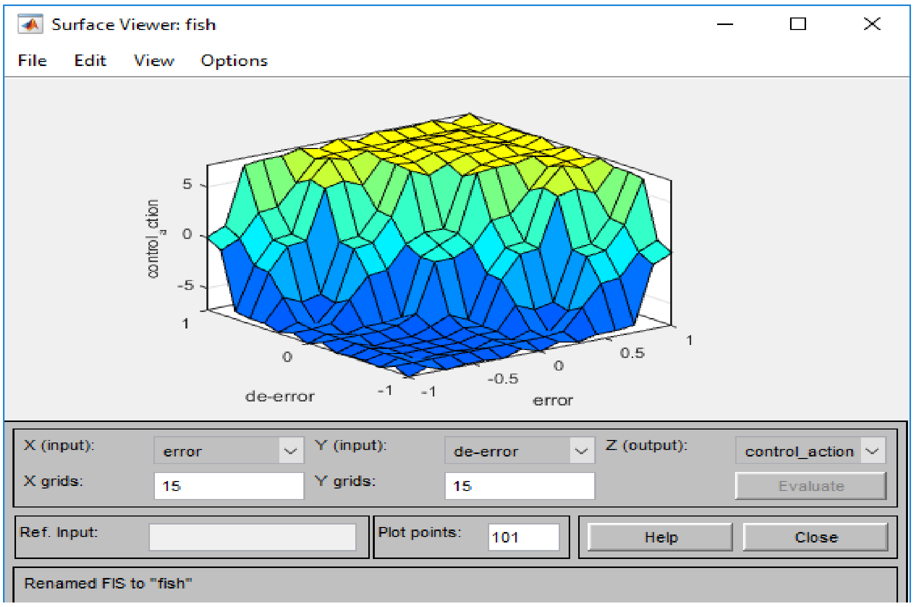


Figure 3.11: Input and output relation for fuzzy controller

The SIMULINK model of PD fuzzy logic controller and considering the design requirements for response of pitch autopilot as follows in Figure 3.12.

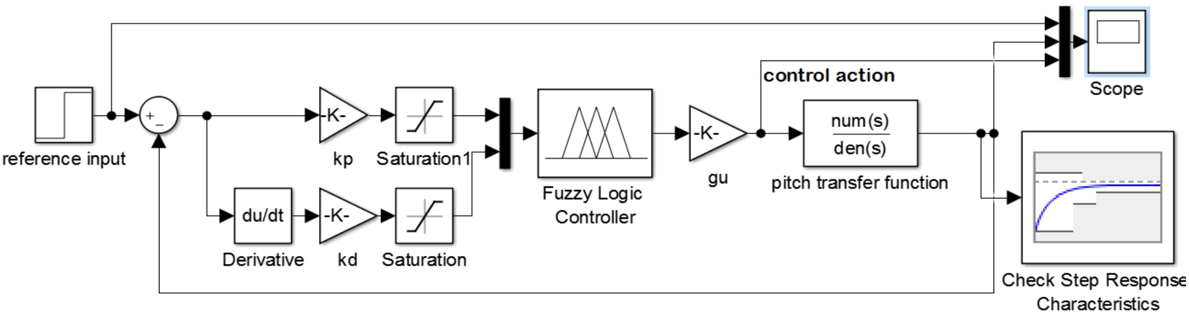


Figure 3.12: The SIMULINK model of the system with FLC

CHAPTER FOUR

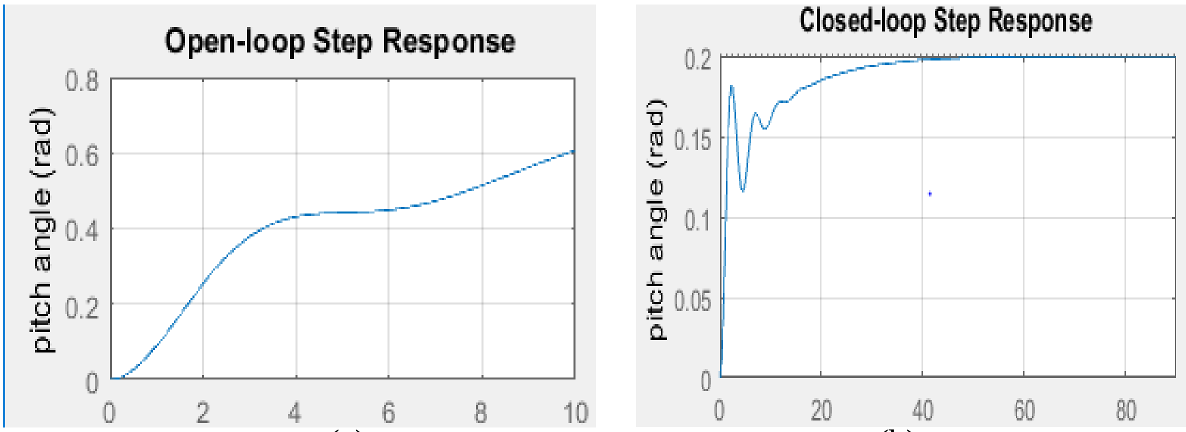
SYSTEM SIMULATION RESULTS

4.1 Introduction

In this section, the proposed of control schemes are implemented and the corresponding results are presented. The conventional PID and FLC controllers are simulated using MATLAB /SIMULINK model R2016a. The simulation results of pitch attitude obtained under different cases.

4.2 System Response without Controller

Using the MATLAB code (see Appendix A), the responses of open and close loop of systems are shown in the Figure 4.1.



(a) (b)

Figure 4.1: The open and close loop response without controller

Examination of the above plot indicates that the open-loop system is unstable for a step input, that is, its output grows unbounded when given a step input. The close loop characteristics are rise time =1.788 sec, settling time =35.0896 sec, overshoot = 0, peak = 0.200 rad, peak time = 98.7611 sec. The closed-loop system is stable, but doesn't meet the design requirements.

4.3 System Response with PID Controller

The performance of the system when Ziegler Nichols, trial and error and automatic tuning methods are used to tuning PID controller parameters are shown in Figures 4.2, 4.3 and 4.4 respectively.

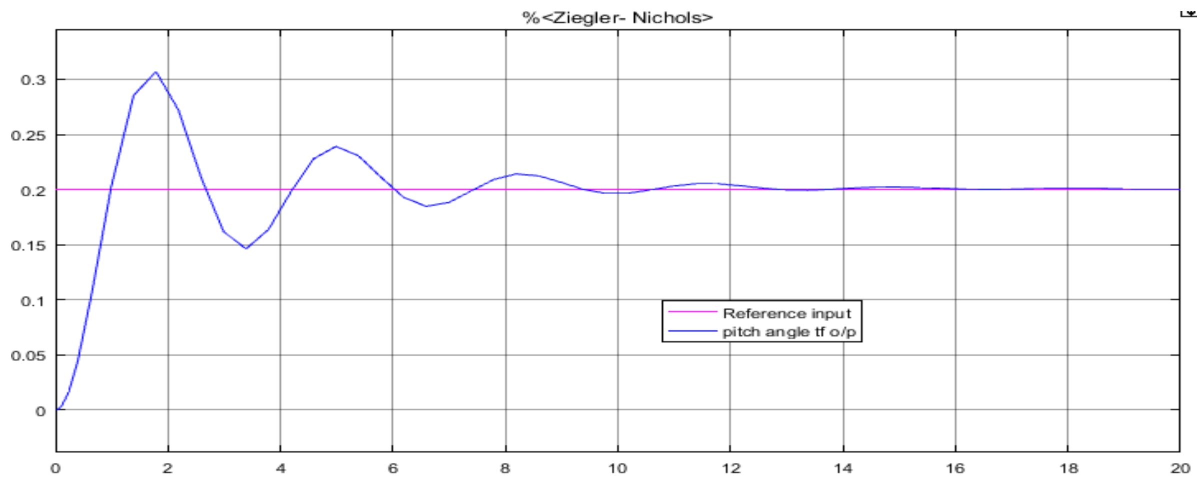


Figure 4.2: The performance of the system when PID controller tuning by Z-N

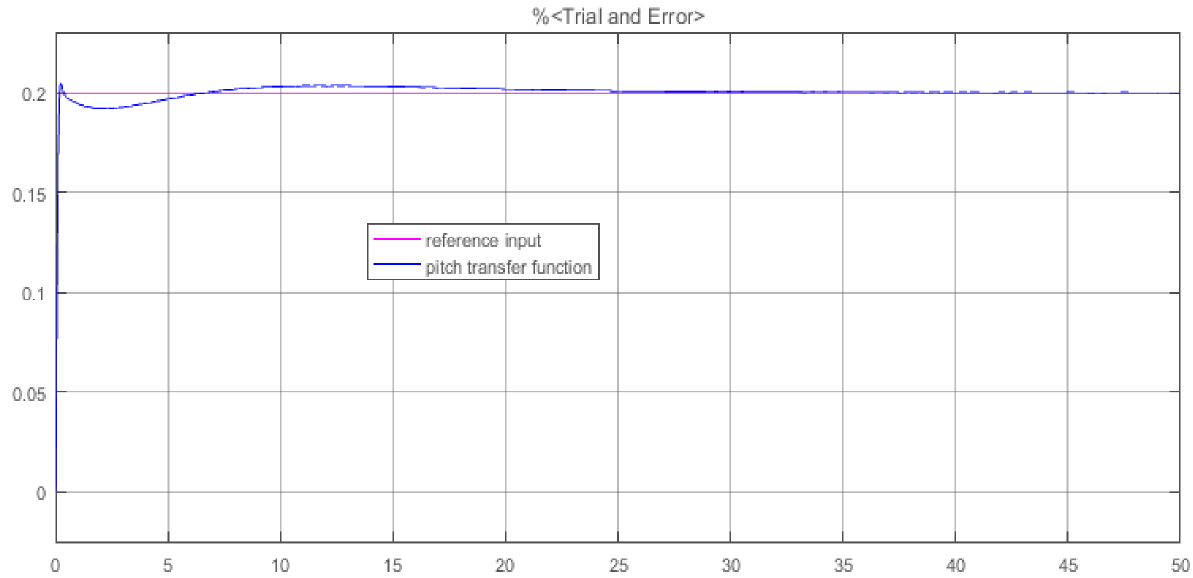


Figure 4.3: The performance of the system when PID tuning by trial and error

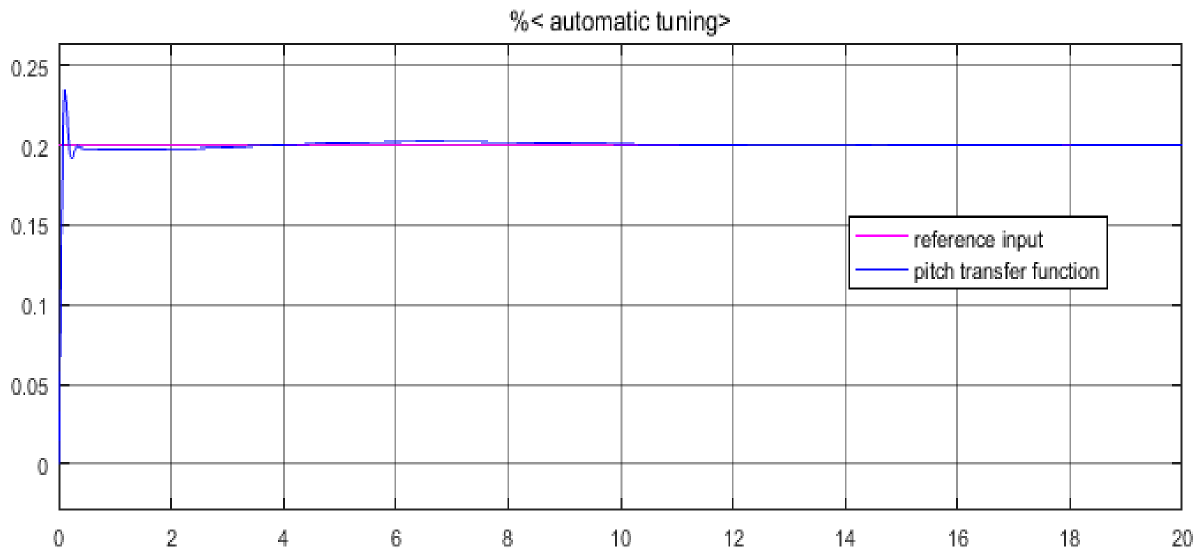


Figure 4.4: The performance of the system when PID tuning by automatic tuning

4.4 System Response with Fuzzy Controller

Refereeing to the Figure 4.3, the block diagram is content from two schemes of pitch control PID and FLC controller. Here is used check step response to tune controller gains. A unit step command is required for pitch angle to follow the reference value of 0.2 radian = 11.5 degree. After tuning the FLC the scaling gains are $k_p = 671.2812$, $k_d = 4.6274$ and $u = 18.8656$. The saturations are used to limit the fuzzy system inputs' minimum and maximum values to the range of their corresponding universe of discourse. The check step response characteristics are used to reach for optimal response of design requirements.

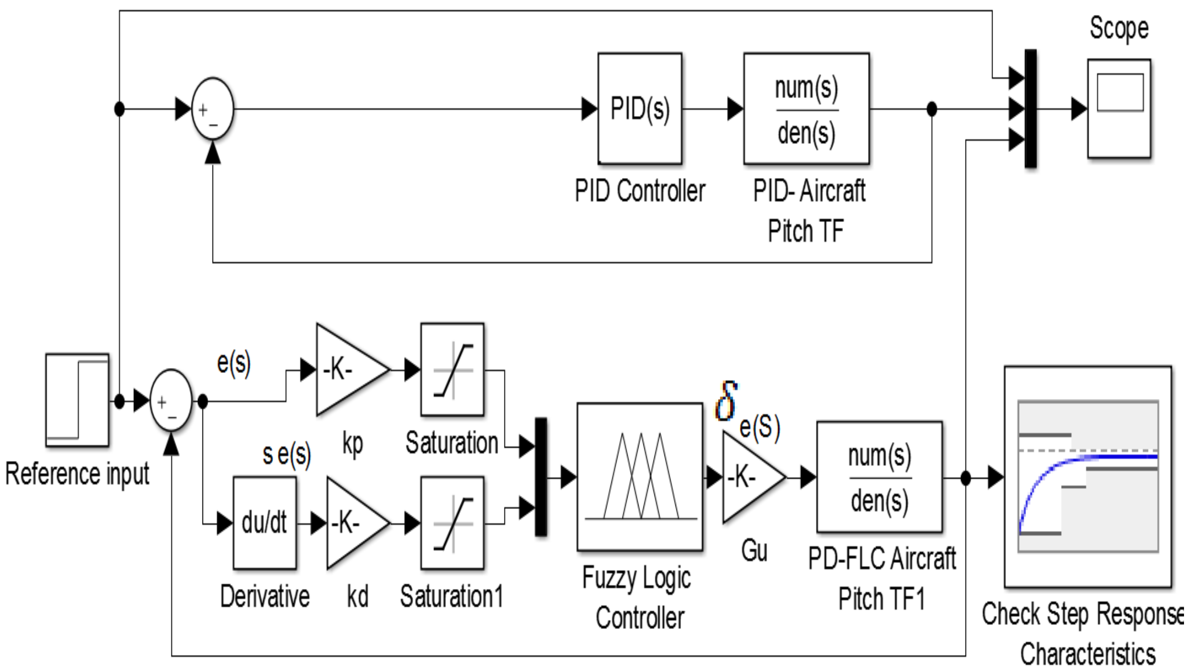


Figure 4.5: The SIMULINK model of the system with PID controller and FLC

4.5 Results Discussion

The response optimization toolbox to optimize the model response to meet design requirements specified in the Bounds tab, by tuning the gains (K_p, K_d, G_u) of PD fuzzy logic controller using genetic algorithm method and gets best response of system as shown in the Figure 4.6. The optimization progress report is shown in Figure 4.7. Figure 4.8 shows the performance of both control schemes conventional PID controller and FLC with respect to the step reference input signal. The system responses performances are compared between PID controller and PD-like fuzzy logic controller. The summary of controller's performance characteristics of the step response for the pitch angle is shown in Table 4.1.

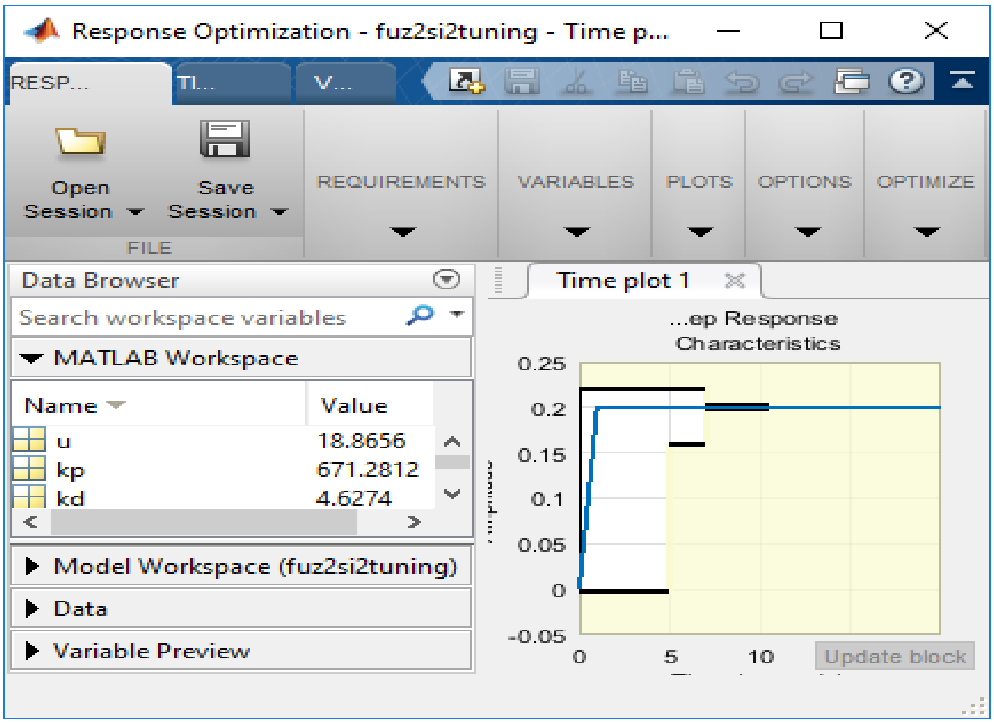


Figure 4.6: The response optimization of best tuning

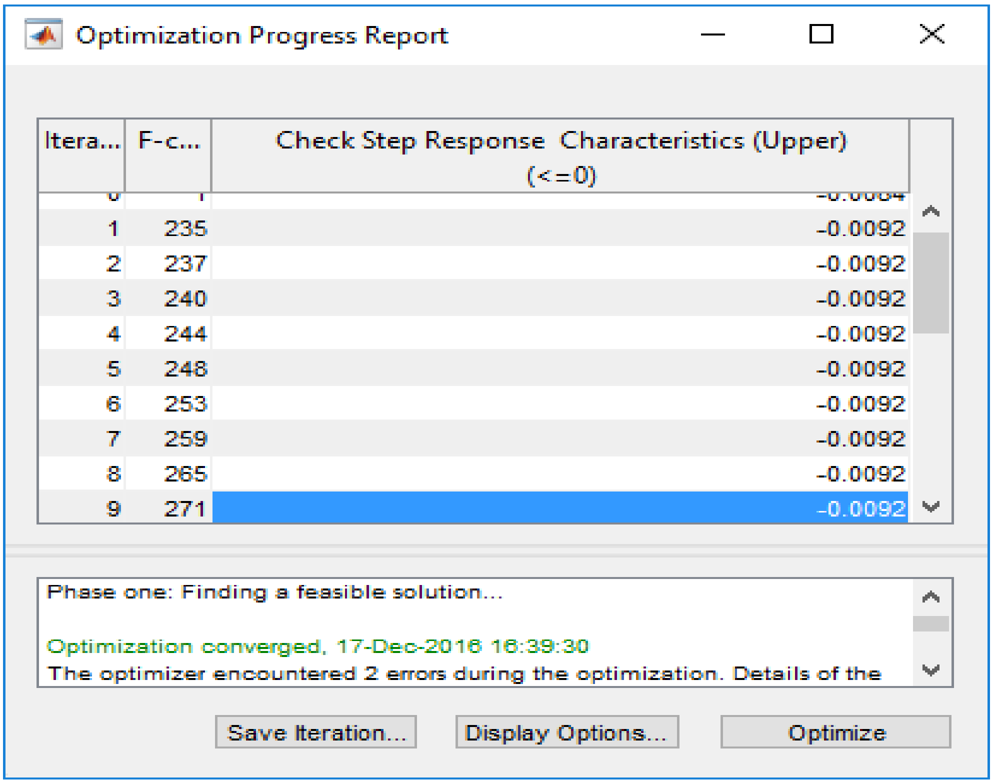


Figure 4.7: The optimization progress report

From Figure 4.8 clearly the performance of PD type fuzzy logic controller is better than conventional PID controller in the overshoot, undershoot and settling time. Table 4.1shows the performance comparison between conventional PID controller and FLC.

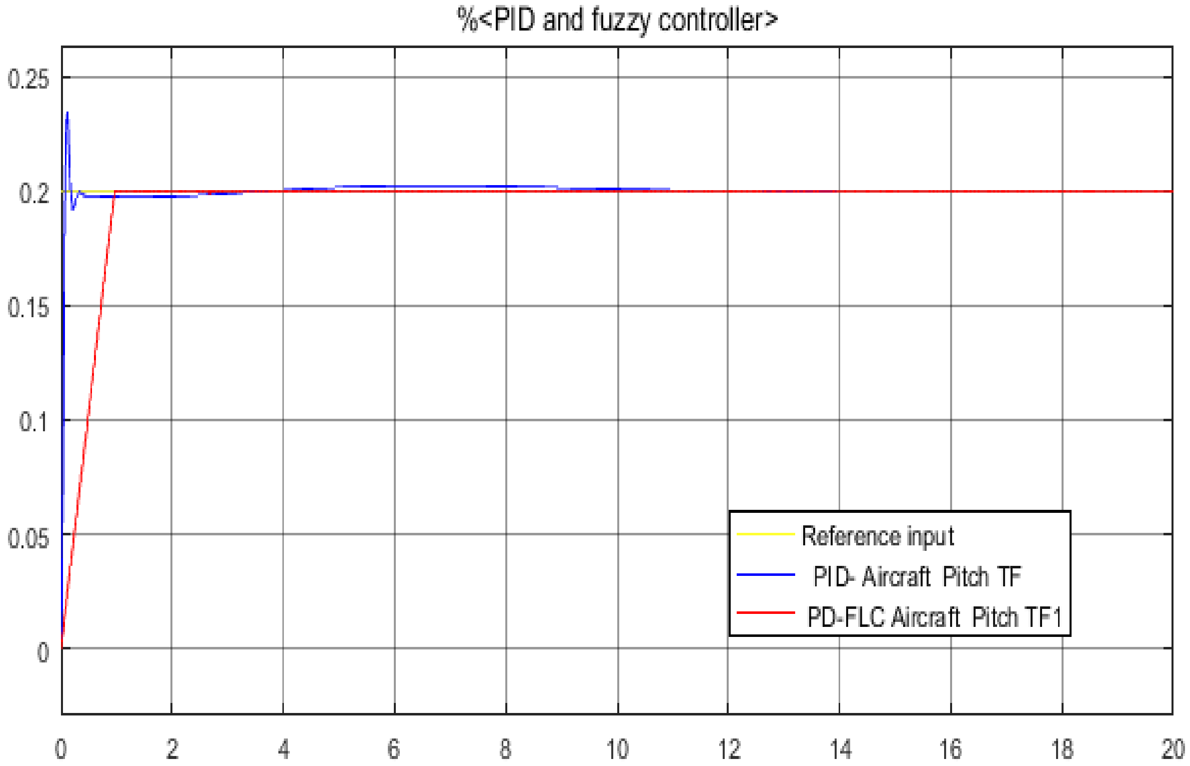


Figure 4.8: The reference input, PID and PD-FLC response of system

Table 4.1: Summary of controller's performance characteristics

Step Response Characteristics	**The Design Requirements**	**Conventional PID Controller**	**FLC**
Overshoot, M_P	Less than 10%	17.059%	0.424%
Rise Time, t_r	Less than 2sec	0.0505 sec	0.746 sec
Settling Time, t_s	Less than 7sec	10.7 sec	0.976 sec
Steady State Error, e_{ss}	Less than 2%	0	0
Undershoot, U_s	Less than 1%	4.434 %	0.786 %
Delay Time, t_d	Minimum Delay	0.0406 sec	0.485 sec

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In this research, the conventional PID controller and PD fuzzy controller were developed to control the pitch angle of aircraft. Modelling is done on an aircraft pitch angle control and fuzzy controller is proposed successfully. The proposed control schemes have been implemented within simulation environment in MATLAB /SIMULINK. Pitch control of an aircraft is a system which requires a pitch controller to maintain the angle at its desired value. Based on the results, the system responses indicate the performance of pitch control system using PD-fuzzy logic controller has been improved and satisfied compared to conventional PID controller. The system responses characteristics when using PD-fuzzy logic controller in most cases are better than that of conventional PID controller in terms of maximum overshoot, undershoot and settling time.

5.2 Recommendations

The possible future work on this topic could be to develop a fuzzy control system by using interval type 2 fuzzy logic controllers to tuning the disturbance rejection of pitch autopilot system.

REFERENCES

[1] Nurbaiti Wahid, “Self-Tuning Fuzzy PID Controller Design for Aircraft Pitch Control”, 2012.

[2] Siddharth Goyal, “Study of a Longitudinal Autopilot for Different Aircrafts”, 2007.

[3] Raghu Chaitanya. M. V, “Model Based Aircraft Control System Design and Simulation”, 2012.

[4] Ropert C. Nelson, “Flight Stability and Automatic Control”, 1989.

[5] Donald McLean, “Automatic Flight Control Systems”, 1990.

[6] Karl Johan A strom,” Control System Design”, 2002.

[7] Neil Kuyvenhoven, “PID Tuning Methods an Automatic PID Tuning Study with Mathcad”, 2002.

[8] TemeL, Semih YAĞLI and Semih GÖREN, “P, PD, PI, PID Controllers”, 2013.

[9] Bai, Y, Zhuang, H, Wag, D. (Eds), “Advanced Fuzzy Logic Technologies in Industrial Application”, 2006.

[10] Michael Negnevitsky, “Artificial Intelligence a Guide to Intelligent Systems”, Second Edition, 2005.

[11] Narenathreyas, K., “Fuzzy Logic Control for Aircraft Longitudinal Motion”, Master Thesis, Faculty of Electrical Engineering, Department of Control Engineering, Czech Technical University, 2013.

[12] F Leonid Reznik, “Fuzzy Controllers”, 1997.

[13] Roland Burns, “Advanced Control Engineering”, 2001.

[14] N. Wahid, N. Hassan, M.F. Rahmat and S. Mansor, “Application of Intelligent Controller in Feedback Control Loop for Aircraft Pitch Control”, 2011.

[15] Amir Torabi, Amin Adine Ahari, Ali Karsaz and Seyyed Hossin Kazem, “Intelligent Pitch Controller Identification and Design”, 2013.

[16] A. B. Kisabo1, F. A. Agboola, C.A. Osheku1, M. A. L. Adetoro1 and A. A. Funmilayo, “Pitch Control of an Aircraft Using Artificial Intelligence”, 2012.

[17] Yamama A. Sheik, “Aircraft Pitch Control Using Type-2 Fuzzy Logic”, 2015.

[18] P. S. Khuntia and Debjani Mitra, “Fuzzy Model Reference Learning Controller for Pitch Control System of an Aircraft”, 2009.

APPENDIX

MATLAB CODE OF OPEN LOOP AND CLOSE LOOP OF PITCH ATTITUDE

```
>> f = figure;
t = [0:0.01:10];
s = tf('s');
P_pitch = (1.151*s + 0.1774)/(s^3 + 0.739*s^2 + 0.921*s);
step(0.2*P_pitch,t);
axis([0 10 0 0.8]);
ylabel('pitch angle (rad)');
title('Open-loop Step Response');
grid
>> f = figure;
sys_cl = feedback(P_pitch,1);
step(0.2*sys_cl), grid
ylabel('pitch angle (rad)');
title('Closed-loop Step Response')
>> stepinfo(0.2*sys_cl)

ans =

        RiseTime: 1.7882
    SettlingTime: 35.0896
     SettlingMin: 0.1155
     SettlingMax: 0.2000
       Overshoot: 0
      Undershoot: 0
            Peak: 0.2000
        PeakTime: 98.7611
```