


Czech Technical University
Department of Control Engineering
Faculty of Electrical Engineering

Fuzzy Logic Control for Aircraft Longitudinal Motion

Master Thesis

Author:
Kashyapa Narenathreyas

Supervisor:
Dr. Petr Hušek

Dept of Control Engineering
Czech Technical University
Karlovo Náměstí, Praha 2
Czech Republic 120 00

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Control Engineering

DIPLOMA THESIS ASSIGNMENT

Student: **Kashyapa Narenathreyas**

Study programme: Cybernetics and Robotics
Specialisation: Systems and Control

Title of Diploma Thesis: **Fuzzy control of an aircraft**

Guidelines:

1. For a nonlinear model of an aircraft design an appropriate Takagi-Sugeno fuzzy model based on linearization around a nominal trajectory and validate it.
2. Based on the Takagi-Sugeno model design a Parallel Distributed Compensator that will guarantee stability of the closed loop.
3. Compare the designed controller with other control approaches (PID, LQ).

Bibliography/Sources:

[1] Johanssen, T. A., R. Shorten and R. Murray-Smith: On the Interpretation and Identification of Dynamic Takagi–Sugeno Fuzzy Models, IEEE Transactions on Fuzzy Systems, Vol. 8, No. 3, 2000
[2] Farinwata, S. S., D. Filev and R. Langari, Fuzzy Control – Synthesis and Analysis, John Wiley & Sons, 2000
[3] Wang, H. O., K. Tanaka and M. F. Griffin, Parallel distributed compensation of nonlinear systems by Takagi-Sugeno fuzzy model, IEEE Transactions on Fuzzy Systems, Vol. 8, No. 3, 2000
[4] K. Tanaka and H.O. Wang, Fuzzy control systems design and analysis, John Wiley & Sons, New York, 2001

Diploma Thesis Supervisor: Doc.Ing. Petr Hušek, Ph.D.

Valid until the summer semester 2013/2014

prof. Ing. Michael Šebek, DrSc.
Head of Department

L.S.

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 15, 2013

I dedicate this work to my parents, family, friends and my Master

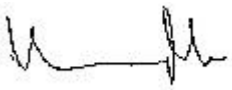
Abstract

Aircraft design consists of many steps such as aerodynamic design, structural analysis and flight control design etc. and flight control is one of the crucial design aspects in modern aircrafts. Modern day aircrafts heavily rely on automatic control systems for most of the functions and there is always a persistent demand for efficient controllers. There are already many control techniques and methods developed in the field of control engineering, but only the conventional control techniques which are more intuitive, are trusted enough in the aviation industry. However, the conventional techniques only work efficiently for linear systems but in real world, the aircraft dynamics are highly nonlinear and thus there is need for a controller which works perfectly for non-linear trajectories. Fuzzy logic control is a nonlinear control technique which uses a linguistic approach for controlling, based on some sets of membership functions and rules. This project attempts to design a Fuzzy Logic controller for the autopilot functions of longitudinal motion of L410 aircraft.

Proclamation

I, Kashyapa Narenathreyas honestly declare that I have completed and worked on this Master thesis by my ownself and I have used only the materials (literature, books, journals etc.) that are stated in the reference section.

I am a student of Czech Technical University and I have complied with the rules and terms held by the university for performing and submitting my thesis.



Prague, 28/05/2013

Signature

Acknowledgement

I would like to thank my parents and my family for supporting me and my studies for all these years and my Master for giving me the strength and knowledge to progress.

I would like to thank my thesis supervisor Dr. Petr Hušek who assisted and encouraged me throughout the period of my thesis. His patience and flexibility towards me helped me to gain knowledge in this new field of research and I am highly grateful of him. I also want to thank Dr. Martin Hromčík for providing me with the nonlinear model of the L410 aircraft through one of his courses.

I also take this opportunity to thank the Erasmus Mundus Consortium for organising this course and supporting my final year with scholarship fund.

Table of Contents

List of Figures

List of Tables

Notations & Abbreviations

A_i	T-S Fuzzy model plant matrix
A_{aug}	Augmented T-S fuzzy plant matrix with reference model
A_c	Reference model plant matrix
B_i	T-S Fuzzy model control matrix
B_{aug}	Augmented T-S fuzzy control matrix with reference model
B_c	Reference model control matrix
B_{caug}	Augmented matrix reference input
$C_{x\alpha}$	Angle of attack derivative of drag coefficient
$C_{x\delta e}$	Elevator angle derivative of drag coefficient
$C_{z\alpha}$	Angle of attack derivative of lift coefficient
$C_{z\delta e}$	Elevator angle derivative of lift coefficient
$C_{m\alpha}$	Angle of attack derivative of pitching moment coefficient
$C_{m\delta e}$	Elevator angle derivative of pitching moment coefficient
C_{mq}	Pitch rate derivative of pitching moment coefficient
D_i	Affine term matrix
D_{aug}	Augmented affine term matrix
D_{trim}	Drag at trimmed conditions (N)
F_t	Engine thrust (N)
g	Acceleration due to gravity (ms^{-2})
H	Aircraft Altitude (m)
I_x	Moment of inertia about x* axis (kgm^2)
I_y	Moment of inertia about y* axis (kgm^2)
I_{xz}	Moment of inertia about x*z* plane (kgm^2)
K_1	Integral gain of hybrid fuzzy controller
K_2	Proportional gain of hybrid fuzzy controller
$K_{i,1,2..n}$	Compensator gain for T-S fuzzy model based on LMI
L_{trim}	Lift at trimmed conditions (N)
m_a	Mass of aircraft (kg)
M	Pitching moment (kgm^2s^{-2})
X_{vx}	Derivative of X with respect to v_x
X_α	Derivative of X with respect to α
X_q	Derivative of X with respect to q
X_θ	Derivative of X with respect to θ
N,n	Number of fuzzy rules
p	Roll rate (rad/s)
q	Pitch rate (rad/s)
$\bar{q}$	Dynamic pressure ($1/2\,\rho U_0^2$) (Pa)
r	Yaw rate (rad/s)
S	Aircraft wing surface area (m^2)
u	Control input

U_0	Resultant aircraft velocity (ms^{-1})
U	Output of fuzzy controller
U_c	Output of hybrid fuzzy controller
v_x	Velocity of aircraft in x* direction (ms^{-1})
v_y	Velocity of aircraft in y* direction (ms^{-1})
v_z	Velocity of aircraft in z* direction (ms^{-1})
w_i	Weighting functions for T-S fuzzy controller
x*	Inertial aircraft axis
$\lambda_{1,2,\ldots,n}$	Input to Mamdani fuzzy system
$\Lambda_{1,2,\ldots,n}$	Mamdani input fuzzy sets
X	Aircraft Drag force (force in x* direction) (N)
X_{vx}	Derivative of X with respect to v_x
X_α	Derivative of X with respect to α
X_q	Derivative of X with respect to q
X_θ	Derivative of X with respect to θ
y*	Inertial aircraft axis
$\omega_{1,2,\ldots,n}$	Output of Mamdani fuzzy system
$\Omega_{1,2,\ldots,n}$	Mamdani output fuzzy sets
Z	Aircraft Lift force (force in z* direction) (N)
Z_{vx}	Derivative of Z with respect to v_x
Z_α	Derivative of Z with respect to α
Z_q	Derivative of Z with respect to q
Z_θ	Derivative of Z with respect to θ
α	Angle of attack (rad)
β	Sideslip angle (rad)
δ_e	Elevator control deflection (rad)
θ	Pitch angle (rad)
ϕ	Roll angle (rad)
μ	Membership functions
σ	Reference signal
χ	Coefficients of reference signal's characteristic equation

FLC	Fuzzy Logic Controller
LQ	Linear Quadratic
LTI	Linear Time-Invariant
MAC	Mean Aerodynamic Chord
MISO	Multi Input Single Output
MPC	Model Predictive Control
PDC	Parallel Distributed Compensator
PID	Proportional Integral Derivative
T-S	Takagi-Sugeno
SISO	Single Input Single Output
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Take-Off and Landing

1. Introduction

After the revolutionary invention of aircrafts by Wright brothers, the aircrafts soon started to adapt the concept of autopilots for making the pilot's job easier. The first automatic flight controller in the world was designed by the Sperry brothers in 1912. The Sperry brothers developed an autopilot that was sensitive to the movements of an aircraft. Currently, the aircraft design relies heavily on automatic control systems to monitor and control many of the aircraft subsystems. Therefore, the development of automatic control systems has played an important role in the growth of civil and military aviation.[1,2]

Although, there have been many developed techniques to control a dynamic system using feedback such as PID control, LQ control and MPC etc. but very few control techniques are actually implemented in the real world flight control applications. The main reason behind not implementing the advanced optimal control techniques is that they are not intuitive and in aerospace where safety is a high priority, unintuitive techniques are not trusted enough to be implemented in real aircrafts. In the advanced modern aircrafts, the conventional PID (Proportional-Integral-Derivative) controllers are used extensively even though they are not very efficient for non-linear dynamic systems, mainly because of their intuitive nature, ease of operation and low cost. To overcome this flaw, an unconventional technique of Fuzzy Logic could be used as it has proven to be more efficient than PID controllers and depends on human experience and intuition.

The Fuzzy control has gained interests of many scientists from various research areas and there have been many successful applications.[17] Fuzzy Logic Controller (FLC) is one of the artificial intelligence methods and its advantages are that it is a nonlinear and rule-based method; therefore no complex model is required. This type of Fuzzy control was expressed by Mamdani and is very popular compared to Takagi-Sugeno type which uses fuzzy sets to define the input variables but the output is defined by means of functions or LTI systems. Therefore, Takagi-Sugeno is considered to be more complicated but stability is guaranteed from this technique.[19, 20]

1.1. Mamdani Fuzzy Logic with PID Combination

The approach of fuzzy PID control has been prominent in Japan, but it has found relatively fewer applications in aerospace field. This controller has the special feature of retaining the same linear structure as conventional PID control, but the control gains are nonlinear functions of the input signals which make it more efficient for nonlinear dynamics.[18] This type of controller was used for conceptual unmanned aerial vehicle (UAV) for longitudinal and lateral autopilots by Institute of Aeronautics and Astronautics, Taiwan in 2011.[3] Here, it was found that the Fuzzy Logic controllers were effective and capable of waypoint navigation, trajectory following and even resist and stabilize from wind/gust disturbance.[1] Many other previous experimentations have been carried out using fuzzy PID combination for control system building a hybrid intelligent control scheme such as controller for VTOL quad-rotor piloting system,[8] small scale helicopters[21] etc.

The biggest advantage of the hybrid fuzzy PID controller is the robustness against noise, and its ease for implementation. There have been lot of experiments and research regarding the implementation and application of fuzzy logic in flight control systems from UAVs to even fighter jets.[3,4,6,7]

1.2. Takagi-Sugeno (T-S) Model

The heuristic technique of Mamdani fuzzy control mentioned in section 1.1 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 T-S model retains the advantages of the fuzzy control, and it is also constructed in a mathematically rigorous method and as a result, stability and control analysis has been developed.[9]

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.[9] 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 in spite of the more intuitive nature of Mamdani fuzzy reasoning in terms of dealing with uncertainty.

The T-S fuzzy model has not been in the research interest of the aerospace field, and not many effective attempts have been made till now to utilize this method for flight control experimentation. The motivation of this project is to demonstrate the T-S modelling of

aircraft dynamics and control techniques for flight handling, and also to demonstrate the advantages and disadvantages of the T-S model over the Mamdani model.

1.3. Objectives

The main objectives of the project are mentioned below:

- Designing a Mamdani Fuzzy Logic Controller (autopilot) for L410 aircraft longitudinal decoupled dynamics using the hybrid Fuzzy PI controllers.
- Design a Takagi-Sugeno model for L410 aircraft longitudinal decoupled dynamics, and develop a Parallel Distributed Controller.
- Simulation of the control systems developed on Simulink.
- Comparing with conventional control techniques used in aerospace industries.

1.4. Report Outline

This thesis is focused mainly on the design of a fuzzy type autopilot controller which will improve the stability of the system. Primary computational tool for the design was MATLAB and Simulink. The Model of the aircraft was provided by Department of Control Engineering at Czech Technical University.

Chapter 2 of the thesis describes briefly the theory and mathematical equations which are necessary to understand and comprehend the work done in this project. Chapter 3 of the thesis explains and demonstrates the model building and controller designing processes.

Chapter 4 presents the results of the performance of T-S model and the application of designed controllers to the nonlinear model. Chapter 5 makes concluding remarks on the results obtained and possible future work for the project.

2. Theory

In order to understand and discuss the modelling and simulations, it is necessary to get the fundamentals dealing with the project. This section will describe the basic information regarding the longitudinal dynamics and fuzzy modelling theories.

2.1. Aircraft Longitudinal Motion

The longitudinal dynamics of the aircraft only considers Pitching Moment *M*, Drag force *X* and Lift force *Z* and the variables affecting these quantities. In Figure 2-1, the variables for both longitudinal and lateral motions are shown. Therefore, longitudinal motion can be visualised to be on x-z plane and the moments are only considered about y-axis.

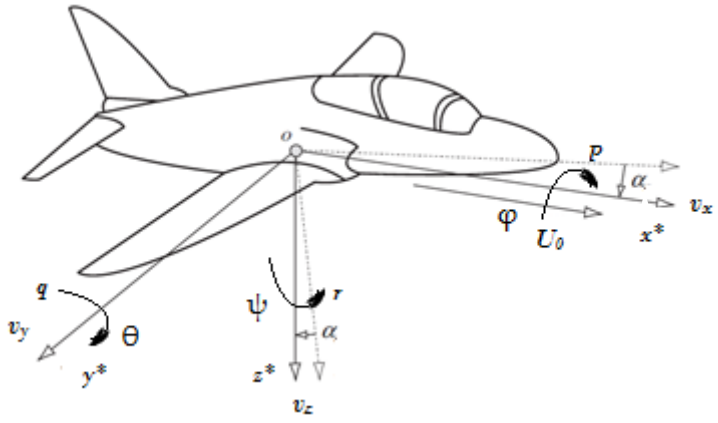


Figure 2-1: Shows the schematic depicting the variables of aircraft motion about different axis[2]

The resultant components of total force and moments on the rigid body are given by equations below, and as only the longitudinal motion is considered the equations for lateral motion is not presented.

$$X = m_a(\dot{v_x} - rv_y + qv_z) \tag{2-1}$$

$$Z = m_a(\dot{v_y} - pv_z + rv_x) \tag{2-2}$$

$$M = I_y\dot{q} + (I_x - I_z)pr + I_{xz}(p^2 - r^2) \tag{2-3}$$

In the above equations m_a is aircraft mass (kg), v_x is velocity component in x^* direction (m/s), r is the yaw rate (rad/s), v_y is the velocity component in y^* direction (m/s), q is pitch rate (rad/s), p is roll rate (rad.s), v_z is velocity in z^* direction (m/s), I_y is moment of inertia about y^* axis (kgm^2), I_x is moment of inertia about x^* axis (kgm^2), I_z is moment of inertia about z^* axis and I_{xz} is moment of inertia about x^*z^* plane. Even though the open loop dynamics might be stable, but there are certain aircraft modes present which produce instability such as phugoid motion. Therefore, there is need for stability augmentation and this is usually done by closed loop feedback method. In longitudinal motion, in order to damp the high amplitude short period oscillations (oscillations in pitch angle excited due to some disturbances or pilot input), a pitch rate (q) damper is introduced through a proportional gain feedback to elevator input (δ_e). In many instances, a wash-out filter is also additionally introduced in the feedback to improve the damping performance.

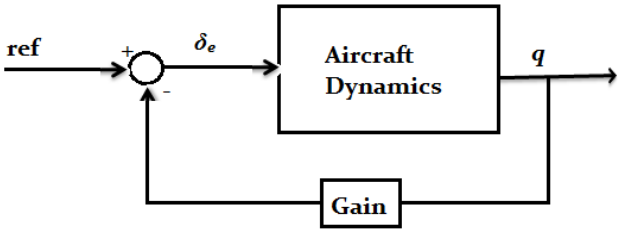


Figure 2-2: Block diagram of pitch damper[2]

Since, this project considers only with the longitudinal dynamics, the Pitch Autopilot is explained in detail here. The Pitch Autopilot by its name concerns with feedback from pitch angle (θ) and produces a reference input angle for the elevator. The block diagram demonstrating the Pitch Autopilot is shown in Figure 2-3.

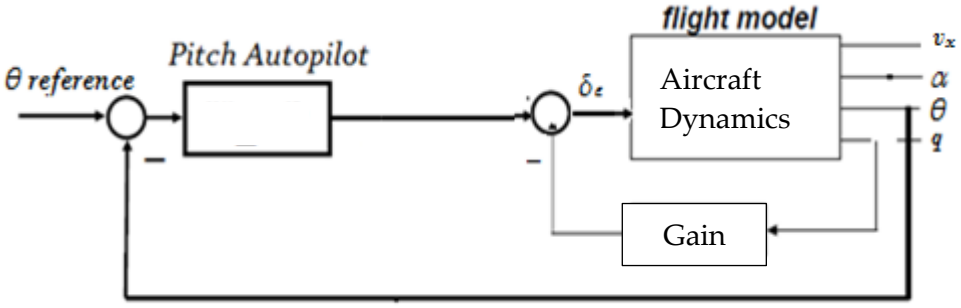


Figure 2-3: Block diagram of Pitch Autopilot also showing the Pitch Damper

The Pitch Autopilot's main function is to control the pitch angle of the aircraft. During climb or other manoeuvres in longitudinal plane, the pitch angle must be constantly controlled for performing the required manoeuvre.

2.2. Fuzzy Systems

The world's first fuzzy controller was developed by Prof. E. H. Mamdani in 1974 and basic idea was to utilise the human operator's knowledge and experience to intuitively construct controllers which imitate or more precisely behave in same manner as a human operator. Fuzzy models are more intuitive and easier to understand than neural network models because fuzzy sets, fuzzy logic, and fuzzy rules are all intuitive and meaningful. However, fuzzy models are not as simple as those models that can be expressed in mathematical formulae.[10]

There are two major types of fuzzy controllers namely Mamdani and Takagi-Sugeno. The classification mainly depends on the output form; Mamdani type produces output in the form of fuzzy sets while Takagi-Sugeno produces output in the form of functions or LTI systems. Both types of fuzzy controllers are described in following subsections of this section.

2.2.1. Mamdani Fuzzy Controllers

In Mamdani type model, the inputs and outputs are defined in fuzzy sets through membership functions which also define the range of the inputs and outputs beyond which the controller will be futile. The basic process involves different stages such as 1) fuzzification of crisp values of the input fuzzy sets, 2) fuzzy inference where the fuzzy sets are mapped according to the fuzzy rules, and 3) defuzzification. The controller process has been shown in Figure 2-4.[10]

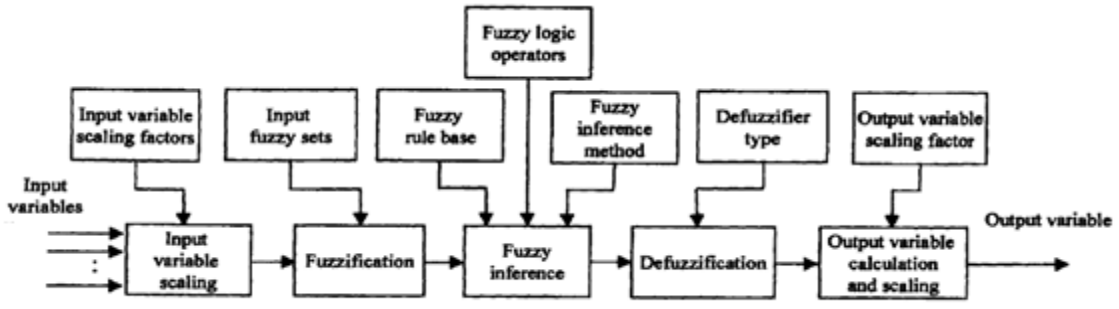


Figure 2-4: Structure of a typical MISO Mamdani fuzzy controller[10]

The rules are defined in a linguistic manner which can be quantified mathematically later. The general form the rules are shown below:[11]

IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_v is Λ_{1v} THEN ω is Ω_1
IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_v is Λ_{2v} THEN ω is Ω_2
…
IF λ_1 is Λ_{n1} AND λ_2 is Λ_{n2} …. AND λ_v is Λ_{nv} THEN ω is Ω_n

where the λ_j ($j = 1,2,\ldots v$) is input to the fuzzy system, Λ_{ji} ($i = 1,2\ldots,n$) are input fuzzy sets, v is the number of inputs, n is the number of rules, ω is the output of the fuzzy system and Ω_j is the output fuzzy set. The fuzzy sets are represented through membership functions. There are number of different membership functions expressed in various shapes such as Triangular, Gaussian and Trapezoidal etc. In this report the membership functions are denoted by μ. The function of the fuzzy inference is to produce an output fuzzy set from the defined rules. Final stage involves with defuzzification of the output fuzzy sets computed in the fuzzy inference. There are many defuzzifiers also but the most popular is the centroid method and the output produced can be expressed mathematically by equation (2-4): [10,11,19,20]

$$U = \frac{\sum_{i=1}^{n} \mu(\Omega_j).c^i}{\sum_{i=1}^{n} \mu(\Omega_j)} \qquad (2\text{-}4)$$

where U is the defuzzified output of the fuzzy system, $\mu(\Omega_j)$ is the output fuzzy set and c^i is the centroid point of the all the fuzzy parts for a particular rule j determined by inference.

The concept of combining the output of the above described fuzzy system with PI controllers is called Hybrid Fuzzy PI Controller. The output from the fuzzy system is passed through a pre-defined PI controller which produces a final value of the combined system. The basic structure of fuzzy PI controller is shown in Figure 2-5 in a block diagram form and as seen here, the feedback inputs are passed through fuzzy system and output from the fuzzy system is the input for PI control. The output of this controller is given by the equation (2-5):[5,4]

$$U_c(t) = K_1 \int_0^t U(\tau)d\tau + K_2 U(t) \qquad (2\text{-}5)$$

Where U is the time dependent output from fuzzy controller, K_1 and K_2 are the integral and proportional gains of the PI controller and U_c is the final output of the combined fuzzy PI controller.

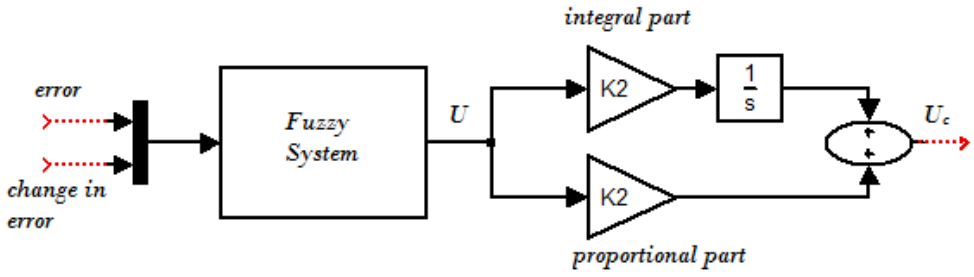


Figure 2-5: Block diagram showing structure of fuzzy PI controller[5]

2.2.2. Takagi-Sugeno Fuzzy Controller

In the Takagi-Sugeno fuzzy model, instead of describing the rules as shown in section 2.2.1 the output is not defined to be a fuzzy set but the output is defined as a LTI system in this case. The example of the IF-THEN rules is shown below:[12]

IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_v is Λ_{1v} THEN $\dot{x} = A_1x + B_1u$
IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_v is Λ_{2v} THEN $\dot{x} = A_2x + B_2u$
…
IF λ_1 is Λ_{n1} AND λ_2 is Λ_{n2} …. AND λ_v is Λ_{nv} THEN $\dot{x} = A_nx + B_nu$

where the λ_j ($j = 1,2,\ldots v$) are premise variables of the dynamic system (premise variables are the variables on which the linearized local submodels depend on), Λ_{ji} ($i = 1,2\ldots,n$) are the fuzzy sets defining the premise variables, v is the number of premise variables, n is the number of rules as in section 2.2.1, A_i ($n \times n$) and B_i ($n \times m$) are plant and control matrices where $i = 1,2,\ldots n$ and these are called local submodels, and x and u are the states and input of the models. Therefore, the IF part is fuzzy but the THEN part is crisp.[12] Here, every rule describes a local model and each model contributes to the global model. The nonlinear model is linearized at some operating points in order to produce the local affine submodels. If the nonlinear system is represented in the form of equation (2-6):

$$\dot{x} = f(x,u) \tag{2-6}$$

At a certain operating point (x',u'), the local linearization of equation (2-6) is given by:

$$\dot{x} = A(x',u')(x - x') + B(x',u')(u - u') \tag{2-7}$$

Here the matrices A and B are the local submodels plant and control matrices at the operating point.[13] The local affine submodels require the affine terms d_i in order to be accurate.

$$A(x',u') = \frac{\partial f}{\partial x}|_{(x',u')} \tag{2-8}$$

$$B(x',u') = \frac{\partial f}{\partial u}|_{(x',u')} \tag{2-9}$$

$$d(x',u') = Ax' + Bu' \tag{2-10}$$

The linearization of the nonlinear dynamics is accurate only if the affine terms are also included in the model. The local submodels expressed in State-Space form are presented in equations (2-11) and (2-12):[13]

$$\dot{x} = \sum_{i=1}^{n}(A_ix + B_iu + d_i)w_i(x,u) \tag{2-11}$$

$$y = \sum_{i=1}^{n}(C_ix)w_i(x,u) \tag{2-12}$$

where i = 1,2 … ,n, and w_i are the weighting functions determined according to the membership functions as shown in equation (2-13):[13,19]

$$w_i(x,u) = \frac{\mu_i(x,u)}{\sum_{j=1}^{n} \mu_j(x,u)} \tag{2-13}$$

where $\mu_j(x,u)$ represent the fuzzy sets which was denoted earlier by Λ_{ji} and equation (2-13) assumes that $\sum \mu_j(x,u) > 0$ for all (x,u).

In the control design, for each local affine model, a linear feedback control is designed. The resulting controller, which is nonlinear is a fuzzy blending of each individual linear controllers.[14] This type of blending of the controllers, when setup in parallel is called Parallel Distributed Compensator (PDC). The idea is that for each controller, the IF statements are the same as the model but the THEN part defines the controller.[15,22] The controller rules are shown below:

IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_n is Λ_{1v} THEN u is K_1x
IF λ_1 is Λ_{11} AND λ_2 is Λ_{12} …. AND λ_n is Λ_{2v} THEN u is K_2x
…
IF λ_1 is Λ_{n1} AND λ_2 is Λ_{n2} …. AND λ_n is Λ_{nv} THEN u is K_nx

where the λ_j (j = 1,2,…v) are premise variables of the dynamic system, Λ_{ji} (i = 1,2…,n) are the fuzzy sets defining the premise variables as earlier and K_i are the controller gains. Hence the fuzzy controller is defined as shown in equation (2-14):

$$u = \sum_{i=1}^{n} (K_i x) w_i(x,u) \tag{2-14}$$

In order to obtain the controller gains K_i which stabilises the system globally, the LMIs (Linear Matrix Inequalities) shown in equations (2-15) and (2-16) are solved using convex LMI programming. The theorem shown below defines the conditions for obtaining the controller gains.[16]

Theorem: *the fuzzy control system is stabilizable in the large via PDC if there exist a positive definite matrix $Q > 0$ and regular matrices W_i, i = 1,2, … , n, such that the following LMI conditions hold:*[23]

$$QA_i^T + A_iQ - B_iW_i - W_i^T B_i^T < 0 \tag{2-15}$$

$$QA_i^T + A_iQ + QA_j^T + A_jQ - B_iW_j - W_j^T B_i^T - B_jW_i - W_i^T B_j^T < 0 \tag{2-16}$$

$$i < j \leq n$$

Here the matrix Q has dimensions (n×n) and matrices W_i have the dimensions (m×n). Once, the Q and W_i matrices are obtained, the controller gain K_i is given by $K_i = W_iQ^{-1}$. This

process is very effective and guarantees stability, but in the autopilot design, the controller has to be designed which can track the given reference. The reference tracking for PDC is not as simple as conventional methods; the process involves augmenting the plant and control matrices of the linear submodels with the reference model. The equations for reference model are shown below:[23]

$$\dot{x_c} = A_c x_c + B_c e \tag{2-17}$$

$$e = y_r - y \tag{2-18}$$

Here, x_c and e are the states and the tracking error input for reference model, y_r is the reference signal and y is the output of the main system described in equation (2-12). The matrices A_c and B_c are calculated by the characteristic equation of the reference signal, i.e. $\sigma(s) = s^l + \chi_{l-1} s^{l-1} + \ldots + \chi_0$, so that it can expressed in canonical form as shown below:[23]

$$A_c = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \vdots & \\ 0 & I_{l-1} & \vdots \\ -\chi_0 & -\chi_1 \cdots & -\chi_{l-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

The final controlled system with reference tracking is expressed as shown in equation (2-20).

$$\begin{bmatrix} \dot{x} \\ \dot{x_c} \end{bmatrix} = \sum_{i=1}^{n} \left(\begin{bmatrix} A_i & 0 \\ -B_c C_i & A_c \end{bmatrix} \begin{bmatrix} x \\ \dot{x_c} \end{bmatrix} + \begin{bmatrix} B_i \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ B_c \end{bmatrix} y_r + \begin{bmatrix} D_i \\ 0 \end{bmatrix} \right) w_i(x, u) \tag{2-19}$$

And now, the PDC will be calculated according to the equation (2-20) and fed back to the original system.

3. Conceptual Model Setup

There have been many phases and milestones in the setup of the project. Firstly, the setup and model of L410 aircraft has been described in this section. Also, the setup of Mamdani and Takagi-Sugeno fuzzy controllers has been described in the later subsections of this section.

3.1. L410 Aircraft Model

The aircraft used to design and model the fuzzy control systems is L410 aircraft which is a twin-engine short-range transport aircraft manufactured by Czech aircraft manufacturer LET. The aircraft is a turbo-propeller type with excellent latent stability. The cost of operation and maintenance is also very low compared to other aircrafts of similar size and operational conditions.

Figure 3-1: L410 aircraft by LET aircraft manufacturer

The basic structural configuration and specifications of L410 aircraft is shown in Table 3-1 and operational conditions in Table 3-2.

Table 3-1: Showing the aircraft's structural dimensions and specifications.

Structural Specifications	
Wing Span	19.98 m
Length	14.424 m
Height	5.83 m
Wing area	34.86 m^2
Passenger capacity	19
Maximum take-off mass	6600 kg

Table 3-2: Showing the operational and trim conditions of the aircraft

Operational Conditions	
Velocity (U_o)	150 m/s
Mach number	0.468
Altitude (H)	5000 m
Aircraft Mass (m_a)	5000 kg
Moment of Inertia (I_x)	6000 kgm^2
Moment of Inertia (I_y)	38000 kgm^2
Moment of Inertia (I_z)	34000 kgm^2
Moment of Inertia (I_{xz})	2750 kgm^2
Trim Conditions	
Angle of Attack (α)	2.287°
Pitch Angle (θ)	2.287°
Elevator Deflection (δ_e)	-0.7742°
Engine Thrust (F_t)	5896.9 N

3.2. Aerodynamic Derivatives and Coefficients

To compute the aerodynamic values and coefficients, a panel method solver called Tornado was used. The Tornado code is a vortex lattice method programmed to be used in conceptual aircraft design and in aerodynamic education. The program is coded in MATLAB and the code is provided under the General Public License.

Geometry of main wing and tail plane of L410 aircraft was created in the Tornado solver for computing the aerodynamic performance. The body of the aircraft was not included as it was not necessary in this case. The basic visualization of the created geometry is shown in Figure 3-2. The term MAC in the figure refers to the mean aerodynamic chord.

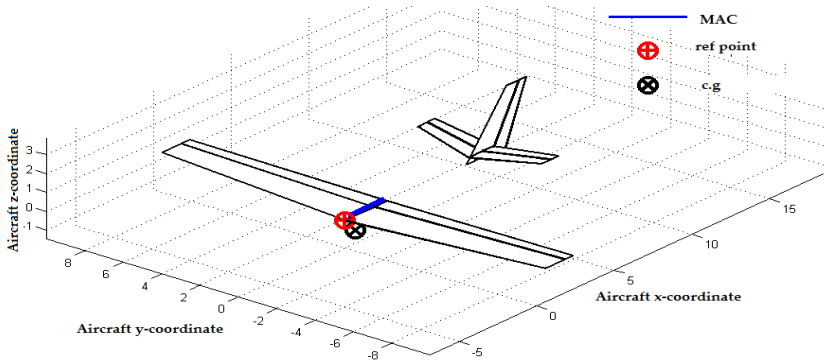


Figure 3-2: L410 aircraft geometry created in the Tornado interface

This geometry was analysed at the operating conditions mentioned in Table 3-2 and keeping the pitch rate $q = 0$ rad/s. For the longitudinal case, the variables regarding lateral states were all kept at zero (i.e. $\beta = p = r = 0$). This way a completely decoupled dynamic motion could be executed. One of the important assumptions here is that the aircraft is in straight flight.

3.3. Autopilot Controller Design

For the control design, two different fuzzy control methods were designed simulated namely Mamdani and Takagi-Sugeno. The performance of these two controllers was compared with conventional PI controller.

In order to design a PI control, a given State-Space model of the nonlinear dynamics was used. The State-Space model is given in equation (3-1) below:

$$\dot{x} = Ax + Bu \tag{3-1}$$

$$\text{where } A = \begin{bmatrix} X_{v_x} & X_\alpha & X_q & X_\theta \\ Z_{v_x} & Z_\alpha & Z_q & Z_\theta \\ M_{v_x} & M_\alpha & M_q & M_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix}$$

The elements of the State-Space matrices are determined by calculating aerodynamic stability derivatives which is done by the method shown in section 3.2, the explanation of these concepts are beyond the scope this report, the values have given in Appendix D. The details can be found in reference [2].

Before the autopilot design, a pitch damper was designed first and main function of the pitch damper is to damp the high amplitude short period oscillations caused by random disturbances or guts or pilot input. A feedback from Pitch Rate is passed through a gain/filter and fed back into elevator input. The Open Loop transfer function (OLTF) of assumed SISO system is shown in equation (3-2).

$$OLTF\ (s) = \frac{-8.7833s(s+0.8354)(s+0.09973)}{(s^2+0.05156s+0.005346)(s^2+1.893s+7.419)} \qquad (3\text{-}2)$$

This synthesis was done using the Root-locus method of the system shown in equation (3-2) considering it to be a SISO system as shown in Figure 2-2. In Figure 3-3 the Root-locus plot and Bode plot of the damper are shown and it can be seen that poles are moved to higher stability region thus decreasing the oscillations and the transfer function of damper is shown in equation (3-3).

$$Damper(s) = 0.8\left(\frac{s}{s+0.5}\right) \qquad (3\text{-}3)$$

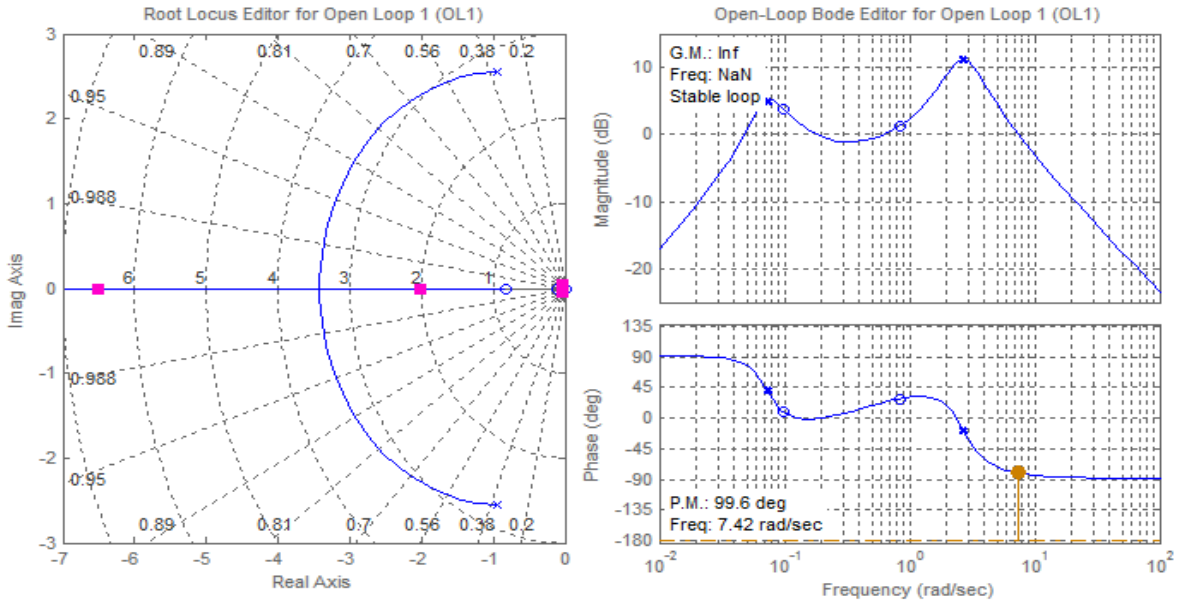


Figure 3-3: Showing the Open loop system and the designed feedback poles for Pitch Damper

The closed system feedback system was the new system now and the transfer function of the system with pitch damper is shown in equation (3-4) below:

$$transfer\ function = \frac{-8.7833(s+0.9973)(s+0.5)(s+0.8354)}{(s+7.5651)(s+1.511)(s+0.3592)(s^2+0.04022s+0.004833)} \qquad (3\text{-}4)$$

Now, based on the new system with pitch damper, a Pitch Autopilot was designed with feedback from θ and using the Root-locus method once again considering a SISO system with θ as output and δ_e as the input, a PI compensator was designed and the transfer function is shown below in equation (3-5).

$$PI(s) = \frac{1.25+0.5s}{s} \qquad (3\text{-}5)$$

In Figure 3-4 the Root-locus and Bode plot for designed PI Autopilot control is shown and actually it is seen that there are complex poles very close to the imaginary axis and these represent the long period Phugoid motion which in reality is hard to control.

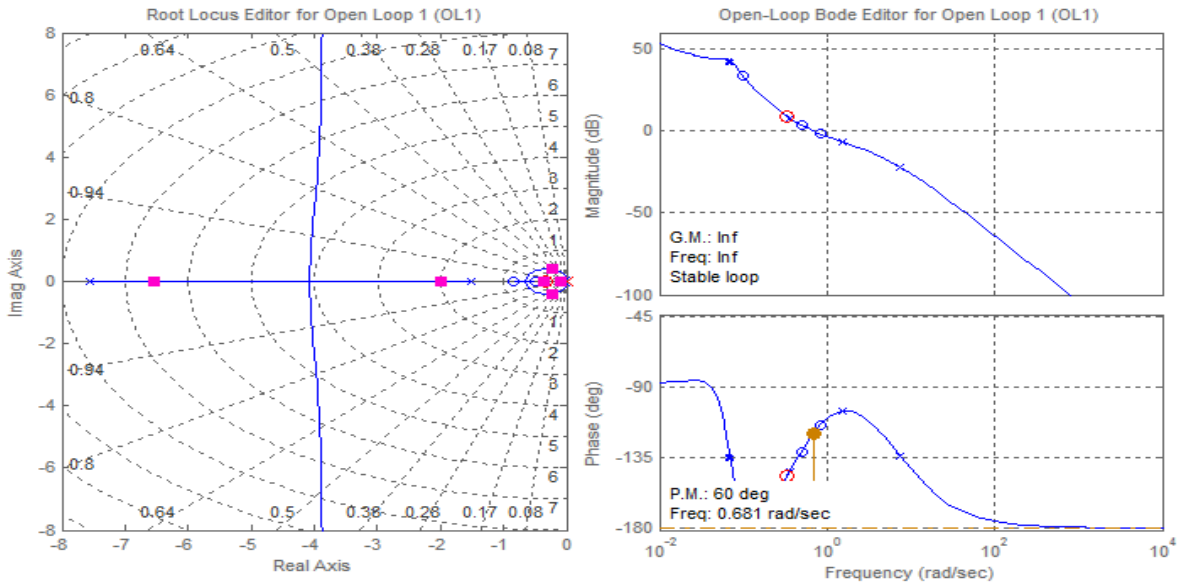


Figure 3-4: Root-locus and Bode plot for PI control with closed-loop pitch damper system

3.3.1. Mamdani Fuzzy System

The Mamdani Fuzzy Logic controller designed for pitch autopilot has two parts as explained in section 2.2.1, the feedback inputs pass through fuzzy controller and the output of fuzzy controller is the input for PI controller. The block diagram shown in Figure 3-5 demonstrates the basic structural setup of the system.

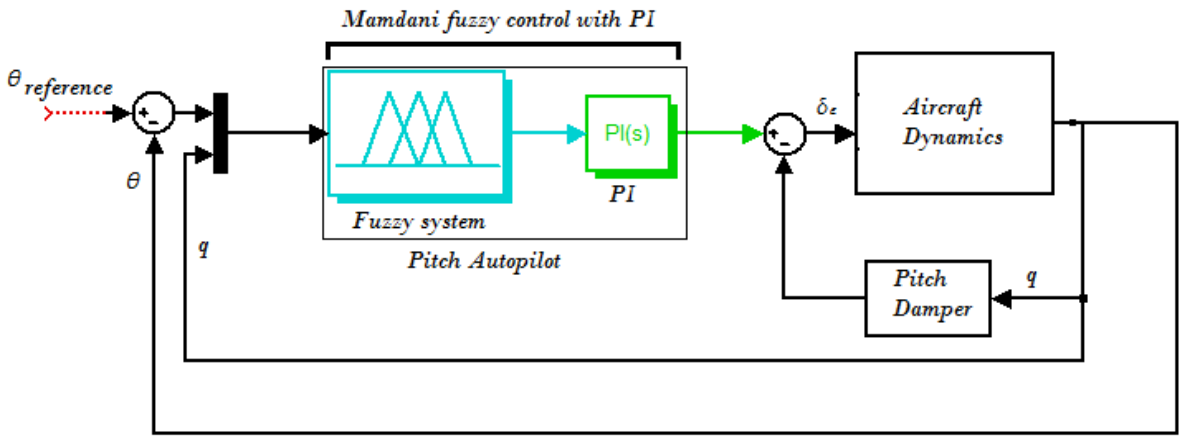


Figure 3-5: Hybrid Fuzzy Logic PI autopilot controllers for longitudinal system

The fuzzy controller was designed in Matlab using the inbuilt fuzzy interface system. As explained in section 2.2.1, the fuzzy inference engine needs two inputs: error and change in error. In the longitudinal system, the two inputs were pitch angle (θ) and pitch-rate (q) and the output of the fuzzy inference engine was the elevator deflection angle (δ_e). The Fuzzy interface system in Matlab is shown in Figure 3-6.

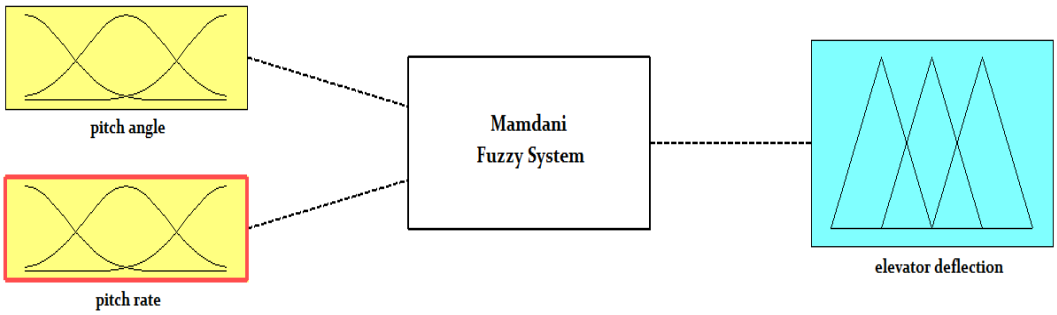


Figure 3-6: Fuzzy interface block diagram showing the connections between input and output

The membership functions used were simple triangular functions with different range of angles for inputs and output as shown in Figure 3-7, and the methods used for fuzzification and defuzzification are shown in Table 3-3 which is the default setup in fuzzy interface system in Matlab. The range of membership functions were chosen by the detailed study of the aircraft and survey of research done previously as mentioned in section 1.1.[4,6]

Table 3-3: Methods used in the fuzzy inference engine

Fuzzy Inference Engine	
AND method	Min
OR method	Max
Implication	Min
Aggregation	Max
Defuzzification	Centroid

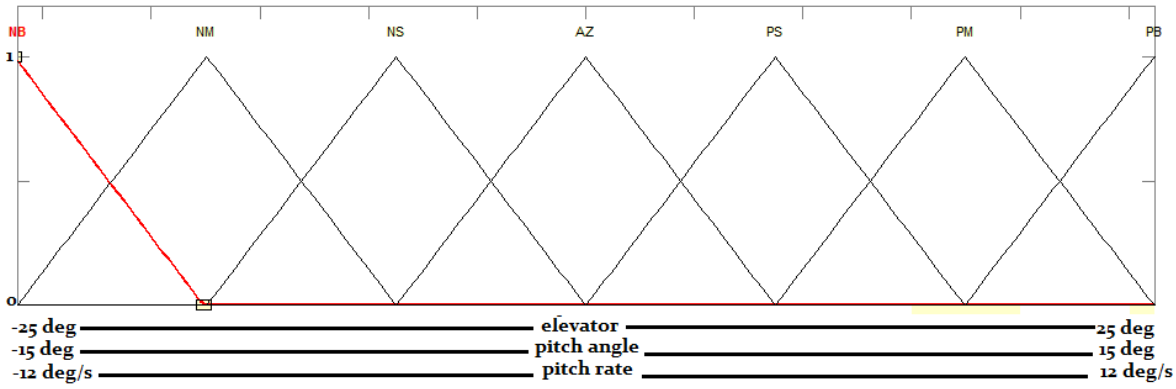


Figure 3-7: Membership functions of the inputs and output for fuzzy system

The fuzzy rules are shown in Table 3-4, here the abbreviations of the membership functions denoting NB – negative big, NM – negative medium, NS – negative small, AZ – around zero, PS – positive big, PM – positive medium and PB – positive big.

Table 3-4: Mamdani Fuzzy rules with 7 membership functions

E		θ						
		NB	**NM**	**NS**	**AZ**	**PS**	**PM**	**PB**
q	**NB**	NB	NB	NB	NM	NM	PS	PM
	NM	NB	NB	NM	NM	NS	PS	PB
	NS	NB	NB	NM	NS	AZ	PM	PB
	AZ	NB	NM	NS	AZ	PS	PM	PB
	PS	NB	NS	AZ	PS	PM	PM	PB
	PM	NB	NS	AZ	PM	PM	PB	PB
	PB	NM	NS	PS	PM	PM	PB	PB

The second part of the system which is a PI controller which was designed earlier and the same controller was used in this system as well.

3.3.2. Takagi-Sugeno Model & Parallel Distributed Compensator

The aircraft dynamics described in section 2.1, was rearranged in such a way that it would represent the form shown in equation (2-6). The equations for longitudinal motion are shown below (the lateral motion terms are kept zero). The complete equations of longitudinal motion are presented in Appendix A: Complete Equations of Motion.

$$\dot{v_x} = \frac{1}{m_a}(F_t\cos(\alpha) + mg(\cos(\alpha)\sin(\theta) + \sin(\alpha)\cos(\theta)) + D_{trim} - X) \quad (3\text{-}6)$$

$$\dot{\alpha} = q + \frac{1}{m_a U_o}\left(-Z - L_{trim} - F_t\sin(\alpha) + m_a g(\cos(\alpha)\cos(\theta) + \sin(\alpha)\sin(\theta))\right) \quad (3\text{-}7)$$

$$\dot{q} = \frac{M}{I_y} \quad (3\text{-}8)$$

$$\dot{\theta} = q \quad (3\text{-}9)$$

$$X = \bar{q}S\left(C_{x_\alpha}\alpha + C_{x_{\delta_e}}\delta_e\right) \quad (3\text{-}10)$$

$$Z = \bar{q}S\left(C_{z_\alpha}\alpha + C_{x_{\delta_e}}\delta_e\right) \quad (3\text{-}11)$$

$$M = \bar{q}S\bar{c}\left(C_{m_\alpha}\alpha + C_{x_{\delta_e}}\delta_e + C_{m_q}\frac{q\bar{c}}{2V_o}\right) \quad (3\text{-}12)$$

where F_t is the engine thrust (N), D_{trim} is drag force (N) at trimmed condition (equal to F_t), L_{trim} is lift force (N) at trimmed condition (equal to $m_a g$), g is the acceleration due to gravity (m/s^2), $\bar{q}$ is the dynamic pressure (Pa), U_0 is the resultant velocity (m/s), S is the wing surface area (m^2) and $\bar{c}$ is the mean aerodynamic chord (m). Therefore, from the equations (3-6) to (3-9), the general form could be written as:

$$[(v_x, \dot{\alpha}, q, \theta)] = f(v_x, \alpha, q, \theta, \delta_e) \qquad (3\text{-}13)$$

The matrices *A, B* and *d* for T-S local submodels for were calculated to be:

$$A_i = \begin{bmatrix} 0 & g(\sin(\alpha')\sin(\theta') + \cos(\alpha')\cos(\theta')) - 1.1794\sin(\alpha') + 116.5903 & 0 & -g(\sin(\alpha')\sin(\theta') + \cos(\alpha')\cos(\theta')) \\ 0 & 0.0654(\cos(\alpha')\sin(\theta') - \sin(\alpha')\sin(\theta')) - 0.0079\cos(\alpha') - 1.7723 & 1 & 0.065(\sin(\alpha')\cos(\theta') - \cos(\alpha')\sin(\theta')) \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} -1.008 \\ -0.1185 \\ -15.2534 \\ 0 \end{bmatrix} \qquad d_i = A_i \begin{bmatrix} v'_x \\ \alpha' \\ q' \\ \theta' \end{bmatrix} + B\delta_e{}'$$

In the above given matrices, α',θ', v_x and q' are the states at the corresponding operating points according to the rule.

From the above shown matrix *A*, it can be seen that the model depends only on two variables namely pitch angle θ and angle of attack α. These are the so called premise variables for Takagi-Sugeno models. Now, the nonlinear model was linearized over three operating points. For both variables, there was maximum value, minimum value and value inbetween for which the trajectory was defined. The trajectory in this case was the straight flight in trimmed condition. The range for θ was (-12,2.287,12)° and for α was (-10,2.287,10)°. The operating points are pictured in Figure 3-8.

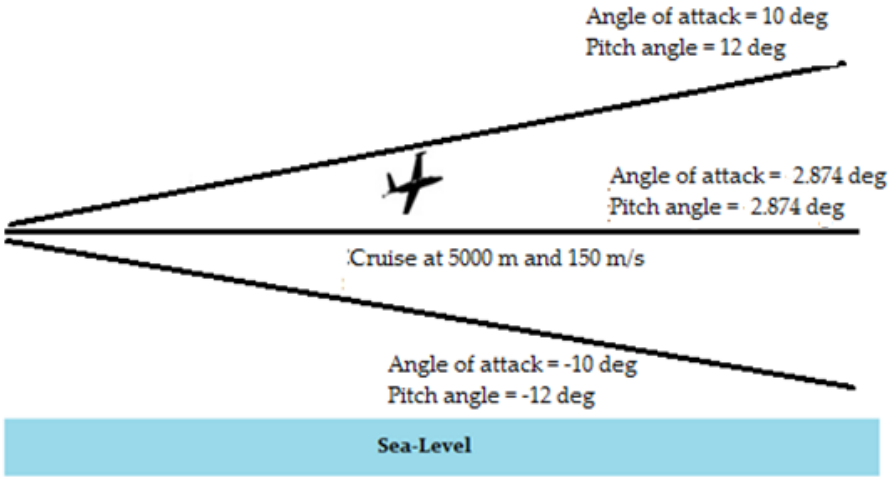


Figure 3-8: Operating points for longitudinal motion to design linear submodels

As there were three membership functions for each premise variables, the total number of rules was 3^2 equal to 9 rules. The membership functions expressed are shown below in Figure 3-9.

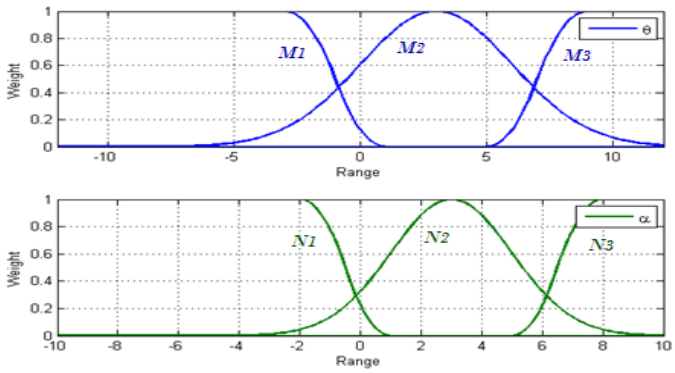


Figure 3-9: Membership functions of pitch angle and angle of attack for T-S models

Since the B matrix is common for all submodels, the rules were based on A_i and D_i matrices as shown below and Figure 3-10 shows the Simulink scheme of the rules. The matrices are given in Appendix B: TS Submodels.

IF θ is $M1$ and α is $N1$, THEN $\dot{x} = A_1x + Bu + D_1$
IF θ is $M1$ and α is $N2$, THEN $\dot{x} = A_2x + Bu + D_2$
IF θ is $M1$ and α is $N3$, THEN $\dot{x} = A_3x + Bu + D_3$
IF θ is $M2$ and α is $N1$, THEN $\dot{x} = A_4x + Bu + D_4$
IF θ is $M2$ and α is $N2$, THEN $\dot{x} = A_5x + Bu + D_5$
IF θ is $M2$ and α is $N3$, THEN $\dot{x} = A_6x + Bu + D_6$
IF θ is $M3$ and α is $N1$, THEN $\dot{x} = A_7x + Bu + D_7$
IF θ is $M3$ and α is $N2$, THEN $\dot{x} = A_8x + Bu + D_8$
IF θ is $M3$ and α is $N3$, THEN $\dot{x} = A_9x + Bu + D_9$

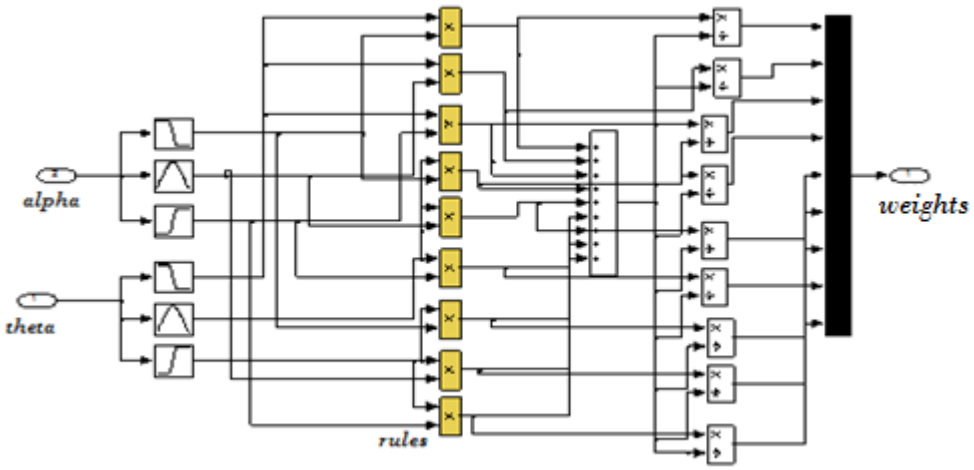


Figure 3-10: Simulink scheme for Takagi-Sugeno model fuzzy rules

The block diagram of the T-S fuzzy with PDC connected in a closed-loop structure with nonlinear dynamics is shown in Figure 3-11. The PDC determined solving the convex LMI conditions. The control matrices are given in the Appendix C: PDC control Gains.

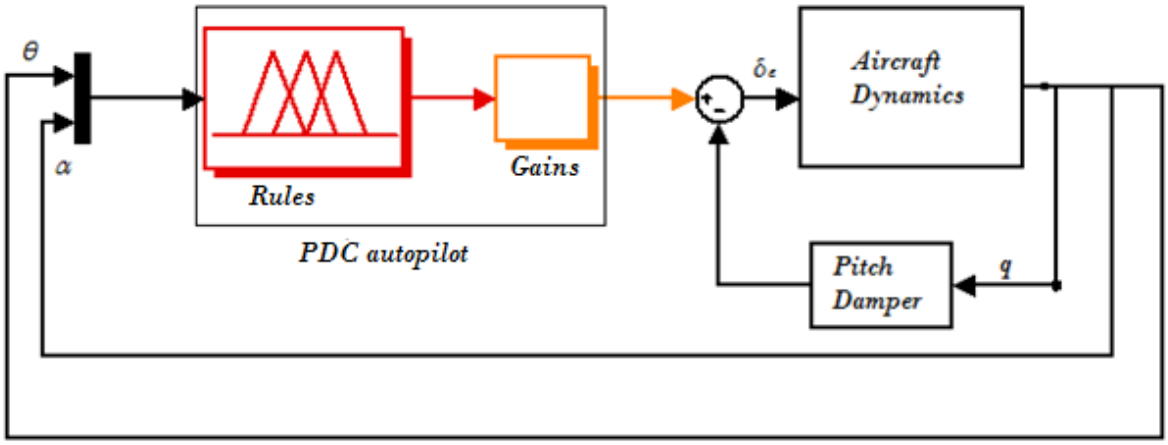


Figure 3-11: Takagi-Sugeno fuzzy model scheme in Simulink

Now when the reference tracking was to be included, the system was augmented with the reference model. The reference signals selected in this case was a step function, and the characteristic equation of a step function is given by $\psi(s) = s$. therefore, the reference model can be written as shown below:

$$A_c = [1] \qquad B_c = [1]$$

As the reference was tracked only for θ, the C_i matrix was chosen so that only the pitch angle will be the output.

4. Simulation Results

In this section, the results of the simulations conducted for the aircraft longitudinal motion using Mamdani and T-S fuzzy controllers are demonstrated and discussed briefly explaining the differences and concluding on the remarks.

4.1. Performance of Takagi-Sugeno Model

The Takagi-Sugeno model had several model building stages such as finding out the plant matrices, control matrix and affine matrices. In many works, the affine terms are usually omitted and designed a controller without any affine terms. It might work in some cases but the model is very inaccurate without the affine terms and especially when considering designing a control for aircraft, the model has to be very accurate.

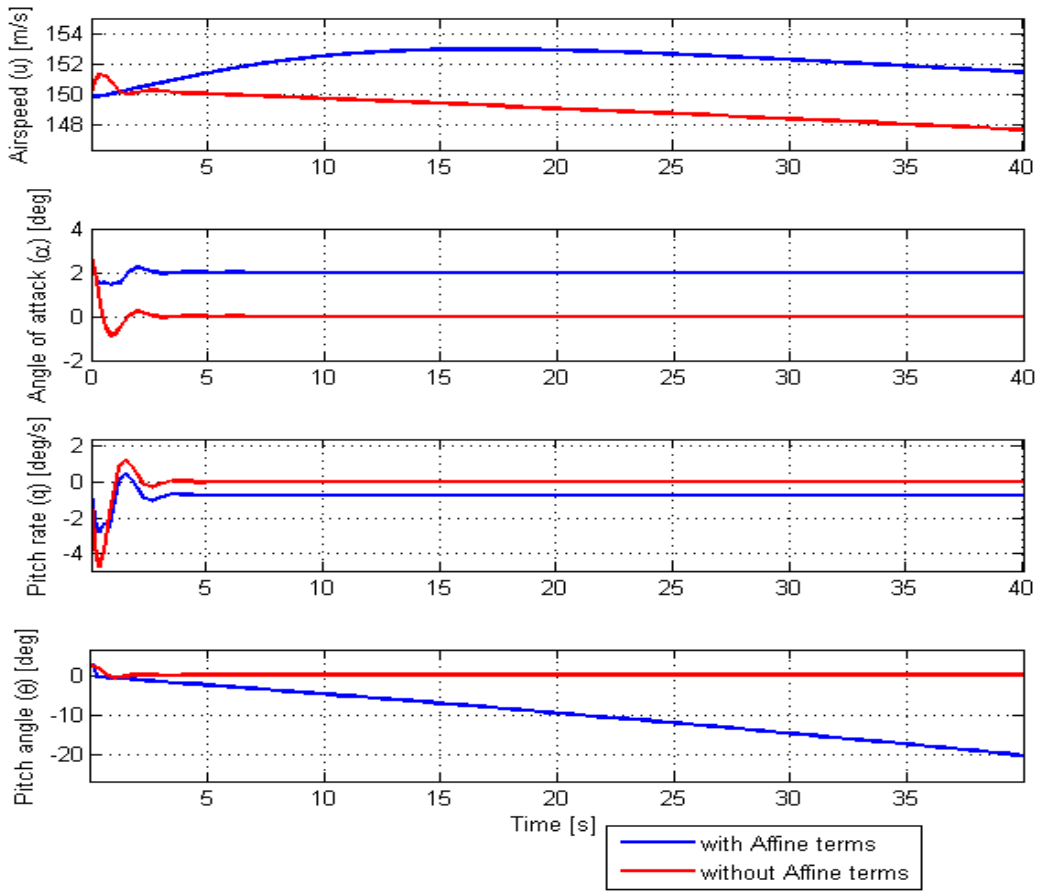


Figure 4-1: Difference between open-loop responses of T-S model with and without affine terms

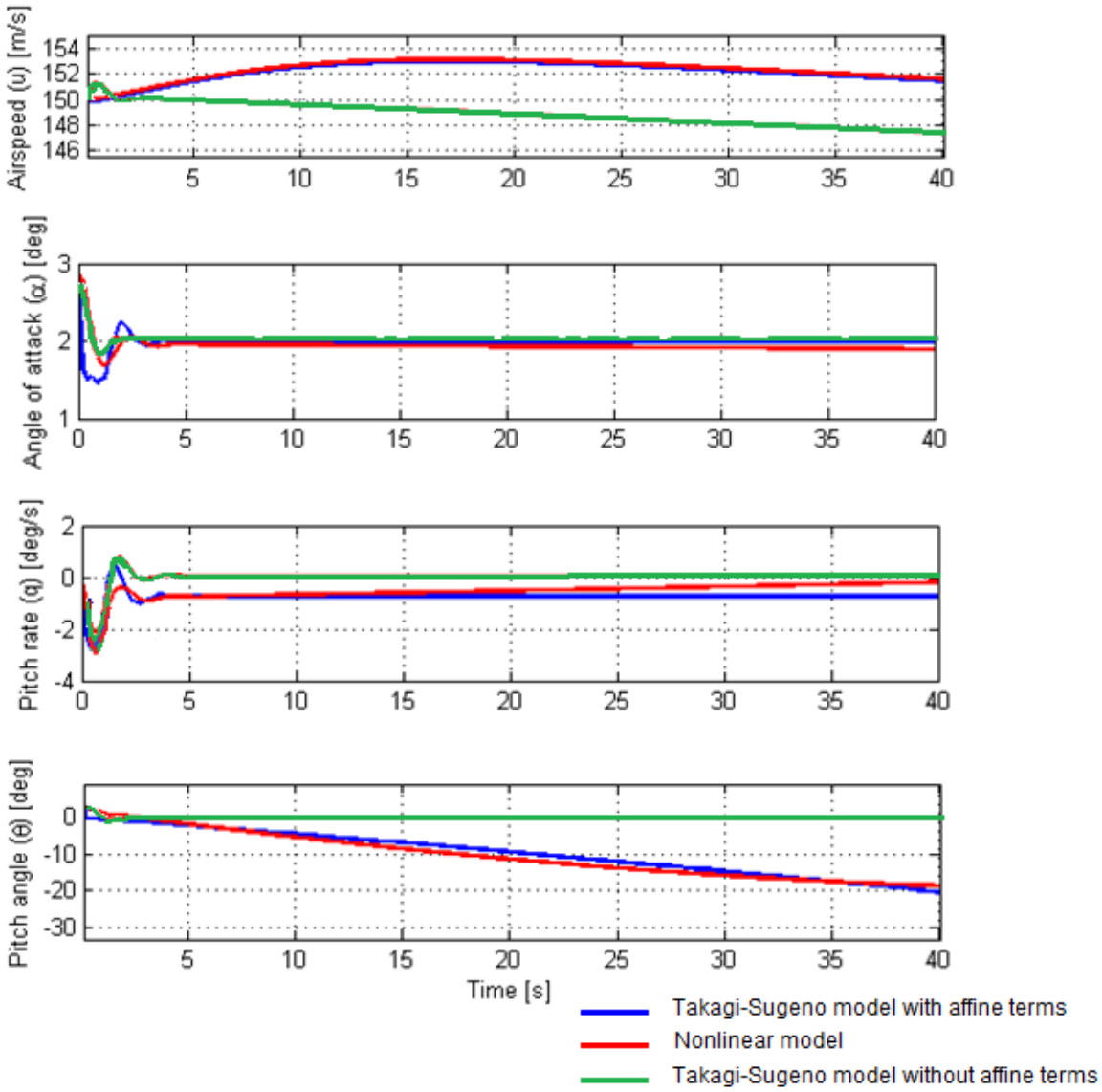


Figure 4-2: Open-loop responses comparing T-S model and Nonlinear model when the elevator deflection is set to 0°

From Figure 4-1 and Figure 4-2, the inaccuracy when the affine terms were not used is clearly seen. The response of T-S model shown in Figure 4-2 was not highly accurate match of the nonlinear model but the curves are close and overlapping and this gave a reasonable approximation of the nonlinear model. The reason for slight inaccuracies was that the nonlinear model provided was not built using the classical flight dynamics equations and some information was missing in order to build an accurate T-S model. However, the responses show that the T-S model was good enough approximation for testing the controls at an early stage project such as in this case.

4.2. PDC Controlled System

The PDC was connected to both T-S model and nonlinear model to visualise the difference in control and stabilising performance of the controller.

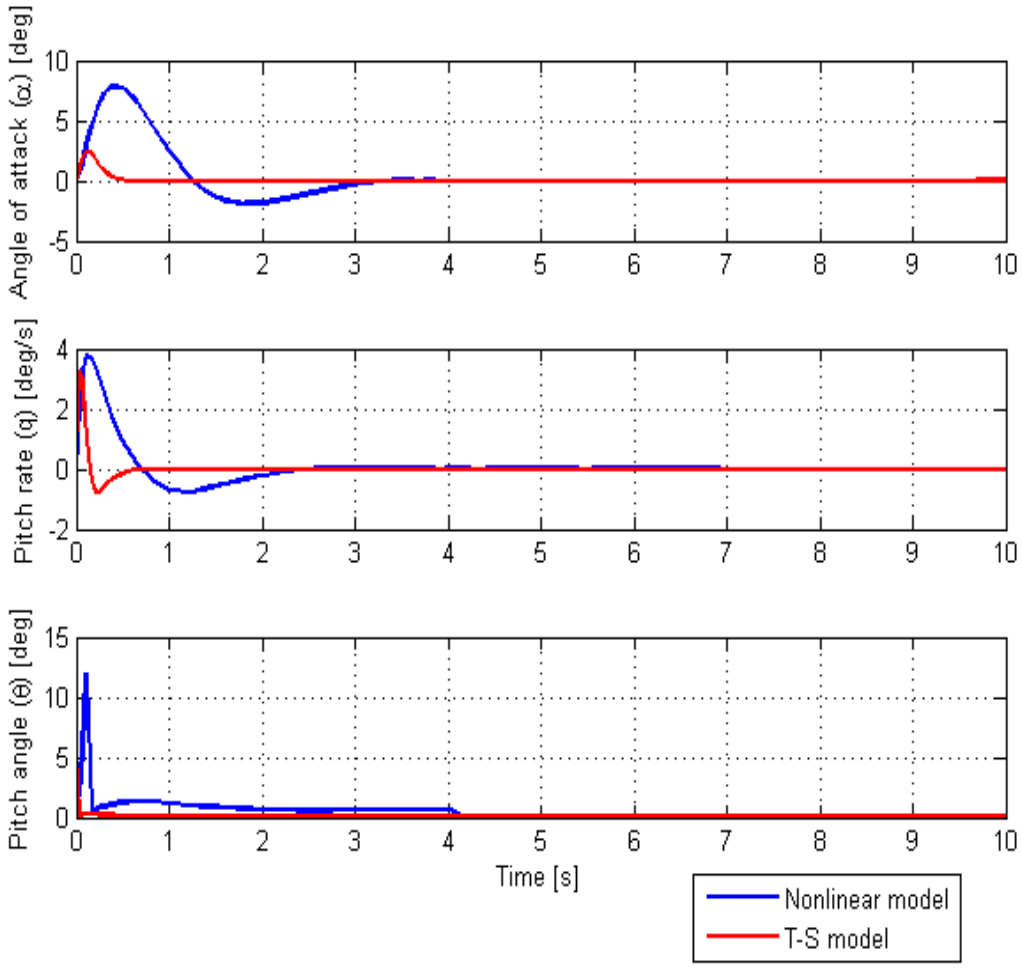


Figure 4-3: Control action stabilising all states

The control system shown in Figure 4-3 stabilised all states to zero. As expected the nonlinear model took longer to stabilise and showed higher overshoot in all states. It was also noted that the overshoots were considerably large and there was no way to reduce this because the PDC guaranteed stability but did not affect the occurring overshoot. The control action required for stabilising the system is shown in Figure 4-4, and it can be concluded that the PDC designed for T-S model was working very good for nonlinear model as well. The control action required to stabilise the states were well within the elevator deflection (input) range. The first state horizontal velocity (v_x) was not controllable with PDC therefore it is not presented in the plots. In real aircrafts, a separate velocity stabiliser is used to control v_x.

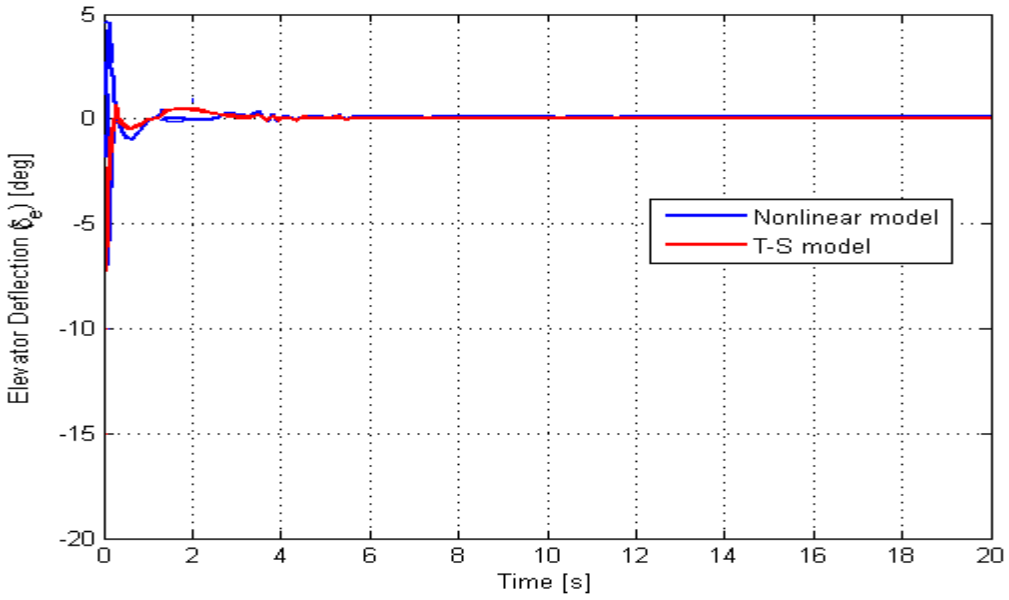


Figure 4-4: Elevator control action for stabilising all states

4.3. Comparison with Mamdani and PI Control

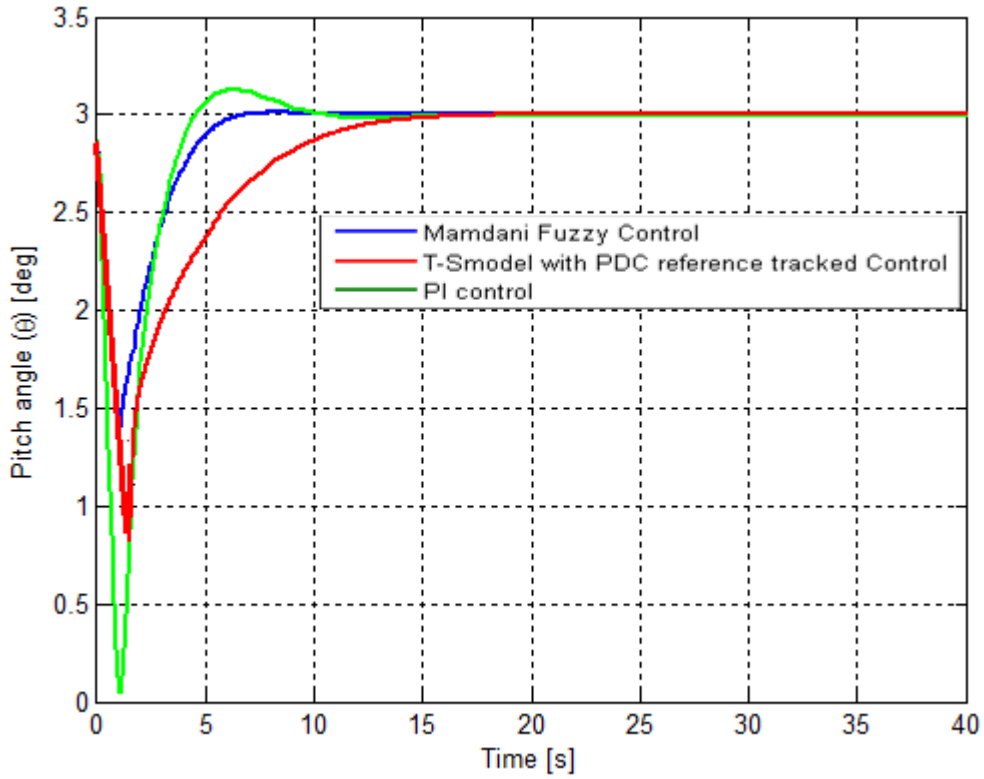


Figure 4-5: Pitch response with reference input of 0°

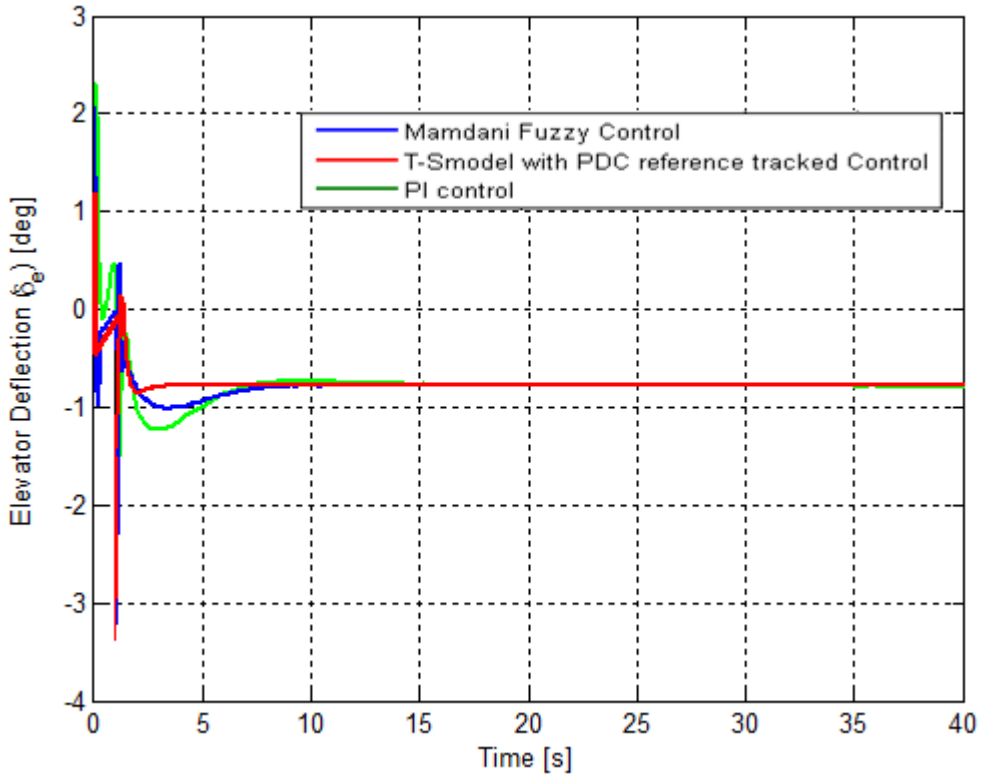


Figure 4-6: Elevator action for stabilising θ to reference angle of 0°

From Figure 4-5 and Figure 4-6, it was seen that PDC controller produced slightly larger overshoot and settling time compared to Mamdani PI controller. Considering, the guaranteed stability of PDC, it would still be more efficient to implement PDC rather than PI controllers which has largest overshoot followed by few low amplitude harmonic motions between the three controllers.

However, the control action with PDC was similar to that of Mamdani PI controller and PI controller with lower amplitude oscillations which in fact makes PDC more efficient in terms of power consumption to perform control action.

5. Conclusions & Future Work

The Takagi-Sugeno fuzzy model was successfully built which demonstrated reasonable accuracy to the nonlinear model. The slight inaccuracies were due to incomplete information about the nonlinear model. A Parallel Distributed Compensator was successfully designed for the T-S model which also works agreeably for the nonlinear model. Since an autopilot control must have reference tracking, the Parallel Distributed Compensator with reference tracking was designed which actually improves the performance of PDC compared to just stabilizing PDC.

The T-S fuzzy control had better efficiency compared to Mamdani PI controller and conventional PI controller in terms of control action. However, the response for pitch angle with PDC was reasonable compared to Mamdani PI controller but the stability is guaranteed only with PDC.

The project was a successful attempt to design a fuzzy control based autopilot system for longitudinal motion. The possible future work on this topic could be to develop a fuzzy control system which could perform a manoeuvre or perhaps follow a given flight path with navigational systems.

The fuzzy control works very efficiently for nonlinear dynamic systems, and its simple and intuitive which is precisely what is required in the current and future aerospace industry.

References

[1] M. Usta, Ö. Akyazi and Akpinar A, "Aircraft roll control system using LQR and fuzzy logic controller," pp. 223, Nov 2011.

[2] M. Cook, Flight Dynamics Principles, Second ed., Elsevier Ltd., 2007.

[3] Y. Lai and F. Hsiao, "Application of fuzzy logic controller and pseudo-attitude to the autonomous flight of an unmanned aerial vehicle," Journal of the Chinese Institute of Engineers, vol. 33, no. 3, pp. 387, 2010.

[4] S. Kurnaz, Cetin and O. Kaynak, "Fuzzy logic based approach to design of flight control and navigation tasks for autonomous unmanned aerial vehicles," J Intell Robot Syst, vol. 54, no. 229, 2009.

[5] N. Hassan, M. Rahmat and S. Mansor, "Application of intelligent controller in feedback control loop for aircraft pitch control," Australian Journal of Basic and Applied Sciences, vol. 5, no. 12, pp. 1065, 2011.

[6] J. Gomez and M. Jamshidi, "Fuzzy adaptive control for a UAV," J Intell Robot Syst, vol. 62, pp. 271, 2011.

[7] M. Zugaj and J. Narkiewicz, "Autopilot supported by nonlinear model following reconfigurable flight control system," Journal of Aerospace Engineers, vol. 23, no. 4, pp. 339, Oct 2010.

[8] B. Erginer and E. Altug, "Design and implementation of a hybrid fuzzy logic controller for a quadrotor VTOL vehicle," International Journal of Control, Automation, and Systems, vol. 10, no. 1, pp. 61, Oct 2012.

[9] B. Stevens and F. Lewis, Aircraft Control and Simulation, John Wiley & Sons, INC, 2004.

[10] H. Ying, Fuzzy Control and Modelling: Analytical Foundations and Applications, Institute of Electrical and Electronics Engineers, Inc., 2000.

[11] G. Chen and T. Pham, Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems, CRC Press LLC, 2001.

[12] L. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis, PTR Prentice Hall, 1994.

[13] T. Johansen, "On the interpretation and identification of dynamic takagi-sugeno fuzzy models," IEEE Transactions on Fuzzy Systems, vol. 8, no. 3, pp. 297, Jun 2000.

[14] H. Wang, K. Tanaka and M. Griffin, "Parallel distributed compensation of nonlinear systems by takagi-sugeno fuzzy model," IEEE, vol. 95, no. 7, pp. 531 1995.

[15] E. Butler, H. Wang and J. Burken, "Takagi-sugeno fuzzy model-based flight control and failure stabilization," Journal of Guidance, Control and Dynamics, vol. 34, no. 5, pp. 1543 October 2011.

[16] H. Wang and K. Tanaka, "A LMI-based stable fuzzy control of nonlinear systems and its application to control of chaos," IEEE, vol. 96, no. 3, pp. 1433 1996.

[17] H. Wang, "An approach to fuzzy control of nonlinear systems: Stability and design issues," IEEE Transactions on Fuzzy Systems, vol. 4, no. 1, pp. 14 1996.

[18] K. Tang, "An optimal fuzzy PID controller," IEEE Transactions on Industrial Electronics, vol. 48, no. 4, pp. 757 2001.

[19] H. Wang and K. Tanaka, Fuzzy Control Systems: Design and Analysis - A Linear Matrix Inequality Approach, John Wiley & Sons, Inc., 2001.

[20] Z. Li, Fuzzy Chaotic Systems: Modeling, Control and Applications, Springer, 2006.

[21] G. Limnaios and N. Tsourveloudis, "Fuzzy logic controller for a mini coaxial indoor helicopter," J Intell Robot Syst, vol. 65, pp. 187 2012.

[22] T. Agustinah, A. Jazidie, M. Nuh and N. Du, "Fuzzy tracking control design using observer-based stabilizing compensator for nonlinear systems," IEEE, vol. 10, no. 6, pp. 275 2010.

[23] H. Wang and K. Tanaka, Fuzzy Control Systems: Design and Analysis - A Linear Matrix Inequality Approach, John Wiley & Sons, Inc., 2001.

[24] S. Boyd, L. Ghaoui, E. Feron and V. Balakrishnan, Linear Matrix Inequalities in System and Control Theory, Society for Industrial and Applied Mathematics., 1994.

[25] C. Ionescu, Matlab – A Ubiquitous Tool for the Practical Engineer, InTech, 2011.

[26] J. Lilly, Fuzzy Control and Identification, John Wiley & Sons, Inc., 2010.

[27] B. Stevens and F. Lewis, Aircraft Control and Simulation, John Wiley & Sons, INC, 2004.

[28] R. Langton, Stability and Control of Aircraft Systems Introduction to Classical Feedback Control, John Wiley & Sons Ltd, 2006.

[29] J. Franklin, Dynamics, Control and Flying Qualities of V/STOL Aircraft, AIAA Education Series, 2002.

Appendix A: Complete Equations of Motion

The equations of motion for longitudinal dynamics shown in section 3.3 for deriving the T-S model was reduced version as unnecessary terms were omitted. If required to consult complete equations, they are shown below.

$$\dot{v_x} = \frac{1}{m_a}\left(F_t\cos(\alpha)\cos(\beta) + m_a g\begin{pmatrix}\cos(\alpha)\cos(\beta)\sin(\theta) + \sin(\beta)\sin(\phi)\cos(\theta)\\ +\sin(\alpha)\cos(\beta)\cos(\theta)\end{pmatrix} + D_{trim} - X\right) \quad \text{(A-1)}$$

$$\dot{\alpha} = q - \tan(\beta)\left(p\cos(\alpha) + r\sin(\alpha)\right) + \frac{1}{m_a U_o}\left(-Z - L_{trim} - F_t\sin(\alpha) + m_a g\begin{pmatrix}\cos(\alpha)\cos(\theta)\\ +\sin(\alpha)\sin(\theta)\end{pmatrix}\right) \quad \text{(A-2)}$$

$$\dot{q} = \frac{1}{I_y}\left(M - (I_x - I_z) - I_{xz}(p^2 - r^2)\right) \quad \text{(A-3)}$$

$$\dot{\theta} = q\cos(\phi) - r\sin(\phi) \quad \text{(A-4)}$$

where β is sideslip angle (rad) and ϕ is roll angle (rad).

Appendix B: T-S Submodels

$$A_1 = \begin{bmatrix} 0 & 126.599 & 0 & -9.804 \\ 0 & -1.7824 & 1 & 0.0023 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_1 = \begin{bmatrix} -18.828 \\ 0.3776 \\ 1.6706 \\ 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 126.0124 & 0 & -9.4813 \\ 0 & -1.797 & 1 & 0.0168 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_2 = \begin{bmatrix} 9.5625 \\ -0.024556 \\ -0.21524 \\ 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 125.4814 & 0 & -9.0959 \\ 0 & -1.8046 & 1 & 0.0245 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_3 = \begin{bmatrix} 25.194 \\ -0.2484 \\ -1.2585 \\ 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0 & 126.3584 & 0 & -9.5634 \\ 0 & -1.7655 & 1 & -0.0146 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_4 = \begin{bmatrix} -21.356 \\ 0.37599 \\ 1.6706 \\ 0 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0 & 126.3411 & 0 & -9.81 \\ 0 & -1.7802 & 1 & 0 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_5 = \begin{bmatrix} 7.0443 \\ -0.022365 \\ -0.21524 \\ 0 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} 0 & 126.1198 & 0 & -9.7343 \\ 0 & -1.7882 & 1 & 0.0081 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_6 = \begin{bmatrix} 22.736 \\ -0.24414 \\ -1.2585 \\ 0 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} 0 & 126.0136 & 0 & -9.2186 \\ 0 & -1.7577 & 1 & -0.0224 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_7 = \begin{bmatrix} -22.844 \\ 0.37288 \\ 1.6706 \\ 0 \end{bmatrix}$$

$$A_8 = \begin{bmatrix} 0 & 126.2654 & 0 & -9.7343 \\ 0 & -1.7721 & 1 & -0.0081 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_8 = \begin{bmatrix} 5.4892 \\ -0.023192 \\ -0.21524 \\ 0 \end{bmatrix}$$

$$A_9 = \begin{bmatrix} 0 & 126.1955 & 0 & -9.81 \\ 0 & -1.7801 & 1 & 0 \\ 0 & -8.3928 & -0.6754 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D_9 = \begin{bmatrix} 21.178 \\ -0.24638 \\ -1..2585 \\ 0 \end{bmatrix}$$

Appendix C: PDC Control Gains

The control gains calculated for T-S submodels using convex LMI programming.

$$K_1 = [1.541 \quad 190.72 \quad -3.399 \quad -232.35]$$

$$K_2 = [1.5438 \quad 191.13 \quad -3.4044 \quad -232.79]$$

$$K_3 = [1.5459 \quad 191.42 \quad -3.4081 \quad -233.1]$$

$$K_4 = [1.537 \quad 190.16 \quad -3.3918 \quad -231.76]$$

$$K_5 = [1.5388 \quad 190.42 \quad -3.3951 \quad -232.03]$$

$$K_6 = [1.5404 \quad 190.64 \quad -3.3979 \quad -232.27]$$

$$K_7 = [1.5357 \quad 189.96 \quad -3.3893 \quad -231.55]$$

$$K_8 = [1.5369 \quad 190.14 \quad -3.3915 \quad -231.74]$$

$$K_9 = [1.5381 \quad 190.32 \quad -3.3938 \quad -231.93]$$

The control gains calculated for augmented T-S models with reference signals using convex LMI programming.

$$K_1 = [0.91899 \quad 93.884 \quad -2.2616 \quad -125.87 \quad -104.68]$$

$$K_2 = [0.91916 \quad 93.908 \quad -2.262 \quad -125.9 \quad -104.7]$$

$$K_3 = [0.91928 \quad 93.925 \quad -2.2623 \quad -125.91 \quad -104.71]$$

$$K_4 = [0.91881 \quad 93.863 \quad -2.2611 \quad -125.85 \quad -104.66]$$

$$K_5 = [0.9189 \quad 93.876 \quad -2.2614 \quad -125.86 \quad -104.67]$$

$$K_6 = [0.91899 \quad 93.888 \quad -2.2616 \quad -125.87 \quad -104.68]$$

$$K_7 = [0.91876 \quad 93.857 \quad -2.2609 \quad -125.84 \quad -104.65]$$

$$K_8 = [0.91881 \quad 93.865 \quad -2.2611 \quad -125.85 \quad -104.66]$$

$$K_9 = [0.91888 \quad 93.874 \quad -2.2613 \quad -125.86 \quad -104.67]$$

Appendix D: Aerodynamics coefficients & Derivatives

Table 0-1: Presenting the values of the aerodynamic derivatives

X_u	-0.03321
X_α	62.01
X_q	-7.523
$-X_\theta$	-9.789
Z_u	-0.0008684
Z_α	-0.9495
Z_q	0.9823
Z_θ	-0.003265
M_u	-0.001673
M_α	-6.623
M_q	-0.9614
M_θ	0.0007055
$C_{x\alpha}$	-2.001
$C_{x\delta e}$	0.0173
$C_{z\alpha}$	4.5627
$C_{z\delta e}$	0.305
C_{zq}	4.5678
$C_{m\alpha}$	-0.4842
$C_{m\delta e}$	-0.88
C_{mq}	-5.1703

Appendix E: Figures

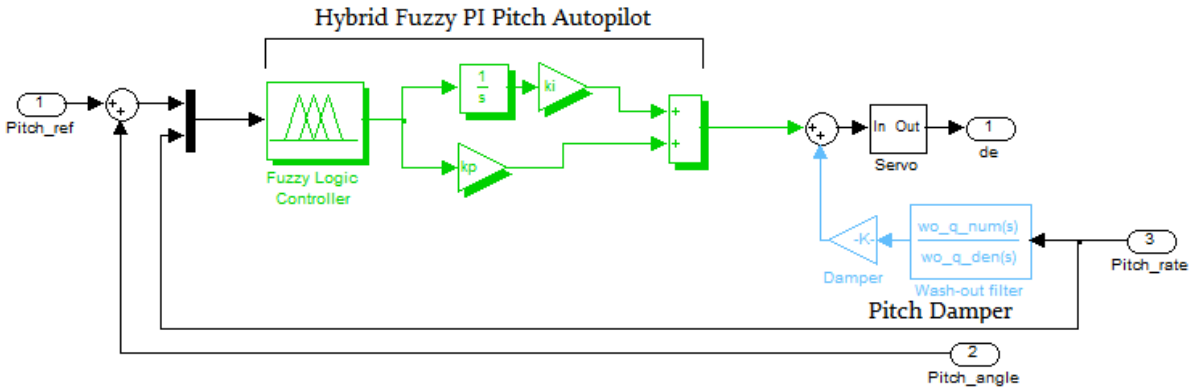


Figure 0-1: Mamdani PI controller

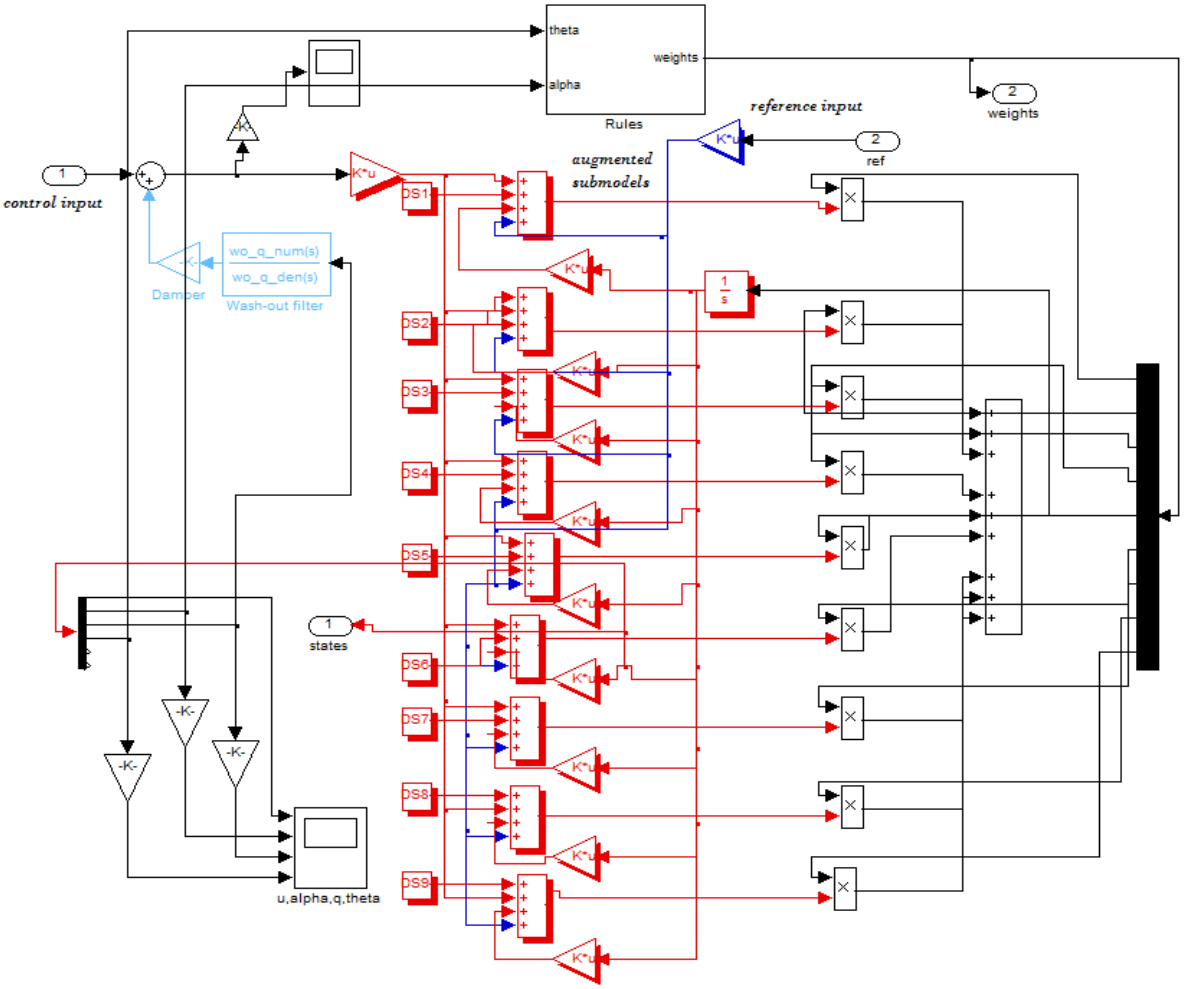


Figure 0-2: T-S fuzzy model with reference tracking

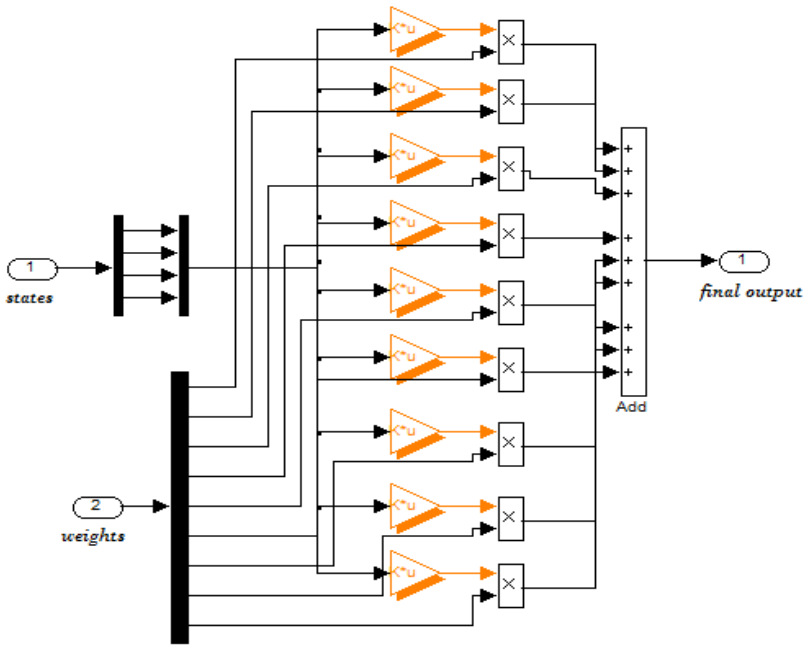


Figure 0-3: Simulink scheme of PDC

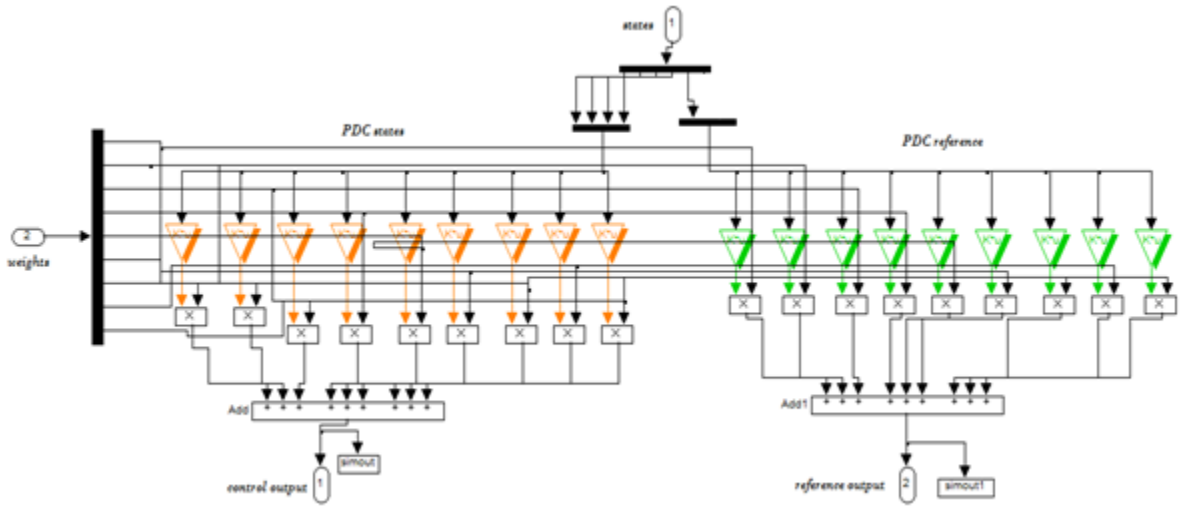


Figure 0-4: Simulink scheme of PDC with reference tracking