Aircraft Longitudinal Automatic Landing Control Using Model Predictive Control

By
Naila Qayyum
NUST201464418MCEME35014F

Supervisor
Dr. Muwahida Liaqat
NUST-CEME

A thesis submitted in partial fulfillment of the requirements for the degree of Masters in Electrical Engineering (MS EE)

In
Department of Electrical Engineering,
College of Electrical and Mechanical Engineering,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(June, 2017)

Aircraft Longitudinal Automatic Landing Control Using Model Predictive Control

Author
Naila Qayyum
Registration Number
NUST201464418MCEME35014F

A thesis submitted in partial fulfillment of the requirements for the degree of MS Electrical Engineering

Thesis Supervisor: Dr. Muwahida Liaqat

Thesis Supervisor's Signature: _____

Department of Electrical Engineering
College of Electrical and Mechanical Engineering
National University of Science and Technology
Islamabad, Pakistan
June, 2017

DECLARATION

"I certify that this research work titled *Aircraft Longitudinal Automatic Landing Control Using Model Predictive Control* is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources has been properly acknowledged/referred".

Signature: _____

Naila Qayyum

NUST201464418MCEME35014f

LANGUAGE CORRECTNESS CERTIFICATE

"This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university".

Signature: _____
Naila Qayyum

Signature: _____
Supervisor: Dr. Muwahida Liaqat

Copyright Statement

Acknowledgments

First and foremost, I am immensely thankful to Almighty Allah for letting me pursue and fulfill my dreams. Nothing could have been possible without His blessings.

I would like to pay gratitude to my parents for their support throughout my educational career. They have always encouraged me to do my best in all matters of my life. I would like to dedicate this work to them.

I thank my supervisor Dr. Muwahida Liaqat who was a great source of help in my research. She constantly encouraged me and gave me strength. Finally, my Co-advisor, Dr. Aamer Iqbal Bhatti who has been a great source of inspiration for me during these years of research. His expert guidance have made this thesis possible. Despite all the assistance provided by Dr. Aamer Iqbal and others, I alone remain responsible for any errors or omissions which may unwittingly remain.

Dedicated to my supportive parents and siblings whose tremendous cooperation led me to this accomplishment

Abstract

In this report, continuous model predictive control using orthonormal functions has been used to control landing trajectory of the aircraft. The purpose of this thesis was to implement constraint control on unmanned aerial vehicle and get improved results as compared to PID technique.

For the simulations in MATLAB, longitudinal state space model of reliance 0.46 trainer aircraft has been used because of its stable natural dynamics. The model has two inputs four states and four outputs. The model is linearized at specific trim conditions.

Model predictive control(MPC) is used for designing controller required in landing for unmanned aerial vehicle. MPC can be simulated in real time and online optimization at every step can be performed. State space model is used and selected outputs are tracked. Landing control requires more than one state to be controlled at the same time so MPC proved to be a good candidate for designing such multi input multi output(MIMO)system.

In real life applications, constraints are always present in all dynamical systems. MPC gives the possibility of adding constraints on inputs, input derivatives, states and output of a system. This gives MPC a priority when designing constraints systems.

Simulation done in MATLAB shows good performance of landing phase of the flight by simply tuning the parameters of the laguerre functions.Also controller handled constraint very well while slightly compromising on the output. But over all it showed improved performance when compared to PID controllers.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Unmanned aerial vehicles are currently one of the main research fields owing to their high demand in military operations. Research is focused on producing highly efficient UAV's for the complicated and dangerous mission where human pilot's physical presence is avoided. Besides military operations, UAV also has immense applications in surveillance missions and air traffic control.

In early days, UAV's were piloted by the operators from the ground. But, now due to advancement in research, operators set certain parameters of UAV's (like speed , altitude etc) and an autonomously operated UAV can now reach itself to its target point. This research has made UAV, a monitored aircraft from constantly controlled aircraft.

1.2 Focus of this Research

In order to make further research in making UAV fully autonomous, landing and takeoff system must also be fully automated. There is already a lot of research published in the field of Automatic landing and takeoff(ATOL). This thesis has the objective to formulate and achieve more accurate and efficient landing system of fixed wing aircraft (A/C) capable of performing landing maneuvers efficiently by keeping control inputs within their physical limitations.

1.3 Literature Review

Landing is one of the most dangerous flight control phase. There are various techniques for landing depending upon specific military missions and A/C size and geometry. Wheeled landing is one of them and proves to be most reliable technique for landing. Work has been done in this field to improve autonomous wheeled landing. In [1],and [3],basic groundwork has been done in landing and take off of fixed wing A/C. The controllers required for landing and takeoff maneuvers are presented in [1].

Both longitudinal and lateral motion of UAV are being considered and researched to gain autonomous control over these phases. Landing controller are considered under guidance controllers where landing trajectory is first designed and relevant command is given to controller for following the trajectory accurately. Way point navigation is one of the method for getting the reference trajectory for landing. Similar work related to path planning are done in [4], [5], [6] and [7].

In [8], various control algorithms for autonomous landing are designed for landing maneuvers in severe weather conditions. It is shown through simulations that UAV followed all longitudinal and lateral position commands and was able to stabilize the system when subjected to changing weather conditions. A backstepping design is developed in [9] for tracking angle of glide slope. This techniques is much simpler and can effectively control the glide path of the aircraft.

Total energy control and cross track control system is being utilized in [10] for accurate landing of UAV in conditions involving crosswinds. In [11], adaptive backstepping technique with PI controller is utilized to generate commands for glide slope and flare. A nonlinear control is designed in [12] using the dynamic inversion approach for automatic landing of unmanned aerial vehicles (UAVs). It is ensured that the sink rate at touchdown remains within specified bounds.

1.3.1 Model Predictive Control

Model Predictive control, which is a costly technique computationally wise is in use these days due to recent advancements in research. In [13], restrictions imposed on control efforts have been considered by implementing MPC over internal loop PID autopilot. Linear quadratic tracker with integral(LQTI) and MPC is utilized in [14] for landing a UAV on moving platform. This technique is useful for finding the control inputs for landing by considering the motion of UAV and moving platform. Non linear model predictive control algorithm is used in [15] for landing of UAV on small space having large angle of attack with less speed which is referred as deep stall landing. 3D prediction model with two inputs is used in [16] for Non Linear Model Predictive Control (NMPC) to predict future states to reach to the commanded height and state. Field programmable gate arrays (FPGAs) are utilized to compute states for NMPC in parallel to save computational cost.

In [17] PID control, LQT, SMC control laws are applied to linearized mathematical model of small fixed wing aircraft with 2 different trim airspeed. [18] employs hierarchical fuzzy control to keep the states at landing within the limits required for safe landing.

Model Predictive control is the most successful technique and is utilized in various applications due to its ability to handle complex systems with hard constraints on states and inputs. In this thesis landing controllers are deigned using this technique. In MPC, current state of the plant is used online to solve a finite horizon problem. the solution will give the optimum control sequence. First control in the sequence is applied to the plant and output is obtained. Main difference of this technique with other technique is that it computes the control law online at each step as opposed to use of already controlled law in other techniques.

The other significance of this technique is its unique capability n handling hard constraints on states and input. In every system there is a possibility of constraints related to actuators and states. Also there is a shortage of techniques which can handle constraints.

In this thesis, continuous model predictive control using orthonor-

mal basis functions is utilized. The control sequence of the plant required in MPC is estimated using set of laguerre functions. The response of the closed loop system depends upon the choice of scaling parameters. These scaling parameter tune the system and depict the accuracy of the results.The state space form of the system is used in MPC. A velocity form model of state space is used. This leads to a embedding of integrator into plant and MPC can be easily implemented on the plant. Also hard constrains can easily be implemented in this velocity form model.

Almost all application in process industry has constraints on its states and control variables. MPC has this most important feature of incorporating constraints on-line at each step of the process without violating their limits. Quadratic cost function in MPC can be solved using quadratic programming methods. Similar work is done in [19],[20],[21],[22]. This thesis uses the velocity form model in MPC algorithm. It uses state space model so that it can handle multivariable systems. Papers related to state space from of the MPC are [23], [24].

1.3.2 Quadratic Programming

Quadratic programming method is one of the most important topic in general life sciences.Linear programming which was used to optimize linear functions was the first programming method and it was used to handle linear constraints in 1940's [25]. Nowadays, Quadratic programming is in use for solving constrained problems. Quadratic programming has been studied by researchers and work has been done [26],[27],[28],[29],[30],[31],[32].There are several methods for solving this quadratic problem. Some of them are primal dual interior point algorithm [33][34],[31], active set method [28], Shor's r-Algorithm[35] and Hildreth programming[36],[20]. This thesis will use Hildreth quadratic programming procedure in implementing constraints in MPC

1.4 Unmanned aerial Vehicle

Unmanned aerial vehicles have been widely used and its predicted that in near future all air space would be flooded with UAV's performing different assigned task of military or commercial use. Due to non availability of pilot in UAV's it is very necessary to have efficient autopilot control system so that in case of any disturbances the aircraft can perform well within limits. The aim of this thesis is to design landing control system for UAV's for performing smooth landing maneuvers.

1.4.1 Aircraft and model

In this section, aircraft dynamic model will be discussed. Reliance 0.46 Size RC Trainer Aircraft model taken from [1] is used in this thesis as a suitable airframe because of its stable natural dynamics for carrying out landing operations.

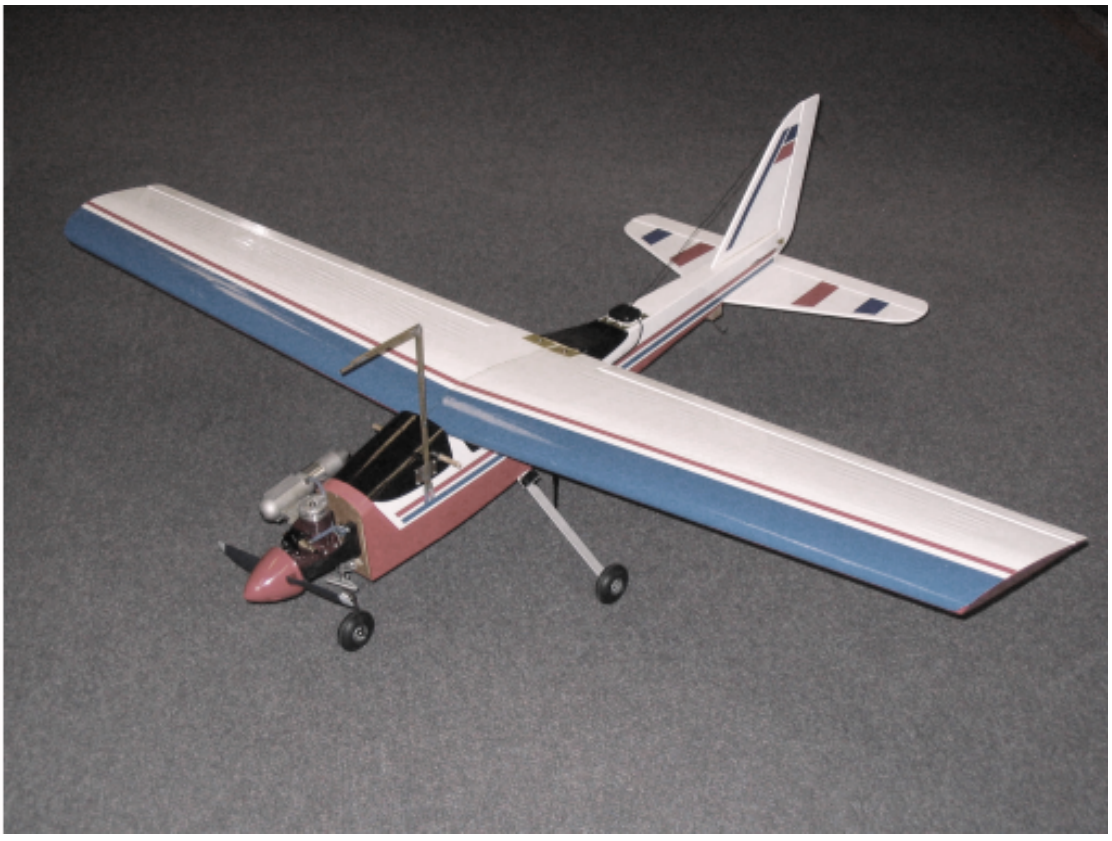


Figure 1.1: Photograph of the Reliance 0.46 Size RC Trainer Aircraft[1]

A 6 DOF model is presented in [1]. In this model, aerodynamic forces and moments(called the airfoils), engine and gravity are the three main sources contributing towards forces and moments acting on the aircraft. Propeller engine contributes towards the engine forces and moments. It's angular momentum is ignored. Constant force per unit mass model act as a model for gravitational forces acting on the aircraft. Aerodynamic forces include lift and drag forces acting on the aircraft.

1.4.2 Axis system

Three reference axes systems are used when modeling the dynamics of the aircraft:

1. The inertial axis system:

   The inertial axis system is a reference frame in which all Newton laws can be applied easily. The origin of the axis is chosen as the aircraft's starting point on a runway for convenience. The X_E axis points in the north direction, the Y_E axis points in the east direction and the Z_E axis points downwards towards the centre of the earth as photographed in [1].

2. The fixed body axis system:

   The origin of the fixed body axis system is the aircraft's centre of gravity. The X_E axis is in the aircraft's plane of symmetry and is taken positive in the forward direction. The Y_E axis lies normal to the plane of symmetry X_EZ_E and points towards the right wing. The Z_E axis is positive in the downward direction.

   The aircraft roll,pitch and yaw about X_E,Y_E and Z_E axis respectively. Conventional Aerodynamic control inputs are defined with respect to this system as:

   (a) Aileron control surface deflection δ_a:
   Aileron Positive deflection creates a negative rolling moment.

   (b) Elevator control surface deflection δ_E:

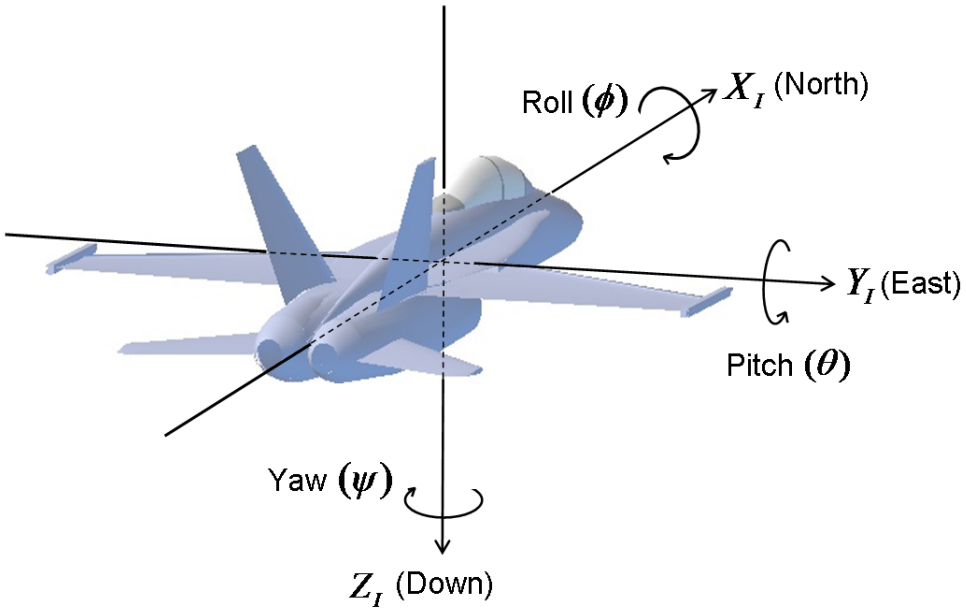


Figure 1.2: Body Axis reference system [2]

Elevator's Positive deflection creates a negative pitching moment.

(c) Rudder control surface deflection δ_R:

Rudder's Positive deflection creates a negative yawing moment.

3. The wind axis system :

The reasons for using the wind-axes system is that aerodynamic forces can be easily expressed in this system. The wind axis shares the same origin with body fixed axis system but its orientation is different. The X_w axis points in the direction of the velocity vector. The Z_w axis points downwards. The Y_w axis points in the direction of the right wing.

The non-linear model given in [1] can be linearized about the trim conditions using small disturbance theory. The system is linearzied so that linear control algorithms can be applied to the model. In small disturbance theory, we assume that aircraft moves with steady flight conditions with very small change in these steady conditions. This theory cannot be applied to systems where there is large change

in steady state motions. The air speed has a huge impact on dynamics of aircraft, the aircraft model given in next section is trimmed and linearized about $20m/s$. The following linearized model will be utilized for applying control algorithms of landing.

1.5 Longitudinal model

The linear longitudinal state space model values of the Reliance 0.46 size trainer aircraft[1] is given below:

$$\dot{x}_l = A_l x_l + B_l u_l \tag{1.5.1}$$

$$\dot{h} = C_{l\dot{h}} x_l \tag{1.5.2}$$

$$\dot{p}_N = C_{lu} x_l \tag{1.5.3}$$

$$\dot{\delta}_t = -\frac{1}{\tau_T}\delta_t + \frac{1}{\tau_T}\delta_{tc} \tag{1.5.4}$$

where

$$A_l = \begin{bmatrix} -0.15 & 0.23 & 0 & -0.17 \\ -0.97 & -12.13 & 3.49 & 0 \\ 0 & -45.56 & -11.18 & 0 \\ 0 & 0 & 10 & 0 \end{bmatrix} \tag{1.5.5}$$

$$x_l = \begin{bmatrix} u & w & q & \theta \end{bmatrix} \tag{1.5.6}$$

$$B_l = \begin{bmatrix} 0 & 1 \\ -0.43 & 0 \\ -24.09 & 0 \\ 0 & 0 \end{bmatrix} \tag{1.5.7}$$

$$u_l = \begin{bmatrix} \delta_e & \delta_t \end{bmatrix} \tag{1.5.8}$$

$$C_{l\dot{h}} = \begin{bmatrix} 0 & -1 & 0 & \frac{20\Pi}{180} \end{bmatrix} \tag{1.5.9}$$

$$C_{lu} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \tag{1.5.10}$$

$$\tau_T = 0.5 \tag{1.5.11}$$

x_l is the vector containing all longitudinal states, where u and w are the axial and vertical velocity components of airspeed expressed in m/s, q is the pitch rate in $degree/sec$, θ and δ_e in degrees, δ_t and δ_{tc} in m/s^2. p_N is the position error along longitudinal axis measured in $metre(m)$, h is the height state in $metres(m)$. τ_T is the engine time

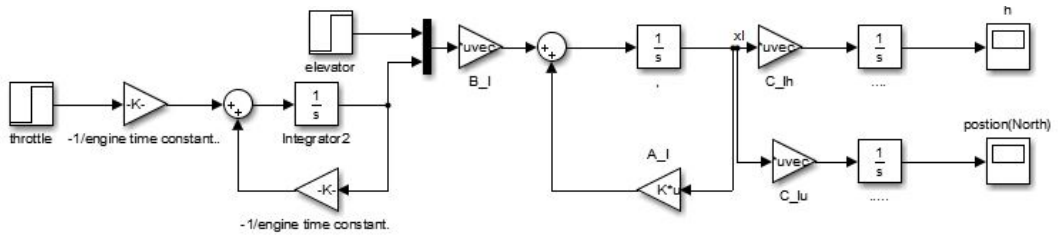


Figure 1.3: Longitudinal Aircraft Model

constant. The linear engine model corresponds to equation (1.5.4). Figure 1.3 shows the block diagram of longitudinal model of the aircraft.

The states are allowed to deflect from the trim conditions within specific limits[1]. Maximum allowed control deflection in δ_e and δ_t is $10deg$ and $5m/s^2$ respectively.

As eigenvalues of the state space system tells us about its dynamic stability modes. There are two longitudinal modes of the aircraft, the phugoid mode and short period mode. Low frequency components corresponds to phugoid mode and high frequency components corresponds to short period mode.

The longitudinal modes are used to describe the motion of the aircraft, when the aircraft is not perturbed about the yaw or roll axis. There are two distinct longitudinal modes of motion.

1.5.1 Phugoid Mode

Phugoid mode is a slow and low frequency. It is a long period mode which involves trade between altitude and airspeed. This mode is lightly damped consisting of many overshoots. In this mode pitch angle and axial velocity varies more than the vertical velocity and pitch rate.

Phugoid mode response of the aircraft under study in this thesis can be found out by setting initial conditions of the state space model equal to real parts of the phugoid mode eigenvector. Real parts of

the eigenvector corresponding to phugoid mode are:

$$\begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} = \begin{bmatrix} -0.0573 \\ 2.189 \times 10^{-3} \\ -6.69 \times 10^{-3} \\ 1 \end{bmatrix} \tag{1.5.12}$$

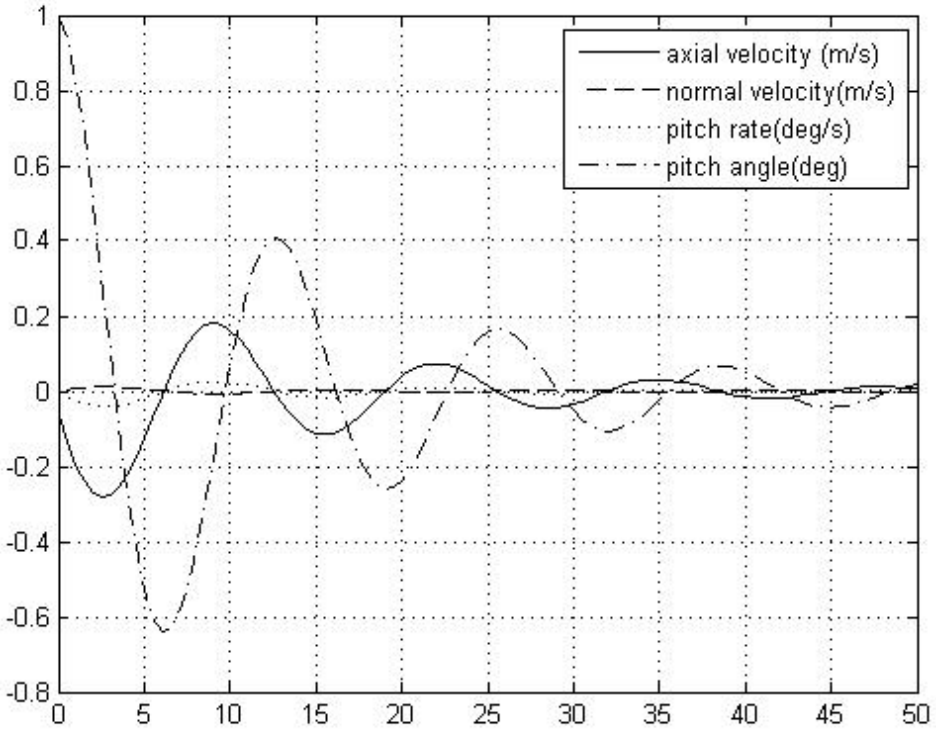


Figure 1.4: Pure Phugoid mode Response

Response of Pure phugoid mode response is shown in Figure1.4

1.5.2 Short Period Mode

Short period mode is fast, having high frequency components and is a stable mode. It is well damped having few overshoots. In this mode, values of pitch rate changes significantly while other values remain considerably constant.Real parts of the eigenvector corresponding to

short period mode are:

$$\begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} = \begin{bmatrix} -2.339 \times 10^{-3} \\ 0.01058 \\ 1 \\ -0.3954 \end{bmatrix} \tag{1.5.13}$$

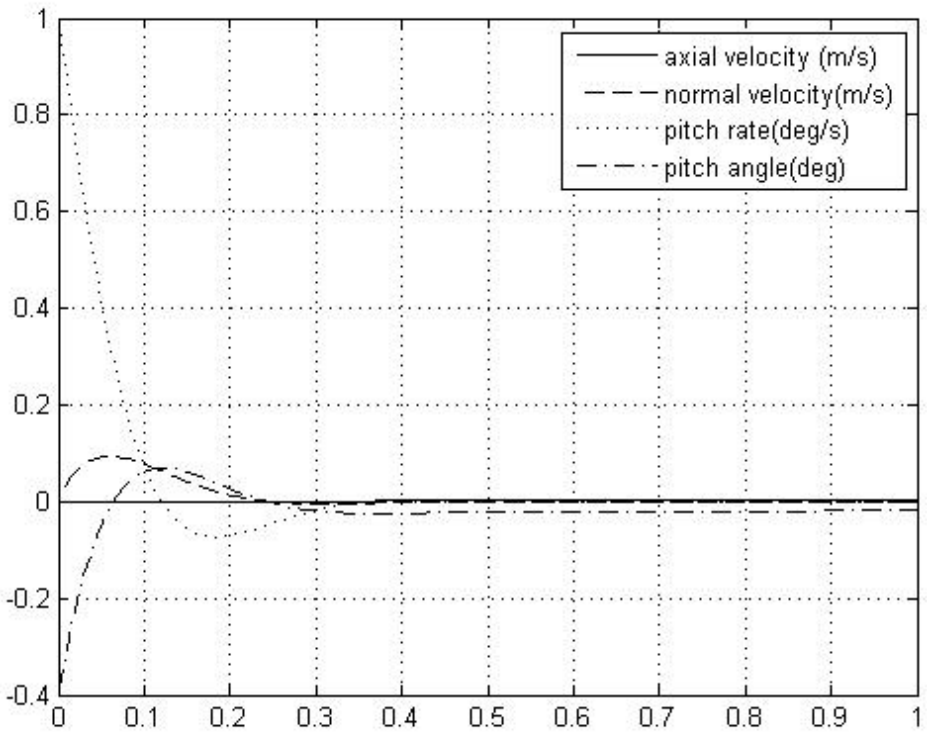


Figure 1.5: Short Period mode Response

Figure 1.5 shows the time response of short period mode when initial conditions corresponding to this mode is put in system's state space vector.

1.6 Lateral Model

Figure 1.6 shows the block diagram of lateral model of the aircraft. The linear Lateral state space model of the Aircraft understudied [1]

is given below again for reference.

$$\dot{x_t} = A_t x_t + B_t u_t \tag{1.6.1}$$
$$\dot{\psi} = C_{tr} x_t \tag{1.6.2}$$
$$\dot{y} = K_t \psi + C_{tv} x_t \tag{1.6.3}$$
$$\tag{1.6.4}$$

where

$$A_t = \begin{bmatrix} -0.57 & 0 & -3.49 & -0.17 \\ -37.22 & -45.15 & 4.72 & 0 \\ 18.81 & -0.36 & -2.52 & 0 \\ 0 & 10 & 0 & 0 \end{bmatrix} \tag{1.6.5}$$
$$x_t = \begin{bmatrix} v & p & r & \phi \end{bmatrix} \tag{1.6.6}$$
$$B_t = \begin{bmatrix} 0 & 0.16 \\ -52.58 & 1.50 \\ 0.53 & -5.23 \\ 0 & 0 \end{bmatrix} \tag{1.6.7}$$
$$u_t = \begin{bmatrix} \delta_a & \delta_r \end{bmatrix} \tag{1.6.8}$$
$$C_{tr} = \begin{bmatrix} 0 & 0 & 10 & 0 \end{bmatrix} \tag{1.6.9}$$
$$C_{tv} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \tag{1.6.10}$$
$$K_{t\psi} = \frac{20\pi}{180} \tag{1.6.11}$$

Cross track error is the shortest distance between present position

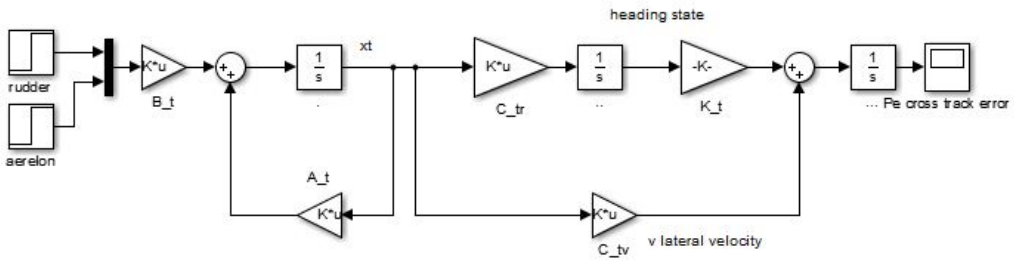


Figure 1.6: Lateral Aircraft Model

of the aircraft and the line joining the origin and the target position. It can be calculated from equation (1.6.4). Unit of Cross track error y is in metres. v is the lateral velocity measured in m/s. r and p are

in $10deg/sec$ while all other states δ_a δ_r ϕ ψ are measured in $degrees$. The maximum δ_a and δ_r deflections are limited to $10deg$. The natural modes of motion corresponding to lateral mode are discussed below:

1.6.1 Roll Mode

The roll mode is generally a non-oscillatory motion consisting of pure rolling motion. There is a differential lift generation when one wing goes down in rolling motion and the other goes up. This lift creates a moment which tries to reinsate the equilibrium. Roll rate value increases when aircraft is disturbed by the rolling moment until the restoring moment balances it.

The following real parts of the roll mode eigenvector when put in the state state of the systems gives the time response of this mode. The value of roll rate is larger because of the phenomenon discussed above. The graph showing roll mode is given in Figure 1.7

$$\begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} = \begin{bmatrix} 1.5 \times 10^{-3} \\ 1 \\ 0.0077 \\ -0.2214 \end{bmatrix} \tag{1.6.12}$$

1.6.2 Spiral Mode

This mode is a slow and unstable mode. If in a level flight, a disturbance occurs and creates a small roll angle. This can move the vehicle downhill as the side slip also increases due to disturbance. If this disturbance is not properly checked, the aircraft experiences yawing moment which would tend to move the aircraft spirally into the ground. A small tail and a dihedral wing is used to reduce the spiral mode affect. The time response of this mode of the aircraft under study is plotted in Figure 1.8.

The real part of the eigenvector corresponding to spiral mode are

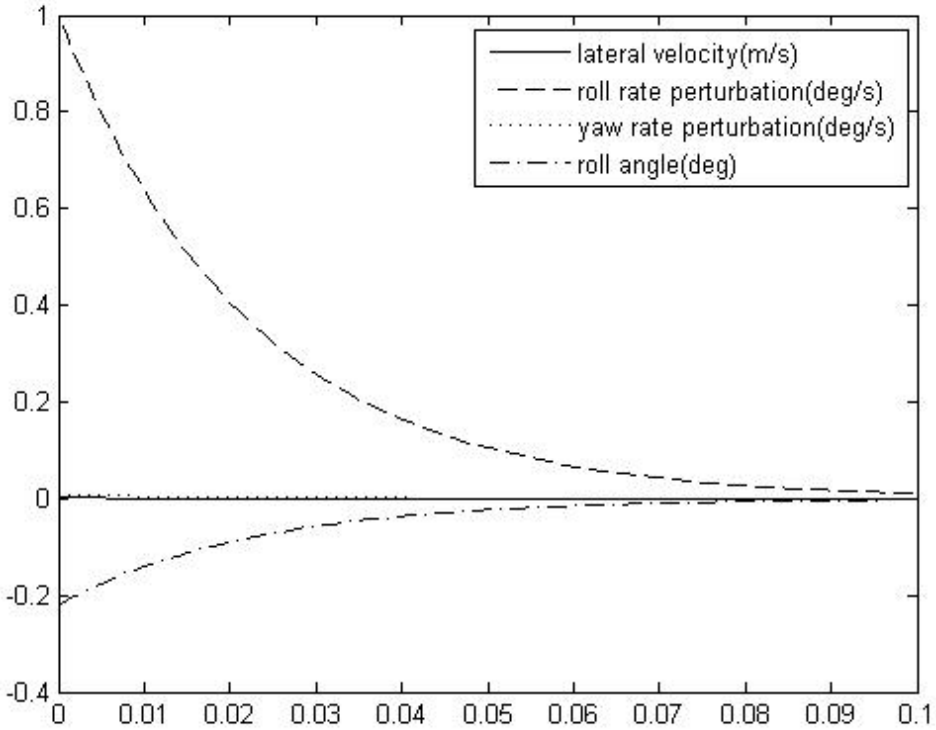


Figure 1.7: Roll mode Response

given below:

$$\begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} = \begin{bmatrix} 0.0064 \\ -3 \times 10^{-4} \\ 0.0480 \\ 1 \end{bmatrix} \tag{1.6.13}$$

1.6.3 Dutch Roll Mode

Yaw rate is mostly affected by this mode, while all other lateral states experience small changes in their values. In this mode, roll rate motion lags the yaw rate motion by 90^o. It is very much similar to longitudinal short period mode. A disturbance oscillations in yaw ψ makes the wings to move back and forth which results in oscillatory differential drag and lift forces. This lift/drag causes aircraft to roll with roll angle (ϕ) that lags yaw angle (psi) by 90^o. the time history response of this mode is shown in Figure 1.9.

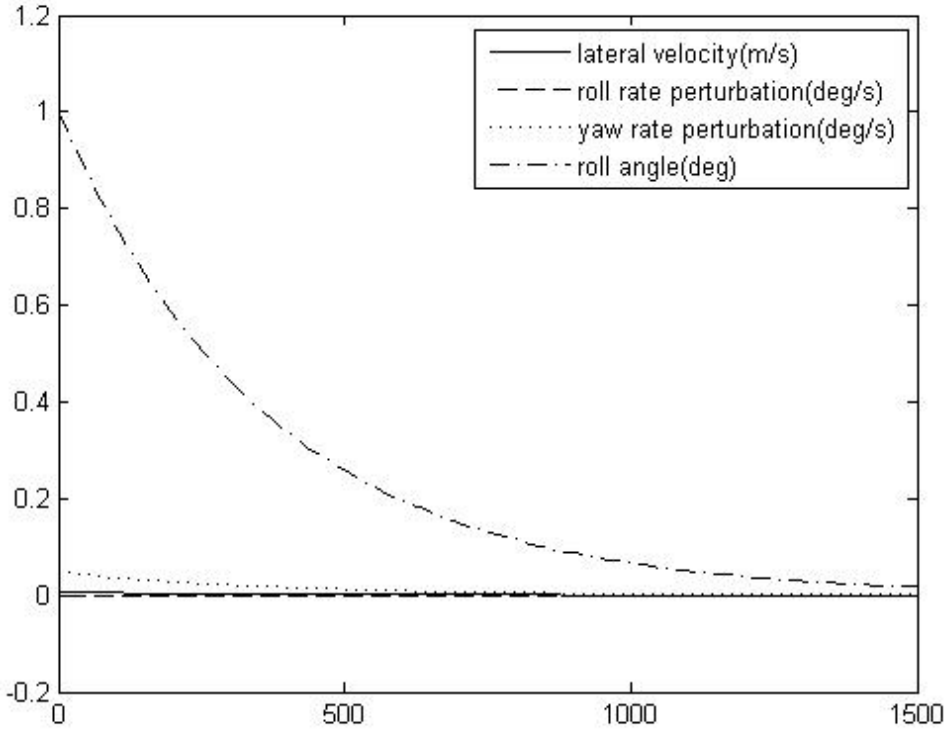


Figure 1.8: Sprial mode Response

The real parts of the eigenvector corresponding to this mode are:

$$\begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} = \begin{bmatrix} 0.052 \\ -4.905 \times 10^{-3} \\ 1 \\ -0.4291 \end{bmatrix} \tag{1.6.14}$$

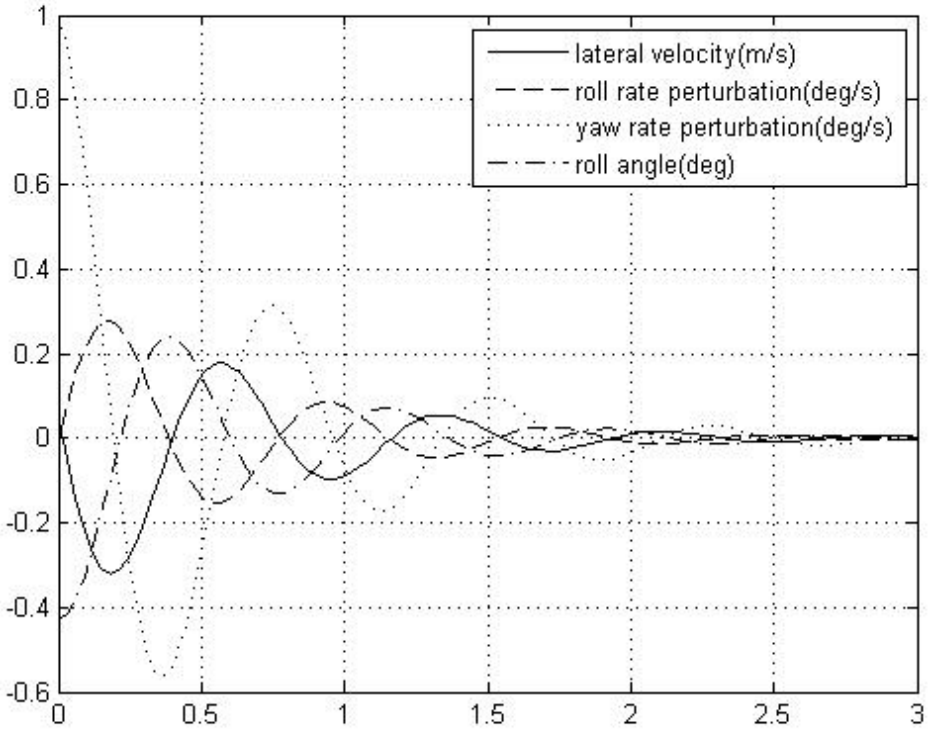


Figure 1.9: Dutch Roll mode Response

1.7 Summary

In this chapter, literature review of the research done in the aircraft Landing phase is discussed. Linear model of the reliance aircraft and its axis systems used is discussed. Various modes of longitudinal and lateral aircraft dynamics have been discussed and their time history response is shown.

Chapter 2

Landing Problem

2.1 Introduction

This chapter discusses landing problem and its different phases. In the thesis, we assume that the aircraft first descends to a certain height. Aircraft then align itself with the runway center-line and point its heading to it. It also decrease its velocity. Then, A/C start its landing maneuver and descends towards the runway with a constant glide path angle and speed. For a smooth landing, It enters a flare phase when it is near to touchdown point. After touching the ground, the A/C follows the centerline of the runway path until its speed reduces to zero and aircraft comes to rest.

2.2 Glide Path Phase

This phase comprise the interval in which A/C descents from its initial height to flare altitude. The glide slope angle (γ) and airspeed (U_o) is kept constant during this phase. The aircraft speed and the flight path angle, γ, determine the trajectory for this phase. For a short landing distance, lower velocity and larger path angle is desired to decrease the landing distance and landing time. This tracking trajectory is based on the steady state constant sink rate which must be maintained during glide phase.

During this phase, the height of the A/C above ground is regulated. The desired height is calculated as a function of forward distance.

The glide slope is a straight line path whose slope is determined by the flight path angle γ.

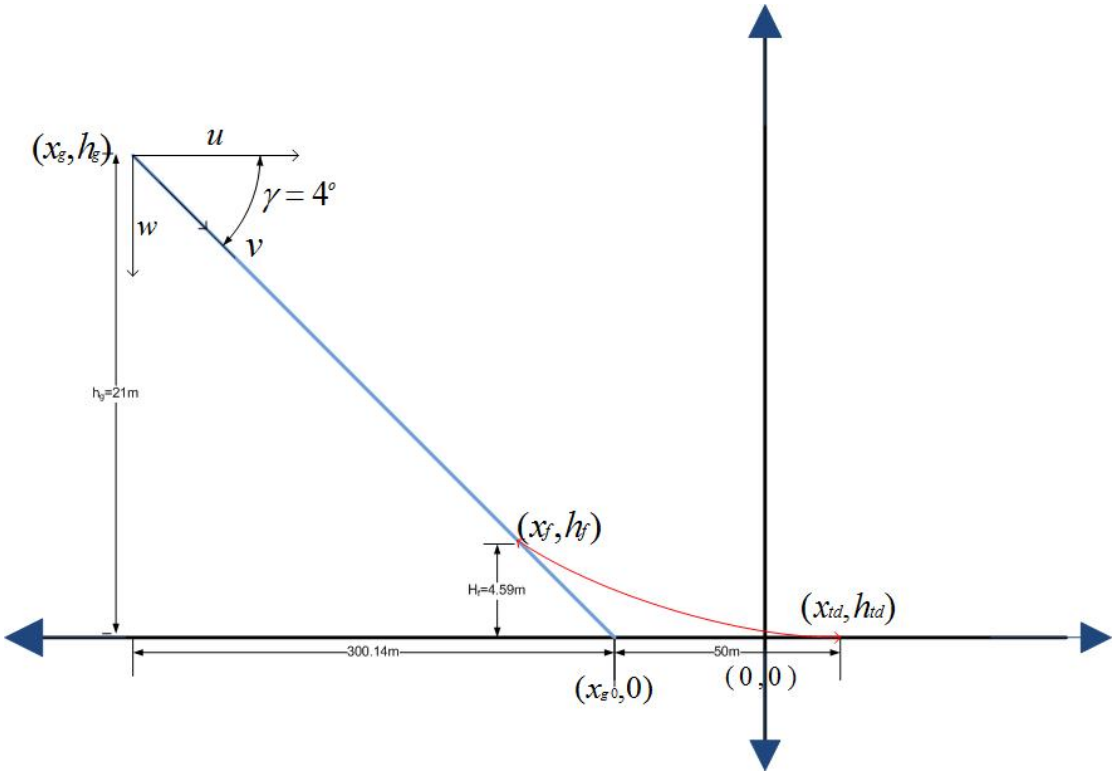


Figure 2.1: Glide path and Flare Generation

For the glide-slope tracking, the aircraft follows a constant vertical speed that represents the sink rate of the aircraft [11]. Glide path graphical representation is given in Figure 2.1. (x_g, h_g) is the start of the glide path. Flare begins at (x_f, h_f). The glide path reference command for the glide-slope tracking is defined as

$$\dot{h}_{ref,GS} = tan^{-1}\left(\frac{h - h_{ref}}{S}\right) U_o k + \frac{-4}{180/\pi} U_o \tag{2.2.1}$$

where S is the horizontal distance and h_{ref} is the desired height of glide slope after which flare starts. h is the starting height of glide slope which is taken to be $21m$. U_o is the constant speed. Speed is taken to be double of stall speed which is $7.5m/s$.The first term in the expression of (2.2.1) is adjusted factor. k is an adjusted factor which adjust time taken for glide maneuver. The simulink block diagram is shown in Figure (2.2).

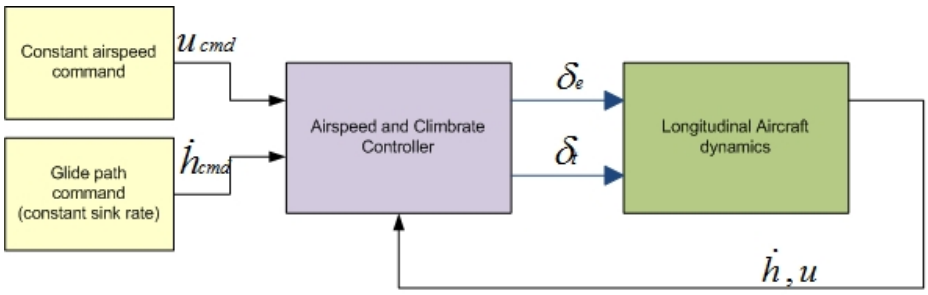


Figure 2.2: Glide slope Controller Block diagram

2.3 Flare Phase

The flare maneuver is commenced after the glide path. The flare phase must start at a correct height to avoid risking damage to the aircraft at the touchdown. That is why it should not be started at much lower height. Also larger flare initiation height results in large horizontal distance towards the touchdown and decrease in airspeed.

During the flare manoeuvre, the descent rate is kept exponential. In this phase, the aircraft follows an exponential path and moves towards the touchdown point in a curved path. The aircraft raises its nose and slides down smoothly while following the curved path. The desired curved path can be found from equation

$$h_{ref}(t) = h_o \, e^{\frac{-t}{\tau}} \tag{2.3.1}$$

h_o is the distance at the start of the flare phase. τ is time constant which defines the time taken to reach the runway. To get the corresponding climb rate command from this equation we can take derivative of equation (2.3.1).

$$\dot{h}_{ref} = -\frac{h_o}{\tau} \, \exp(-t/\tau) \tag{2.3.2}$$

As it is assumed that the airspeed remains same during glide and flare phase so flight path angle during flare can be calculated as

$$\gamma_{ref}(t) = -\frac{h_o}{U_o \tau} \, \exp(-t/\tau) \tag{2.3.3}$$

Initial flare height can be calculated by putting $t = 0$ in equation (2.3.3)

$$h_o = -U_o \tau \gamma(0) \tag{2.3.4}$$

$\gamma(0)$ is the path angle at the start of flare which is equal to glide path angle that is 4 deg because of the steady descent between glide mode and flare mode. From equation (2.3.3) and (2.3.4), γ required at touchdown can be calculated as

$$\gamma_{ref,TD} = \gamma_{ref}(0)\ \exp(-t/\tau) \tag{2.3.5}$$

As the airspeed is constant so we can replace t by $(D - D_o)/U_o$ where D_o is the distance from starting point of flare mode and D is the horizontal distance. by manipulating the terms, we can get expression for time constant τ.

$$\tau = -\frac{D - D_o}{U_o[ln(\gamma_{ref,TD}/\gamma_{ref}(0)]} \tag{2.3.6}$$

$D - D_o$ is the distance between touchdown and flare starting mode. $\gamma_{ref,TD}$ can be calculated by considering climb rate at touchdown to be $0.4572m/s$ by the following formula

$$\gamma_{ref,TD} = \dot{h}_{TD}/U_o \tag{2.3.7}$$

Graphical description of the flare phase is shown in Figure 2.1. The climb rate command obtained for flaring as in (2.3.2) is then fed to climb rate controller to regulate it. Flare controller block diagram is shown in Figure 2.3.

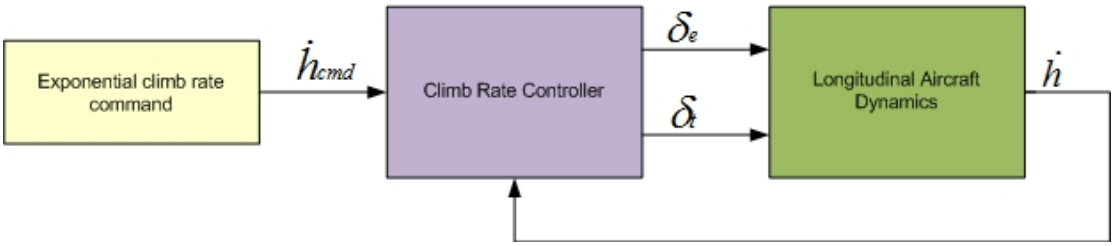


Figure 2.3: Flare Controller Block Diagram

2.4 Taxi Phase

This phase starts just after the touchdown. It is required to keep the plane on the runway in particular direction, so heading controllers are utilized to keep the heading error zero and make the A/C follow the center line of the runway.

2.5 Challenges

There are certain constraints and points which can be summed as challenges which are needed to be followed while landing. This section discusses these challenges faced in landing problem.

The pilot must predict the altitude and distance accurately from which a gradual descent will result in landing at the desired spot on the runway. This is called base leg. The distance will depend on the two factors i.e effect of wind and position of base leg. If a strong wind is present on final approach or the flaps are used to produce a steep angle of descent, the base leg must be positioned closer to the approach end of the runway. The opposite is required to be done if light wind is present.

After moving onto the base leg, the pilot should start the descent with reduced power and airspeed of nearly 1.4 V_{SO} ((V_{SO} is the stalling speed with power off, landing gears and flaps down). An Airplane will stall when critical angle of attack is exceeded. Pilots find it difficult to find angle of attack at flight so airspeed is used to find when stall will occur. The stall speed of aircraft under study is $7.3m/s$. The aircraft landing and takeoff speed is then $15m/s$ for landing.

As a aircraft moves through the air, the wing of the aircraft is inclined at some particular angle with the flight direction. The angle which is between the chord line and the direction of the flight is called the angle of attack and has a large effect on the lift produced by a wing. Hence, these factors affects the speed and it varies. The descent angle is affected by four basic and fundamental forces that acts on an aircraft (lift, drag, thrust, and weight). If all these forces are constant, the descent angle will stay constant in a no-wind condition. The pilot can control and adjust these forces by adjusting the airspeed, attitude, power, and drag.

Many runways or landing paths are such that landings must be executed while the wind is blowing across the landing path rather than parallel to the landing direction. All pilots should be prepared to cope with these situations when they occur.

All these challenges like maintaining speed double than stall speed have been taken in account while designing controllers for glide and flare. Also, Microburst a type of weather condition which occurs in thunderstorms has been considered and controller efficiency is assured by incorporating microburst condition in simulation.

2.6 Summary

This chapter discusses different phases of landing. Glide and flare path trajectories required for designing controllers have been discussed and derived. It also discusses challenges faced while landing. With the aircraft model and trajectories defined, next step is to design controllers for landing.

Chapter 3

PID Controller Design and Simulations

3.1 Introduction

In controls systems, controller is the device which can regulate, alter and maintain the operating dynamics of the system. The linearized state space models about the trim conditions are used to design controllers to control various states of the aircraft. The control consists of two blocks containing longitudinal and lateral Controllers. Longitudinal block uses throttle and elevator while lateral block uses aileron and rudder as control input channels. Longitudinal autopilot regulates the speed and pitch angle of the aircraft, and lateral autopilot regulate the roll and yaw angle of the aircraft. Controller design in this thesis is broken into stability augmentation controllers, attitude regulation controllers and trajectory controllers.

The stability augmentation controllers forms the inner loop and increase the natural damping of the dynamical system by introducing artificial damping into the system. Thus controlling high frequency modes of the aircraft. The attitude controllers regulate the aircraft's pitch and roll angles. They are the outer loop of stability controllers. Attitude controller forms the outer most loop and are responsible for regulating trajectory motion states like altitude and heading.This thesis concerns with the design of longitudinal controllers allowing to regulate altitude and pitch states of the aircraft. The longitudinal controller designed in [1],and [3] are reproduced for comparison with

MPC and are discussed in subsequent sections.

3.2 Aircraft Longitudinal Controllers

Speed and altitude controllers come under the category of longitudinal controllers These states are very dependent and correlated with one another. In landing it is required for speed and altitude to be in specific states at a certain time, so airspeed and climb rate controller designed in this section process both correlated states (u, h) at the same time.

3.2.1 Pitch Rate Damper

Artificial damping in aircraft is required for rejecting wind gust disturbances. So pitch rate damper is designed to give artificial damping to the system. Pitch rate damper is designed in [3]. Its simulink block diagram is shown in Figure 3.1 where q is pitch rate and is fed back to the controller. The value K_q chosen for artificial damping in

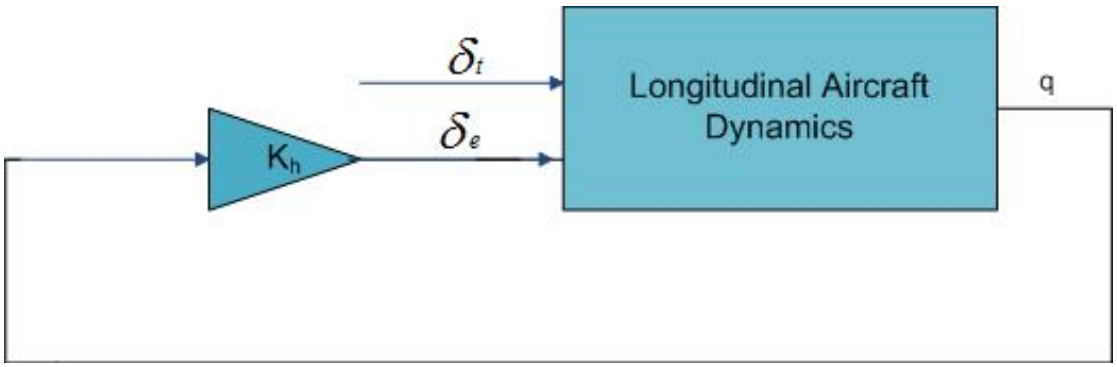


Figure 3.1: Pitch Rate Damper

landing controller is -0.06. The closed loop system equation is as follows

$$\dot{x} = [A - B[K_q\ 0]^T]x + Bu \tag{3.2.1}$$

3.2.2 Climb Rate Controller

Climb rate Controller designed in [3] is used in flaring phase of landing to control climb rate. Decreasing exponential climb rate is re-

quired in flaring,which must be controlled for smooth landing. The designed controller showed slow time response.

The Simulink block diagram of the above mentioned controller is shown in Figure 3.2 where plant dynamics are taken from equation (3.2.1). C is a matrix extracting climb rate state ($\dot{h}$) from output state vector in m/s. The trim velocity at which the model is being linearized is $14m/s$. Feedback gain K_p used is -0.9 . A integral value of -0.3 is used in the model to decrease overshoot.

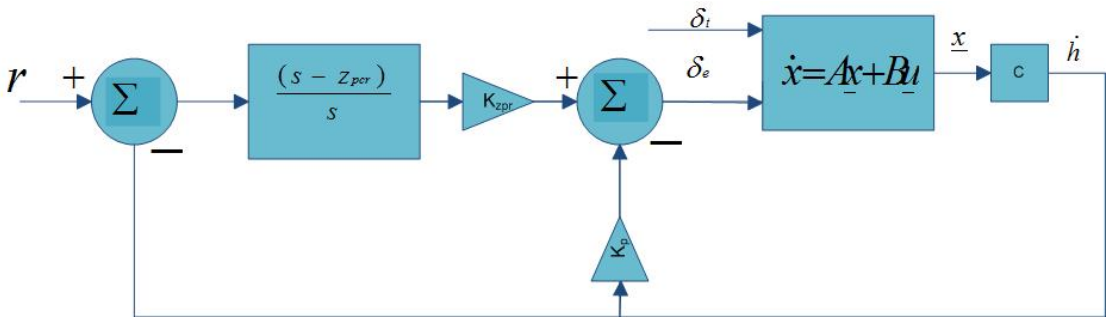


Figure 3.2: Climb Rate Controller

The designed controller reaches the final state value very slowly. In flaring we require fast response as it consists of only few seconds so the designed controller must be very efficient in regulating the states a early as possible which is not the case when flare controller is designed using PID. That's why this controller was not utilized by [3]. Also sink rate of less than 0.5 m/s should be maintained so that flare occurs smoothly. The time to reach this sink rate using PID controller is very slow. The time history response of the climb rate controller to the reference command of $1m/s$ is shown in Figure 3.3. The controller is following the commanded value in $10sec$ with some set point error.

3.2.3 Airspeed and Climb rate Controller

The Airspeed and Climb rate controller is designed in [1]. The Simulink block diagram of the airspeed and climb rate controller is shown in Figure 3.5. It is a Multi-Input-Multi-Output (MIMO) controller which is regulating both airspeed and climb rate simultaneously. The values K_p and K_i of the controller is designed by

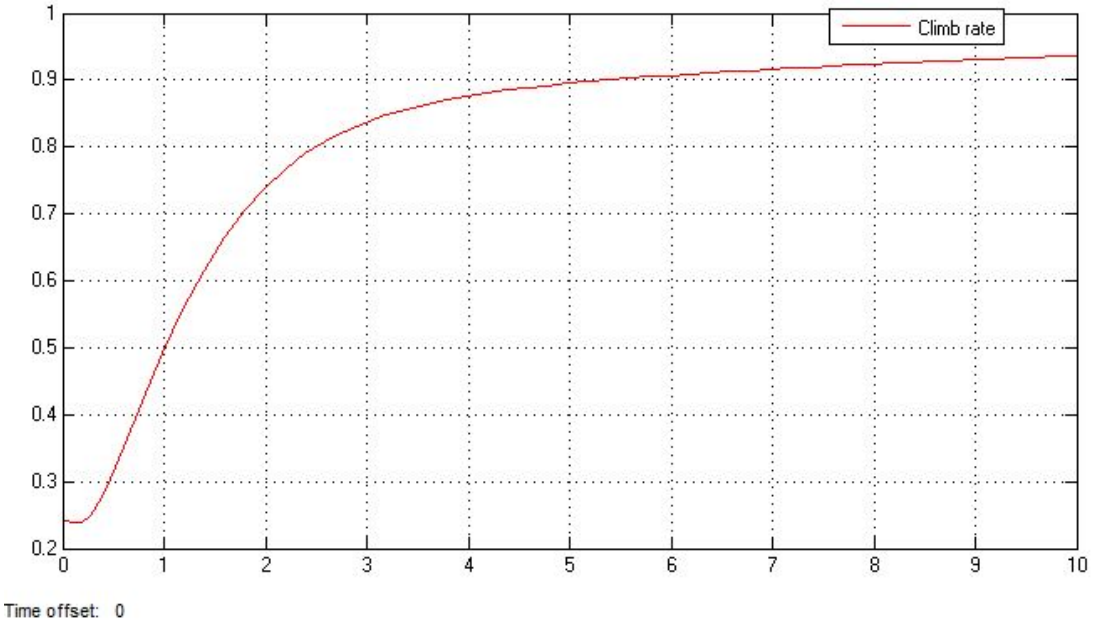


Figure 3.3: Climb Rate Controller Simulation

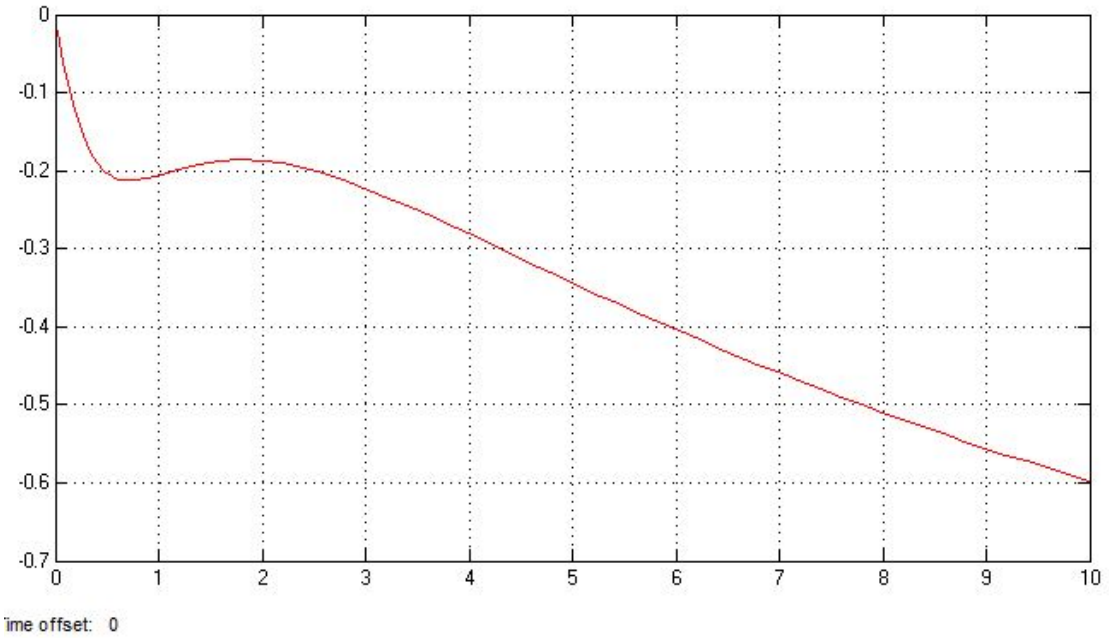


Figure 3.4: Climb Rate Controller Elevator Response

iterative procedure. In this process value of k_p and k_i are changed and corresponding change in response and root loci graph are noticed. The values where satisfactory response and pole locations are obtained are chosen for the controller. The controller is designed for various trim speeds. The values of K_p and K_i for various trim speed

are given in [1]. In landing, we will be using trim speed of $14m/s$. The values of K_p and K_i corresponding to this trim speed are:

$$K_p = \begin{bmatrix} 0.08 & -0.98 \\ 0.88 & 0.18 \end{bmatrix} \qquad K_i = \begin{bmatrix} 0.24 & -0.18 \\ 0.14 & 0.64 \end{bmatrix} \tag{3.2.2}$$

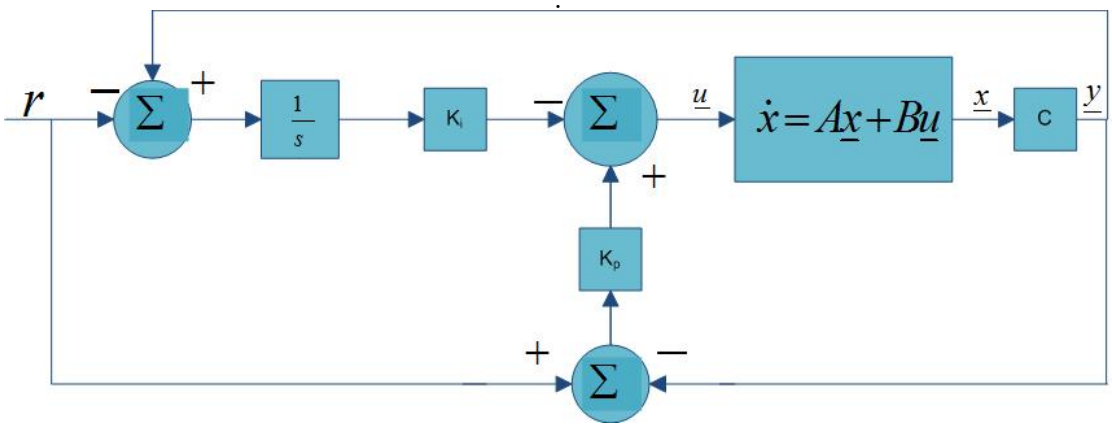


Figure 3.5: Airspeed and Climb rate Controller Simulink diagram

The closed loop equation of the controller is:

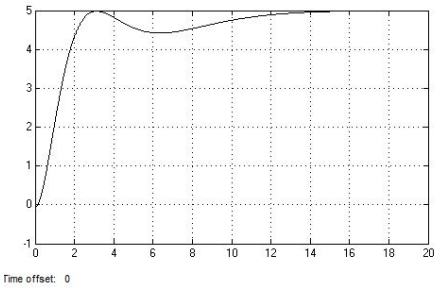
$$\begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A - K_P C & -BK_I \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix} + \begin{bmatrix} BK_P \\ -I \end{bmatrix} r \tag{3.2.3}$$

where A,B,C matrices are longitudinal state space equation taken from [1]. The response of air speed and climb rate state to $5m/s$ airspeed command is shown in Figure 3.6. similarly , Figure 3.7 shows response of both these states to $2m/s$ climb rate command.

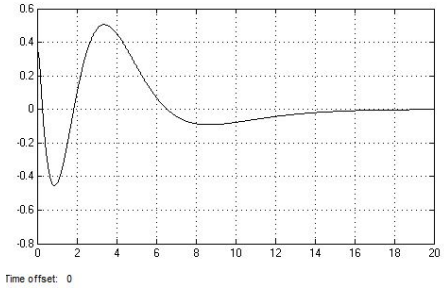
3.2.4 Altitude Controller

Altitude controller designed in [1] come under trajectory controllers. In this climb rate command is generated by the controller which is fed to the system and system achieves the desired height. The saturation block is used to limit the climb rate input signal in excess of $\pm 2m/s$.

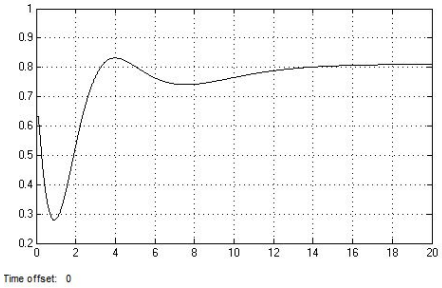
The Simulink block diagram of altitude controller is shown in Figure 3.8. The plant model used is from equation (3.2.3). Output matrix C extracts climb rate from the output. The value of K_h is taken to



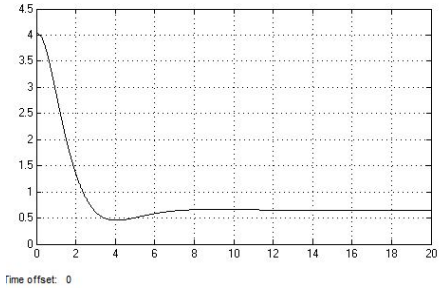
(a) Airspeed Response



(b) Climbrate Response

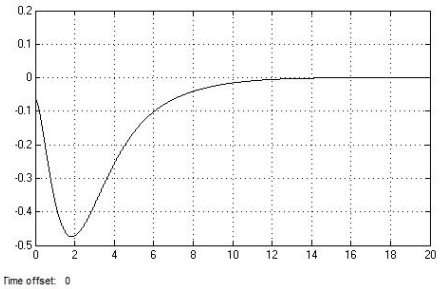


(c) Elevator Response

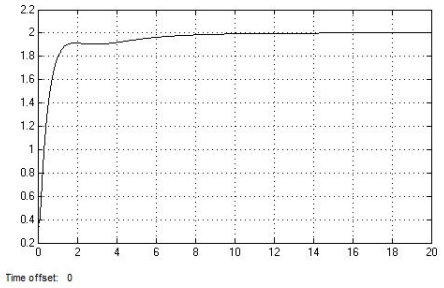


(d) Throttle response

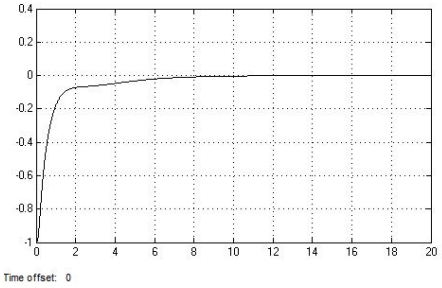
Figure 3.6: Response to $5m/s$ step command in airspeed



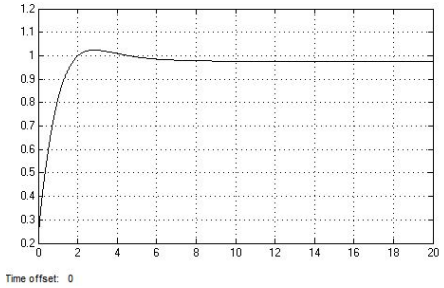
(a) Airspeed Response



(b) Climbrate Response



(c) Elevator Response



(d) Throttle response

Figure 3.7: Response to $2m/s$ step command in Climbrate

be 0.48. The natural integrator changes the state into height state which is then fed back to controller as a feedback. The time history response to a $10m$ height command is shown in Figure 3.9.

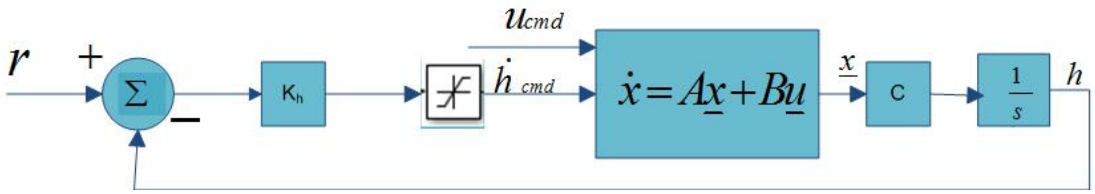


Figure 3.8: Altitude Controller Simulink Diagram

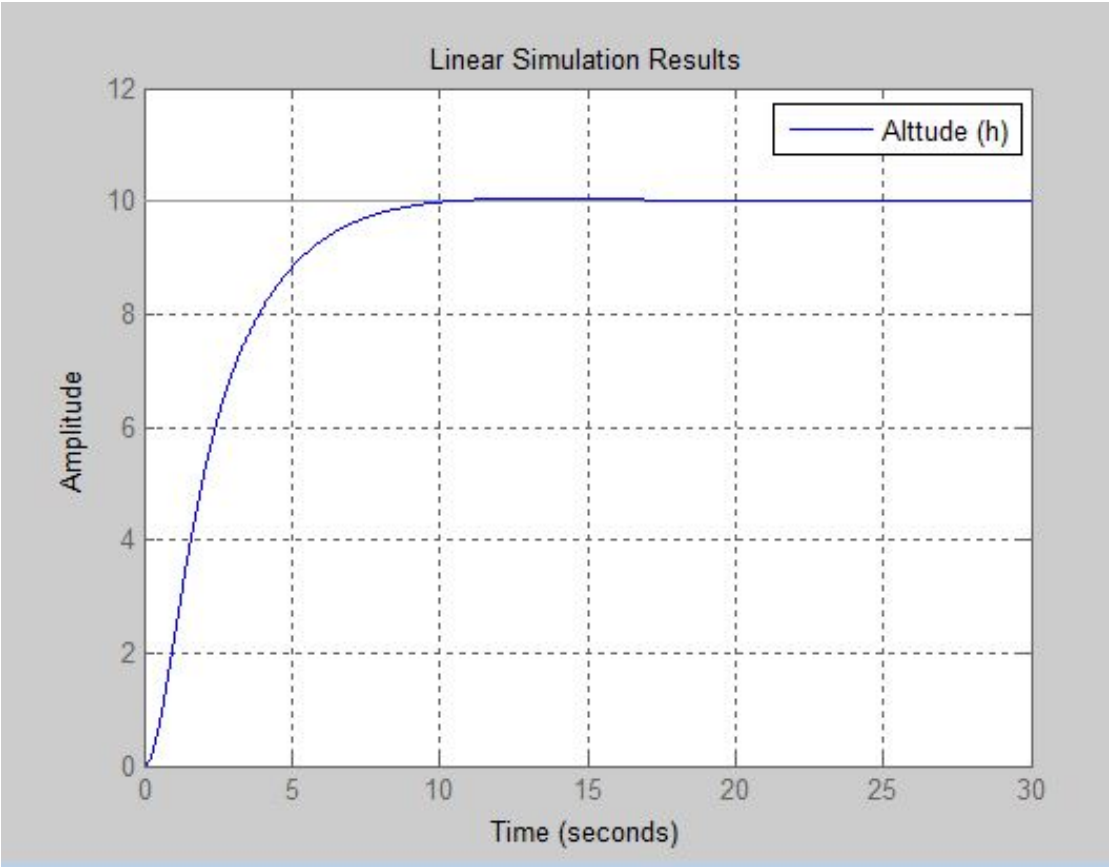


Figure 3.9: Altitude Controller Response to $10m$ Altitude Command

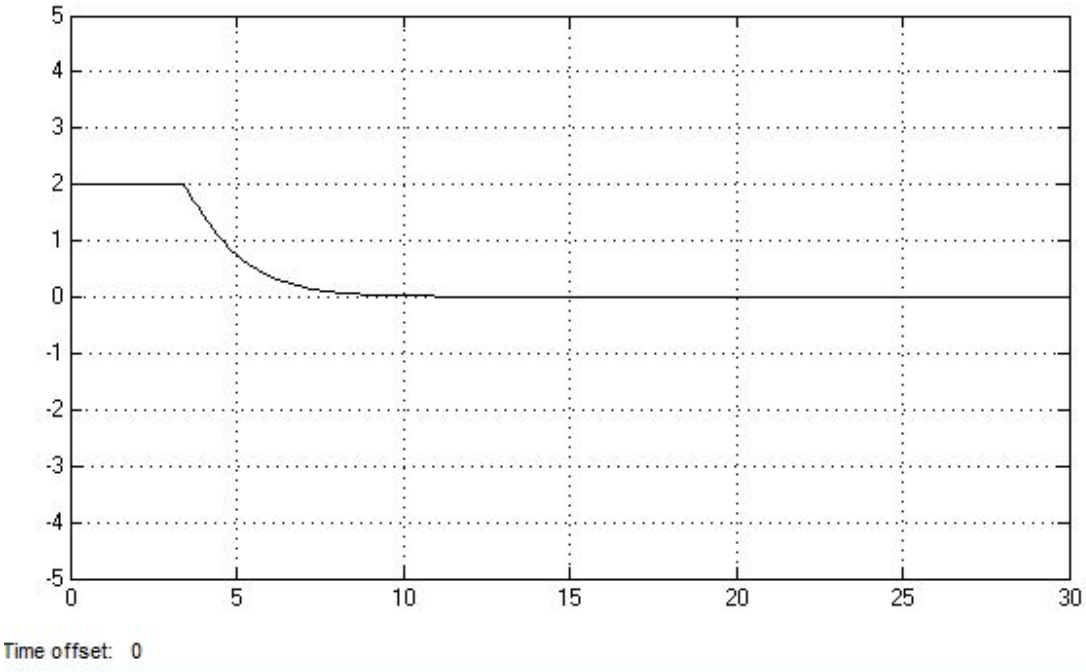


Figure 3.10: Altitude Controller Throttle Response to $10m$ Altitude Command

3.3 Summary

The controllers required for landing maneuver have been redesigned in this chapter. All controllers are designed using PID control. All controllers can regulate the motion variables of the system efficiently.

Chapter 4

Controller Design using Model Predictive Control

In this chapter, Model predictive control design is introduced. The benefits of MPC is its ability to find out predictions of systems, use of state space model and to handle hard and soft constraints of the system under study. The controller can reduce the cost of computation by using series of laguerre functions. This chapter also describes the solution of cost function under systems constraints. To find optimal solution of the problem, optimization techniques for solving cost function will be discussed.

Section 4.1 discusses the design principle and use of laguerre functions in approximating control input series. Section 4.2 discusses equations used to predict future control inputs. Section 4.3 defines optimal cost function calculation. section 4.4 discuses constrained solution of the problem. Section 4.5 discussed quadratic programming methods utilized in solving constrained cost function. Programming procedure to solve constrained problem is discussed in section 4.6. The basic diagram depicting MPC is shown in Figure4.1.

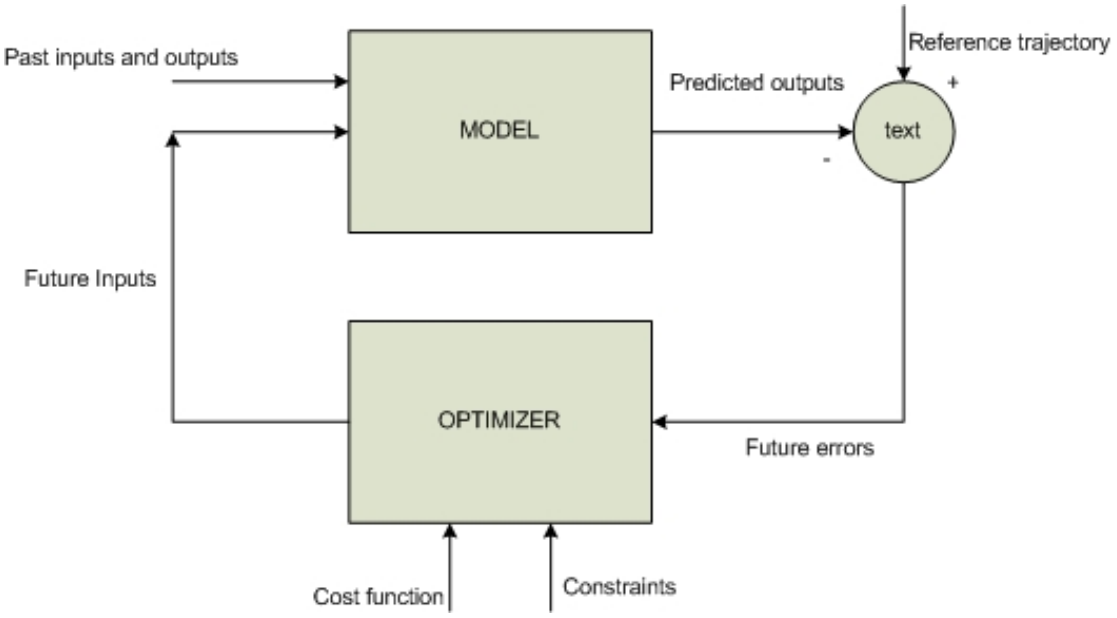


Figure 4.1: Basic block diagram of Model Predictive Control

4.1 Design Principle

4.1.1 Laguerre Functions

An arbitrary function f(t) can be expressed a series expansion

$$f(t) = \sum_{n=1}^{\infty} \wp_i l_i(t) \tag{4.1.1}$$

where $\wp_i = 1, 2, 3, \ldots n$ and $l_i(t) = 1, 2, 3, \ldots n$ are orthogonal if satisfy the following properties

$$\int_0^{\infty} {l_i}^2 dt = 1 \tag{4.1.2}$$

and

$$\int_0^{\infty} l_i(t) l_j(t) dt = 0, i \neq j \tag{4.1.3}$$

Assume that

$$\int_0^{\infty} f(t)^2 dt < \infty \tag{4.1.4}$$

and for any $\epsilon > 0$, there exists an integer N such that such that for all $k \geq N$

$$\int_0^\infty (f_(t) - \sum_{n=1}^{\infty} \wp_i l_i(t))^2 dt < \epsilon \tag{4.1.5}$$

Thus, we can approximate any function $f(t)$ that satisfies (4.1.4) arbitrarily closely by $\sum_{n=1}^{\infty} \wp_i l_i(t)$ with an increasing number of terms, N.

The laguerre functions are the orthonormal functions used and can be defined as

$$\int_0^\infty l_i(t) e^{-st} dt = \sqrt{2p} \frac{(s-p)^{i-1}}{(s+p)^i} \tag{4.1.6}$$

where p is a positive number and is called the tuning parameter. $L(t) = [l_1(t)\ l_2(t)\ ...l_N(t)]^T$. Initial conditions of the Laguerre parameter vector as $L(0) = \sqrt{2p}[11...1]^T$. The Laguerre functions satisfy the following state-space equation:

$$\begin{bmatrix} \dot{l}_1(t) \\ \dot{l}_2(t) \\ \vdots \\ \dot{l}_N(t) \end{bmatrix} = \begin{bmatrix} -p & 0 & \cdots & 0 \\ -2p & -p & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ -2p & \cdots & -2p & -p \end{bmatrix} \begin{bmatrix} l_1(t) \\ l_2(t) \\ \vdots \\ l_N(t) \end{bmatrix} \tag{4.1.7}$$

The solution of the above differential equation gives the set of Laguerre functions for $i = 1, 2, ..., N$ as

$$L(t) = e^{A_p t} L(0) \tag{4.1.8}$$

where

$$A_p = \begin{bmatrix} -p & 0 & \cdots & 0 \\ -2p & -p & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ -2p & \cdots & -2p & -p \end{bmatrix} \tag{4.1.9}$$

4.2 Prediction

MPC utilizes state space model of the plant. The linear dynamic model of the process is defined as

$$\dot{X_m} = A_m X_m(t) + B_m u(t) \tag{4.2.1}$$

$$Y(t) = C_m X_m(t) \tag{4.2.2}$$

where X_m is state vector having dimension n. In MPC we use rate of control input which can be approximated using set of laguerre function ad can be

$$\dot{u} = L^T \eta \tag{4.2.3}$$

We can incorporate this rate of change of control input into our main model by taking derivative of the plant model. The augmented model having the same output as that of the original model can be written as

$$\begin{bmatrix} \ddot{x}(t) \\ \dot{y}(t) \end{bmatrix} = \begin{bmatrix} A_p & o_m \\ C_p & o_{q\times q} \end{bmatrix} \dot{x}(t) + \begin{bmatrix} B_p \\ o_{q\times q} \end{bmatrix} \dot{u}(t) \tag{4.2.4}$$

$$y(t) = \begin{bmatrix} o_m & I_{q\times q} \end{bmatrix} \begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} \tag{4.2.5}$$

Define $\dot{X}(t) = Z(t)$ and $X(t) = [Z(t)^T \ y(t)^T]^T$. The augmented model takes the following form

$$\dot{X}(t) = AX(t) + B\dot{u}(t) \tag{4.2.6}$$

$$y(t) = CX(t) \tag{4.2.7}$$

Assume that at the current time t_i, the state variable $x(t_i)$ is available. Then at the future time τ , $\tau > 0$, the predicted state variable $x(t_i + \tau|t_i)$is described by the following equation:

$$x(t_i + \tau|t_i) = e^{A\tau} x(t_i) + \int_0^\tau A(\tau - \gamma) B\dot{u}(\gamma) d\gamma \tag{4.2.8}$$

Let the control signal of the plant can be written as

$$u(\tau) = [\dot{u}_1(\tau) \ \dot{u}_2(\tau) ... \dot{u}_m(\tau)]^T \tag{4.2.9}$$

and the input matrix B be written as $B = [B_1 B_2 ... B_m]$, where B_i is the ith column of the B matrix. The i_{th} control signal $\dot{u}_i(t)(i = 1, 2...m)$ is expressed as :

$$\dot{u}_i(\tau) = L_i(\tau)^T \eta_i \tag{4.2.10}$$

where the i_{th} vector $L_i(\tau)^T$ consists of

$$L_i(\tau)^T = l_1^i(\tau) l_2^i(\tau) ... l_{Ni}^i(\tau) \tag{4.2.11}$$

and the i_{th} coefficient vector: $\eta_i = [c_1^i c_2^i ... c_{Ni}^i]^T$ The values of p_i and N_i are tuned for all systems inputs separately and are defined for each $L_i(\tau)$ individually.

Then, the predicted future state at time τ is

$$x(t_i + \tau | t_i) = e^{A\tau} x(t_i) + \int_0^\tau A(\tau - \gamma)[B_1 L_1(\tau)^T \dots B_m L_m(\tau)^T] d\gamma \eta$$

$$x(t_i + \tau | t_i) = e^{A\tau} x(t_i) + [\phi_1(\tau) \phi_2(\tau) \dots \phi_m(\tau)] \eta$$

Thus the prediction of the future state trajectory is partly expressed in terms of $\phi_i(\tau)^T$ with $1 \leq i \leq m$.

Let define $I_{int}(\tau)^i$ having dimension $(n + q) \times N_i$ denote the the convolution integral

$$I_{int}(\tau) = \int_0^\tau A(\tau - \gamma)[B_1 L_1(\tau)^T \dots B_m L_m(\tau)^T] d\gamma \tag{4.2.12}$$

$$A I_{int}(\tau)^T - I_{int}(\tau)^T A_p^T = -B L(\tau)^T + e^{A\tau} B L(0)^T \tag{4.2.13}$$

By taking I_i as the column of I_{int} and M_i as the column of right hand side of the above equation, the elements in I_{int} can easily be determined. For $i = 1$

$$(A + pI) I_1 = M_1$$

For i=2,3,... N

$$(A + pI) I_i = M_i - 2p \sum_{k=1}^{i-1} I_k$$

4.3 Optimal Control Strategy

The future desired plant response is known because we need our plant to follow some reference trajectory. The reference trajectory can be defined as $r(t_i + \tau) = [r_1(t_i + \tau)\ r_2(t_i + \tau) \dots r_q(t_i + \tau)], 0 < \tau < T_p$. T_p is the prediction horizon. It dictates how far we wish to make predictions. The purpose of MPC is to make the plant response equal or nearly equal to the reference trajectory. Therefore, we need to minimize the following cost function

$$J = \int_0^{T_p} (r(t_\tau - y(t_i + \tau))^T Q(r(t_\tau) - y(t_i + \tau))d\tau + \int_0^{T_p} \dot{u}^T(\tau)^T R\dot{u}(\tau)d\tau \tag{4.3.1}$$

R and Q are symmetric matrices with both $Q, R > 0$. Q can be taken as I for simplicity. R is a diagonal matrix with $R = diag(\lambda(k))$, $0 \leq k \leq r$. From (4.1.4), the second term of the cost function can be written in terms of η

$$\int_0^{T_p} \dot{u}_k(\tau)^T \dot{u}_k(\tau)D\tau = \eta_k^T \eta_k$$

By putting the above equation in cost function, we get

$$J = \int_0^{T_p} (r(t_\tau - y(t_i + \tau))^T Q(r(t_\tau) - y(t_i + \tau))d\tau + \eta^T R\eta \tag{4.3.2}$$

$R = diag(R_i)$ and $R_i = \lambda_i I_{N_i \times N_i}$. $I_{N_i \times N_i}$ is unit matrix with dimension N_i.

4.4 Constrained Solution

The handling of constraints increases the systems performance and system can perform well by keeping constraints into account. There are constraints imposed on control inputs, outputs and states. In this thesis hard constraints on control inputs and its rate of change will be applied. For the constrained solution, all the constraints should be written in terms of linear inequalities. The manipulated

magnitude constraint can be written as

$$u_{low}(t_i+\tau) \leq u(t_i+\tau) = \int_0^{\tau_i} L(\gamma)^T d\tau\eta + u(t_i-\Delta t) \leq u_{high}(t_i+\tau) \tag{4.4.1}$$

Substituting the prediction of the state variables $x(t_i+\tau|t_i)$ equation into cost function. This will result in the following form of equation

$$J = \eta\left[\int_0^{Tp} \Phi(\tau)Q\Phi(\tau)^T d\tau + R_L\right]\eta + 2\eta^T \int_0^{Tp} \Phi(\tau)Qe^{A\tau}d\tau x(t_i) + x(t_i)^T \int_0^{Tp} e^{A^T\tau}Qe^{A\tau}d\tau x(t_i)$$

Let us define variables which depend on τ only and are independent of η

$$\Pi = \int_0^{Tp} \Phi(\tau)Q\Phi(\tau)^T d\tau + R_L$$

$$\psi_1 = \int_0^{Tp} \Phi(\tau)Qd\tau$$

$$\psi_2 = \int_0^{Tp} \Phi(\tau)QCe^{A\tau}d\tau$$

$$\Phi(\tau) = [C[I_{int}^1\ I_{int}^2\ \ldots I_{int}^r]]$$

so cost function becomes

$$J = \eta^T\Pi\eta - 2\eta^T[\psi_1 r(t_i) - \psi_2 x(t_i)] + \int_0^{T_p} \dot{u}^T(\tau)^T R\dot{u}(\tau)d\tau \tag{4.4.2}$$

The value of η that minimized J can be found by

$$\eta = \Pi^{-1}(\psi_1 r(t_i) - \psi_2 x(t_i)) \tag{4.4.3}$$

subject to

$$\begin{bmatrix} \int_0^{\tau_i} L_k(\gamma)d\gamma \\ -\int_0^{\tau_i} L_k(\gamma)d\gamma \end{bmatrix}\eta \leq \begin{bmatrix} u_{low}(t_i+\tau_i) + u(t_i-\Delta t) \\ -u_{low}(t_i+\tau_i) + u(t_i-\Delta t) \end{bmatrix}$$

MPC in case of unconstrained solution is just like linear quadratic regulator. When constraints are applied on the problem, quadratic

programming is utilized in solving the problem.Stable performance can be guaranteed using MPC. The delays in response can occur due to constraints on the control inputs. but it can be improved and tuned with the help of tuning parameters which is the main feature of designing MPC using laguerre functions.

4.5 Quadratic Programming methods

Optimization algorithms are used to treat inequalities equations of constraints. In optimization, the set of independent variables are introduced and their acceptable values are obtained at particular conditions. These conditions are called constraints. These optimization algorithms are applied to cost functions and optimum values of control signal will be obtained.

The concept of stationary point for which $g(x) = 0$ and $\delta(f(x) = 0$ is fundamental to the subject of unconstrained optimization and is a necessary condition for a local minimizer. Lagrange multipliers arise when similar necessary conditions are sought for the solution x^* of the constrained minimization problem.

In MPC, there may exist a point at which no output is available for particular inputs because of constraints. In this case system return without a result and will show systems in feasibility. So it is always better to impose some constraints known as hard constraints while relaxing other ones. The active set method applies this technique. But this method is not feasible in case of large number of constraints. So primal dual method is utilized. To make the problem easily solvable in computer, Hildreth quadratic programming method is useful for solving this dual problem. For solving non linear constraints, shor's algorithm is useful but in this thesis, only linear inequality constraints are dealt with Hildreth programming procedure. The Active Set method and Hildreth's Quadratic Programming method produce the same results. They both can handle constraints well as compared to other methods. Shor's algorithm gives noise at the derivative of the input and does not give feasible solution. Also more iterations are required to get the feasible solution. It is not a good solution for optimization online. The active set method is best for

MPC using laguerre functions in problems in which constraints are fixed or changing. The method eliminates the inactive constraints and solution converges earlier. But it is not useful when constraints are larger and while eliminating constraints, it may be eliminating the feasible solutions.

4.5.1 Lagrangian Dual Function

The Lagrange Multiplier theorem lets us translate the original constrained optimization problem into an ordinary system of simultaneous equations at the cost of introducing an extra variable: The primal problem can be written as

$$Minimize\ \frac{1}{2}x^TQx + c^Tx \tag{4.5.1}$$

$$subject\ to\ Ax \geq b$$

$$i.e\ \ Minimize\ f(x)$$
$$s.t.\ g(x) \leq 0$$

where

$$f(x) = \frac{1}{2}x^TQx + c^Tx$$
$$g(x) = b - Ax \leq 0$$
$$x \in R^n$$

Assume that Q is positive semi definite, so that $f(x)$ is convex.Lagrangian multipliers is the strategy for finding the local maxima and minima of a function subject to equality constraints.

$$L(x, \lambda) = f(x) + \lambda^T g(x)$$
$$L(x, \lambda) = \frac{1}{2}x^TQx + c^Tx + \lambda^T(b - Ax)$$

For each value of λ, an unconstrained minimization of a convex quadratic function must be performed. Because of the convexity of

the Lagrangian function, the optimal x must be a stationary point of the Lagrangian function.

$$\Delta_x L(x(\lambda), \lambda) = 0 \tag{4.5.2}$$

i.e for each λ, we must choose x to satisfy

$$\Delta_x L(x(\lambda), \lambda) = Qx + c - A^T\lambda = 0 \tag{4.5.3}$$

In this problem, we need to minimize dual objective function while x must satisfy (4.5.3). Therefore the problem becomes Lagrangian dual of the original quadratic problem which can be written as

$$Maximize \;\; L(\lambda) = \lambda^T b - \frac{1}{2}x^T Q x \tag{4.5.4}$$

$$subject\ to \;\; Qx + c - A^T\lambda = 0 \tag{4.5.5}$$

$$\lambda \geq 0 \tag{4.5.6}$$

If is positive definite i.e f(x) is convex then Q must be non singular and can be solved as

$$Qx + c - A^T\lambda = 0;$$
$$x(\lambda) = Q^{-1}[A^T\lambda - c]$$

This gives minimal value of x for particular Lagrange multiplier. This value of x can be inserted into Lagrangian dual function and an equation independent of x can b found as

$$Maximize \;\; e^T\lambda + \frac{1}{2}\lambda^T D\lambda - \frac{1}{2}c^T Q^{-1}c$$
$$subject\ to \;\; \lambda \geq 0$$

where

$$e = b + AQ^{-1}c$$
$$D = -AQ^{-1}A^T$$

In dual problem, number of unknown variables to be found decreases to number of constraints. So the primal problem when converted to its dual is can be dealt easily and is computationally attractive when number of constraints are larger than the state variables.

4.6 Hildreth's quadratic programming procedure

A cyclic search coordinate search method is applied to this quadratic problem. This problem satisfies following Kuhn tucker optimality conditions.

$$\Delta L(\lambda) = D\lambda + e \leq 0$$
$$\lambda_i \frac{\delta L}{\lambda_i} = 0$$

steps of programming

1. Select an initial λ^0, ,e.g ${\lambda^0}_i = 0$
2. Let $i = 1$ and $\lambda = \lambda^o$
3. Search for maximum in direction parallel to the λ_i -axis by fixing λ_j, $j \neq 1$ and solving

$$\frac{\partial L}{\partial \lambda_i} = 0 \ for \ \lambda_i$$

4. If $\lambda_i < 0$, then fix $\lambda_i = 0$
5. Increment i. if $i \leq n$ go to step 2
6. if $\lambda \neq \lambda^o$ then let $\lambda^o = \lambda$ and go to step 1.

The current λ will satisfy the KKT optimality conditions for Dual problem. And optimum value of decision variable can be found using this λ.

4.7 Summary

This chapter discussed in detail the design principle of MPC and how predictions are made using state space model of the process. Constrained cost function has been defined and solved through quadratic programming procedure.

Chapter 5

MPC Simulation Results

5.1 Introduction

This chapter discusses the simulation results of the controller designed in this thesis for landing. section 5.2 discuses glide controller simulation results. Section 5.3 discusses flare controller results while section 5.4 discusses whole landing trajectory both glide and flare included. Section 5.5 discusses wind shear and its affect on simulation results.

5.2 Glide Controller Simulation Results:

In glide controller, UAV follows a glide path defined in section 2.1. It is a straight path line starting from some initial height and ends at the point where flare begins. UAV has to follow the glide path with minimum error. Calculations related to glide controller are discussed in section 2.1. According to the calculations the aircraft must reach from height $21m$ to height $4.58m$ covering $250m$ horizontal distance. Result of MPC glide path controller is shown in Figure 5.1 and Figure 5.2 . It can be seen from the diagram that the controller is covering the glide phase in specified distance.

Now the response results of the controller at different values of parameters are discussed below.

Case I: Effect of pole location tuning parameter: Here, the effect of pole location on the system response will be discussed in

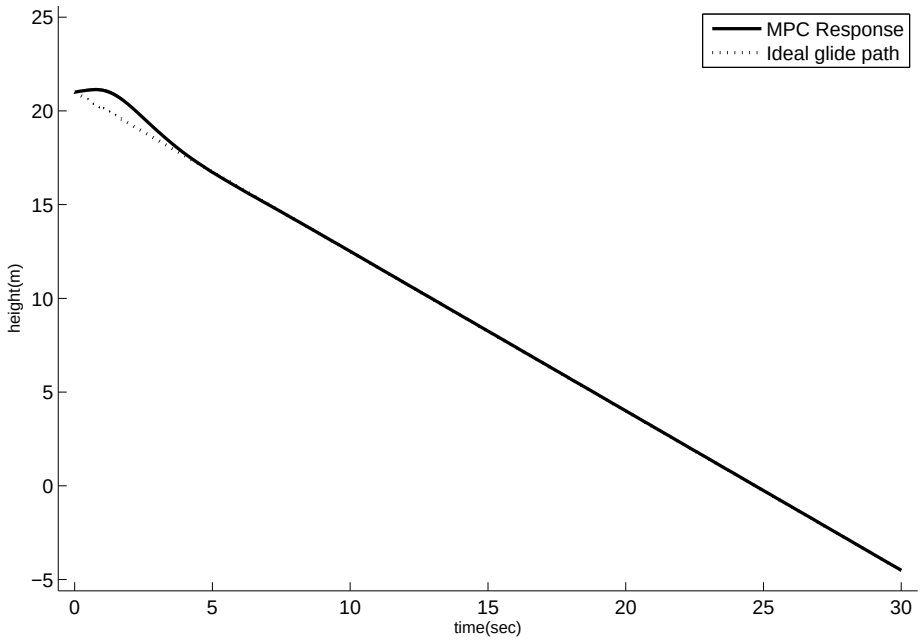


Figure 5.1: Glide Simulation(height vs time)

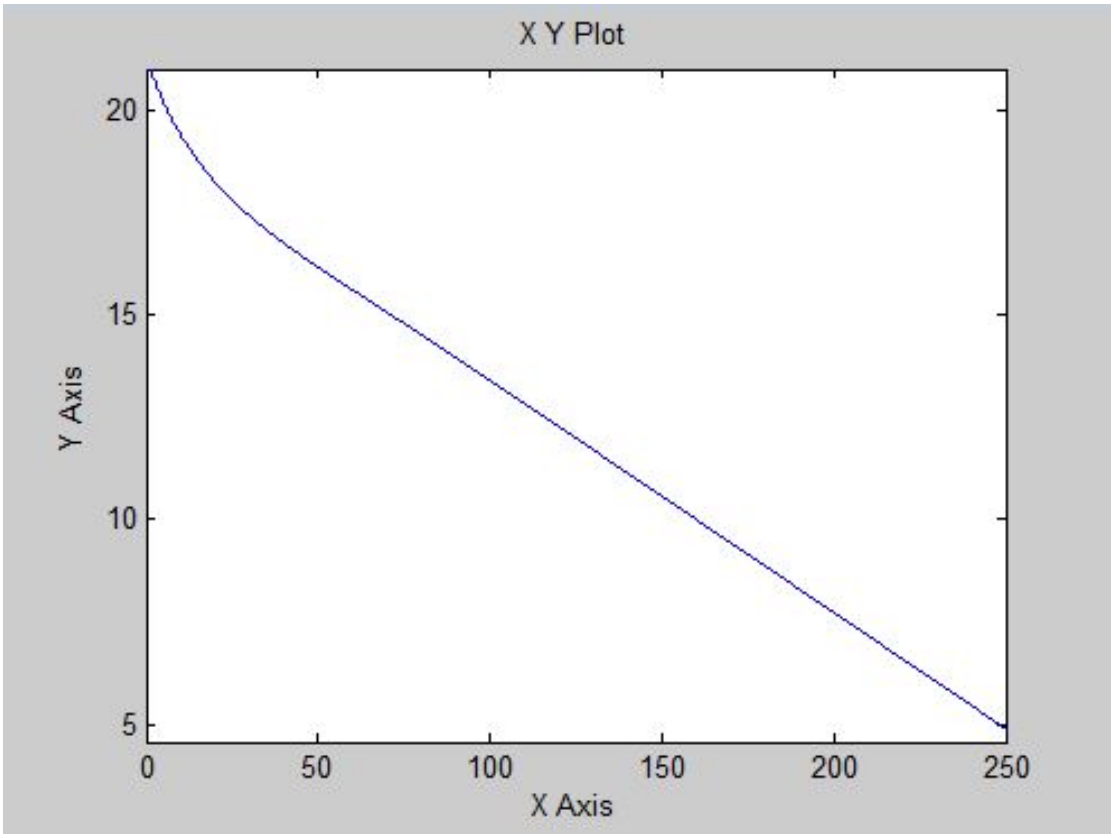
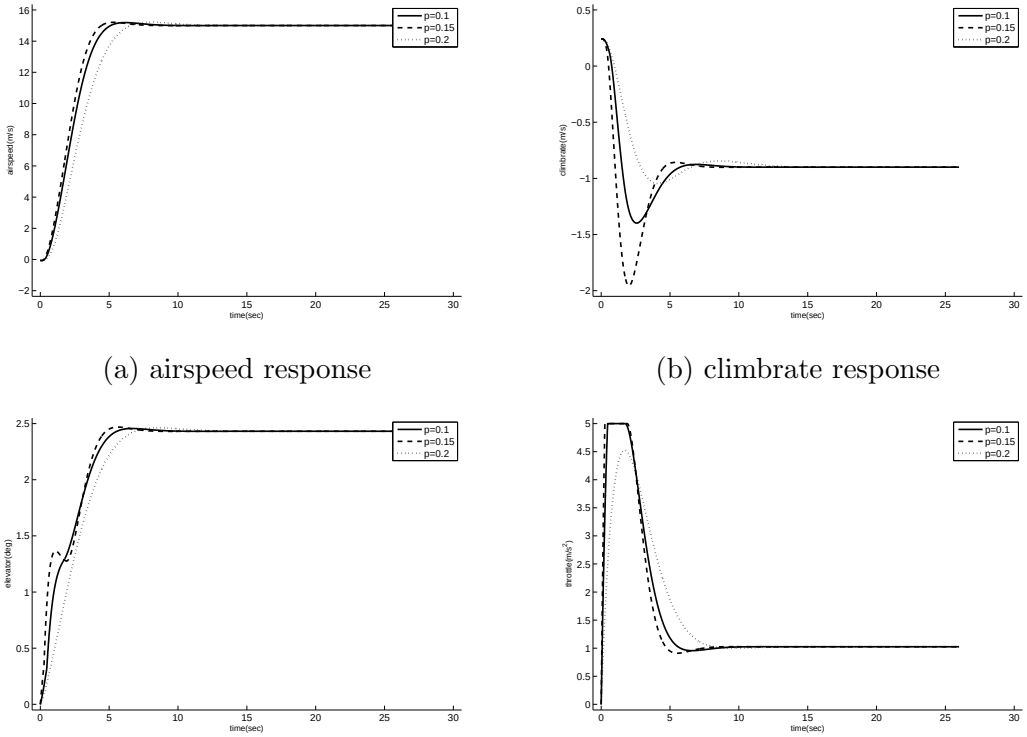


Figure 5.2: height vs distance plot

Figure 5.3. Prediction horizon T_p and tuning parameters N is kept constant while changing p.

$$N_1 = N_2 = 11 \quad T_{p1} = T_{p2} = 15$$

Conclusion: There is increase in response time when we increase value of pole locating parameter p. Small values of pole parameter give smooth response to all inputs. so small values are preferable.



(a) airspeed response

(b) climbrate response

(c) elevator response

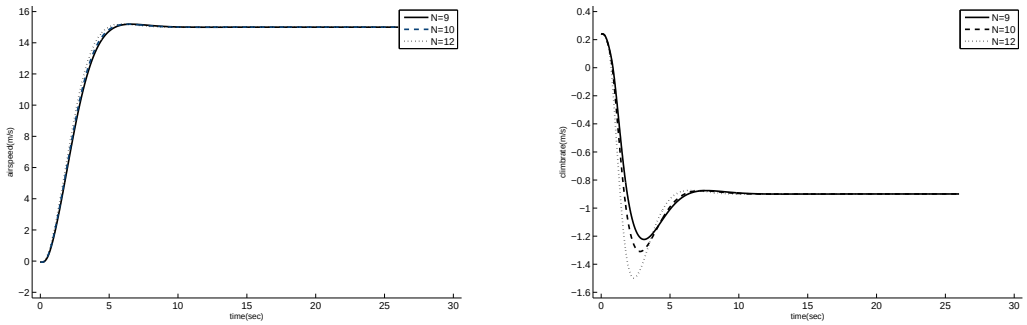
(d) throttle response

Figure 5.3: airspeed and climbrate response(top), control signal (elevator and throttle)(bottom) of the UAV glide path ($N = 11, T_p = 15$): with $p = 0.1; p = 0.15; p = 0.2$

Case 2: Effect of tuning parameters N: Here, the effect of number of Tuning parameters of Laguerre functions on the system response will be discussed. Prediction horizon T_p and pole locating tuning parameter p is kept constant while changing N.

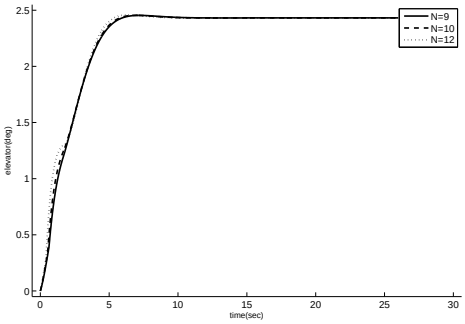
$$p = 0.1 \quad T_{p1} = T_{p2} = 15$$

Conclusion: Large values of N give faster response with minimal fluctuations as can be seen in Figure 5.4.

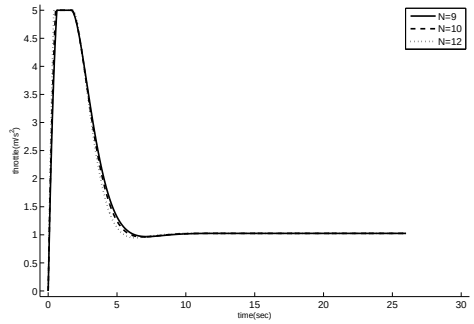


(a) airspeed response

(b) climbrate response



(c) elevator response



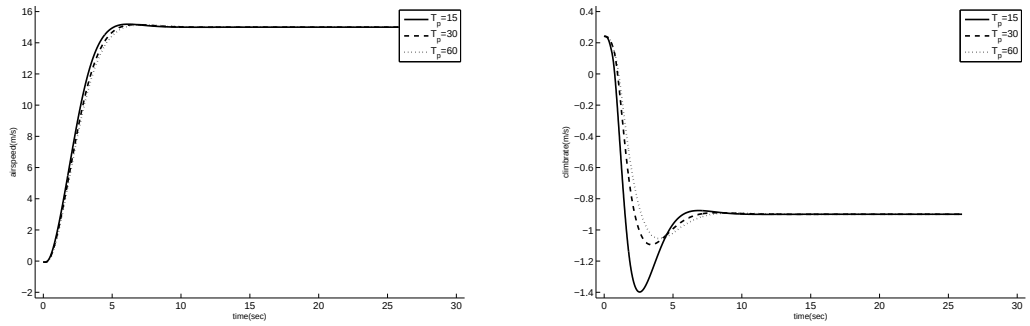
(d) throttle response

Figure 5.4: airspeed and climbrate response(top), control signal (elevator and throttle)(bottom)of the UAV glide path ($p = 0.1, T_p = 15$): with $N = 9; N = 10; N = 12$

Case 3: Effect of prediction horizon T_p: Here, the effect of prediction horizon on the system response will be discussed. Pole location p and number of tuning parameters N is kept constant while changing T_p. Results are shown in Figure 5.5

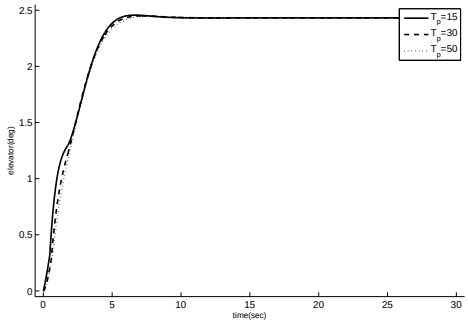
$$p = 0.1 \quad N = 11$$

Conclusion: Small values of T_p give fast response

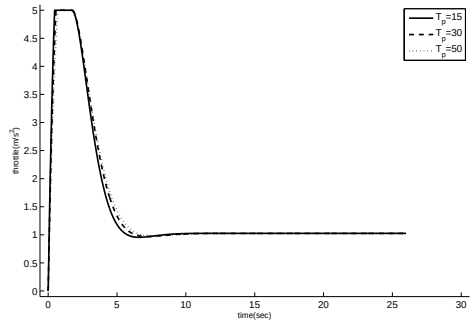


(a) airspeed response

(b) climbrate response



(c) elevator response



(d) throttle response

Figure 5.5: airspeed and climbrate response(top), control signal (elevator and throttle)(bottom) of the UAV glide path ($p = 0.1, N = 11$): with $T_p = 15; T_p = 30; T_p = 50$

5.3 Flare Controller Simulation Results

After glide path, UAV follows the flare path which exponentially decrease the height of plane and allow it to smoothly land on the ground. Figure 5.6 shows flare controller results.

In this section, effect of various factors of MPC (pole location parameter, number of tuning parameters and prediction horizon) will be discussed.

Case I: Effect of pole location tuning parameter Figure 5.7 depicts the effect of pole location parameters on results of flare path.

Conclusion: In case of flare, response improves by increasing value of pole locating parameter.

Case 2: Effect of tuning parameters N Here effect of number of laguerre paramters is discussed as shown in Figure 5.8.

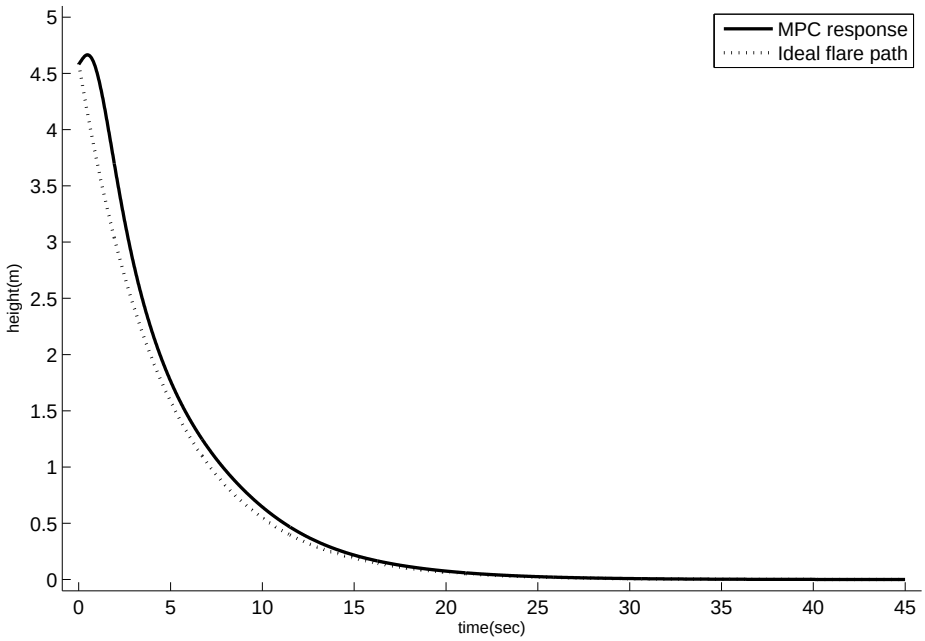
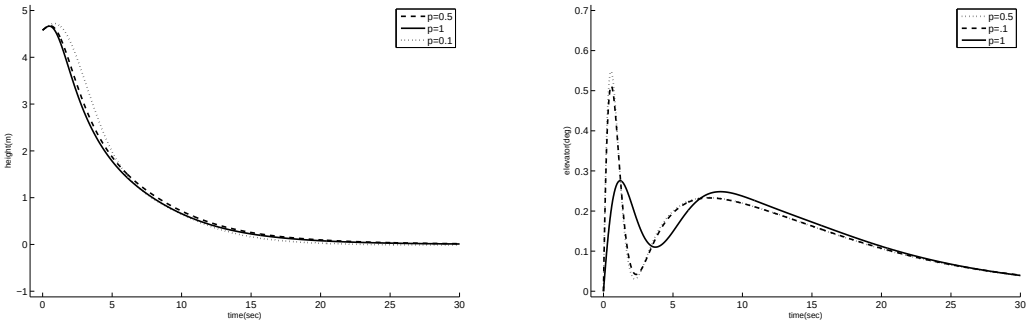


Figure 5.6: flare path trajectory



(a) flare height vs time

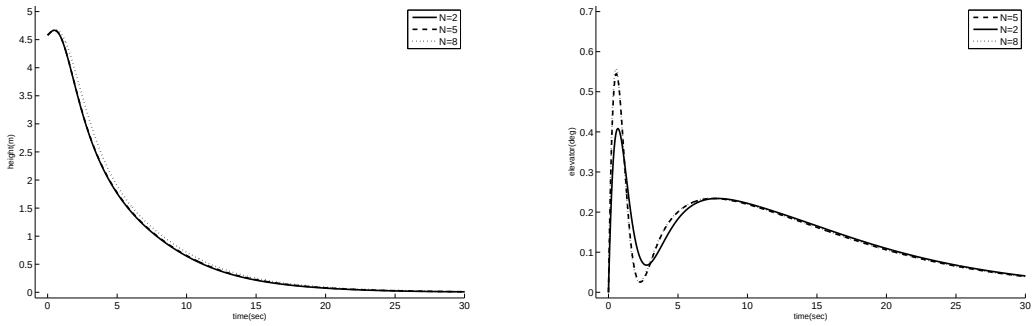
(b) elevator response

Figure 5.7: UAV flare path ($p = 0.1, 0.5, 1$) with $T_p = 3$ and $N = 3$

Conclusion: In case of flare, response improves by decreasing number of pole locating parameters N.

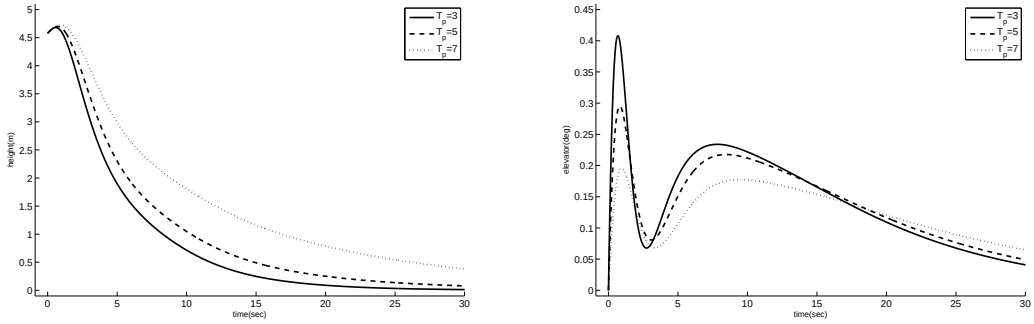
Case 3: Effect of prediction horizon T_p Prediction horizon effect on flare path of landing is discussed and analyzed in this subsection in Figure 5.9.

Conclusion: Small values of prediction horizon T_p gives good results with less settling time.



(a) height vs time (b) elevator response

Figure 5.8: UAV flare path ($T_p = 3; p = 1$): with $N = 2, 5, 8$



(a) height vs time (b) elevator response

Figure 5.9: UAV flare path ($N = 2; p = 1$): with $T_p = 3, 5, 7$

5.4 Landing (glide and flare path) simulations

The whole landing trajectory consists of glide path and flare path combined. In this section glide and flare embedded simulations will be discussed. Both glide path and flare controllers are combined and simulations are run one after another. The whole trajectory of landing i.e glide path and flare path combined is obtained. Glide path starts at $21m$ height. After reaching height of $4.58m$, controllers from glide path are switched to flare controllers which then control the trajectory in exponential path. The simulation diagram is shown in Figure 5.10. Black solid line shows the measured landing path while dotted lines shows the ideal path which must be followed. The controllers are following the trajectory with minimal errors. Thus it can be concluded that MPC can handle the landing path trajectory very well.

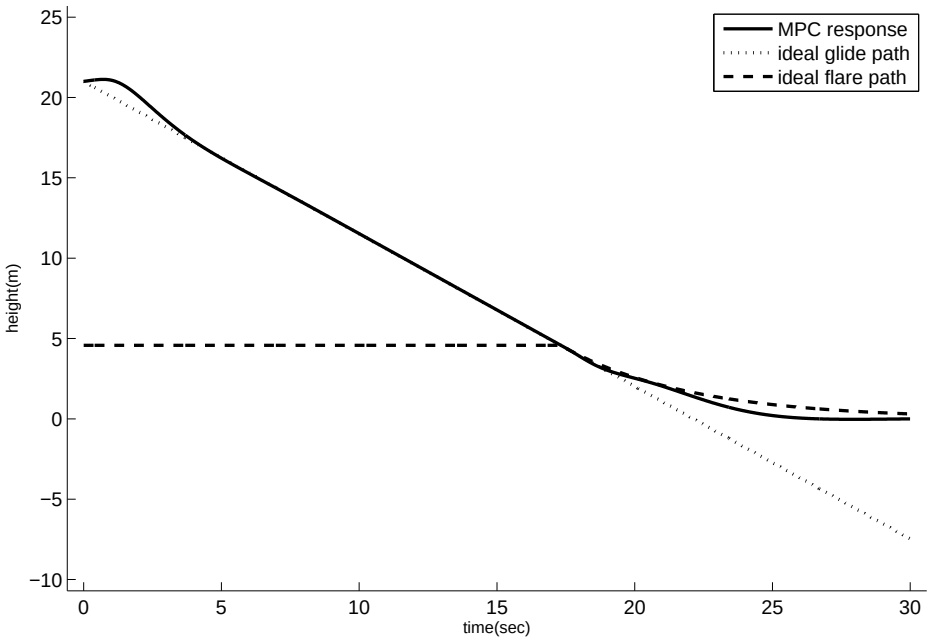


Figure 5.10: Landing Trajectory

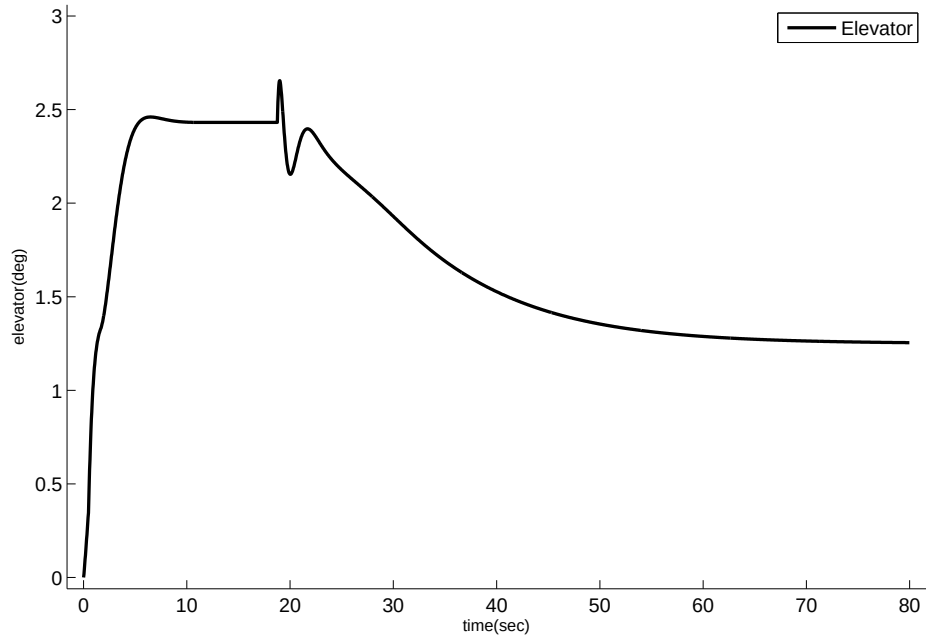


Figure 5.11: elevator control input data in landing path

Control efforts corresponding to elevator and throttle are shown in figure Figure 5.11 and Figure 5.12. The algorithm is handling constraints as it can be seen in Figure 5.12, the control when reached

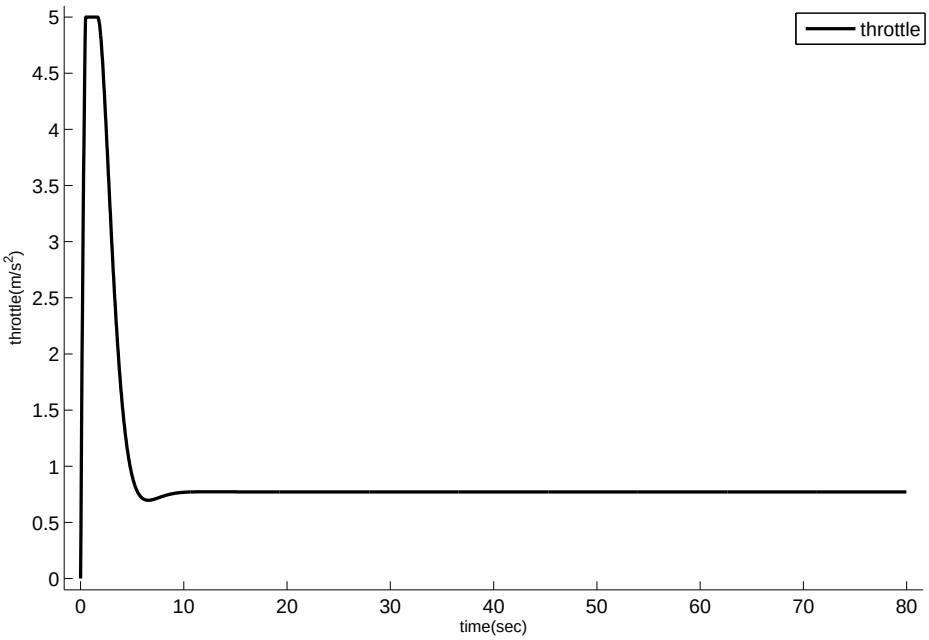


Figure 5.12: throttle control input data in landing path

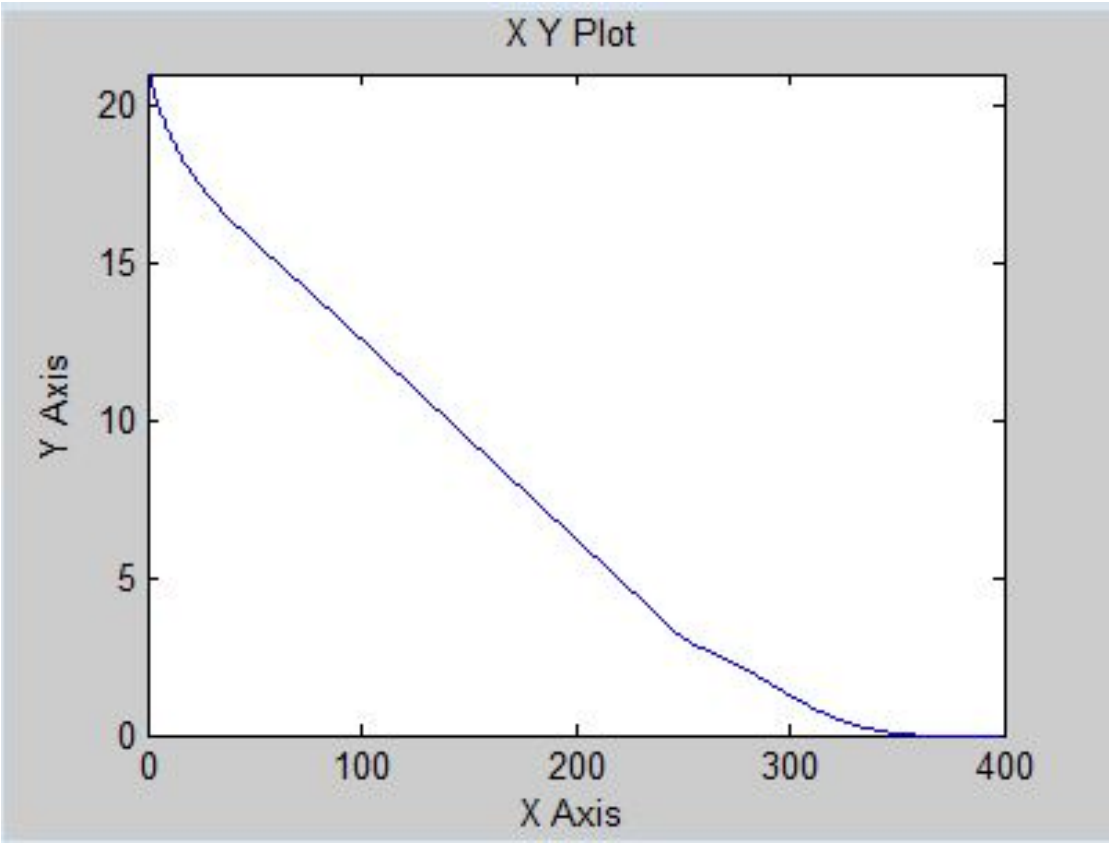


Figure 5.13: height vs distance plot

above its constraint value of 5, trimmed to its constant value and not allowed it to go above it. The controller performance degraded when constraint(throttle) was active in the start as it can be seen in Figure 5.10, there is a hump in the start of the landing trajectory. Overall the controller gives satisfactory results under constraints as well. Height vs distance plot is shown in Figure 5.13 which shows that UAV covered $350m$ distance horizontally while landing which is correct according to the calculations done for glide and flare path in section 2.1 and section 2.2.

5.5 Disturbance Rejection in Glide Controller

In this section, disturbances will be added in the plant and simulations will be performed. Effectiveness of MPC glide controller under disturbance will be studied in this section and analyzed.

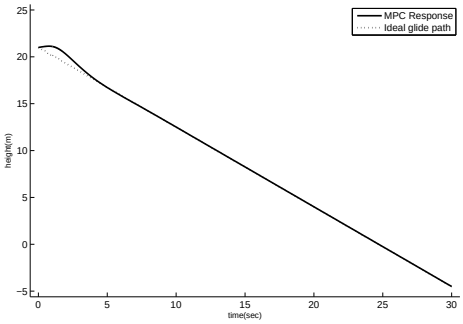
5.5.1 Windshear

Wind shear is a change in speed of the wind over short distances. Wind shear can change a regular approach towards ground into emergency situation. Aircraft is affected by this windshear because it changes the speed and motion of the aircraft relative to the ground. If the aircraft is stable and moving on landing path, when it encounters the head wind, there is significant decrease in the speed and lift of the aircraft which causes it to descend earlier. Now the pilot must compensate for this loss of the lift, but the critical factor is that whether the aircraft has sufficient altitude to complete a recovery.

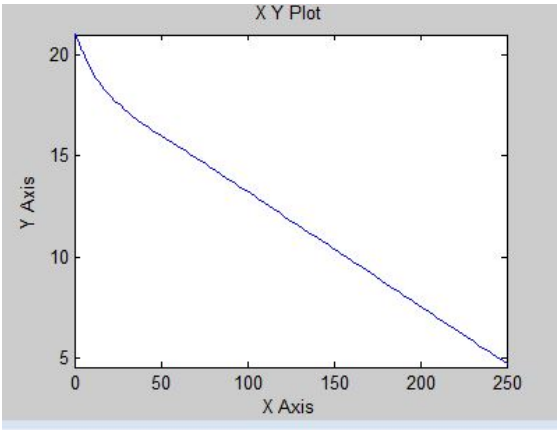
The aircraft drops below the glide path if it encounters the wind shear. The aircraft starts to sink. The airspeed is decreased equivalent to shear value. In this situation, the pilot considers this a deviation from the glide path and tries to correct this deviation by more power and increased pitch. Mostly the correction is large and the aircraft rises above the glide path and starts to decent and follow the glide path afterwards.

There are two cases, one in which relative speed of aircraft increases

and other in which relative speed decreases. Decrease in relative speed causes the aircraft to fall below the glide slope path, while increases in relative velocity causes the aircraft to rise above the glide slope trajectory. This can cause accidents. several cases have been reported in which shear forces were a contributing factor in major aircraft crash.



(a) Glide path with no windshear effect



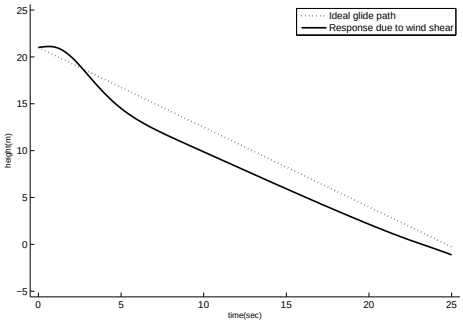
(b) height vs distance graph with no windshear effect

Figure 5.14: Simulations without any disturbance added

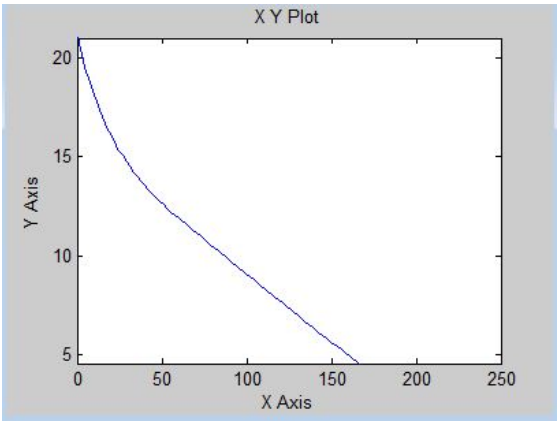
According to calculation the aircraft under study must complete horizontal distance of $250m$ while covering vertical distance from $21m$ to $4.58m$. Figure 5.14 shows the simulations under no windshear effect. Now we will incorporate windshear force and see the impact on simulations.

5.5.2 Disturbance Rejection

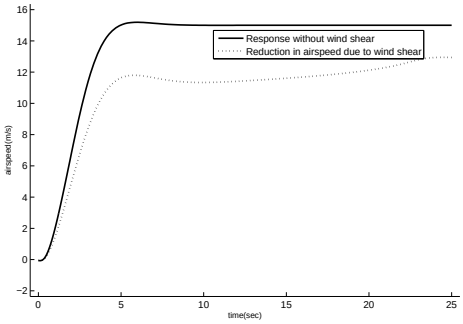
Suppose we apply head wind sheer force of $5m/s$ at an altitude of $6m$. Due to headwind the speed of the aircraft will decrease, airlift lift decreases and sink rate increases and it starts to drop off below its glide path. As can be seen from Figure 5.15, MPC can handle this disturbance well and tries to move the aircraft in defined glide trajectory. Aircraft is trying to reduce the effect of disturbances.



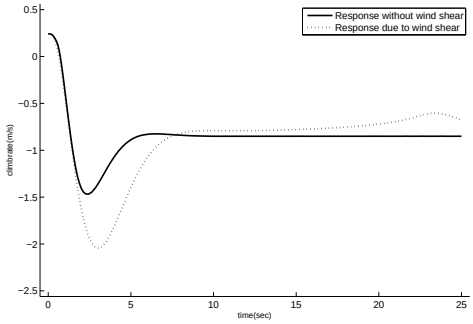
(a) glide trajectory in $5m/s$ windshear force



(b) height vs distance in $5m/s$ windshear force



(c) airspeed change due to windshear

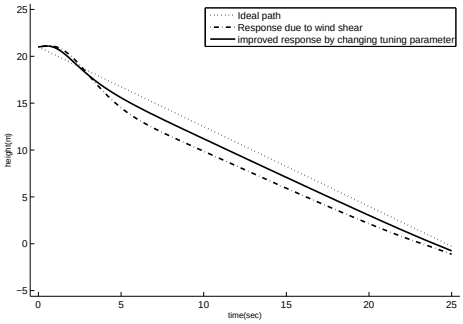


(d) climbrate change due to windshear

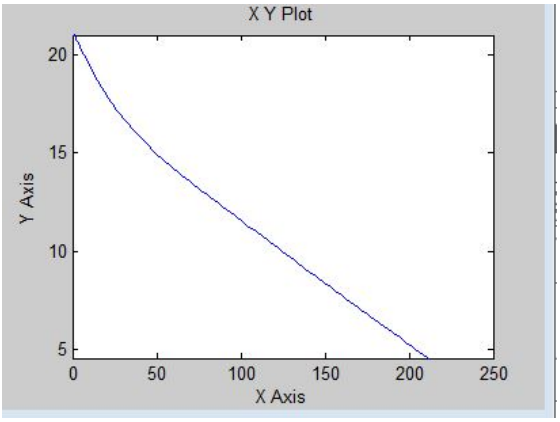
Figure 5.15: Change in response of glide controller due to windshear

5.5.3 Reducing the effect of disturbance

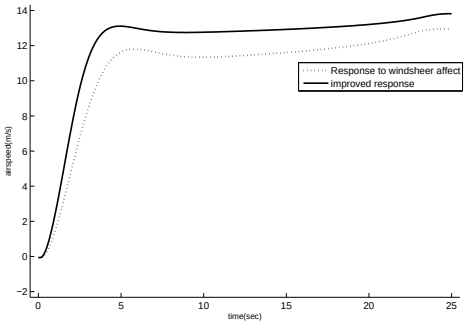
The purpose of this simulation is to check how affect of disturbances can be reduced by changing tuning parameters. Increasing prediction horizon improves the result. The simulation in Figure 5.16 show improved results when prediction horizon is increased.



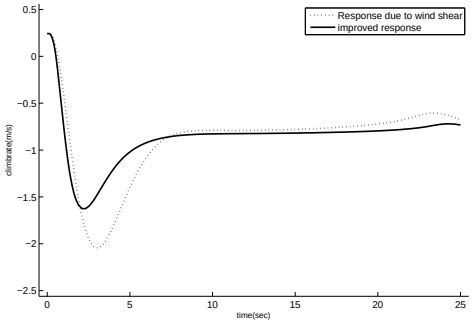
(a) glide path simulation



(b) height vs distance plot



(c) airspeed simulation



(d) Climbrate simulation

Figure 5.16: improved results by increasing prediction horizon to 25

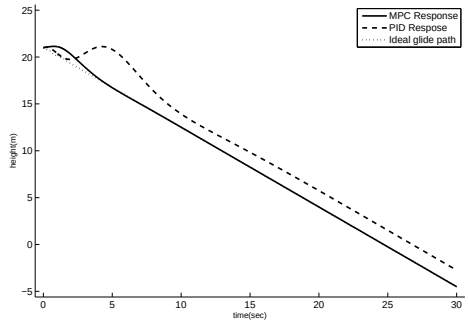
5.6 PID and MPC results comparison

In this section comparison of results of two control techniques i.e PID and MPC will be done through simulations.

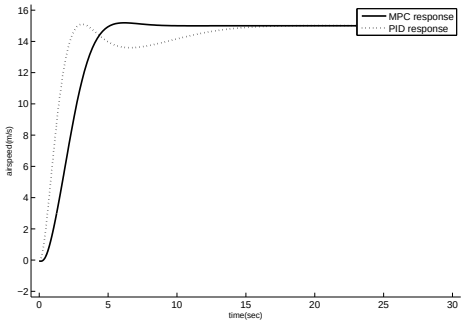
5.6.1 Glide Controller

Figure 5.17 shows simulations of PID ad MPC controller results of glide controller. In Figure 5.17a, glide path trajectory is very well followed by MPC, while PID gives results with certain set point error. In Figure 5.17b, MPC has attained the speed for landing which is 15m/s(double of stall speed) and constant climbrate command with less rise time as compared with PID. Throttle input simulation in Figure 5.17d shows constrained input control. MPC algorithm has prevented the input from going above its set limit(i.e $5m/s^2$) and trimmed its value without affecting the output. While PID has no

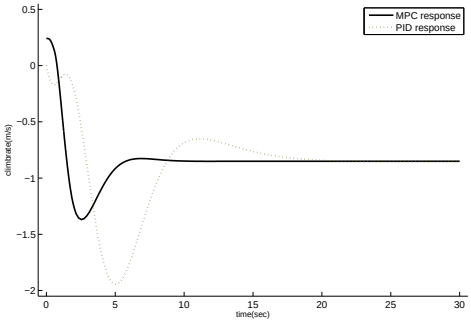
mechanism to limit the control input within its constrained value.



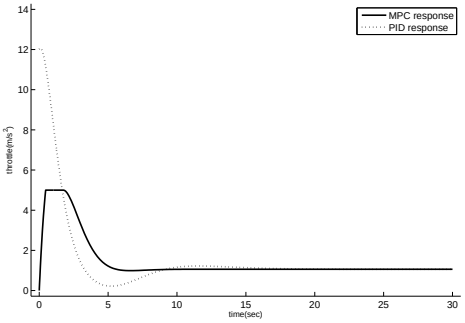
(a) Glide path results of MPC and PID



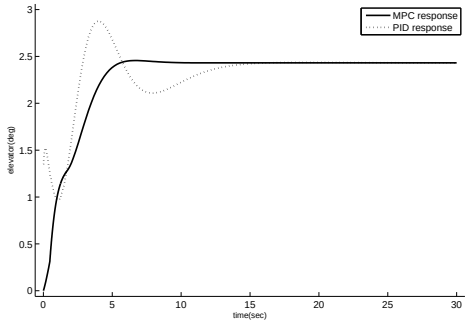
(b) Airspeed simulation



(c) Climbrate simulation



(d) Throttle input simulation



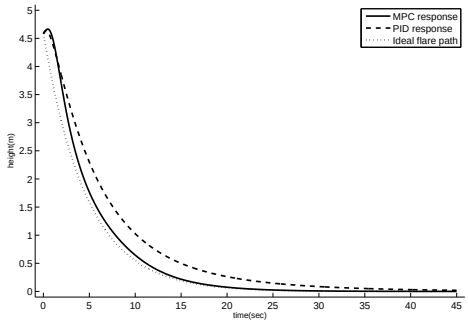
(e) Elevator input simulation

Figure 5.17: Comparison of PID and MPC Glide path controllers

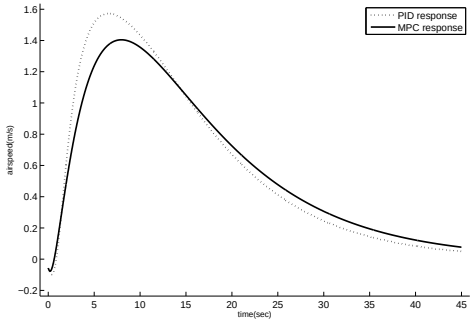
5.6.2 Flare Controller

Simulations of comparsion of PID and MPC for flare path is shown in Figure 5.18. In Figure 5.18a, MPC has followed the flare path very well as compared to PID. In flare throttle is completely closed and airspeed gradually slows down. . so the trhottle input control is

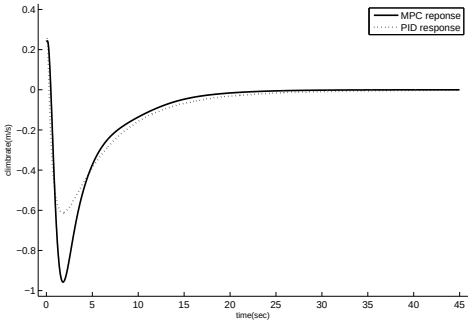
zero. Climbrate simulation of flare in Figure 5.18c shows attainment of zero climbrate faster as compared to PID. Elevator and throttle control input simulation in Figure 5.18e shows control inputs within its limit bounds.



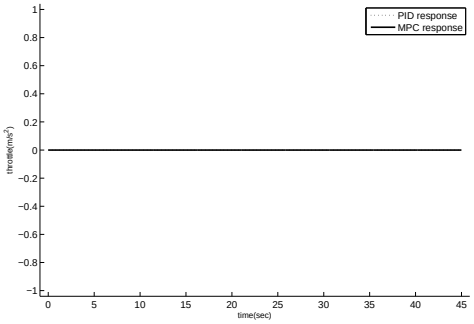
(a) Glide path results of MPC and PID



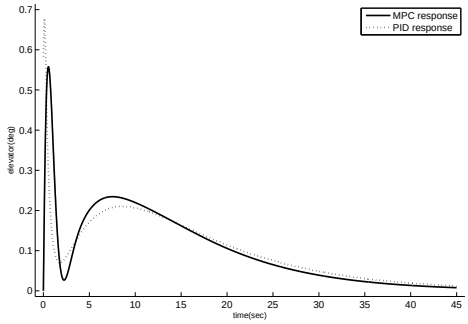
(b) Airspeed simulation



(c) Climbrate simulation



(d) Throttle input simulation



(e) Elevator input simulation

Figure 5.18: Comparison of PID and MPC Flare path controllers

5.7 Summary

In this chapter, MPC concepts are reviewed by running simulations in simulink. Glide and flare controller simulations are performed and affect of different parameters on response of the process is checked. Constraints are applied to the controllers and quadratic programming is applied to ensure that the response does not violate the constraints imposed on the control inputs. This confirmed the evidence of ability of MPC to handle constraints. Results of PID and MPC are compared. MPC shows better performance and can handle constraints as compared to PID. The slight delay in response may have occurred due to constraints on the control inputs. But the response can be improved by changing the tuning parameters which is key success of using MPC using laguerre functions. MPC also gives faster response when disturbances are added. MPC can handle windshear disturbances between $1m/s$ to $5m/s$ in glide controller. The response can further be improved by changing the tuning parameters.

Chapter 6

Conclusion

6.1 Conclusion

In this thesis, orthonormal functions are used to describe the control signal of the plant and then continuous model predictive control is built. A multi input multi output linear state space model of the unmanned aerial vehicle is used for designing the controller. It is shown through simulations that model predictive control is capable of handling constraints imposed on the control input very effectively.

The controller works very well when designed using MPC. It follows the reference trajectory while taking constraints into account. The simulations showed that the controller will never override the input constraints. It should be noted that when constraints was active in landing, the controller showed slightly degraded performance and was not able to follow the reference trajectory required for landing. But overall the response was satisfactory. If the constraints are active for a longer period then the controller will try to compensate by adjusting the other control input or state so that overall controller gives satisfactory response. for example if increase in speed is required, the controller will not only increases the throttle to its full limit but also adjust the pitch of the aircraft to increase the speed instead of waiting for the aircraft model to gain its desired speed.

The work in this thesis shows that continuous time MPC using orthonormal functions to design control signals gives good performance when applied to linear state space model of unmanned aerial vehicle for landing phase.

Bibliography

[1] I. K. Peddle, "Autonomous flight of a model aircraft," Ph.D. dissertation, Stellenbosch: Stellenbosch University, 2005.

[2] "Body axis description," "http://www.chrobotics.com", accessed: 2010-09-30.

[3] J.-C. Roos and I. Peddle, "Autonomous take-off and landing of a low cost unmanned aerial vehicle," *University of Stellenbosch*, 2007.

[4] G. Ducard, K. Kulling, and H. Geering, "A simple and adaptive on-line path planning system for a uav," in *Control & Automation, 2007. MED'07. Mediterranean Conference on*. IEEE, 2007, pp. 1–6.

[5] M. Niculescu, "Lateral track control law for aerosonde uav," in *AIAA, Aerospace Sciences Meeting and Exhibit, 39 th, Reno, NV*, 2001.

[6] S. Park, J. Deyst, and J. P. How, "A new nonlinear guidance logic for trajectory tracking," in *AIAA guidance, navigation, and control conference and exhibit*, 2004, pp. 16–19.

[7] G. J. Ducard, *Fault-tolerant flight control and guidance systems: Practical methods for small unmanned aerial vehicles*. Springer Science & Business Media, 2009.

[8] K. VolKan, "Design of an autonomous landing control algorithm for a fixed wing uav," Ph.D. dissertation, Master of Science Thesis, Ankara: Middle East Technical University, 2007.

[9] H.-S. Ju, C.-C. Tsai, and C.-T. Lee, "Flight path control design for glide-slope tracking by backstepping," in *IEEE International*

Conference on Mechatronics, 2005. ICM'05. IEEE, 2005, pp. 887–892.

[10] C. Le Roux and J. Engelbrecht, "Automated landing of an intelligent unmanned aerial vehicle in crosswind conditions using total energy control," 2014.

[11] H.-S. J. C.-C. Tsai, "Glidepath command generation and tracking for longitudinal autolanding."

[12] S. Singh and R. Padhi, "Automatic path planning and control design for autonomous landing of uavs using dynamic inversion," in *2009 American Control Conference.* IEEE, 2009, pp. 2409–2414.

[13] J. A. de Bonfim Gripp and U. P. Sampaio, "Automatic landing of a uav using model predictive control for the surveillance of internal autopilot's controls," in *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on.* IEEE, 2014, pp. 1219–1224.

[14] S. Koo, S. Kim, and J. Suk, "Model predictive control for uav automatic landing on moving carrier deck with heave motion," *IFAC-PapersOnLine*, vol. 48, no. 5, pp. 59–64, 2015.

[15] S. H. Mathisen, T. I. Fossen, and T. A. Johansen, "Non-linear model predictive control for guidance of a fixed-wing uav in precision deep stall landing," in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on.* IEEE, 2015, pp. 356–365.

[16] A. Joos, P. Heritier, C. Huber, and W. Fichter, "Method for parallel fpga implementation of nonlinear model predictive control," *IFAC Proceedings Volumes*, vol. 45, no. 1, pp. 73–78, 2012.

[17] O. Durmaz, "Robust flight and landing autopilot," Ph.D. dissertation, MIDDLE EAST TECHNICAL UNIVERSITY, 2015.

[18] L. Zhi and W. Yong, "Intelligent landing of unmanned aerial vehicle using hierarchical fuzzy control," in *Aerospace Conference, 2012 IEEE.* IEEE, 2012, pp. 1–12.

[19] C. V. Rao, S. J. Wright, and J. B. Rawlings, "Application of interior-point methods to model predictive control," *Journal of optimization theory and applications*, vol. 99, no. 3, pp. 723–757, 1998.

[20] L. Wang, "A tutorial on model predictive control: Using a linear velocity-form model," *Developments in Chemical Engineering and Mineral Processing*, vol. 12, no. 5-6, pp. 573–614, 2004.

[21] L. Wang, P. J. Gawthrop, C. Chessari, and T. Podsiadly, "Continuous-time system identification of a food extruder: experiment design and data analysis," 2003.

[22] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[23] G. C. Goodwin, D. E. Quevedo, and E. I. Silva, "Architectures and coder design for networked control systems," *Automatica*, vol. 44, no. 1, pp. 248–257, 2008.

[24] J. A. Rossiter, B. Kouvaritakis, and M. Rice, "A numerically robust state-space approach to stable-predictive control strategies," *Automatica*, vol. 34, no. 1, pp. 65–73, 1998.

[25] G. Astfalk, I. Lustig, R. Marsten, and D. Shanno, "The interior-point method for linear programming," *IEEE software*, vol. 9, no. 4, pp. 61–68, 1992.

[26] F. Kappel and A. V. Kuntsevich, "An implementation of shor's r-algorithm," *Computational Optimization and Applications*, vol. 15, no. 2, pp. 193–205, 2000.

[27] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "Chapter iii constrained nonlinear programming," *Handbooks in Operations Research and Management Science*, vol. 1, pp. 171–210, 1989.

[28] R. Fletcher, *A general quadratic programming algorithm*. Citeseer, 1970.

[29] ——, "Quadratic programming," *Practical Methods of Optimization, Second Edition*, pp. 229–258, 2000.

[30] S. Nash and A. Sofer, "Linear and nonlinear programming. 1996."

[31] S. Mehrotra, "On the implementation of a primal-dual interior point method," *SIAM Journal on optimization*, vol. 2, no. 4, pp. 575–601, 1992.

[32] S. J. Wright, *Primal-dual interior-point methods*. SIAM, 1997.

[33] P. S. Stanimirović, N. V. Stojković, and V. V. Kovačević-Vujčić, "Stabilization of mehrotra's primal–dual algorithm and its implementation," *European journal of operational research*, vol. 165, no. 3, pp. 598–609, 2005.

[34] R. Bartlett, A. Wachter, and L. Biegler, "Active set vs. interior point strategies for model predictive control," in *American Control Conference, 2000. Proceedings of the 2000*, vol. 6. IEEE, 2000, pp. 4229–4233.

[35] N. Shor, "Monotone modifications of r-algorithms and their applications," *Cybernetics and Systems Analysis*, vol. 38, no. 6, pp. 855–872, 2002.

[36] D. G. Luenberger, *Optimization by vector space methods*. John Wiley & Sons, 1969.