GUIDANCE OF AEROSPACE VEHICLES: A SLIDING MODE APPROACH

by

Muhammad Zamurad Shah
PE083005

A thesis submitted to the
Department of Electrical Engineering
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY IN ELECTRICAL ENGINEERING

Faculty of Engineering
Mohammad Ali Jinnah University
Islamabad
July 2015

To my Father (Late)

Your guiding hand on my shoulder will remain with me forever,
I miss you

ABSTRACT

Recent advances in guidance technologies have enabled Unmanned Aerial Vehicles (UAVs) to execute the missions autonomously without the human interaction. A typical guidance problem is path-following, which is concerned with the design of control laws that force a vehicle to reach and follow a geometric path defined in 3-D space. A subset of 3-D path-following problem is the 2-D lateral track following; the objective is to ensure accurate ground track following of the vehicle. Path-following with bounded control input in the presence of uncertainties (e.g. in velocity) and input disturbances like wind is a challenging task. Another challenging task is good performance of guidance algorithm for both large and small track deviations without using gain scheduling. Addressing these challenges, this thesis presents novel nonlinear guidance schemes for 2-D and 3-D path-following of aerial vehicles.

This dissertation considers the development and application of sliding mode theory in the area of guidance law design for UAVs. For lateral path-following problem, the limitations of a linear sliding surface are indicated here and it has been shown that traditional linear sliding surface based design is not a viable solution. Nonlinear sliding surfaces for lateral and longitudinal planes are thereafter presented here which overcomes these limitations, and its stability is guaranteed using the Lyapunov theory. For the selection of optimum sliding parameters, work and energy principle based criterion is defined here considering the UAV maneuvering capabilities. The presented criterion for optimized sliding coefficients is related to the minimization of work done and hence it also minimize the reaching time to the desired path. Using numerical techniques, the sliding surface coefficients that corresponds to an optimal path can be selected; this working is demonstrated for a research UAV.

Based on nonlinear lateral sliding surface (that meets the criterion of a ‘good helmsman’), a novel lateral guidance scheme is thereafter presented here for straight and circular path following cases. The proposed guidance logic is derived from the sliding mode control technique, and is particularly suited for unmanned aerial vehicle (UAV) applications. Control boundedness is also proved to ensure that the controls are not saturated even for large track errors. To demonstrate the effectiveness of this algorithm, a test platform (scaled YAK-54 UAV) is developed that have a generic MPC-565 processor based flight control computer for programming of different guidance and control algorithms. The proposed guidance law is programmed in the flight control computer of the test vehicle and different scenarios of large and small cross track errors are generated in the flight. Flight results are presented here that confirm the effectiveness and robustness of the proposed guidance scheme. Moreover, the effect of wind is also analyzed and flight results in the presence of wind is presented to show the robustness of proposed algorithm. The flight test results are also compared with that of 6-dof simulation.

The work is then further extended for generalized 3-D path-following problem. Generalized 3-D kinematic equations are considered here during the design process to cater far the coupling between longitudinal and lateral motions. Using the presented optimized nonlinear manifolds, guidance scheme is then derived for the multiple-input multiple-output (MIMO) system considering the coupling between longitudinal and lateral planes. The Proposed scheme is implemented on a 6-degrees-of-freedom (6-dof) simulation of a UAV and simulation results are presented here for different 3D trajectories with and without disturbances.

Keywords: Sliding Mode Control (SMC), Unmanned Aerial Vehicles (UAVs), Guidance & Control, Sliding Manifolds, Nonlinear Sliding Surface, Path-Following, Trajectory Tracking, Cross Track Error, Lateral Guidance, 3-D Guidance, Scaled YAK-54.

LIST OF PUBLICATIONS

Journal Publications

1. S. Ussama Ali, Raza Samar, M. Zamurad Shah, A.I. Bhatti, ... , "*Lateral Guidance and Control of UAVs using Second-Order Sliding Modes*", Submitted to Journal of Aerospace Science and Technology, 2015.
2. S. Ussama Ali, Raza Samar, M. Zamurad Shah, A.I. Bhatti & K. Munawar, "*Higher-order sliding mode based lateral guidance for unmanned aerial vehicles*", Accepted for publication in Transactions of the Institute of Measurement and Control, 2015.
3. M. Zamurad Shah, M. Kemal Özgören & Raza Samar, "*Optimized Sliding manifolds for 3D path following of UAVs*", Ready for submission to Journal, 2015.
4. M. Zamurad Shah, M. Kemal Özgören & Raza Samar, "*3D Guidance of Unmanned Aerial Vehicles using Sliding Mode Approach*", WASET International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering Vol. 8 No. 10, pages 1644-1650, 2014.
5. M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Lateral track control of UAVs using the sliding approach: from design to flight testing*", Transactions of the Institute of Measurement and Control, Vol. 37 No. 4, pages 457-474, 2015.
6. M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Guidance of air vehicles: a sliding mode approach*", IEEE Transactions on Control Systems Technology, Vol. 23 , No. 1, pages 231244, 2015, http://dx.doi.org/10.1109/TCST.2014.2322773
7. Raza Samar, M. Zamurad Shah & M. Nzar, "*Lateral Control Implementation for an Unmanned Aerial Vehicle*", ISRN Journal of Aerospace Engineering, 2013.

Conference Publications

1. M. Zamurad Shah & Raza Samar, "*Altitude controller design and flight demonstration for a research UAV*", Accepted for presentation in 8th Ankara International Aerospace Conference, Sep 10-12, 2015, Ankara, Turkey.

2. Ussama Ali, M. Zamurad Shah, A.I. Bhatti & Raza Samar, "*Robust Nonlinear Control Design For Scaled Yak-54 Unmanned Aerial Vehicle*", Proceedings of 26th Chinese Control and Decision Conference (CCDC), May 31-June 2, 2014, Changsha, China.
3. M. Zamurad Shah, M. Kemal Özgören & Raza Samar, "*Longitudinal guidance of UAVs using sliding mode approach*", Proceedings of 10th International Conference on Control (CONTROL 2014), July 8-11, 2014, Loughborough, United Kingdom.
4. M. Zeeshan Baber, Ussama Ali, M. Zamurad Shah, Raza Samar, A.I. Bhatti & W. Afzal, "*Robust control of UAVs using* H^{∞} *control paradigm*", Proceedings of 9th International Conference on Emerging Technologies, December 9-10, 2013, Islamabad, Pakistan.
5. Ussama Ali, M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Lateral Control of UAVs: Trajectory Tracking via Higher Order Sliding Modes*", Proceedings of 9th Asian Control Conference, June 26-28, 2013, Istanbul, Turkey.
6. Olena Kuzmych, A. Aitouche & M. Zamurad Shah "*The Geometrical Criterion for Stability of Linear Hybrid Automata on the Plane*", Proceedings of 2013 International Conference on Control, Decision and Information Technologies (CoDIT'13), May 6-8, 2013, Hammamet, Tunisia.
7. M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Cross-Track control of UAVs during circular and straight path following using sliding mode approach*", Proceedings of 12th International Conference on Control, Automation and Systems (ICCAS), Oct 17-21, 2012, Jeju Island, South Korea.
8. Ussama Ali, M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Robust Level Flight Control Design for Scaled YAK-54 Unmanned Aerial Vehicle using Single Sliding Surface*", Proceedings of 24th Chinese Control and Decision Conference (CCDC), May 23-25, 2012, Taiyuan, China.
9. M. Ismail, M. Zamurad Shah, & M. Anwar Mughal, "*Deep Sea Motion Under Higher Sea States*", Proceedings of 9th International Bhurban Conference on Applied Sciences & Technology (IBCAST), Jan 9-12, 2012, Islamabad, Pakistan.
10. M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Lateral Control for UAVs using Sliding Mode Technique*", Proceedings of 18th IFAC World Congress, August 28 September 2, 2011, Milano, Italy.
11. M. Zamurad Shah, Raza Samar, & A.I. Bhatti, "*Sliding Mode based Lateral Control for UAVs using Piecewise Linear Sliding Surface*", Proceedings of 2011 International Conference on Communications, Computing and Control Applications, March 3-5, 2011, Hammamet, Tunisia.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

UAV	Unmanned aerial vehicle
SMC	Sliding mode control
SS	Sliding surface
VSS	Variable structure system
PF	Path following
TT	Trajectory tracking
PD	Proportional-derivative
PID	Proportional-integral-derivative
LQR	Linear quadratic regulator
MPC	Model predictive control
NMPC	Nonlinear model predictive control
WP	Way-point
BTT	Bank-to-turn
INS	Inertial navigation system
IMU	Inertial measuring unit
GNSS	Global navigation satellite system
GPS	Global Positioning System
MIMO	Multi-input-multi-output
AHRS	Attitude and heading reference system
MEMS	Micro electro-mechanical system
ECEF	Earth centered earth fixed
ENU	East north up
NED	North east down
ARF	Almost ready to fly
RoC	Rate of climb
RoD	Rate of decent
RC	Remote control
RF	Radio frequency
AGL	Above ground level
MSL	Mean sea level
WD	Work done

LIST OF SYMBOLS

Symbol	**Description**	**Units**
π	3.14159265358979323384	rad
y_e	Lateral cross track deviation	m
z_e	Deviation from path in longitudinal plane (normal)	m
h_e	Error in altitude	m
m	Mass of UAV/Aircraft	kg
g	Gravitational acceleration	m/s^2
$V = (V_g)$	Velocity of UAV relative to ground	m/s
x, y, z	Inertial position of UAV	m
V_x, V_y, V_z	Velocity components of UAV in Inertial frame	m/s
α, β	Angle of attack and side-slip	rad
V_a	Velocity of UAV relative to air	m/s
V_w	Velocity of wind	m/s
χ_p	Reference course angle	rad
χ	Course angle of UAV	rad
χ_e	$= \chi - \chi_p$	rad
γ_p	Reference flight path angle	rad
γ	Flight path angle of UAV	rad
γ_e	$= \gamma - \gamma_p$	rad
ϕ_{req}	Required/reference roll angle	rad
$(\phi,\, \theta,\, \psi)$	Roll, pitch and yaw angles of UAV	rad
$(p,\, q,\, r)$	Body angular velocities about x, y and z-axis	rad/s
u_b, v_b, w_b	Body linear velocities along x, y and z-axis	m/s
L	Aerodynamic lift force	N
L_{req}	Required aerodynamic lift force	N
R	Radius of turn of UAV	m
u	Control input	–
u_{eqv}	Equivalent control	–
s	Sliding surface	–
s_1	Lateral plane sliding surface	–
s_2	Longitudinal plane sliding surface	–
c_1, c_2	Non-linear lateral sliding surface parameters	–
c_3, c_4	Non-linear longitudinal sliding surface parameters	–
F_{thrust}	Engine thrust force	N
$F_{lateral}$	Aerodynamic force in lateral plane	N
$F_{longitudinal}$	Aerodynamic force in longitudinal plane	N

Chapter 1

INTRODUCTION

1.1 Background and Motivation

Unmanned Aerial Vehicles (UAVs) are typically known as powered vehicles that do not carry human operators, can be operated autonomously or remotely, and can carry a variety of payloads depending on their type, functionality, operational characteristics, and mission objectives. UAVs have many extraordinary abilities, like ultra-long endurance, high-risk mission acceptance and accurate trajectory following, which cannot be easily performed by manned aircraft. This is mainly made possible due to the advancement in technology in the last two decades; nowadays UAVs are the back bone of high-tech military arsenals. Besides military, their use is also in the rise in commercial and civilian applications. These include search and rescue, surveillance and law enforcement, environmental studies, gas/water/oil pipeline and power line monitoring, field spraying/crop dusting and agriculture growth patterns monitoring, mapping and surveying, media and traffic reporting, forest fire monitoring and control, radio/communications relays and disaster surveying and management operations.

In the last two decades, the usage of unmanned systems has seen its most rapid growth. This is primarily due to improvements in computing and sensing technologies that are becoming more powerful, while being smaller, requiring less power, and costing less. This has led, not only to the development of highly complex and capable unmanned systems for military applications, but has also opened the door to commercial and private applications. In recent years, the work on UAV autonomy has added a new dimension to the utility of these vehicles. Autonomy is the ability to perform a task (mission) without being directly or remotely controlled by a human operator (Wise, 2006). Autonomy technology that will become important to future UAV development falls under the following categories: Sensor fusion (combining information from different sensors for use on board the vehicle), communications (handling communication and coordination between multiple UAVs/ground stations in the presence of incomplete and imperfect information), path/motion planning (determining an optimal path for the UAV while meeting certain objectives and constraints), trajectory generation or path following (determining an optimal control maneuver to follow a given path or to go from one location to another), task allocation and scheduling (determining the optimal distribution of tasks amongst a group of agents, with time and equipment constraints) and cooperative tactics (formulating an optimal sequence and spatial distribution of activities between agents in order to maximize chance of success in any given mission scenario).

When focusing on unmanned system autonomy, improved system autonomy requires, and will require new and innovative guidance and control techniques to fulfill mission requirements and to take decisions autonomously. A fundamental capability required to enable high-level command of autonomous systems is the ability for the system to be able translate high-level specifications into low-level descriptions of how the vehicle must move. The autonomous system must then plan a path which passes through the desired goal states, ideally in some optimal manner, and then follow that path. In both of these implementation approaches, the vehicle is required to follow a specified path. Autonomous vehicles must have sufficiently advanced path planning algorithms, combined with effective and robust guidance and automatic control systems. Successful control system design for high performance UAVs requires efficient and effective techniques for the design of guidance and control algorithms that ensure satisfactory operation in the face of system uncertainties and environmental disturbances. A key performance criterion for the guidance and control system is to have the ability of precise path following control in the presence of disturbing forces. A criterion for assessing the lateral performance and handling qualities for such vehicles is presented in (Capello et al., 2012).

In guiding UAVs, motion control problems for UAVs can be classified into two categories: path following and trajectory (reference) tracking. It is of primary importance to steer the UAV along a desired path. The speed or dynamic behavior along the path may be of secondary interest. Path-following problems are primarily concerned with the design of control laws that drive an object (UAV, mobile robot, ship, aircraft, etc.) to reach and follow a geometric path. On the other hand, the trajectory-tracking problem is concerned with the design of control laws that force a vehicle to reach and follow a time parameterized reference (i.e., a geometric path with an associated timing law). Typically, in path-following, smoother convergence to the path is achieved and the control signals are less likely to be pushed into saturation, as compared to trajectory-tracking. An example is to automatically drive a car along a road. This can be achieved by making the car track a point that moves along the road with a certain speed. However, by instead emphasizing that the main task is to make the car stay on and follow the road, one can let the desired speed be of secondary interest and sacrificed if necessary. In case of air vehicles, the primary objective is to maintain sufficient relative airspeed for lift requirement, and hence ground speed variation is obvious due to unpredictable nature of winds. Trajectory-tracking problem becomes more complicated in case of UAVs due to the varying ground speed, and hence path-following is a simple, efficient and secure way for motion control problems.

Motivated by these considerations, we propose a solution to the path following problem for unmanned vehicles in this thesis for both 2- and 3-D spaces. We are especially interested in situations for which there is parametric uncertainty in the model of the vehicle or there are external disturbances like winds. Inspired from the inherent nature of sliding mode theory to cater for parametric uncertainties and input disturbances, the proposed guidance schemes for path following are based on the sliding mode approach.

1.2 Literature Review

Two approaches are usually employed for ground track control of unmanned aerial vehicles. In the first approach, the 'guidance' and 'control' design problems are separated into an outer-loop guidance, and an inner-loop control design problem (Samar et al., 2008; Ratnoo, 2015; Rysdyk, 2006; Niculescu, 2001; Dadkhah and Mettler, 2012). In the second approach, the guidance and control problems are addressed together in an integrated and unified framework (Dadkhah and Mettler, 2012; Park, 2004; Yamasaki1 and Balakrishnan, 2010; Yu et al., 2011; Yamasaki et al., 2012). A good background to the integrated approach and associated challenges is presented in (Kaminer et al., 1998; Yamasakia et al., 2012). This approach is more complicated due to the coupling of the different guidance and control variables, and has not been as popular amongst the practising aerospace community. In most applications, the first approach is employed owing to its simplicity and intuitive appeal. The guidance law resides in the outer loop, uses position information (ground track and altitude measurements), and generates appropriate commands for the inner loop to follow. These commands can be in the form of required/reference roll angle (or reference lateral acceleration) and required/reference lift (or reference longitudinal acceleration) commands. The inner loop consists of a tracking controller that accepts the reference commands generated by the guidance loop, and performs control and stability augmentation tasks. Design of the outer loop guidance logic has been inspired by techniques as diverse as proportional navigation, cognitive and intelligent methods, vision based techniques, and others (Yamasaki1 and Balakrishnan, 2010; Cesetti et al., 2009; Siouris, 2004; Ren and Beard, 2004; Zhang et al., 2013; Oland and Kristiansen, 2014). The objective remains to drive the off-track track deviation of the vehicle to a minimum. The inner loop control design problem is beyond the scope of this thesis, numerous well established techniques exist for the (inner loop) control design problem, such as linear, nonlinear, robust and intelligent techniques (Skogestad and Postlethwaite, 2005). The reader may refer to any text on advanced control for further reading.

The computed path by mission planning is essentially defined in terms of a sequence of waypoints connected by straight lines or arcs through which the vehicle must traverse. Three dimensional (3-D) guidance refers to following the mission path in both the lateral (horizontal) and longitudinal (vertical) planes (Breivik and Fossen, 2005). It covers both 2-D ground track following, as well as altitude profile following of the desired mission. In the 3-D case, guidance commands are generated for the lateral/directional control system in terms of reference roll or heading angles, and for the longitudinal control system in terms of pitch or altitude commands. A subset of the general 3-D problem is the 2-D lateral guidance problem in which the guidance objective is to ensure accurate ground track following of the vehicle. This is to say that the vehicle must exactly fly over lines and arcs joining mission waypoints as projected on plane ground, with minimum cross-track or lateral deviation. It is relevant to point out that most of the results in literature only solve the problem in the horizontal plane. Only a few authors have tackled this problem in 3-D space. The existing algorithms on path-following can be broadly classified into two categories: geometrical, and classical control theory based. In this section, we will discuss the existing work in literature related to 2-D lateral and 3-D guidance problems. In recent years, a few survey papers have been published that cover the work done in this area in detail, the reader is referred to references (Sujit et al., 2014; Dadkhah and Mettler, 2012; Goerzen et al., 2009; Sujit et al., 2013).

Linear controllers based on classical control theory are commonly used for the guidance logic design of unmanned systems. Typically, the cross-track error as

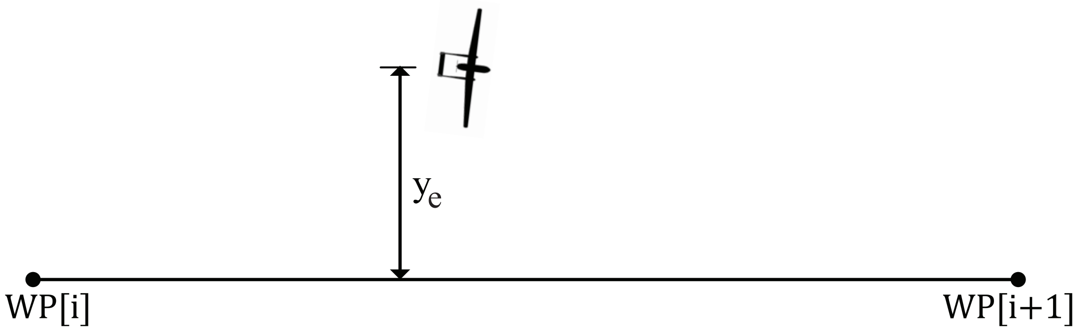


Figure 1.1: Traditional linear controller based guidance for Cross-track minimization.

shown in Figure 1.1 is used for feedback to generate the guidance commands, i.e., required lateral acceleration or bank angle. Linear proportional-derivative (PD) control for the outer guidance loop has been used in some UAV applications (Pappoullis, 1994; Siouris, 2004; Valavanis and Vachtsevanos, 2015). In the presence of a persistent disturbance (e.g., constant winds) however, the path following performance of the PD control degrades. Another difficulty associated with these linear controllers is switching decision logic when following a series of way-points for a smooth transition from one waypoint to the next. A conventional proportional-derivative lateral control scheme with some modifications for performance enhancement is discussed in (Samar et al., 2008, 2007); stability is however not formally proved. Mixed integer linear programming (MILP) based guidance for UAVs has also been considered (Mettler et al., 2003; Schouwenaars et al., 2005); however here the optimization program generates a sequence of waypoints (positions) and velocities for the vehicle to follow. In other words a mission plan is generated and deviations from this plan need to be corrected through a lower level guidance algorithm. Waypoint guidance schemes for marine vessels are suggested in (Kurowski, 2015; Pettersen and Lefeber, 2001; Brhaug et al., 2011); the qualities of a 'good helmsman' are discussed along with conditions for convergence in (Pettersen and Lefeber, 2001). In case of marine vehicles, the control problem is that of following the position and heading, using only the rudder input (Rysdyk, 2003). Guidance laws for UAVs following the 'good helmsman' concept are discussed in (Rysdyk, 2006, 2003), along with simulations of different scenarios.

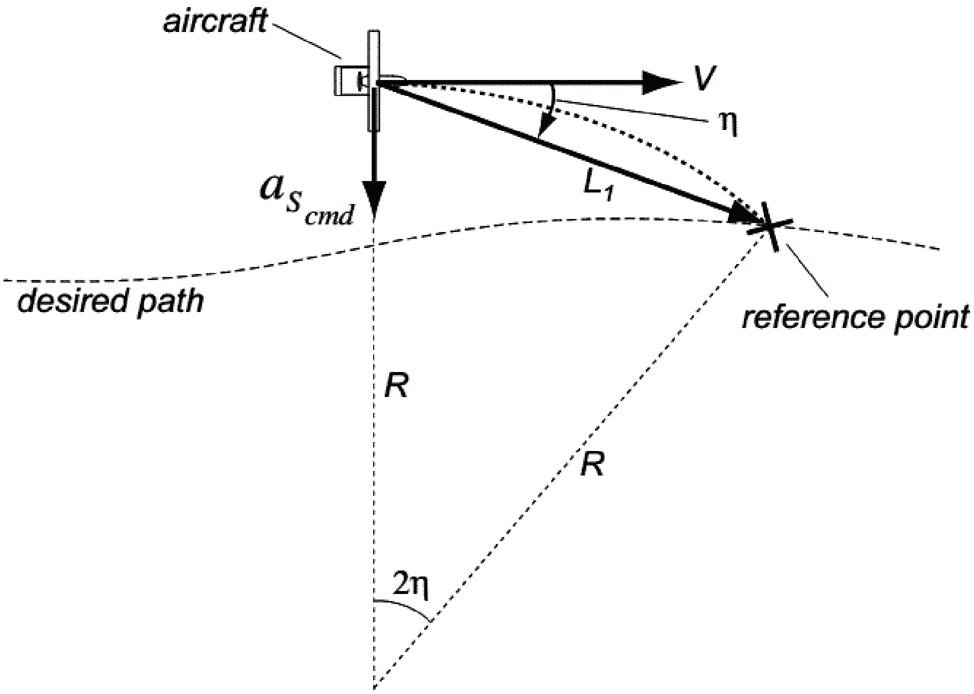


FIGURE 1.2: Nonlinear guidance law proposed by (Park et al., 2004).

A lateral track control law for small UAVs has been discussed in (Niculescu, 2001); this is based on a pure geometrical concept. The idea is to make the ratio of lateral deviation to lateral velocity equal to the ratio of longitudinal distance to longitudinal velocity. Simulation results indicate that the yaw-rate command generated by the guidance law exhibits oscillations in the vehicle roll channel, this could be a problem for implementation on a real vehicle. In reference (Park et al., 2004), a nonlinear guidance algorithm (see Figure 1.2) is suggested for the path following problem and its stability is discussed in detail in (Deyst et al., 2005). For curved as well as straight path following, the performance of the proposed algorithm is better than traditional PD logic. However, control input saturates in case of large track errors and therefore stability is not guaranteed in this case. A ‘reference point’ (ahead of the current position) on the desired path is also required for computation as shown in Figure 1.2; a discontinuity may arise if the desired mission gets changed online.

With reference to UAV formation flying, a guidance algorithm is presented in (Regina and Zanzi, 2011) for a follower UAV. For moving target following, the algorithm generates lateral acceleration commands as a reference for the control system to follow. A path following algorithm is proposed in (Sasongko et al., 2011) for a small UAV; the guidance output is in the form of heading angle corrections that is achieved by using bank-to-turn maneuvers. The inner roll control loop is designed by LQR (linear quadratic regulator) method. Effectiveness of the algorithm is demonstrated by simulation of different scenarios. Pythagorean Hodograph (PH) curve based generated trajectory generation and its tracking using back-stepping law is discussed in (Zhang Yi and Weiwei, 2015); effectiveness is shown using simulation results.

Crosswinds are a major source of disturbance for accurate lateral track control. The wind effect is ignored by most authors in design of guidance logic. UAV dynamics in the presence of winds is discussed in references (Sarras and Siguerdidjane, 2014; Osborne and Rysdyk, 2005; McGee and Hedrick, 2006; Ceccarelli et al., 2007); however the emphasis here is not on guidance but how to modify the planned mission to best cater for wind conditions. A composite controller that is able to compensate the effect of winds is discussed in (Liu et al., 2012); wind information is estimated first by a nonlinear observer and this is then incorporated in the path following controller. Robustness of these algorithms depends on the correct estimation of wind speed and direction, this is not straightforward specially in the presence of gusts. (Beard and McLain, 2012) discusses the effect of wind on guidance performance of UAVs and aircraft. In (Brezoescu et al., 2013) an adaptive backstepping based guidance law (considering the effect of winds) for UAVs is presented using a skid-to-turn concept. Waypoint Fuzzy Guidance Scheme (FGS)

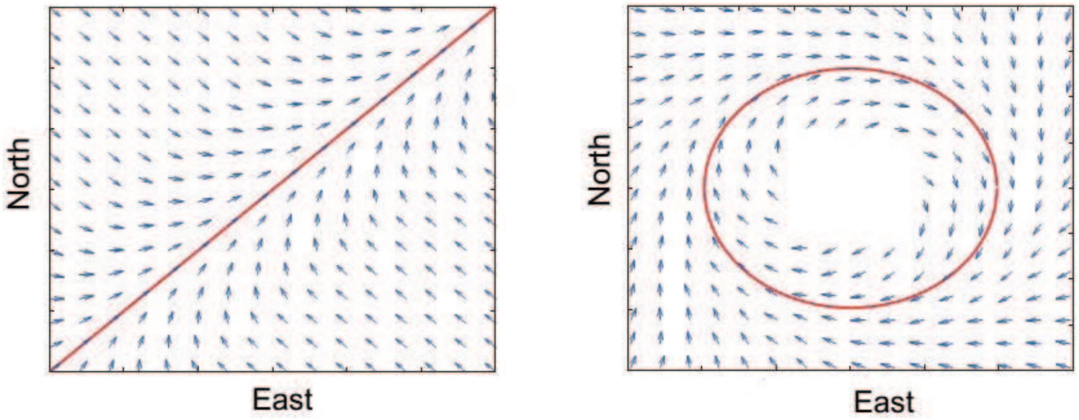


FIGURE 1.3: Vector field approach for path following problems.

is discussed in (Baralli et al., 2002), it is based on the three standard Takagi-Sugeno fuzzy controllers. The three controllers generate velocity, heading and flight path angle references for the autopilots; simulation results are presented to show the efficacy of the scheme. A sliding mode based guidance law for soft landing is discussed in (Zexu and etc., 2012), and working of the algorithm is demonstrated via numerical simulations. In (Shtessel et al., 2003) and (Shtessel et al., 2007), application of sliding mode control in the outer guidance loop is discussed.

A conceptually different guidance scheme employing vector fields as shown in Figure 1.3 for curved path following has also been pursued, see for example (Griffiths, 2006; Pisano et al., 2007; Lim et al., 2014; D. R. Nelson and Beard, 2007). In this case, a vector field indicating directions for course commands is computed as a function of vehicle deviation from the desired path. The difference between the computed and actual course angle is fed to the control algorithm for regulation. The computed course commands using the VF approach may give rise to sudden and large commands in some cases, that may be difficult for the control system to follow.

In recent years there has been a lot of interest in guidance laws based on model predictive control (MPC) and nonlinear model predictive control (NMPC) methods

(Rohan C. Shekhar and Shames, 2015). The computational and implementation complexity makes use of MPC based techniques quite challenging on-board UAVs, particulary low cost systems. A trajectory tracking algorithm for UAVs based on the nonlinear model predictive control (NMPC) approach is discussed in (De Filippis et al., 2012). The proposed algorithm tracks the path by solving an optimal control problem online using a genetic algorithm. A high level controller for small fixed-wing UAVs is designed in (Kang and Hedrick, 2006) using NMPC, minimization of the proposed cost function results in minimization of the cross track error. To avoid overshoot, knowledge of the entire mission is assumed beforehand, and the error is minimized N steps ahead. An adaptive NMPC based path tracking control for UAVs with a variable control horizon is discussed in (Yang et al., 2009), the objective here is to minimize the mean and maximum cross track error. High level control for UAVs with a non-quadratic cost function is discussed in (Poh et al., 2010); the solution is shown to approach the Dublins path under certain conditions. In (Garcia and Keshmiri, 2011), the authors discuss an integrated navigation, guidance and control algorithm using NMPC and suggest that 'MPC and NMPC applications to unmanned vehicles are more recent and there are still promissory advances to be made'. Model Predictive Control (MPC) laws to solve the trajectory-tracking problem and the path-following problem for constrained underactuated vehicles is discussed in (Alessandretti et al., 2013). Most of these works provide computer simulation results, real time implementation on actual hardware remains a challenging task.

The subject of guidance related to motion behaviour in 2-D plane and 3-D space is discussed in detail in (Breivik and Fossen, 2005). Path generation and tracking algorithm for UAVs in 3-D space is discussed in (Ambrosino et al., 2009). The nonlinear path tracking algorithm resembles the line-of-sight guidance algorithm and lacks robustness in the presence of parametric uncertainties. In (Kaminer et al., 2006), the authors address the problem of coordinated control of multiple

UAVs in 3-D space under tight spatial and temporal constraints. The path-following problem of steering an autonomous vehicle along a desired path while tracking a predefined velocity profile is discussed in (Cunha and Silvestre, 2005). Both these algorithms rely on accurate measurement of the velocity vector and any uncertainty in velocity is not addressed in the derivation. A backstepping based nonlinear guidance algorithm is proposed in (Ahmed and Subbarao, 2010) for 3-D path tracking of UAVs. The proposed guidance law assumes a very simple first order dynamics for heading and elevation angles, and the coupling between the two planes is ignored. A 3-D guidance law based on robust feedback linearization (RFL) concept for UAVs is discussed in (Chen et al., 2014), simulation results are shown for demonstration.

1.3 Contributions of the Dissertation

The contributions of this dissertation are summarized as follows:

- Sliding mode based guidance law design starts with the selection of sliding surfaces. A linear sliding surface (linear combination of states) is a natural choice for most applications. One contribution of this thesis is evaluation of linear sliding surfaces for sliding mode based lateral guidance for UAVs. It is shown in this work that linear sliding surfaces are not a viable option in this case, their stability issues are highlighted. As an adhoc solution, piece-wise linear sliding surfaces are proposed for the lateral guidance scheme for UAVs. These proposed sliding surfaces overcome the stability problems of linear sliding surfaces for the entire phase portrait.

- High performance non-linear sliding surfaces are proposed for lateral and longitudinal guidance of UAVs, and their stability is proved using Lyapunov theory. Nonlinear lateral sliding surfaces have two tunable parameters, it is shown that performance for small and large cross track errors can be achieved by adjustment of these parameters. Moreover, the proposed nonlinear sliding manifold fulfills the criterion of a good helmsman. An initial concept of these sliding surfaces is presented in the 'IFAC World Congress' and the 'UKACC Control Conference', and later published in detail in Journals.

- Another contribution of this thesis is the nonlinear sliding surface based lateral guidance law, and its validation through flight tests. The effect of winds is also analyzed and flight results in the presence of winds are presented to show the robustness of the proposed algorithms. The guidance scheme caters for measurement uncertainty in speed due to winds / sensor accuracy. To demonstrate its effectiveness, a test-platform (scaled YAK-54) is developed that employs a generic MPC-565 processor based flight control computer for programming of different guidance and control algorithms. The proposed algorithm is successfully demonstrated in various flights, results are published in 'IEEE Transactions on Control Systems Technology' and 'Transactions of the Institute of Measurement and Control'.
- For the 3-D path following problem, a generalized 3-D guidance scheme based on sliding mode theory is introduced taking into consideration lateral and vertical plane coupling effects. In this case, we have a multi-input multi-output (MIMO) system with two control variables. We therefore use two nonlinear sliding manifolds for guidance logic design. A paper on this topic is published in the 'International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering'.
- Another important contribution of this thesis is the optimization of non-linear sliding surfaces. The proposed sliding surfaces have tunable parameters that define how the UAV will come back onto the desired path. For selection of these sliding coefficients, work and energy principle based criterion is used, while taking into account the UAV maneuverability constraints. Using

numerical techniques, the sliding coefficients can be optimized so that they yield the shortest path without violating the constraints.

1.4 Dissertation Structure

This dissertation is organized as follows. Chapter 2 sets the foundation for the work by providing basic information about navigation, guidance & control structure for UAVs including basic software flow, important definitions and formulae. Additionally, basics about sliding mode theory is also discussed here and control design based on SMC is elaborated with the help of an example. For any control law design, the first step is the derivation of state equations and this is done in Chapter 3. Basic notations and guidance variables are introduced first, and kinematic models are derived for 2-D and 3-D guidance logic design. The chapter ends with an explicit definition of the guidance problem.

In Chapter 4, a lateral guidance law based on a linear sliding surface is derived, and its limitations are highlighted. Later, a piece-wise linear surface is introduced that overcomes the linear surface stability limitations; its effectiveness is demonstrated via simulation results. High performance nonlinear sliding surfaces for lateral and longitudinal planes are proposed in Chapter 5. Stability of these manifolds is proved using Lyapunov theory, followed by the criterion for optimized sliding parameters for both straight and circular paths. Chapter 6 describes the design of a nonlinear lateral guidance law based on nonlinear sliding manifold for straight and circular path following. Stability and control boundedness proofs for the proposed law are also discussed. The guidance law is implemented on a test aircraft and various flight tests carried out. Flight results show efficacy of the proposed law, these are also presented in this chapter. The generalized 3-D guidance law is derived in Chapter 7 and its effectiveness is shown using 6-dof nonlinear simulations.

Finally, in Chapter 8 we conclude our work and propose future work that can extend the usefulness of the techniques presented here.

Chapter 2

PRELIMINARIES

2.1 Introduction

As the basic theme of this dissertation is the application of sliding mode theory in the guidance of aerospace vehicles, there are some terms and mathematical tools of these two areas (SMC and guidance) that are used extensively within this context. Before proceeding to the problem formulation, fundamental knowledge of these areas is required. This chapter describes the basic concepts and background material related to the guidance & control of aerospace vehicles and sliding mode theory.

2.2 Guidance and Control of UAVs

The basic system architecture for unmanned aerial systems is shown in Figure 2.1. The first block in this architecture is the 'path planner' (also known as mission planner) that has input of maps, obstacles and destinations from the user. On the basis of this information the 'path planner' computes the mission plan in terms of way-points connected by straight lines or circular arcs, and this mission plan is forwarded to the 'path manager' for following. The basic responsibility of 'path manager' is switching between way-points and generation of an active straight / circular path that the vehicle should follow. Based on active path definition, the role of 'path following' (also known as guidance block) is to generate suitable commands (e.g., reference heading/roll angle, speed, altitude etc.) for following the desired mission. The autopilot (or control block) actuates the control surfaces / engine throttle to follow the reference commands generated by the guidance block. In simulation, the 'UAV' block is the mathematical model of the UAV. In actual implementation, the same procedure is followed, however the UAV means the actual flying vehicle. The role of sensors is to feedback the system states for generating the mentioned commands. However, if some state variables are not measurable then they may be estimated using observers.

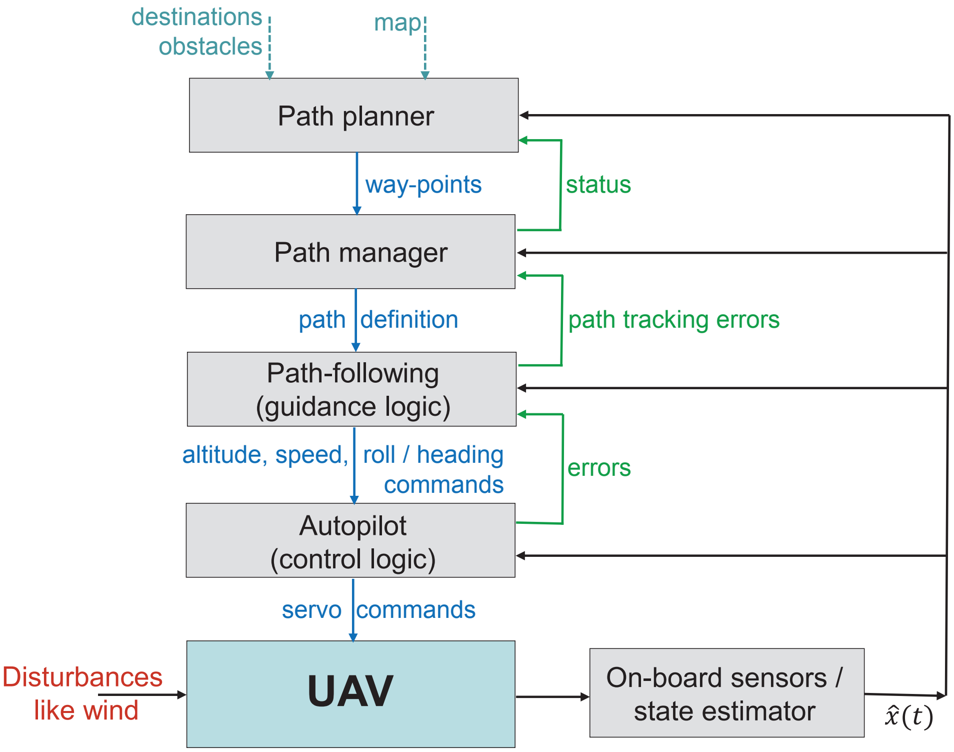


FIGURE 2.1: Software architecture for UAVs.

In aerospace systems, three terms 'Navigation', 'Guidance' and 'Control' are commonly used; these are defined below.

2.2.1 Navigation system

It is a sensor system that measures and outputs the vehicle's full state vector estimate, or its subset. The full state vector comprises of UAV position in space (3-components), velocity (3-components), attitude angles (ϕ, θ, ψ), airspeed (V_a), angle of attack, sideslip angle, and rotation rates (p, q, r). The objective of a navigation system is to find out the UAV position, velocity and attitude at any time; this is known as the navigation state (Gleason and Gebre-Egziabher, 2009; Valavanis and Vachtsevanos, 2015). Attitude determination is an integral part of the navigation system so that the specific measured force by the accelerometers can be transformed to the navigation frame in which the velocity and position are calculated. The sensors (inertial measurement unit (IMU) and global positioning system (GPS) receiver) used to estimate these variables are called navigation sensors. In the IMU, orthogonally-mounted gyroscopes and accelerometers are used for rotation rate and force measurement along the sensitive axes.

Inertial sensors exhibit biases and noise that lead to positional drift/errors over time (Mohamed and Mamatas, 2012). For better position accuracy, it is necessary for most applications using medium-grade and low-grade IMUs to employ aiding systems like the Global Navigation Satellite System (GNSS). By incorporating aiding systems, a dynamic model is used to predict error states in the navigation parameters, which are rendered observable through external measurements of position and velocity. The goal of the aiding system is therefore to help estimate the drifts and correct them. The simplest way to achieve this, of course, is to simply use the calculated positions and velocities from the GNSS directly in place of the results from the mechanization. This is the so-called reset filter, although from the standpoint of optimal filtering, it has many undesirable effects such as introducing sudden jumps in the navigation states. Moreover, the complementary filter places all the weight on the GNSS-derived values, which themselves are subject to error. Alternatively, Kalman filtering is used to optimally estimate the error states of the INS, with updates coming from the GNSS in one of several architectures: loosely-coupled integration, tightly-coupled integration or deeply-coupled integration.

In the last decade, inertial navigation systems using MEMS sensors have been a subject of great interest. Automotive-grade Micro-Electro-Mechanical System (MEMS) based navigation system is a six-degree of freedom inertial measurement unit consisting of three orthogonally aligned accelerometers and three orthogonally aligned gyroscopes confined in a very small chip. Due to its small size, low cost, light weight, low power consumption and ruggedness, MEMS based navigation systems are commonly used in small unmanned aerial vehicles (UAVs). The INS provides high relative accuracy but the absolute accuracy deteriorates with time if the system runs in stand-alone mode and if no external measurement aids are available. The large errors in position, velocity, and attitude come about mainly due to sensor bias and noise. GPS serves as an aiding device to tune the MEMS inertial sensor's measurements through a Kalman filter, while extra sensory feedbacks from magnetometers are employed to improve the system performance. Such an integrated system is known as the GPS aided MEMS Inertial Navigation System. Although there are specialized GPS receivers that can generate data up to 100 Hz, most low-cost GPS receivers provide output data at 1-10 Hz update.

2.2.2 Guidance

According to Wikipedia definition,"A guidance system is a virtual or physical device, or a group of devices implementing a guidance process used for controlling the movement of a ship, aircraft, missile, rocket, satellite, or any other moving object". One other definition is: "Guidance is the process of calculating the changes in position, velocity, attitude, and/or rotation rates of a moving object required to follow a certain trajectory and/or attitude profile based on information about the object's state of motion". Guidance is the "driver" of a vehicle. It takes input from the navigation system (position, speed, direction, etc) and uses targeting/path information (where do I want to go) to send signals to the flight control system that will allow the vehicle to reach its destination (within the operating constraints of the vehicle).

In general, the output of the navigation system, the navigation solution, is an input for the guidance system, and the guidance system computes the instructions for the control system. Some environmental conditions (gravity, density, etc) and the vehicle's characteristics (mass, inertia, etc) are known and available to the guidance system. However, variation in these parameters and other inputs like winds act as disturbances for the guidance system. A guidance system has three major parts: Inputs, Processing, and Outputs. The input section includes sensors, course data, radio and satellite links, and other information sources. The processing section integrates this data and determines what actions, if any, are necessary to maintain or achieve a proper heading. This is then fed to the outputs which can directly affect the system's course. The outputs may control speed by interacting with devices such as engine, or they may more directly alter course by actuating ailerons, rudders, or other devices.

In manned aircraft (flying in manual mode), the pilot does the job of the 'Guidance system'. Using visual and sensor displayed information, the pilot issues reference commands to keep the aircraft on the desired path. For example, he may issue pitch and roll reference commands by moving the stick to keep the aircraft on the desired altitude and ground path. Similarly, he may generate reference commands for the engine by moving the stick to achieve the desired speed. In unmanned systems, all these functions are performed by 'guidance system'; it generates suitable reference commands for the inner control loop.

2.2.3 Flight control

Aerospace vehicle control is defined as "the methods and/or equipment used for correlating the command signals received from the guidance system with the signals received from its attitude stabilization sensors to compute and command actuation of the servos (aerodynamic surfaces, engine throttle etc)". A flight control system is required to ensure that the UAV exhibits stable behavior and delivers desired performance, such as following a desired trajectory with sufficient accuracy in the presence of external disturbances. Thus, a UAV flight control system is typically safety/mission critical, as an incorrect design, lack of robustness to model variations, or a component failure could result in poor performance (e.g., payload pointing capability), or even loss of the vehicle. Using guidance commands and measured vehicle states, the flight control system performs speed control, height and height rate control, bank angle control, heading control, turn compensation and elevation control.

The goal of the control system is to track the desired reference command in presence of external disturbances. The control system achieves this by attempting to eliminate the tracking error between the reference commands and the measured response of the UAV. The control is typically achieved by using estimates of the UAV states, which include position, velocity, angular rate, and attitude of the UAV. These estimates are obtained by the navigation system that fuses noisy measurements from several sensors. A control system attempts to minimize the tracking error between the desired and estimated states in the presence of disturbances.

Sometimes the control loops are nested in an inner-loop outer-loop configuration; this leads to some challenges. The general rule of thumb is to ensure that the inner control loops has "fast" dynamics than the outer loop, and each successive loop added is "slower than the previous one". The primary difficulties here are to define what is meant by fast and slow, and how to determine if there is too much interaction between the loops being closed (e.g., closing the outer loop might reduce the performance of the inner loop, requiring a redesign). The flight control generates servomotor signals for the rudder, elevator, aileron and engine throttle. A servomotor is a compact electromechanical device consisting of a DC motor with a built-in feedback circuit. These servomotors accept pulse-width modulated (PWM) signals as inputs.

2.2.4 The mission plan

The desired mission generated by the ‘path planner’ consists of a number of waypoints connected in series which define the path to be followed. A typical mission profile includes series interconnection of two consecutive waypoints via straight lines or arcs, and there are circles (for loiter). The desired mission is generally planned off-line and can be modified on-line anytime. Path planning is not in the scope of this thesis and it is assumed here that the desired waypoint information is available for the path following algorithm; the main objective of this section is to discuss practical aspects related to guidance and control to follow a given mission.

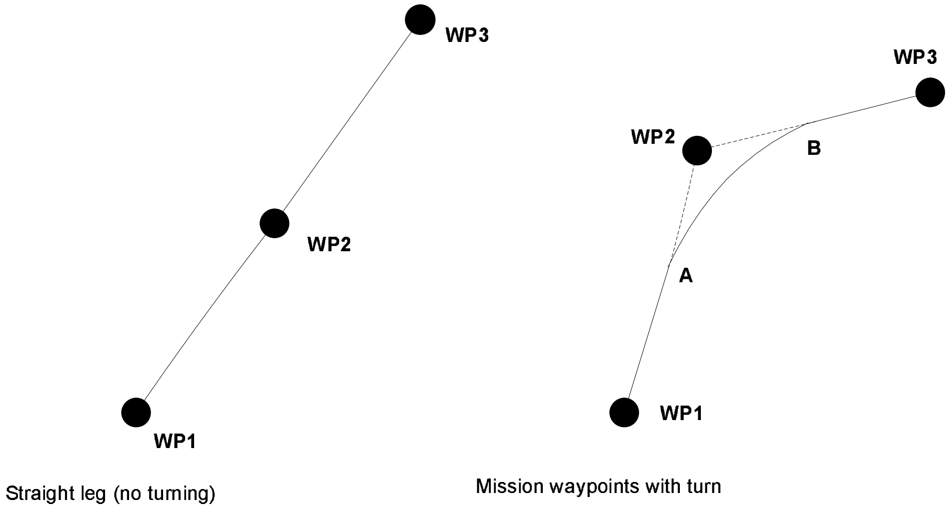


FIGURE 2.2: Waypoints for straight and turning flight.

Figure 2.2 describes two scenarios of a typical mission: a straight line segment, and waypoints with a turn. On the left hand side of the figure the three waypoints lie on a straight line, the UAV has to fly over the central waypoint $WP2$. A scenario of a turn is shown on the right side of Figure 2.2. In this case a sharp turn at the central waypoint ($WP2$) is beyond the capability of the vehicle, and hence a circular arc AB (close to $WP2$) is introduced. The turn starts at point A and ends at point B. Similarly loitering around a given waypoint is also possible

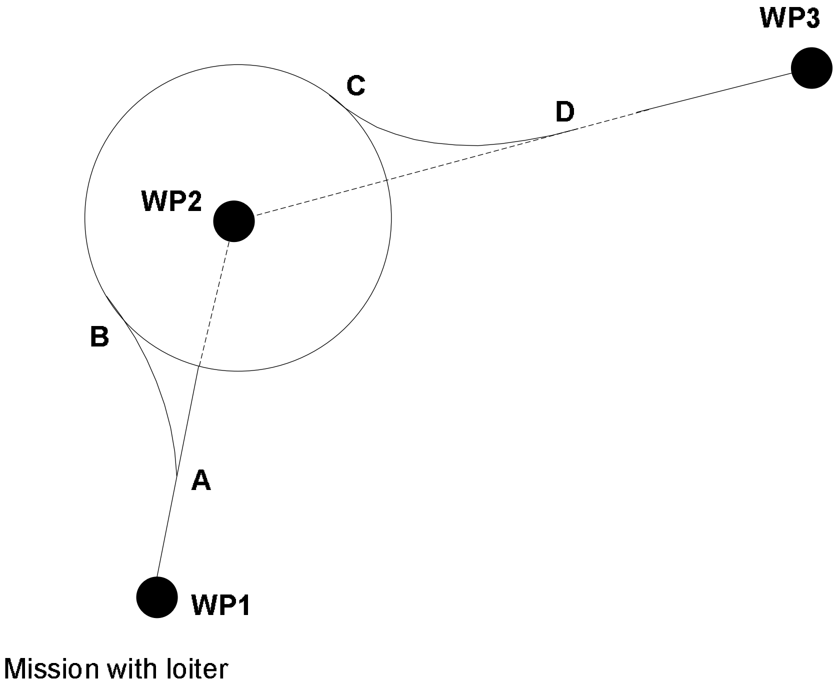


Figure 2.3: Mission with loiter about WP2.

as shown in Figure 2.3. Two additional arcs AB and CD are introduced here for smooth transitioning between the straight segments and the circular loiter. Waypoint generation is beyond the scope of this study and we assume here that the desired mission plan in terms of waypoints is available for the path following algorithm.

We now formulate the problem in more specific terms. The task of the path manager (for the lateral guidance and control system) is defined as to:

- Compute the down-range covered from one waypoint to the next, and the cross-track deviation from the designated track/trajectory,
- Compute the error in heading angle (or the course angle) of the vehicle,
- Determine if a waypoint is achieved,
- Compute ‘turn start’ and ‘turn stop’ decision flags for the autopilot,

- Solve the guidance equations to calculate the roll reference command for steering the vehicle back onto the desired path, and
- Stabilize and control the lateral-directional dynamics of the vehicle across the flight envelope in the presence of disturbances.

All the above computations have to be done in real time. The control inputs available to the vehicle for lateral control are the ailerons and the rudder. Note that the cross-track deviation and heading error (or the course angle error) are used in the guidance equations.

2.2.5 Flight path computations

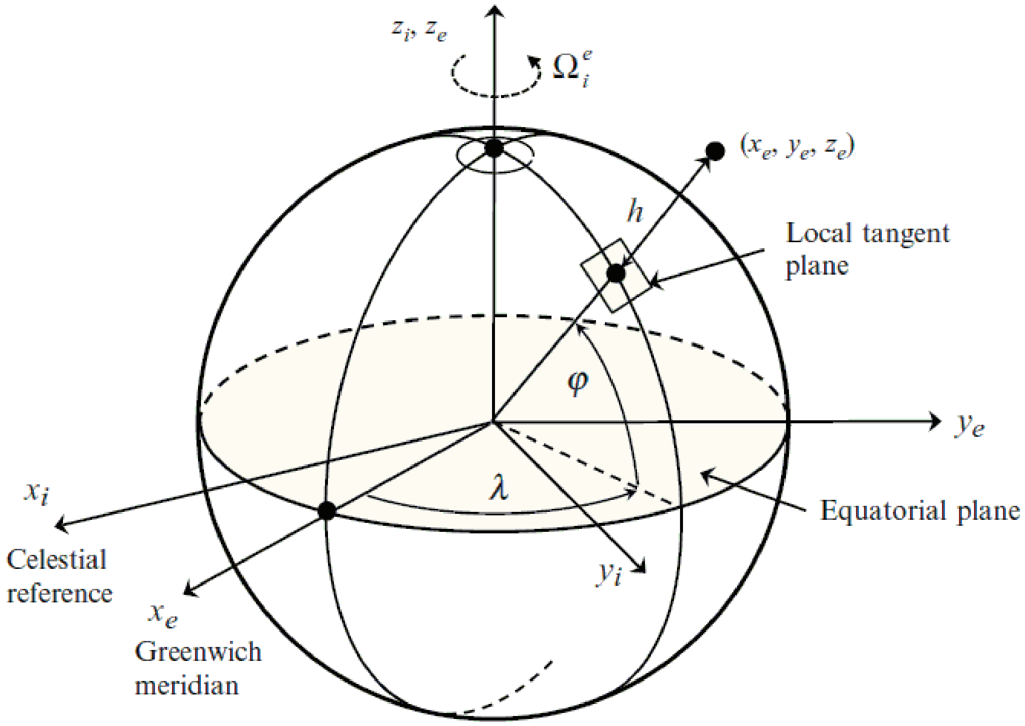


Figure 2.4: Earth fixed reference frames (Valavanis and Vachtsevanos, 2015).

After availability of the mission plan, the next step is to navigate the UAV through the desired waypoints. In order to accomplish this task, some basic computations are performed in real time which are discussed here. The lateral waypoints are generally defined in terms of latitude φ and longitude λ. The first step is to identify turning waypoints and this is done be checking the azimuth difference of $WP1 - WP2$ and $WP2 - WP3$ for any three consecutive waypoints. If the turn angle (difference between azimuths of $WP1 - WP2$ and $WP2 - WP3$) is small, then $WP2$ is referred to as a *straight* waypoint and the segment from $WP1$ to $WP2$ is called a straight line. In case the turn angle is significant (say $> 5°$), the central waypoint $WP2$ is marked as a *turning* waypoint; in this case the complete path from $WP1$ to $WP3$ includes two straight segments and a curved part as shown in Figure 2.2. This process is then repeated for the next three consecutive waypoints $WP2$, $WP3$ and $WP4$, and so on. The other flight path computations are worked out separately for straight and turning parts of the flight.

The algorithm described here computes the lateral deviation (cross-track error) from the desired path as well as the heading error (or the course angle error) for straight and circular paths. The lateral cross-track and course (or heading) errors are fed to the lateral guidance scheme to generate suitable roll commands for accurate path following. Some additional decision flags like 'turn start', 'turn stop' and 'waypoint achieved' are also computed here for the mission manager and the autopilot.

Let $WP1(\varphi_1, \lambda_1)$, $WP2(\varphi_2, \lambda_2)$ and $WP3(\varphi_3, \lambda_3)$ be three consecutive waypoints (φ and λ denote the latitude and longitude, respectively), and let $M(\varphi_M, \lambda_M)$ be the current position of the vehicle. We define an Earth-Centred Earth-Fixed (ECEF) frame (see Figure 2.4) with its origin at the centre of mass of the earth, the z-axis directed northward along the polar axis, the x-axis in the equatorial plane and passing through the Greenwich Meridian, and the y-axis also in the equatorial plane and passing through 90° east longitude. The position vector of a point on the earth's surface with latitude φ and longitude λ in ECEF frame is given by:

$$r_x = a \cos\varphi_r \cos\lambda, \quad r_y = a \cos\varphi_r \sin\lambda, \quad r_z = b \sin\varphi_r \tag{2.1}$$

where a is the earth's equatorial radius, b is the polar radius, and $\varphi_r = \tan^{-1}\left(\frac{b}{a}\tan\varphi\right)$ is the reduced latitude of the point under consideration.

2.2.5.1 Straight path formulae

Distance covered:

Here we consider the straight path from $WP1$ to $WP2$ (Figure 2.5). Let $\overrightarrow{r_1}$ and $\overrightarrow{r_2}$ be the position vectors of $WP1$ and $WP2$ in ECEF frame, respectively. The distance between $WP1$ and $WP2$ can be calculated as:

$$r_{12} = \frac{1}{2}(|\overrightarrow{r_1}| + |\overrightarrow{r_2}|)\cos^{-1}(\hat{r}_1 \cdot \hat{r}_2),$$

where $\hat{r}$ is a unit vector along $\overrightarrow{r}$.

Cross-track deviation:

Cross-track error is the instantaneous normal displacement of the vehicle from the desired track. To compute the cross-track error we first define a vector $\overrightarrow{r_N}$ normal to the trajectory plane (plane containing $WP1$, $WP2$ and the centre of the earth) as:

$$\overrightarrow{r_N} = \hat{r}_1 \times \hat{r}_2.$$

If $\overrightarrow{r_M}$ is the position vector of the vehicle in ECEF frame, then the cross-track error is:

$$\text{Cross-track} = -\overrightarrow{r_M} \cdot \hat{r}_N$$

.

Course error:

Let V_E denote the east velocity and V_N the north velocity of the vehicle, then the

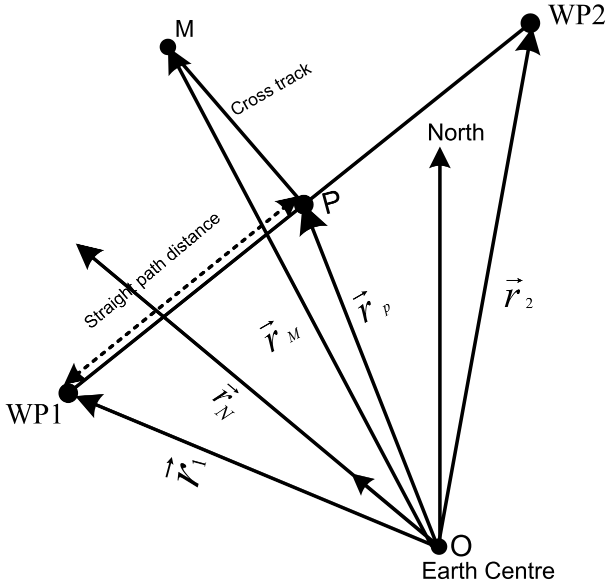


FIGURE 2.5: Straight flight path definitions.

velocity heading of the vehicle is given by $\chi = \tan^{-1}(\frac{V_E}{V_N})$. The course error is the difference between the actual course angle χ and the desired course angle χ_r, i.e., $\chi_e = \chi - \chi_r$.

If P is the projection of the vehicle's current position on the trajectory plane (Figure 2.5), then the desired course angle (χ_r) is defined as the angle between the trajectory plane and the meridian plane passing through the projected point P. The position vector $\overrightarrow{r_P}$ of P in ECEF frame is given by:

$$\overrightarrow{r_P} = \overrightarrow{r_M} - (\overrightarrow{r_M} \cdot \hat{r}_N)\hat{r}_N.$$

The desired course angle χ_r works out to be:

$$\chi_r = \tan^{-1}\left[\frac{\left|(\hat{r}_P \times \hat{k}) \times \hat{r}_N\right|}{(\hat{r}_P \times \hat{k}) \cdot \hat{r}_N}\right], \tag{2.2}$$

where $\hat{k}$ is a unit vector along the earth's spin axis. The course error χ_e can now be readily computed.

Waypoint achieved criterion:

The straight waypoint $WP2$ is achieved, when the straight path distance covered exceeds r_{12}. The straight distance covered from $WP1$ to the current position of the vehicle is $\frac{1}{2}\left(|\overrightarrow{r_1}| + |\overrightarrow{r_P}|\right)\cos^{-1}(\hat{r}_1 \cdot \hat{r}_P)$.

2.2.5.2 Curved path formulae

For turning paths (Figure 2.6), we define another coordinate system (the East-North-Up or ENU frame) at the central waypoint $WP2$. The origin of the ENU frame is at $WP2$, the positive x-axis points towards east, the y-axis points towards north, and the z-axis points up along the local vertical. The position vector of a point on the earth's surface with latitude φ and longitude λ can be expressed in the ENU frame as:

$$x = R_N(\lambda - \lambda_2)\cos\varphi_2, \quad y = R_M(\varphi - \varphi_2), \quad z = 0$$

where R_N and R_M are the normal and meridian radii of curvature (of the earth's ellipsoidal model), respectively. These are given by:

$$\begin{aligned} R_N &= \frac{a}{(1 - e^2 \sin\varphi_2)^{\frac{1}{2}}} \\ R_M &= \frac{a(1 - e^2)}{(1 - e^2 \sin\varphi_2)^{\frac{3}{2}}} \end{aligned} \tag{2.3}$$

where e denotes the eccentricity of the ellipsoidal earth:

$$e = \frac{\sqrt{a^2 - b^2}}{a}.$$

Let $\overrightarrow{q_1}$ and $\overrightarrow{q_2}$ be the position vectors of $WP1$ and $WP3$ in the ENU frame, i.e.,

$$\begin{aligned}\overrightarrow{q_1} &= \begin{bmatrix} R_N(\lambda_1 - \lambda_2)\cos\varphi_2 & R_M(\varphi_1 - \varphi_2) & 0 \end{bmatrix} \\ \overrightarrow{q_2} &= \begin{bmatrix} R_N(\lambda_3 - \lambda_2)\cos\varphi_2 & R_M(\varphi_3 - \varphi_2) & 0 \end{bmatrix} \end{aligned} \tag{2.4}$$

The turn angle β ($\leq 180°$) is given by:

$$\beta = \cos^{-1}(\hat{q}_1 \cdot \hat{q}_2).$$

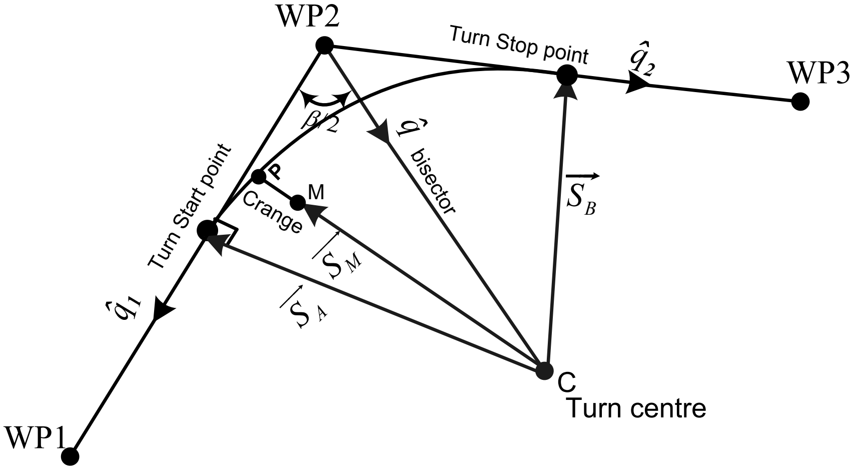


FIGURE 2.6: Flight path with turn.

Turn centre

The turn centre will be at a distance of $R/\sin(\beta/2)$ from $WP2$ on the line bisecting the turn angle β, R being the turn radius. The vector $\overrightarrow{q}$ along this line in the ENU frame is given by $\overrightarrow{q} = \hat{q}_1 + \hat{q}_2$. Thus the position of the turn centre C in the ENU frame is:

$$\overrightarrow{C} = R/\sin(\beta/2)\hat{q}.$$

The projection of the vehicle on the earth's surface (φ_M, λ_M) in the ENU frame is given as:

$$\overrightarrow{M} = \begin{bmatrix} R_N(\lambda_M - \lambda_2)\cos\varphi_2 & R_M(\varphi_M - \varphi_2) & 0 \end{bmatrix},$$

and the position vector of the vehicle's projection w.r.t. the turn centre C is $\overrightarrow{S_M} = \overrightarrow{M} - \overrightarrow{C}$.

Turn start criterion

The turn start point is at a distance of $R/\tan(\beta/2)$ from $WP2$ on the line joining $WP1$ and $WP2$. Its position vector $\overrightarrow{A}$ in the ENU frame is $\overrightarrow{A} = R/\tan(\beta/2)\hat{q}_1$, and its position w.r.t. the turn centre C is $\overrightarrow{S_A} = \overrightarrow{A} - \overrightarrow{C}$. Turning is started when the cross-product $\overrightarrow{S_M} \times \overrightarrow{S_A}$ changes sign. During turn the total distance (down-track) from $WP1$ to the current position is computed as:

$$\text{Distance from } WP1 = r_{12} - R/\tan(\beta/2) + R\cos^{-1}(\hat{S}_A \cdot \hat{S}_M).$$

Cross-track deviation

While turning the cross-track deviation is given by $R - |\overrightarrow{S_M}|$.

Course error

During turn the desired course angle (χ_r) is given by $\chi_r = \chi_s + \chi_T$, where χ_s is the desired heading at the turn start point (computed using equation (2.2) at the turn start point), and χ_T is the angular displacement from the turn start point to the current position:

$$\chi_T = \cos^{-1}(\hat{S}_A \cdot \hat{S}_M)$$

.

Waypoint achieved criterion

The waypoint $WP2$ is said to be achieved when the vehicle crosses the bisector line, i.e., when the vector cross-product $\overrightarrow{S_M} \times \overrightarrow{C}$ changes sign.

Turn stop criterion

The turn stop point is at a distance of $R/\tan(\beta/2)$ from $WP2$ on the line joining $WP2$ and $WP3$. Its position vector $\overrightarrow{B}$ in the ENU frame is $\overrightarrow{B} = R/\tan(\beta/2)\hat{q}_2$, and its position w.r.t. the turn centre C is $\overrightarrow{S_B} = \overrightarrow{B} - \overrightarrow{C}$. Turning is stopped when $\overrightarrow{S_M} \times \overrightarrow{S_B}$ changes sign.

2.3 Sliding Mode Control

Sliding mode control (SMC), a technique derived from variable structure systems (VSS), was originated in the former USSR in the 1950's, and gained appreciation in the wider analytical controls community after the historic paper by Utkin in English (Utkin, 1977). SMC has gathered great popularity amongst the nonlinear control community in recent times and has played a remarkable role not only in theoretical developments but also in practical applications (Edwards and Spurgeon, 1998; Utkin, 1977; Slotine and Li, 1991; Durmaz et al., 2012). This increased interest in SMC is mainly due to the fact that robustness has become a major requirement in modern control applications; and due to its inherent robustness property, SMC provides desired performance even in the presence of parametric uncertainties and external disturbances. The major advantage of sliding mode control is low sensitivity to plant parameter variations and disturbances which eliminates the necessity of exact modeling.

In this section, we cover the basic terminologies of SMC, the procedure of SMC design, and finally an illustrative example.

2.3.1 Basics of sliding mode control

In the formulation of any practical control problem, there will always be a discrepancy between the actual plant and its mathematical model used for controller design. Imprecision in the mathematical model of a system may come from simplified representation of the system dynamics (known as unstructured/unmodeled uncertainties), or from actual uncertainty about plant parameters (structured or parametric uncertainties) (Slotine and Li, 1991). Designing control laws that provide desired performance to the closed-loop system in the presence of disturbances and uncertainties is in general a very challenging task. Modeling inaccuracies can have strong and adverse effects on nonlinear control systems and may even lead to instability. One nice approach to dealing with model uncertainty is robust control. The typical structure of a robust controller is composed of a nominal part, similar to a feedback control law, and additional terms aimed at dealing with model uncertainty. The sliding mode control technique is one approach to robust control design. For a particular class of systems, sliding mode control provides a systematic approach to the problem of maintaining stability and consistent performance in the face of modeling imprecision. Also by allowing tradeoffs between modeling and performance to be quantified in a straightforward fashion, it can illuminate the whole design process.

Sliding mode control is a high-speed switching control in which the gains switch between two values according to some rule. A high speed switching control law is employed to drive the nonlinear plant's state trajectory onto a specified and user-chosen surface in the state space (called the switching or sliding surface), and to maintain the trajectory on this surface for all subsequent time. In order to maintain the trajectory on the surface (sliding motion), the control input is switched at very high frequency between a positive high magnitude and a negative high magnitude (a phenomenon known as chattering) (Young et al., 1999; Boiko, 2013; Burton and Zinober, 1986). While sliding, the plant dynamics is restricted (or forced) to lie on the sliding surface; the performance of the controller depends on the design of the surface (Slotine and Li, 1991). The idea is to force the trajectory of the system towards the sliding surface in the reaching phase, and once achieved, the system states must be constrained to remain on the surface. Sliding mode systems operate in two main phases during runtime:

- Reaching phase: the phase during which the system state trajectory is driven from any initial condition to the sliding surface (in a finite time).
- Sliding phase: the phase during which the system is forced to slide along the sliding manifold and reach the origin, i.e., the switching surface becomes an attractor.

These two phases in turn give rise to a two-step design strategy:

- Sliding surface selection: a switching (or sliding) manifold is selected with prescribed dynamical characteristics. Common candidates are linear hyperplanes.
- Discontinuous control design: a discontinuous control strategy is formulated to ensure finite time reachability and sliding on the switching manifold.

2.3.2 Sliding surface design

Sliding mode or sliding motion may be defined as “the evolution of the state trajectory of a system confined to a specified non-trivial sub-manifold of the state space with stable dynamics”. According to another definition, “Selection of stable hyperplane(s) in the state/error space on which motion should be restricted, is called the switching function (or sliding surface)”. From a geometrical point of view, the equation $s = 0$ defines a surface in the error space, that is called the “sliding surface” (also known as the sliding manifold, switching surface, discontinuous surface). Once sliding occurs, robustness to a certain type of uncertainty is guaranteed and the system behaves as a reduced order system independent of the control. The sliding surface decides the dynamics, therefore it should be designed such that it addresses all requirements. In general, the design of a sliding mode controller (SMC) involves the determination of a sliding surface that represents desired stable dynamics for the closed-loop system.

The trajectories of the controlled system are forced onto the sliding surface; the sliding surface dynamics should meet the design specifications. The trajectories starting from a given initial condition should be guided onto the sliding surface. This is known as the reaching or hitting phase and the system is sensitive to parametric variations in this part of the phase trajectory. When convergence to the sliding surface occurs, the sliding phase starts. The trajectories in this phase are insensitive to parametric variations and disturbances (Utkin and Shi, 1999; Young et al., 1999). Various methods have been suggested to eliminate or reduce the system sensitivity by minimizing or removing the reaching phase (C. and Y., 2000; Choi S.-B. and S., 1994; M.L. and G., 2007). Conclusively with SMC we can say that:

- The system behaves as a reduced order system which (apparently) does not depend on the control signal $u(t)$. Suppose we have a SISO second order system with two state variables x_1 and x_2, if we choose a linear sliding surface $s = x_2 + \lambda x_1 = 0$ then the system dynamics during sliding is given by the (reduced) first order differential equation $s = 0$.
- The equation of motion of the sliding mode can be designed linear and homogenous, even if the original system is governed by non-linear equations.
- Once the sliding motion starts, the system has invariant properties which makes the motion independent of certain system parametric variations and disturbances. Thus the system performance can be completely determined by the dynamics of the sliding manifold.
- The closed-loop sliding motion depends only on the choice of the sliding surface. The sliding mode does not depend on the process dynamics, but is determined by parameters selected by the designer.

After switching surface design, the next important aspect of sliding mode control is guaranteeing the existence of a sliding mode. A sliding mode exists, if in the vicinity of the switching surface, $s = 0$, the velocity vectors of the state trajectory are always directed towards the switching surface, this is discussed in the following subsection.

2.3.3 Control law design

The objective of the sliding mode is to ensure sliding motion in finite time, starting from any point in the phase portrait. The second important step is design of a switching control that will drive the system's state trajectory towards the sliding surface, and upon interception maintain sliding motion on it for all subsequent time. During this phase, if the plant's state trajectory is just above the surface, the feedback path has one gain, and if the state trajectory drops below the surface, a different gain is chosen. The following reaching laws have been proposed in the literature (Gao and Hung, 1993):

- The constant rate reaching law:

$$\dot{s} = -K \operatorname{sgn}(s)$$

- The constant plus proportional rate reaching law:

$$\dot{s} = -K_1 s - K_2 \operatorname{sgn}(s)$$

- The power rate reaching law:

$$\dot{s} = -K|s|^{\alpha} \operatorname{sgn}(s)$$

Equivalent Control

From a geometrical point of view, equivalent control means replacement of the discontinuous control term by a continuous control in the intersection of the

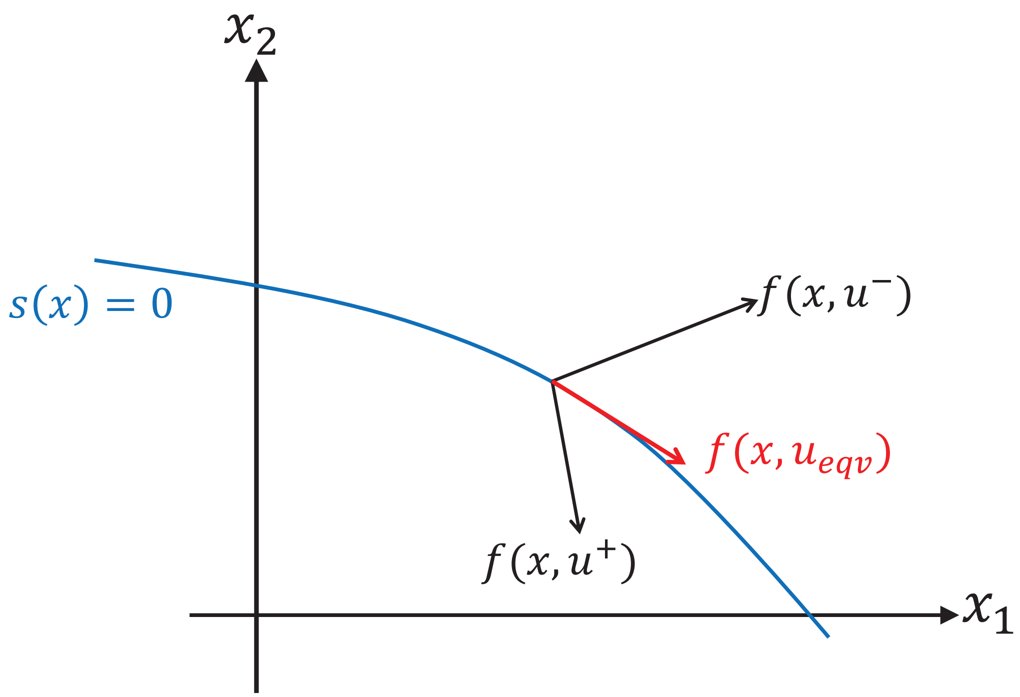


FIGURE 2.7: Filippov's construction of the equivalent dynamics.

switching surface such that the state velocity lies tangential to the sliding manifold, as shown in Figure 2.7. Using discontinuous control u^+ and u^-, the state trajectory is not tangential to the sliding surface. The system's motion on the sliding surface is basically an 'average' of the system dynamics on both sides of the surface. The equation $\dot{s} = 0$ represents the dynamics when in sliding. By solving the equation $\dot{s} = 0$ for the control input, we obtain an expression for u called the *equivalent control* ($u_{equivalent}$).

In sliding mode literature, $u(t)$ is said to be designed in order to satisfy the reachability condition. The reachability condition specifies that the system state trajectory must always point towards the sliding surface. For the case of a single input system, this can be expressed as:

$$\lim_{s \to 0^+} \dot{s} < 0, \qquad \lim_{s \to 0^-} \dot{s} > 0, \tag{2.5}$$

or more compactly as:

$$s\dot{s} < 0. \tag{2.6}$$

This is often referred to as the *reachability condition* (Utkin and Shi, 1999). A more strict reachability condition to ensure that the control law $u(t)$ is designed so that the sliding surface is reached despite the presence of uncertainties, and in finite time is given by:

$$s\dot{s} < -\eta|s|, \tag{2.7}$$

where η is a positive scalar. Equation (2.7) is called the 'η reachability condition'.

In order to guarantee desired behavior of the closed-loop system, the sliding mode controller requires an infinitely fast switching mechanism. However, due to physical limitations in real-world systems, directly applying such a control is not possible, hence approximation to the infinitely fast switching will lead to oscillations in vicinity of the sliding surface: the so called chattering problem. The main limitations come from implementation of controllers in digital computers which work on discrete sample times and cannot allow infinitely fast switching, and from physical actuators. Different approaches have been proposed to avoid chattering, such as boundary layer approximation (Young et al., 1999; Edwards and Spurgeon, 1998), higher order sliding modes (Levant, 1993) and adaptive sliding mode control (Shtessel et al., 2003). Saturation and sigmoid functions are also used as 'filters' to smoothen the discontinuous control signal so that it is realizable by mechanical hardware, interested readers are referred to (Young et al., 1999) for a detailed discussion.

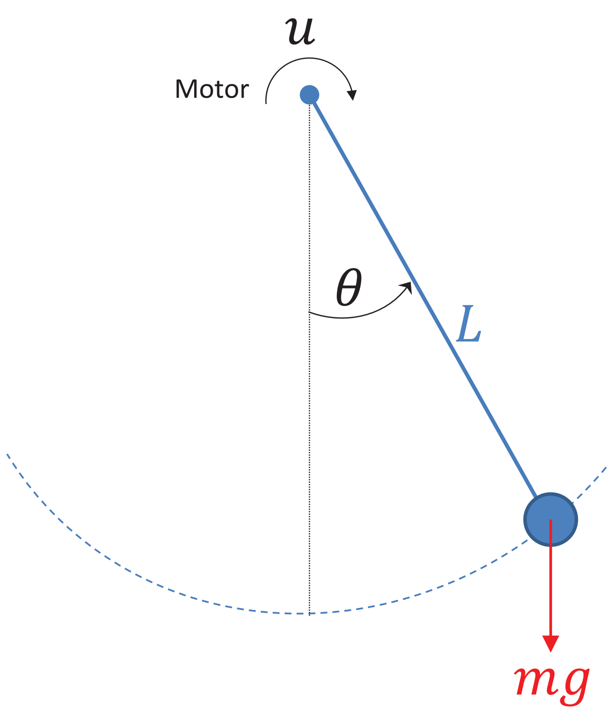


FIGURE 2.8: A simple pendulum system.

2.3.4 An illustrative example

The motion control of the simple pendulum system has been used by many researchers to demonstrate their ideas. To give some insight into the above mentioned sliding mode based principles, here we will apply them to the simple pendulum problem. A schematic of the system is shown in Figure 2.8 consisting of a (weightless) shaft and a mass which is driven by a motor (torque) at the point of suspension. A simple pendulum system is a mechanical system that exhibits periodic motion in the absence of any control action. The objective of control law design is to bring the pendulum to its vertically downwards equilibrium point when released from any initial position.

The equation of motion is given by:

$$mL^2\ddot{\theta} = -mg\sin(\theta)L + u, \tag{2.8}$$

where m is mass of the bob, g is the gravitational acceleration and L is length of the rod. In this second order differential equation, the control variable is denoted by u and the state variables are θ and it its rate of change. Denoting the two state variables θ (angular displacement) and $\dot{\theta}$ (angular velocity) by x_1 and x_2, respectively, we have the following state equations:

$$\left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{mL^2}\left(-mg\sin(x_1)L + u\right). \end{aligned}\right\} \tag{2.9}$$

In sliding mode based control law design, the first step is selection of a stable sliding manifold. A common choice is the linear combination of state variables as discussed earlier. In this case, we select the following linear sliding surface:

$$s = x_2 + c_1 x_1 = 0, \tag{2.10}$$

where c_1 is a real positive constant. To prove stability of the sliding surface, substituting $x_2 = -c_1 x_1$ from equation (2.10) in state equations (2.9) we get:

$$\dot{x}_1 = -c_1 x_1. \tag{2.11}$$

Analytical solution $x_1 = x_{1_0} e^{-c_1 t}$ of the above equation indicates that magnitude of x_1 will decrease exponentially as time t increase. Ultimately $x_1 \to 0$ as $t \to \infty$, and the second state variable $x_2 = -c_1 x_1$ will also approach zero once x_1 approaches the origin. From the analytical solution, it is clear that both state variables will converge to the origin while following the sliding surface $s = x_2 + c_1 x_1$, and hence we have a stable sliding surface for $c_1 > 0$.

The second step in SMC is the selection of a reaching law to reach the sliding surface and then subsequently maintain sliding motion. A common choice is the discontinuous control law $u = -k\,\mathrm{sgn}(s)$ that is used here. For simulation we

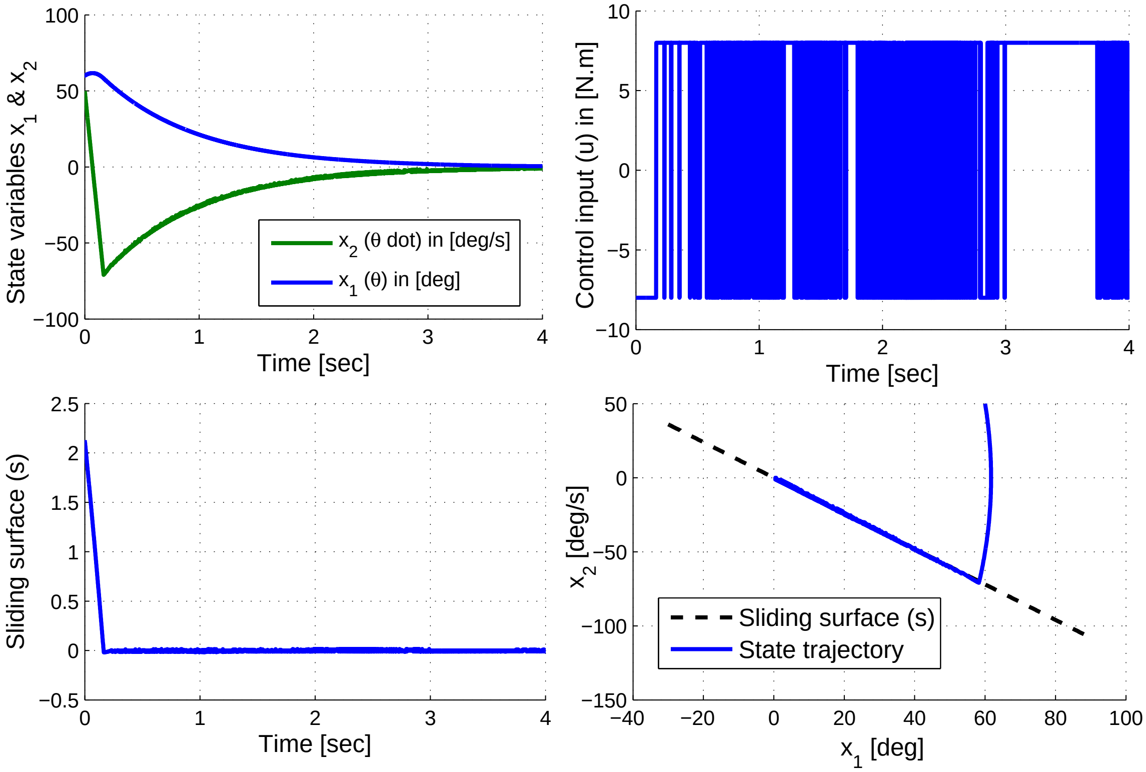


FIGURE 2.9: Simulation results for a pendulum system with $u = -k\,\mathrm{sgn}(s)$ control law.

choose $m = 0.5kg$, $L = 1.5m$, $g = 9.78m/s^2$ and $k = 8$. In Figure 2.9, simulation results are shown for initial state values $\theta = 60deg$ and $\dot{\theta} = 50deg/s$. It take approximately $0.2sec$ for the state trajectory to reach the sliding surface, and then it slides to the origin in approximately $3.5seconds$. The control input (u) has visible chattering in this case, as expected. To reduce the chattering, a more practical control law is required that is next derived.

A practical sliding mode based control law will have two terms: a continuous control term known as equivalent control, and a discontinuous term $-k\,\mathrm{sgn}(s)$. The equivalent control is derived from $\dot{s} = 0$:

$$\dot{s} = \dot{x}_2 + c_1\dot{x}_1 = 0. \tag{2.12}$$

Substituting the states variables from equation (2.9), we have:

$$\frac{1}{mL^2}\left(-mg\sin(x_1)L + u_{equivalent}\right) + c_1 x_2 = 0, \tag{2.13}$$

and hence we have the following expressing for the control law:

$$u_{equivalent} = -c_1 mL^2 x_2 + mg\sin(x_1)L, \tag{2.14}$$

The total control is the sum of the two terms:

$$\begin{aligned} u &= u_{equivalent} + u_{disc} \\ &= -c_1 mL^2 x_2 + mg\sin(x_1)L - k\,\mathrm{sgn}(s). \end{aligned} \tag{2.15}$$

To reduce chattering in control signal, the discontinuous term can be approximated by boundary layer approximation $\mathrm{sgn}(s) \approx \frac{s}{|s|+\epsilon}$, where ϵ is a small positive number. Hence we have the following final expression for the total control law:

$$u = -c_1 mL^2 x_2 + mg\sin(x_1)L - k\frac{s}{|s|+\epsilon}. \tag{2.16}$$

Simulation results with control law (2.16) is shown in Figure 2.10 for $\epsilon = 0.1$. In this case, it takes a slightly longer time for the state trajectory to reach the sliding surface, and then slides along the surface to reach the origin in approximately $4.0 seconds$. The prominent difference is observed in the control signal u. We have a smooth control signal that is physically realizable by actuator hardware if implemented in an actual system.

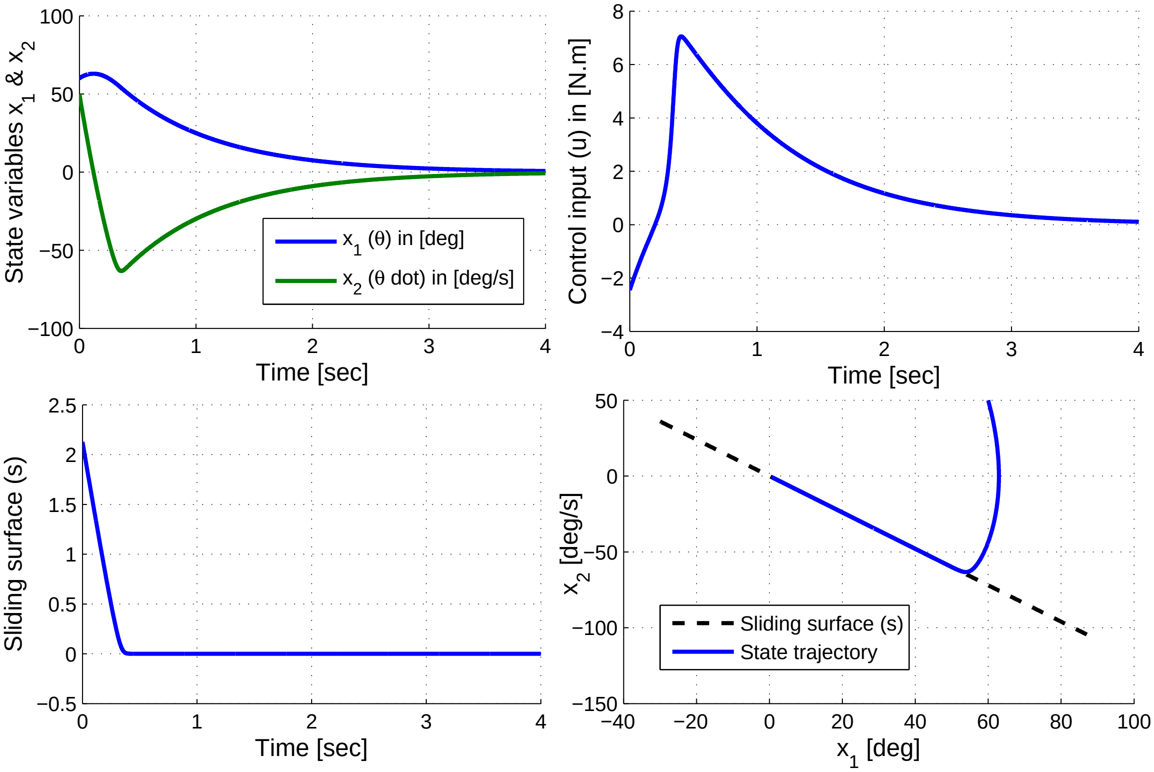


FIGURE 2.10: Simulation results for a pendulum system with a practical SMC law.

Chapter 3

KINEMATIC MODELS FOR UAV GUIDANCE

3.1 Introduction

In this chapter, mathematical models for bank-to-turn aerospace vehicles are formulated as a basis for guidance logic design and analysis. This thesis deals with the problem of UAV path following in a ground coordinate frame attached to the earth. When the equations of motion for a system become complex, it is often necessary to develop design models that have significantly less mathematical complexity, but still capture the essential behavior of the system. It is important to mention that it is not a goal of this dissertation to present detailed mathematical models for flight vehicles, but we present simplified kinematic models that capture most of the features of flight vehicles that are essential for guidance logic design. In deriving reduced-order guidance models, the main simplification we make is to eliminate the force- and moment-balance equations of motion (those involving $\dot{u}_b, \dot{v}_b, \dot{w}_b, \dot{p}, \dot{q}, \dot{r}$), thus eliminating the need to calculate the complex aerodynamic forces acting on the airframe. These general equations are replaced with simpler kinematic equations derived for the specific flight conditions of a coordinated turn and an accelerating climb.

A path planning algorithm generates a desired 3-D curved path that the UAV should follow. Desired path information is passed on to the guidance algorithm in the form of sequence of 3-D waypoints along with connectivity information. Two consecutive waypoints are either joined by a straight line or circular arc with defined radius and center. The guidance algorithm generates suitable commands for the vehicle control system so that the desired path is followed even in the presence of uncertainties and disturbances.

Three dimensional (3-D) guidance refers to following the mission path in both the lateral (horizontal) and longitudinal (vertical) planes (Breivik and Fossen, 2005). It covers both 2-D ground track following, as well as altitude profile following of the desired trajectory. In the 3-D case, guidance commands are generated for the lateral/directional control system in terms of reference roll or heading angles, and for the longitudinal control system in terms of pitch or angle of attack commands, taking into consideration the coupling between these two planes. A subset of the general 3-D problem is the 2-D lateral guidance problem in which the guidance objective is to ensure accurate ground track following of the vehicle. This is to say that the vehicle must exactly fly over lines and arcs joining mission waypoints as projected on plane ground, with minimum cross-track or lateral deviation. Another subset is the longitudinal guidance scheme that ensures accurate altitude following of the vehicle. Wind is an important disturbance parameter for the guidance system that should be properly taken into account during the design process. However, generally the wind vector lies in the horizontal plane and has negligible vertical component, so here the effect of wind is also analyzed for the lateral plane only. In the following sections, kinematic models are derived for all these cases. In section 3.2, kinematic equations are derived for the lateral guidance logic design to ensure ground tracking. Similarly a mathematical model for the design of longitudinal guidance scheme is derived in section 3.3. More generalized 3-D kinematic equations considering the coupling between longitudinal and lateral motions are discussed in the section 3.4.

3.2 Lateral Guidance Model

Before the derivation of kinematics model for guidance logic design, it is important to understand the sense of different variables and coordinate systems. We need to understand how the UAV is oriented with respect to the earth. The subsequent subsection describes the various variables, the coordinate systems used to describe the position and orientation of the aircraft and the transformation between these coordinate systems.

3.2.1 Reference frames and variables definition

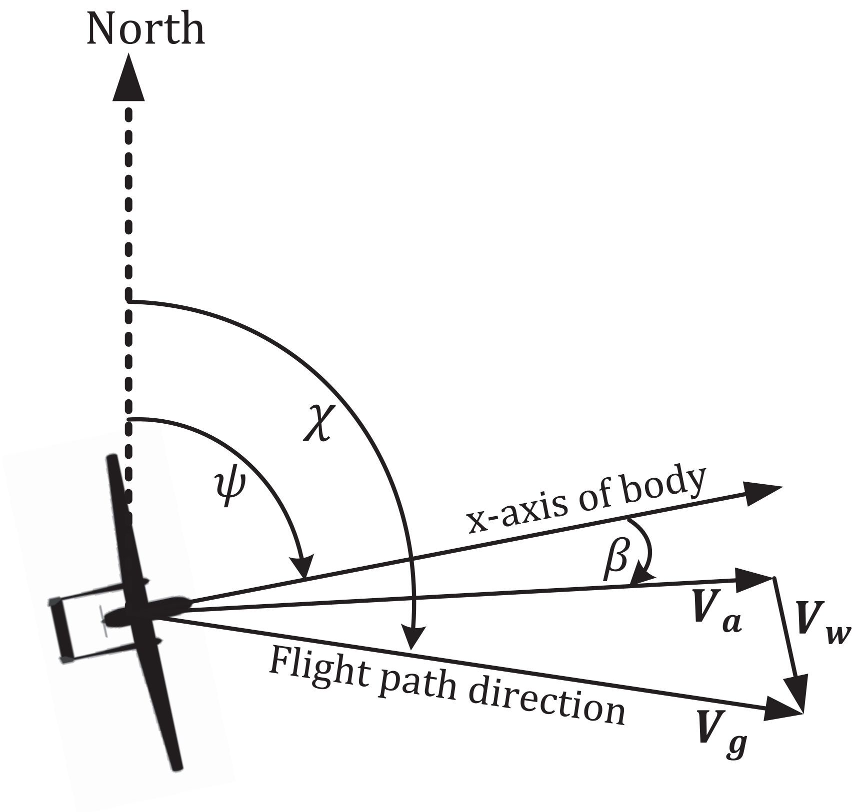


FIGURE 3.1: Definition of various angles for 2-D lateral plane.

In order to set up the guidance problem, description of three important lateral plane angles (see Figure 3.1) is required. The *body heading* angle (ψ) describes the direction of nose of the UAV relative to North and is defined as "the angle between body x-axis and the true North in the horizontal plane". The airspeed vector ($\mathbf{V}_a$) provides the motion of the UAV relative to the incoming air in 3-D space, and the *sideslip* angle β is defined as the angle between the UAV nose and the relative airspeed vector $\mathbf{V}_a$ in the horizontal plane. In steady level flight with zero rudder deflection and no body asymmetries, the nose of the UAV is aligned with the airspeed vector $\mathbf{V}_a$ and hence there is zero sideslip angle in this case.

The rate of change of the vehicle's position in 3-D space is defined as the ground velocity vector $\mathbf{V}_g$ and is not necessarily equal to $\mathbf{V}_a$. The ground velocity ($\mathbf{V}_g$) is the vector sum of the wind velocity vector ($\mathbf{V}_w$) and the airspeed vector ($\mathbf{V}_a$), as shown in Figure 3.1. Obviously, the vector $\mathbf{V}_g$ is equal to $\mathbf{V}_a$ in the absence of wind. Another important angle is the *course* angle (χ) defined as "the angle of the ground velocity vector ($\mathbf{V}_g$) with respect to north in the horizontal plane". As discussed earlier in Chapter 2, the outer loop is much slower than the inner loop in a multi-loop configuration. Dynamic variables such as β respond much more quickly in time scale and hence are considered in inner control loop design. On the other hand, kinematic variables like χ respond slowly and are considered in the outer guidance loop design.

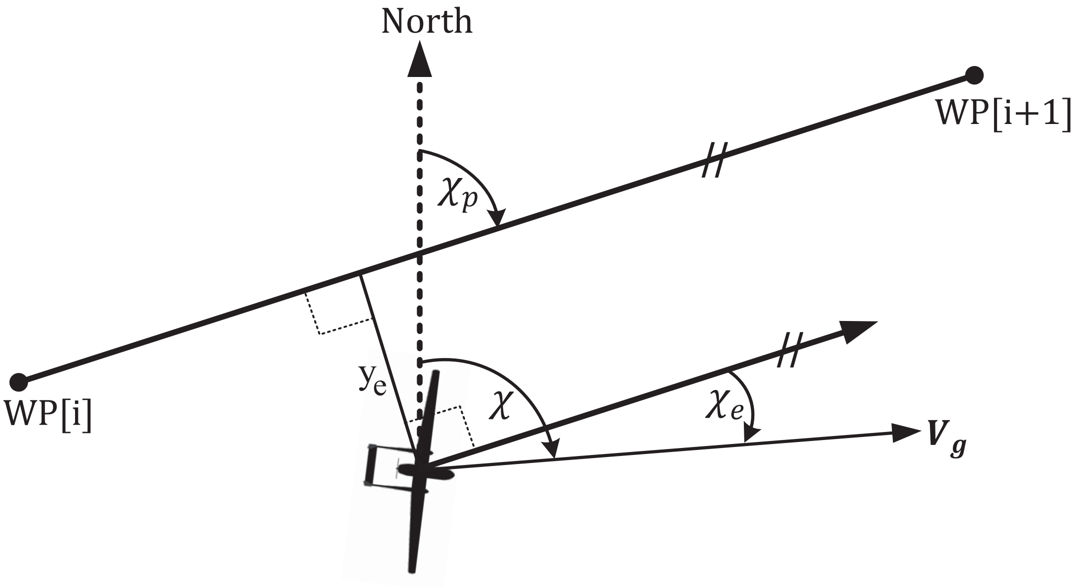


FIGURE 3.2: The 2-D lateral guidance problem definition for a straight path.

Definition and sign convention of different lateral guidance variables for straight and curved paths are shown in Figures 3.2 and 3.3, respectively. In these figures, all the guidance variables are indicated in the positive sense. In case of straight path (Figure 3.2), the angle χ_p of the line joining two successive ground waypoints $WP[i]$ and $WP[i+1]$ with respect to north defines the desired or reference course

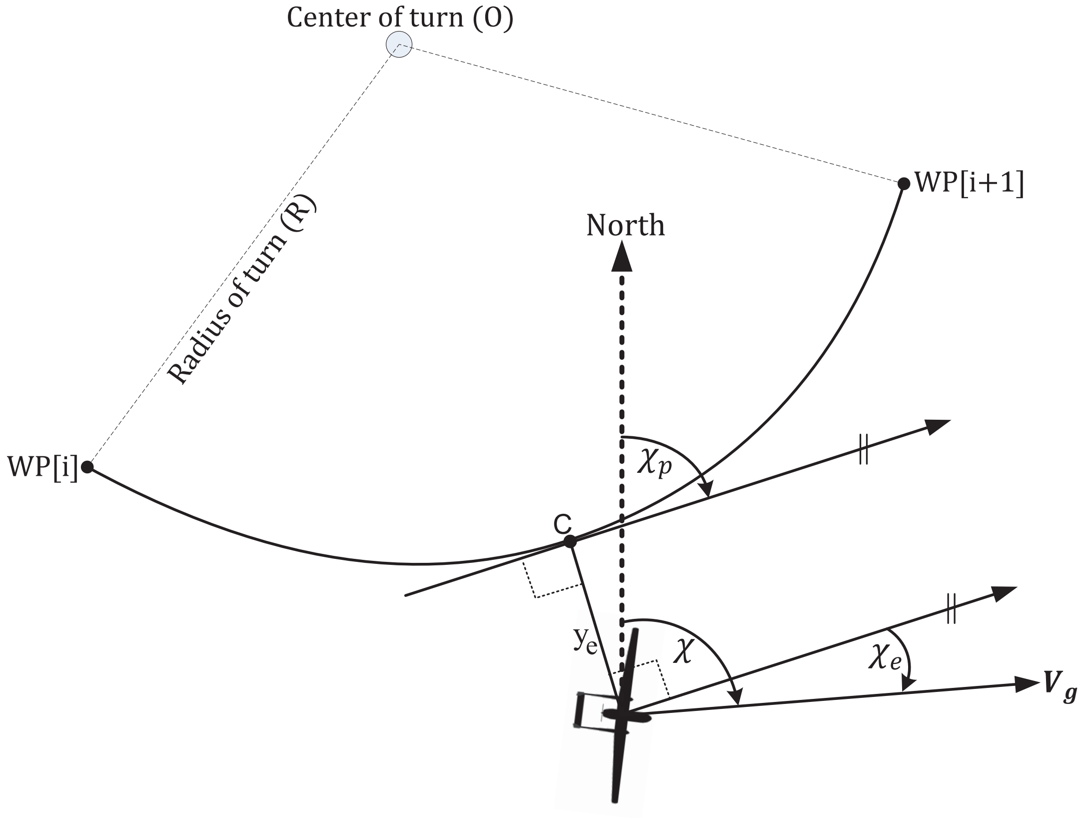


FIGURE 3.3: The 2-D lateral guidance problem definition for a circular path.

angle. In case of a circular trajectory of radius R centered at O (Figure 3.3), the desired course angle χ_p is defined as the angle of the tangent line at point C (the closest point on the desired trajectory to the UAV). In both figures, y_e (cross-track error) denotes the lateral deviation of the UAV from the desired path. Another important guidance parameter $\chi_e = \chi - \chi_p$ is defined in literature as the *intercept course*. In case of any deviation from the reference path, the objective of lateral guidance scheme is to bring y_e to zero through manipulation of χ_e (by generating suitable roll angle commands).

3.2.2 Wind effect

As discussed above, the ground velocity $\mathbf{V}_g$ is the vector sum of the airspeed vector $\mathbf{V}_a$ and the wind velocity vector $\mathbf{V}_w = (V_{w_n}, V_{w_e}, V_{w_d})^T$, i.e., $\mathbf{V}_g = \mathbf{V}_a + \mathbf{V}_w$.

Considering a fixed and level North-East-Down inertial coordinate system, the wind triangle can be expressed as (Beard and McLain, 2012):

$$V_g \begin{pmatrix} \cos\chi\cos\gamma \\ \sin\chi\cos\gamma \\ -\sin\gamma \end{pmatrix} = \begin{pmatrix} V_{w_n} \\ V_{w_e} \\ V_{w_d} \end{pmatrix} + V_a \begin{pmatrix} \cos\psi\cos\gamma_a \\ \sin\psi\cos\gamma_a \\ -\sin\gamma_a \end{pmatrix}, \tag{3.1}$$

where V_g and V_a are the magnitudes of the ground and airspeed vectors, respectively. Variables γ and γ_a denote the inertial referenced and air-mass-referenced flight-path angles respectively, in the longitudinal (vertical) plane. For level flight ($\gamma = 0$) the projection of the these vectors on the North-East (2-D ground) plane is written as:

$$V_g \begin{pmatrix} \cos\chi \\ \sin\chi \end{pmatrix} = \begin{pmatrix} V_{w_n} \\ V_{w_e} \end{pmatrix} + V_a \begin{pmatrix} \cos\psi \\ \sin\psi \end{pmatrix}, \tag{3.2}$$

or alternatively

$$V_g \begin{pmatrix} \cos\chi \\ \sin\chi \end{pmatrix} - \begin{pmatrix} V_{w_n} \\ V_{w_e} \end{pmatrix} = V_a \begin{pmatrix} \cos\psi \\ \sin\psi \end{pmatrix}. \tag{3.3}$$

Taking the square of equation (3.3), we have:

$$V_g^2 - 2V_g(V_{w_n}\cos\chi + V_{w_e}\sin\chi) + V_w^2 - V_a^2 = 0. \tag{3.4}$$

Using this equation, the relation between V_g and V_a can be worked out for selected values of the course angle (χ) and wind vector $\mathbf{V}_w = (V_{w_n}, V_{w_e}$. The aerodynamic lift and drag forces of an aerospace vehicle mainly depend on the airspeed (V_a), this is usually maintained either through a feedback controller, or a preset lookup table of throttle settings. V_a can be maintained approximately constant even in the presence of wind, variation of V_g can be derived using the expression (3.4). As

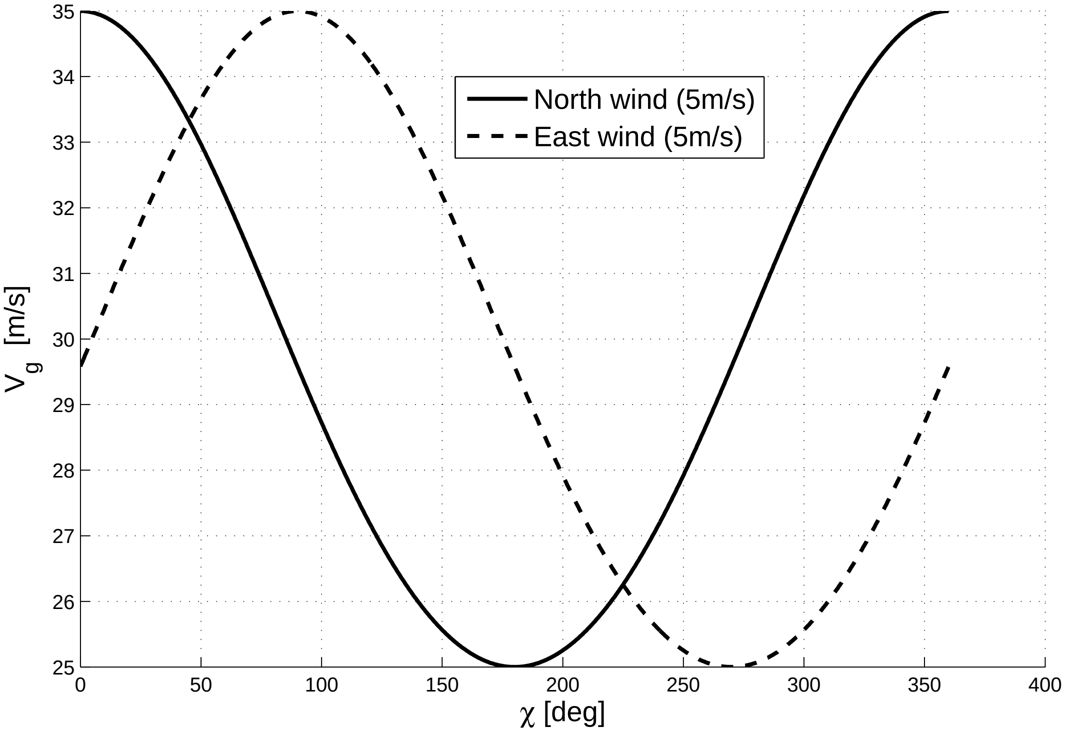


FIGURE 3.4: Variation of V_g for constant V_a in the presence of wind.

an example, variation of V_g is shown in Figure 3.4 for $V_a = 30$ m/s, depending on χ and wind. Here V_g is plotted versus χ for two different cases of wind: a north wind of 5 m/s ($\mathbf{V}_w = (5,0)^T$), and an east wind of 5 m/s ($\mathbf{V}_w = (0,5)^T$).

The relation between ψ and χ can also be derived using the expression (3.3), multiplying both sides of the equation on the left hand side by the row vector $(-\sin\chi, \cos\chi)$ yields:

$$V_{w_n}\sin\chi - V_{w_e}\cos\chi = V_a\left(-\sin\chi\cos\psi + \cos\chi\sin\psi\right), \tag{3.5}$$

which can be written in simplified form as:

$$\psi - \chi = \sin^{-1}\left(\frac{1}{V_a}\left(V_{w_n}\sin\chi - V_{w_e}\cos\chi\right)\right). \tag{3.6}$$

Variation of $\psi - \chi$ for a constant airspeed of 30 m/s is shown in Figure 3.5 for

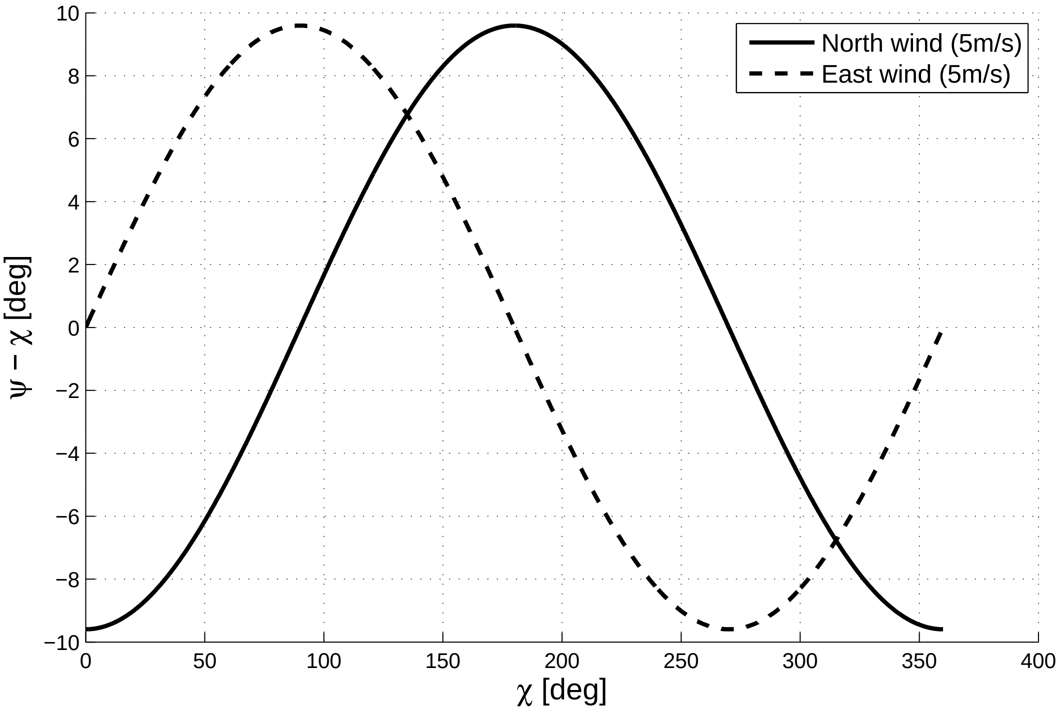


FIGURE 3.5: Variation of $\psi - \chi$ for constant V_a in the presence of wind.

north and east wind. Difference of upto ~ 10 deg in ψ and χ is possible depending on wind speed and course angle χ.

3.2.3 Kinematic equations

The concept of coordinated turn in the presence of winds is discussed in detail in (Beard and McLain, 2012). The term 'coordinated turn' is used in the sense that there is no side force in the body frame of the vehicle, thus implying zero sideslip angle. Aerial vehicles use a component of the aerodynamic lift to generate lateral accelerations to correct the lateral (cross-track) errors during flight (Siouris, 2004); these are balanced by centrifugal accelerations in a coordinated turn. Lateral acceleration is produced by rolling or banking the vehicle so that a component of the lift vector is tilted in the required direction (the *bank-to-turn* maneuver). The guidance loop generates appropriate bank (or roll) commands ϕ_{req} for the inner control loop to follow. Reference bank angle command (ϕ_{req}) therefore serves as the control input in our case.

Consider a coordinated turn ($\beta = 0$) in the presence of wind, the aerodynamic lift vector $\mathbf{L}$ can be resolved into two components as shown in Figure 3.6. The component ($L \cos \phi$) balances the weight of the vehicle, and a part of the other component balances the centrifugal force during a turn (Beard and McLain, 2012). Summing the forces in the vertical and radial directions:

$$L \cos \phi = mg, \qquad L \sin \phi \cos(\chi - \psi) = \frac{m V_g^2}{R}. \tag{3.7}$$

Here m denotes the mass, g is the gravitational acceleration and R is the radius of turn of the vehicle. From above equation (3.7), we get:

$$\tan \phi \cos(\chi - \psi) = \frac{V_g^2}{Rg} \tag{3.8}$$

During a steady turn $V_g = R\dot{\chi}$, so the equation (3.8) becomes:

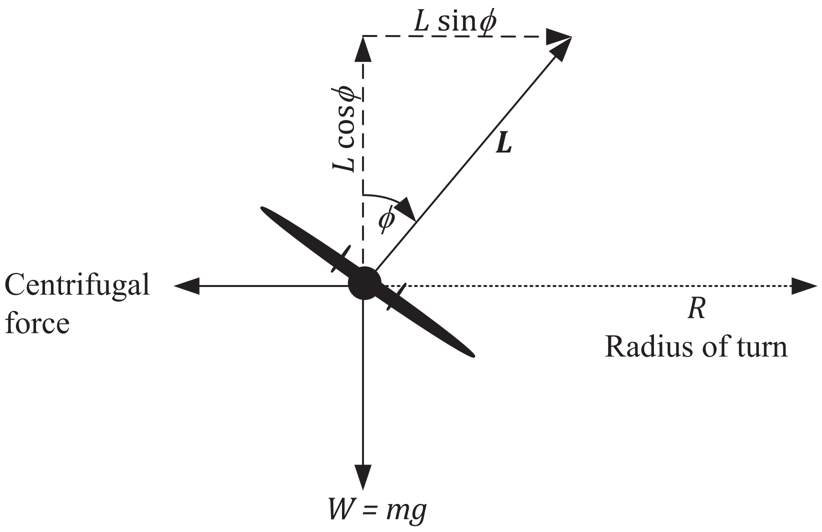
$$\tan \phi \cos(\chi - \psi) = \frac{V_g \dot{\chi}}{g}. \tag{3.9}$$

We assume that the inner loop dynamics from ϕ_{req} to ϕ are significantly faster (e.g., $5 \sim 10$ times faster) than the outer (guidance) loop dynamics. This assumption implies that $\phi_{req} \approx \phi$, and so the expression (3.9) can be approximated as:

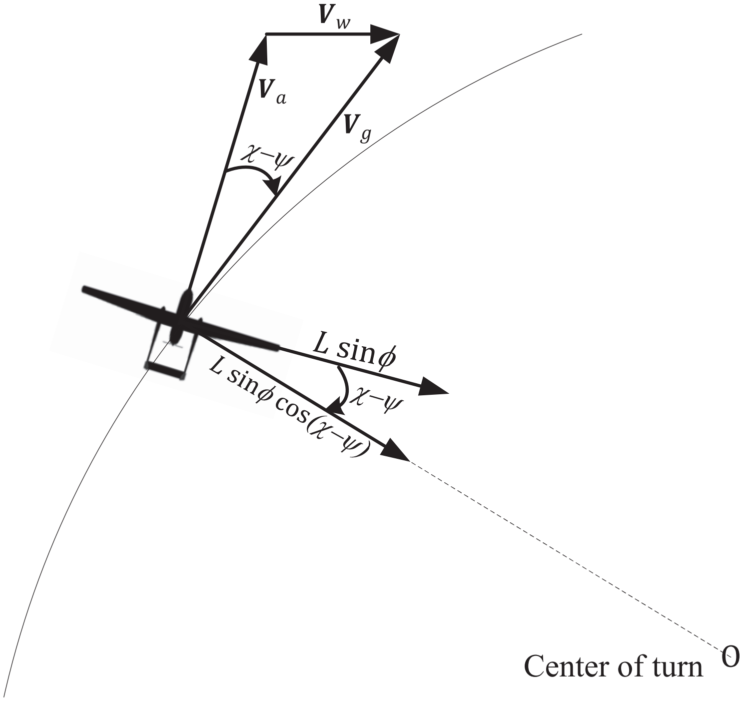
$$\tan \phi_{req} \cos(\chi - \psi) = \frac{V_g \dot{\chi}}{g}. \tag{3.10}$$

Now since $\chi_e = \chi - \chi_p$, therefore we have:

$$\tan \phi_{req} \cos(\chi - \psi) = \frac{V_g (\dot{\chi}_e + \dot{\chi}_p)}{g}. \tag{3.11}$$



(a) Back view during coordinated turn



(b) Top view during coordinated turn

FIGURE 3.6: Components of the lift vector L during a steady turn of radius R.

Rearranging equation (3.11), we get

$$\dot{\chi}_e = \frac{g \tan \phi_{req}}{V_g} \cos(\chi - \psi) - \dot{\chi}_p, \tag{3.12}$$

Using equation (3.6), the above state equation becomes:

$$\dot{\chi}_e = \frac{g \tan \phi_{req}}{V_g} \cos\left(\sin^{-1}\left(\frac{1}{V_a}\left(-V_{w_n} \sin \chi + V_{w_e} \cos \chi\right)\right)\right) - \dot{\chi}_p, \tag{3.13}$$

where χ_e is a state variable and $\dot{\chi}_p$ is the rate of change of the desired course/path – this information is available from mission data. During straight flight, $\dot{\chi}_p$ is equal to zero, but it can become appreciable during sharp turns.

The inertial horizontal position $(P_n, P_e)^T$ of the UAV in the fixed North-East-Down frame can be derived using:

$$\begin{pmatrix} \dot{P}_n \\ \dot{P}_e \end{pmatrix} = V_g \begin{pmatrix} \cos \chi \\ \sin \chi \end{pmatrix} \tag{3.14}$$

To compute the cross-track deviation (y_e) at any point (see Figures 3.2 and 3.3), we can use the relation:

$$\dot{y}_e = V_g \sin \chi_e. \tag{3.15}$$

Equations (3.15) and (3.13) represent the overall dynamics for the outer loop guidance design problem. For simplicity, we may define $u = \tan \phi_{req}$ as the control input, in state space form we can then write:

$$\begin{pmatrix} \dot{y}_e \\ \dot{\chi}_e \end{pmatrix} = \begin{pmatrix} V_g \sin \chi_e \\ -\dot{\chi}_p \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{g}{V_g} \cos\left(\sin^{-1}\left(\frac{1}{V_a}\left(-V_{w_n} \sin \chi + V_{w_e} \cos \chi\right)\right)\right) \end{pmatrix} u \tag{3.16}$$

Here y_e and χ_e are the state variables, and $u = \tan \phi_{req}$ is the control signal that the guidance loop has to generate for track following. The gravity g is a known constant. Measurement of the ground speed V_g will be required for our proposed guidance law. Most sensors onboard UAVs measure the airspeed of the vehicle. This will differ from the ground speed by an amount equal to the wind speed. Also in most cases a GPS[1] receiver will be available on board the aircraft which gives position measurements, usually with an update rate of 1 Hz. These can be used to estimate the ground speed of the vehicle, with some estimation error. In either case, we shall assume an upper bound on the error in the measurement of V_g, and use it to derive bounds on guidance parameters.

Simplified state equations (neglecting wind)

Generally wind speed and direction measurements are not available in low cost UAVs, and these are estimated in real time. A simplified kinematic model can be utilized for guidance logic design neglecting the wind in above state equations (3.16). Effect of wind can be compensated through parametric uncertainty in ground speed V_g. In the absence of wind ($V_{w_n} = V_{w_e} = 0$), state equations for lateral guidance logic take the following form:

$$\begin{pmatrix} \dot{y}_e \\ \dot{\chi}_e \end{pmatrix} = \begin{pmatrix} V_g \sin \chi_e \\ -\dot{\chi}_p \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{g}{V_g} \end{pmatrix} u \tag{3.17}$$

3.3 Longitudinal Guidance Model

The objective of this section is to derive the kinematic equations required for longitudinal guidance logic design of UAVs. To derive the desired equations, we will consider a UAV pull-up maneuver along an arc in the vertical plane. The different longitudinal variables, their convention and reference frames are discussed in detail in subsection 3.3.1 followed by derivation of kinematics equations in subsection 3.3.2.

---

[1] The Global Positioning System.

3.3.1 Reference frames and variables definition

In derivation of the kinematics equations, it is assumed that the earth is flat, non-rotating, and an approximate inertial reference frame. Standard convention of coordinate systems and their orientations is assumed here (Hull, 2007). The *body axes system* $Ox_by_bz_b$ is fixed to the UAV as shown in Figure 3.7. The *local horizontal system* $Ox_hy_hz_h$ moves with the UAV (O is the UAV center of gravity), but its axes remain parallel to the ground axes. The *ground axes system* E_{xyz} is fixed to the surface of the earth at mean sea level, and the xz plane is the vertical plane. Let $WP[i]$ and $WP[i+1]$ be the two successive waypoints in vertical plane, and γ_p be the angle of the line $WP[i]$-$WP[i+1]$ with respect to the local horizontal (called the *reference* or *desired flight path* angle). Flight path angle (γ) denotes the angle of ground velocity vector $\mathbf{V}_g$ of the UAV with respect to the local horizontal as shown in Figure 3.7. Generally, deviation from the desired line ($WP[i]$-$WP[i+1]$) is either measured in vertical direction (h_e) or normal to the desired line (z_e) as shown in Figure 3.7. It is to mention here that sign convention of h_e and z_e is different. If UAV position is above the desired line $WP[i]$-$WP[i+1]$ then h_e will be positive in this case, however z_e will be negative. The main theme of the guidance logic is to generate suitable commands for the inner loop to follow the line $WP[i]$-$WP[i+1]$ as closely as possible by either controlling the variable h_e or z_e. In case of any deviation the guidance logic must bring the UAV back gracefully to the desired path without violating constraints.

3.3.2 Kinematic equations

During accelerating climb/decent (pull-up/down maneuvers along an arc in vertical plane), the forces acting on an aerospace vehicles are summarized in Figure 3.8. During pull up maneuver along an arc, the vehicle lift force has to be balanced by the weight component and the centripetal acceleration. Summing up all the forces, we have:

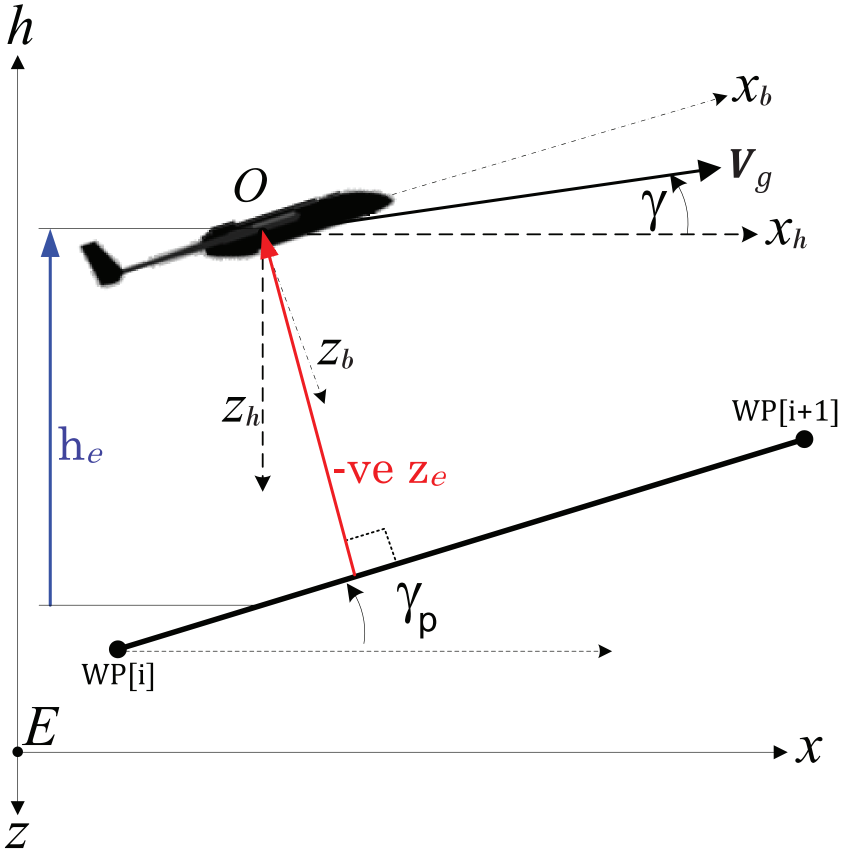


FIGURE 3.7: Coordinate systems and their conventions (vertical plane).

$$L = mg\cos\gamma + mV_g\dot{\gamma}, \tag{3.18}$$

where m is the mass, g is the gravitational acceleration and V_g is the velocity of the vehicle relative to ground. Equation (3.18) can be written as:

$$\dot{\gamma} = \frac{g}{V_g}\left(\frac{L}{mg} - \cos\gamma\right). \tag{3.19}$$

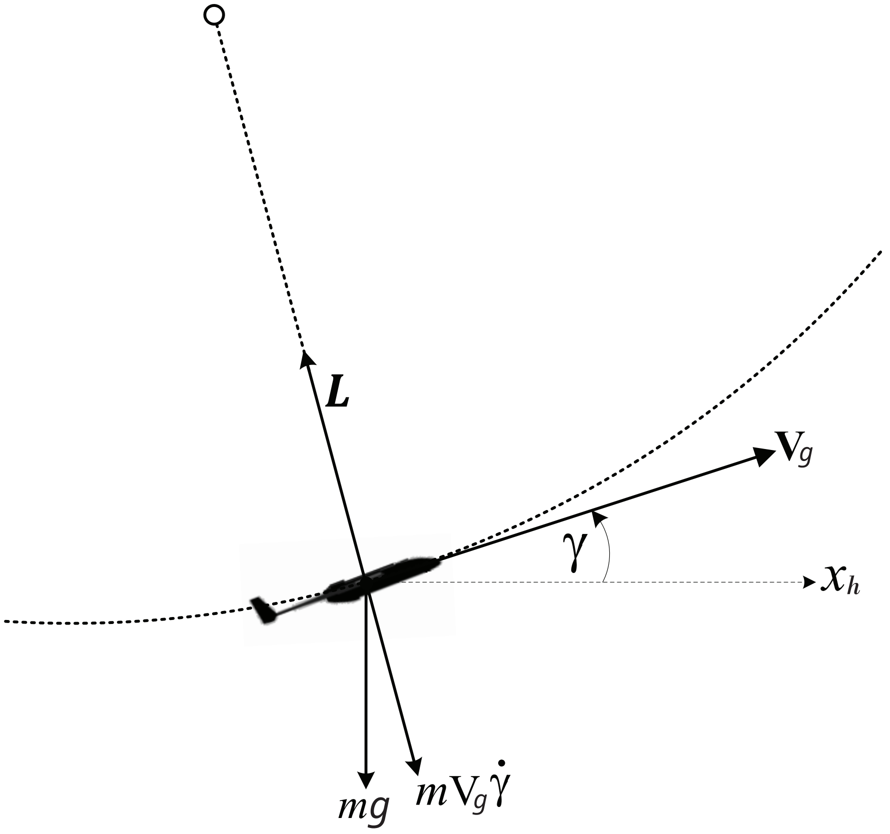


FIGURE 3.8: Forces acting on UAV during accelerating climb in vertical plane.

where the dimensionless term $\frac{L}{mg}$ is known as the *load factor*, i.e. the number of normal $g's$ that an aerospace vehicle experiences during flight.

Assuming an inner loop faster than the outer guidance loop, we have $L_{req} \approx L$ and this implies that equation (3.19) can be approximated as:

$$\dot{\gamma} = \frac{g}{V_g}\left(\frac{L_{req}}{mg} - \cos\gamma\right). \tag{3.20}$$

To derive the other state equation, the ground velocity V_g has two components in the *ground axes system* E_{xyz} (vertical plane):

$$\dot{x} = V_g \cos\gamma, \qquad \dot{h} = V_g \sin\gamma. \tag{3.21}$$

Using equation (3.21), we can derive an expression for the rate of change of deviation from the desired path. As discussed in the above subsection 3.3.1, the deviation from the desired path can be either measured in terms of vertical error (h_e) or deviation normal to the desired path (z_e) as shown in Figure 3.7. Here we have mentioned equations for both errors and any one can be used depending on the application:

$$\dot{z}_e = -V_g \sin\gamma_e \ , \qquad \dot{h}_e = V_g \cos\gamma_p \sin\gamma_e + V_g \sin\gamma_p \cos\gamma_e - \dot{h}_{ref}. \tag{3.22}$$

where $\gamma_e = \gamma - \gamma_p$ denotes the error in flight path angle.

Above equation (3.22) can be simplified by assuming that $\dot{h}_{ref} \approx V_g \sin\gamma_p \cos\gamma_e$, implying the following simplified equations for rate of change of deviation:

$$\dot{z}_e = -V_g \sin\gamma_e \ , \qquad \dot{h}_e = V_g \cos\gamma_p \sin\gamma_e. \tag{3.23}$$

Combining equations (3.20, 3.23), we have the following state equations for guidance law design:

$$\left.\begin{aligned} &\dot{z}_e = -V_g \sin\gamma_e \quad \mathit{or} \quad \dot{h}_e = V_g \cos\gamma_p \sin\gamma_e, \\ &\dot{\gamma}_e = \frac{g}{V_g}\left(\frac{L_{req}}{mg} - \cos\gamma\right) - \dot{\gamma}_p \end{aligned}\right\} \tag{3.24}$$

Here deviation from path (h_e or z_e) and flight path angle error γ_e are the two state variables, and L_{req} is the control variable that the guidance loop generates for inner control loop to follow. It may be mentioned here that the term $\dot{\gamma}_p$ is basically related with the rate of change of commanded RoC/RoD (rate of climb/decent) and it is zero during level cruising as well as during climb/decent with constant RoC/RoD. However, it is non-zero during variable RoC/RoD but its value is very small. Hence the term $\dot{\gamma}_p$ can be taken as zero for simplicity during derivation of the guidance logic implying the following simplified kinematic equations for longitudinal guidance logic design:

$$\begin{pmatrix} \dot{z}_e \\ \dot{\gamma}_e \end{pmatrix} = \begin{pmatrix} -V_g \sin \gamma_e \\ -\frac{g \cos \gamma}{V_g} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{mV_g} \end{pmatrix} L_{req} \tag{3.25}$$

Equation (3.25) represents the state space form of the kinematics equations for longitudinal guidance logic design. Here L_{req} is the output of guidance loop for inner control loop to follow, and the two state variables are z_e and γ_e.

3.4 Guidance Model for 3-D Flight

Many flight mechanics books and papers (Beard and McLain, 2012; Hull, 2007; Miele, 1962) discuss the equations of motion for guidance and control design of aerospace vehicles in detail. In this section, we discuss the kinematic equations for 3-D flight briefly; interested readers are referred to (Beard and McLain, 2012) for a detailed discussion. Extending the concept of 2-D equations in the last two sections (3.2, 3.3), the convention and definitions of different variables is discussed in subsection 3.4.1, followed by kinematic equations for 3-D flight in subsection 3.4.2.

3.4.1 Reference frames and variables definition

In order to set up the guidance problem, we first define the reference frames and important parameters. Consider a UAV modeled as rigid-body, define $\{I\}$ as the inertial frame attached to the earth at mean sea level and $\{B\}$ as the body frame

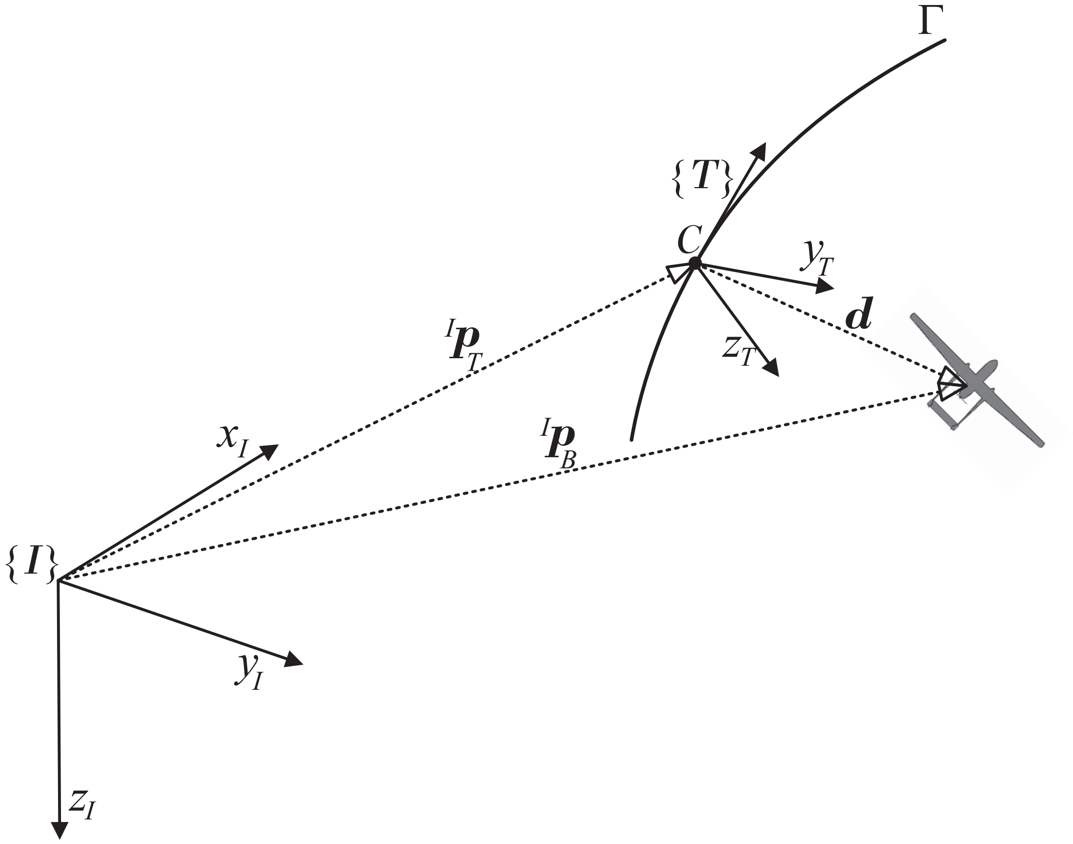


FIGURE 3.9: Coordinate systems and their conventions for 3-D space.

attached to the vehicle's center of mass. Inertial frame $\{I\}$ components x_I, y_I and z_I are pointed towards *north*, *east* and *downward* directions, respectively. Let $^I\mathbf{p}_B = [x, y, z]^T \in \mathbb{R}^3$ and $\mathbf{V} = [\dot{x}, \dot{y}, \dot{z}]^T = [V_x, V_y, V_z]^T \in \mathbb{R}^3$ denote the inertial position and velocity vectors of $\{B\}$ relative to $\{I\}$, respectively. Flight path angle (elevation angle) γ denotes the orientation of ground velocity vector ($\mathbf{V}$) relative to the local horizontal, and the course angle χ is the angle between the ground velocity vector's component projected on the ground relative to North. These two angular variables can be computed as:

$$\chi = \arctan\left(\frac{\dot{y}}{\dot{x}}\right), \qquad \gamma = \arctan\left(\frac{-\dot{z}}{\sqrt{\dot{x}^2 + \dot{y}^2}}\right). \tag{3.26}$$

In order to define the error space, let Γ denote the desired/reference 3-D geometric path that UAV should follow and the point C on the reference path is the closest point to the current position of the UAV at any instant t as shown in Figure 3.9. The tangent frame $\{T\}$ at point C can be defined as a coordinate frame that moves along Γ, whose x-component (x_T) is tangent to the path, y-component (y_T) is pointed towards right (parallel to the local horizontal) and the third z-component (z_T) is perpendicular to both x_T and y_T following the right-hand rule. Let the position of $\{T\}$ relative to $\{I\}$ be denoted by vector ${}^{I}\mathbf{p}_T$, and hence the vector $\mathbf{d} = {}^{I}\mathbf{p}_B - {}^{I}\mathbf{p}_T$ represents the off-track error vector in the inertial frame. It follows that the off-track error vector $\mathbf{d}$ has no $x-$component when expressed in $\{T\}$, i.e.,

$$
{}^{T}_{I}R\mathbf{d} = [0, y_e, z_e]^T, \tag{3.27}
$$

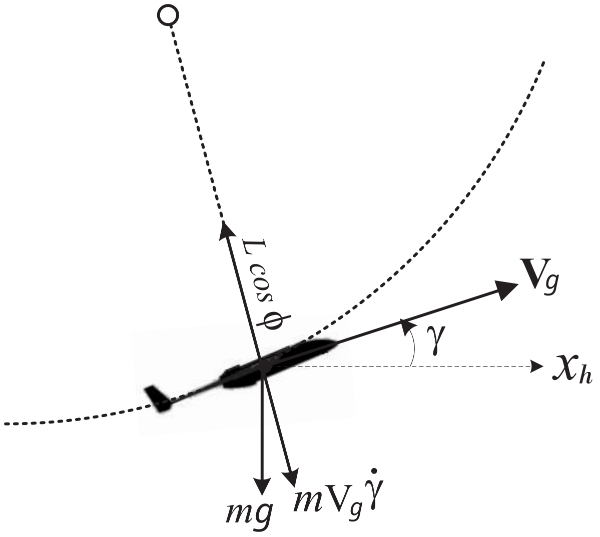
where y_e, $z_e \in \mathbb{R}$ and ${}^{T}_{I}R$ is the transformation matrix from $\{I\}$ to $\{T\}$. The main task of the guidance algorithm is to keep the errors (y_e and z_e) as small as possible by generating suitable commands for the vehicle control system.

3.4.2 3-D Kinematic equations

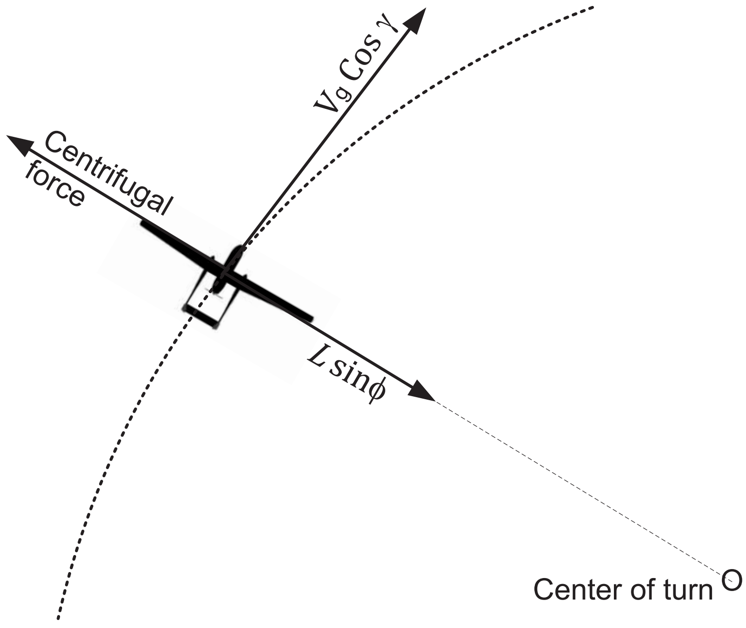
Here we present a simplified 3-D kinematic model that capture most of the features of flight vehicles that are essential for guidance logic design. During bank to turn while climb/decent maneuvers, forces acting on an aerospace vehicle are summarized in Figure 3.10. Summing up all the forces in the vertical and radial directions, we have:

$$
L\cos\phi = mV_g\dot{\gamma} + mg\cos\gamma, \qquad L\sin\phi = m(V_g\cos\gamma)\dot{\chi}, \tag{3.28}
$$

where L is the lift force, ϕ is the body roll angle, m is the mass, g is the gravitational acceleration and V_g is the velocity of the vehicle relative to ground. Rearranging



(a) side view during coordinated turn while climb/decent



(b) Top view during coordinated turn while climb/decent

Figure 3.10: Forces acting on a UAV during accelerating climb while coordinated turn.

the terms, equation (3.28) can be written as:

$$\left.\begin{aligned}\dot{\gamma} &= \frac{g}{V_g}\left(\frac{L\cos\phi}{mg} - \cos\gamma\right)\\ \dot{\chi} &= \frac{L\sin\phi}{mV_g\cos\gamma},\end{aligned}\right\} \tag{3.29}$$

As the main theme of this paper is outer guidance loop design, so we assume here that the inner control loop is already designed and is fast enough (at least $5 \sim 10$ times faster than outer loop). Based on this assumption (ignoring inner loop dynamics), we can approximate $L_{req} \approx L$ and $\phi_{req} \approx \phi$ for guidance design, this implies that the equation (3.29) can be approximated as:

$$\left.\begin{aligned}\dot{\gamma} &= \frac{g}{V_g}\left(\frac{L_{req}\cos\phi_{req}}{mg} - \cos\gamma\right)\\ \dot{\chi} &= \frac{L_{req}\sin\phi_{req}}{mV_g\cos\gamma}.\end{aligned}\right\} \tag{3.30}$$

In these two equations, γ (flight path angle) and χ (course angle) are the state variables. Two control variables are L_{req} (the required or reference Lift force) and ϕ_{req} (the required/reference roll angle). The task of the outer guidance loop is to generate these two control variables for the inner control loop to follow the desired 3-D path. Other two state equations can be derived from the components of ground velocity (V) in vertical and radial directions. As discussed earlier in section 3.4.1, ${}^{I}\mathbf{p}_B = [x, y, z]^T$ denotes the 3-D position of an aerospace vehicle in inertial *ground axes system* $\{I\}$ that can be determined as:

$$\dot{x} = (V_g\cos\gamma)\cos\chi, \quad \dot{y} = (V_g\cos\gamma)\sin\chi, \quad \dot{z} = -V_g\sin\gamma. \tag{3.31}$$

Let Γ be the desired path in 3-D space as shown in Figure 3.9, χ_p and γ_p are the desired course and flight path angles (from mission plan) respectively, at any instance. With $\chi_e = \chi - \chi_p$ and $\gamma_e = \gamma - \gamma_p$, we have the following state equations in error form:

$$\left.\begin{aligned} \dot{z}_e &= -V_g \sin\gamma_e \\ \dot{y}_e &= V_g \cos\gamma \sin\chi_e \\ \dot{\gamma}_e &= \frac{g}{V_g}\left(\frac{L_{req}\cos\phi_{req}}{mg} - \cos\gamma\right) - \dot{\gamma}_p \\ \dot{\chi}_e &= \frac{L_{req}\sin\phi_{req}}{mV_g\cos\gamma} - \dot{\chi}_p, \end{aligned}\right\} \tag{3.32}$$

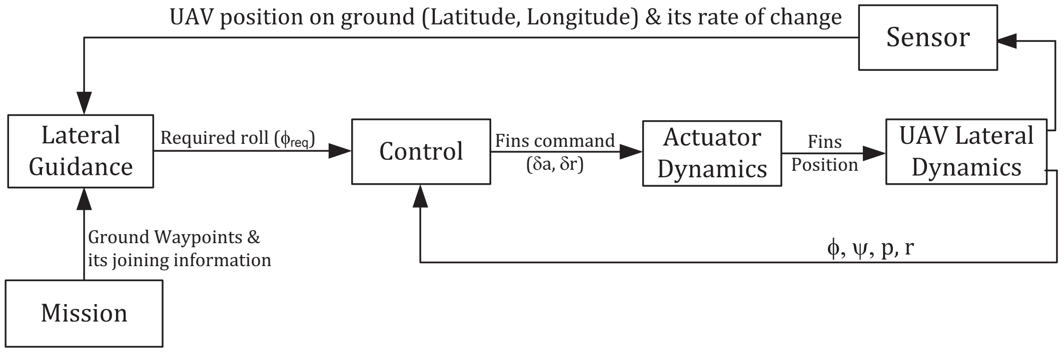
where z_e (error in vertical plane, normal to the reference path), y_e (cross-track error), γ_e (error in flight path angle) and χ_e (error in course angle) are the four state variables. Control inputs are L_{req} (required Lift force) and ϕ_{req} (required roll angle), that the guidance loop has to generate for following the desired mission and correcting any deviations.

3.5 Problem Formulation

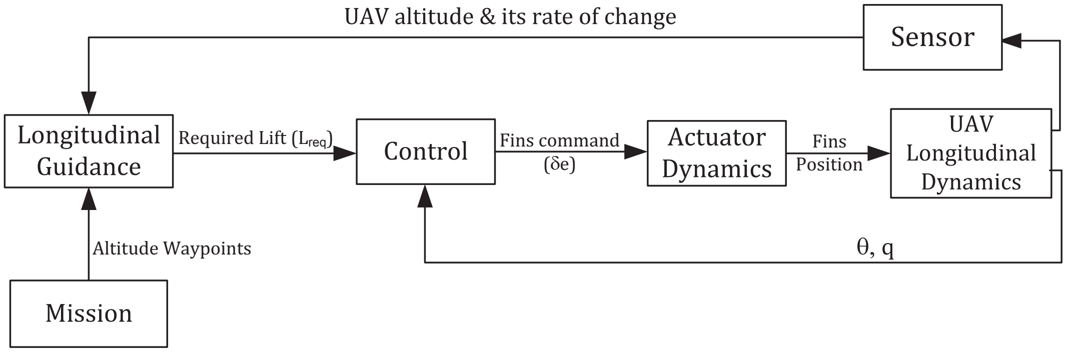
After definition of state space variables, we now define the guidance problem for UAV path following. The UAV path following problem can be solved using two approaches. In the first simplified approach, the 3-D path following problem is separated into two 2-D problems (Lateral and Longitudinal) and guidance algorithms are designed to cater for the deviations in the two planes independently (ignoring the coupling between these two planes). In the second approach, the 3-D path following problem is considered in a single framework and a single guidance algorithm is designed to keep the vehicle on track. In the subsequent subsections problem formulation is discussed for both these cases.

3.5.1 Decoupled 2-D guidance problem

In this approach, the desired mission information is split into lateral and longitudinal waypoints as shown in Figure 3.11. Based on separated waypoints, two independent guidance and control schemes for lateral and longitudinal planes are designed



a) Lateral guidance and control scheme



b) Longitudinal guidance and control scheme

FIGURE 3.11: Decoupled simplified guidance and control structure for UAVs.

ignoring the coupling between these two planes. Decoupled kinematic equations for lateral and longitudinal planes are described by equations (3.16 and 3.25), respectively. Lateral guidance scheme is responsible to keep the lateral cross track deviation as minimum as possible by generating required roll angle (ϕ_{req}) command for the inner control loop to follow. Inputs of lateral guidance scheme are ground waypoints from mission plan and the ground position information from sensors, i.e., Latitude & Longitude of the UAV along with its rate of change. Inner lateral control loop actuates aileron (δa) to follow the required roll angle (ϕ_{req}), and also issues commands to the vertical rudder (δr) to improve the dutch roll damping. The main task of the lateral guidance algorithm is to keep the cross-track error y_e

as small as possible, and also to keep $\chi_e \approx 0$ when $y \approx 0$. In case of a non-zero y, the guidance algorithm will manipulate χ_e by banking the vehicle to bring y to zero. Generally for very large track errors, a constant χ_e ($\leq \frac{\pi}{2}$) is desired, and when the track error reduces, χ_e is adjusted accordingly (a criterion known as the 'good helmsman' in literature (Rysdyk, 2006; Pettersen and Lefeber, 2001)).

Longitudinal guidance logic is separately designed to follow the altitude waypoints. Figure 3.11 shows the basic structure of the longitudinal guidance and control scheme. The outer guidance block gets current altitude (h) and its rate of change (to compute flight path angle (γ)) inputs from sensors, and the desired/reference altitude information from the mission path (mission can be modified during flight). On the basis of this information, the guidance block generates the required lift (L_{req}) command for the inner control loop to follow. The control loop generates commands to control surfaces (elevators) to meet the lift force required. Outer guidance logic should ensure tracking of the reference altitude during level flight, climb and decent phases.

3.5.2 3-D Guidance problem

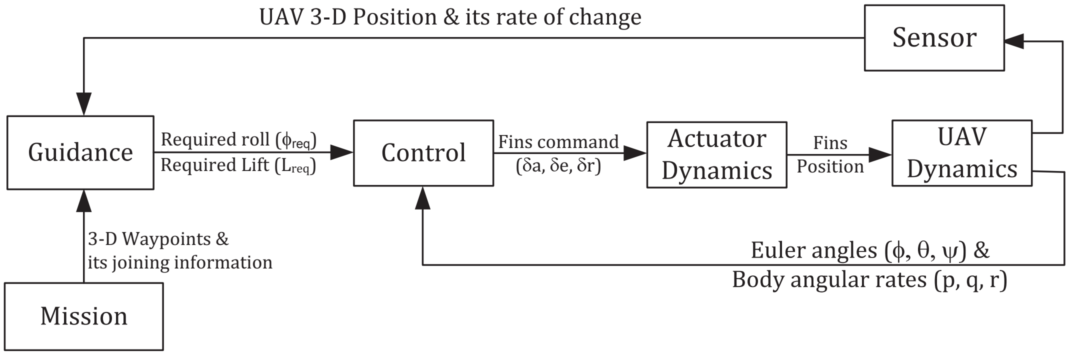


FIGURE 3.12: 3-D guidance and control structure for UAVs.

In 3-D path following case, complete mission information is sent to the 3-D guidance algorithm and the problem is solved in a single framework by considering the whole mission. In this case, generalized kinematics equations (3.32) are considered to design a single 3-D guidance scheme considering the coupling between longitudinal and lateral planes. Structure of the overall guidance and control scheme for 3-D path following is shown in Figure 3.12. The outer guidance block gets current inertial position (${}^{I}\mathbf{p}_B$) and its rate of change (ground velocity vector) $\mathbf{v}$ inputs from sensors and the desired 3-D path information from the mission plan. On the basis of this information, the guidance block computes the required longitudinal and lateral forces to keep the vehicle on the desired 3-D path. Output of the guidance block i.e., required lateral & longitudinal forces can be expressed in the form of required roll angle (ϕ_{req}) and required lift force (L_{req}), respectively. In this case inner control loop generates commands to the Aileron (δa), elevator (δe) and vertical rudder (δr) to follow the reference commands generated by the outer loop. Outer guidance loop is responsible to keep the UAV on the desired path and minimize any deviations.

Chapter 4

LINEAR SLIDING SURFACES FOR UAV GUIDANCE

4.1 Introduction

As discussed in detail in Chapter 2, sliding mode based design can be divided into two parts; the design of a stable sliding surface (also known as switching surface) followed by the design of a control law to force the system states trajectory onto the chosen sliding surface in finite time. System states trajectory starting from any point in the phase portrait is first attracted towards the sliding surface (reaching phase) and once achieved is subsequently forced to remain on the sliding surface (sliding phase). As it has already been mentioned, the sliding surface (or in general the switching hypersurface) completely determines the plant dynamics in the sliding phase. Therefore, the selection of this surface is one of the two major tasks in the process of sliding mode based design. The number of sliding surfaces should be equal to the number of control variables of the plant. Generally the sliding surface is a linear combination of the system state variables (known as linear sliding surface).

For UAV guidance problem, linear combination of state variables is not a feasible solution due to its poor performance and may lead to instability in some cases. Here in this chapter selection of a linear surface for the UAV guidance problem is discussed in detail. For decoupled lateral and longitudinal plane guidance, one sliding surface is required for each plane. In case of a 3-D guidance problem, we have two control variables implying the requirement of two sliding surfaces. Linear sliding surface design for this particular guidance problem is discussed in section 4.2. After the design of sliding surface, its performance limitations and stability is discussed. To address the instability problem in linear switching surface, an adhoc solution in terms of piece-wise linear sliding surface is proposed in section 4.3. Finally, the chapter is summarized in section 4.4.

4.2 Linear Sliding Surfaces and its Limitations

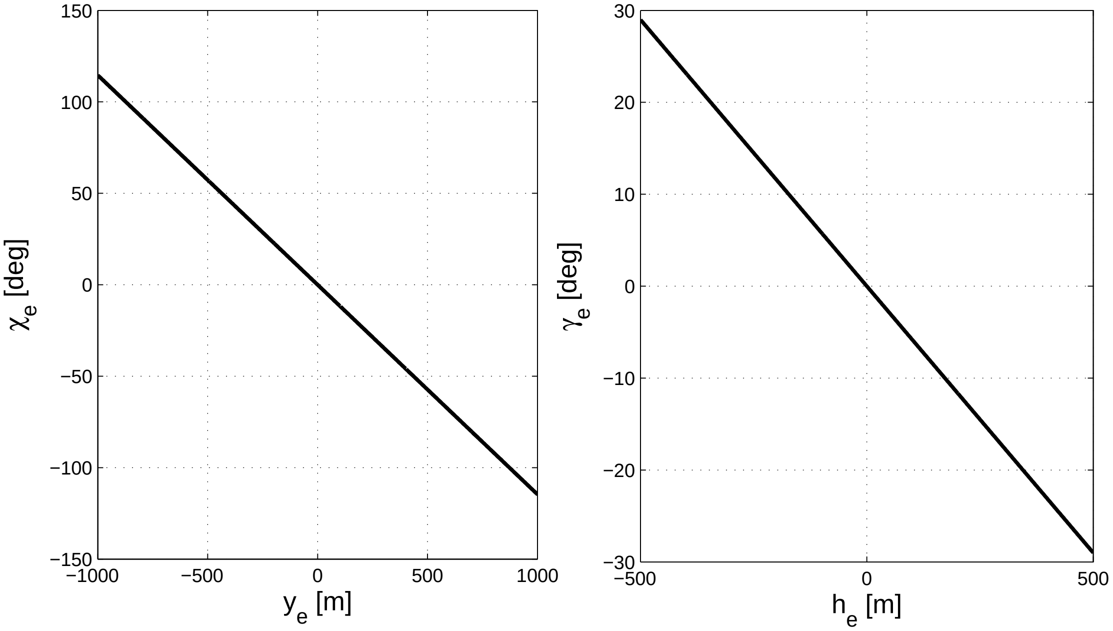


FIGURE 4.1: Linear sliding sufaces for lateral and vertical planes.

First let us choose two linear sliding surface functions $s_1 = \chi_e + \lambda_1 y_e$ and $s_2 = \gamma_e + \lambda_2 h_e$ (or $s_2 = \gamma_e - \lambda_2 z_e$) for some positive scalars λ_1 and λ_2 as shown in Figure 4.1. For the lateral sliding surface (s_1), the state variables y_e (cross-track error) and χ_e (error in course angle) are taken along the horizontal and vertical axes, respectively. Similarly for longitudinal sliding surface, the states variables z_e (vertical normal error to the path) and γ_e (error in flight path angle) are taken along the horizontal and vertical axes, respectively. In case of decoupled SMC guidance logic design for lateral and vertical planes, the first sliding surface s_1 can be utilized for the lateral plane and the second (s_2) for vertical plane guidance design. Both sliding surfaces will be required for the SMC based 3-D guidance design.

In the subsequent subsections (4.2.1 & 4.2.2), performance analysis, stability and control effort issues are discussed in detail for the lateral sliding surface (s_1). Similar arguments hold for the longitudinal (vertical) sliding surface (s_2) too and it is discussed in the subsection 4.2.3 briefly to avoid the duplication.

4.2.1 Performance with lateral SS: Simulation results

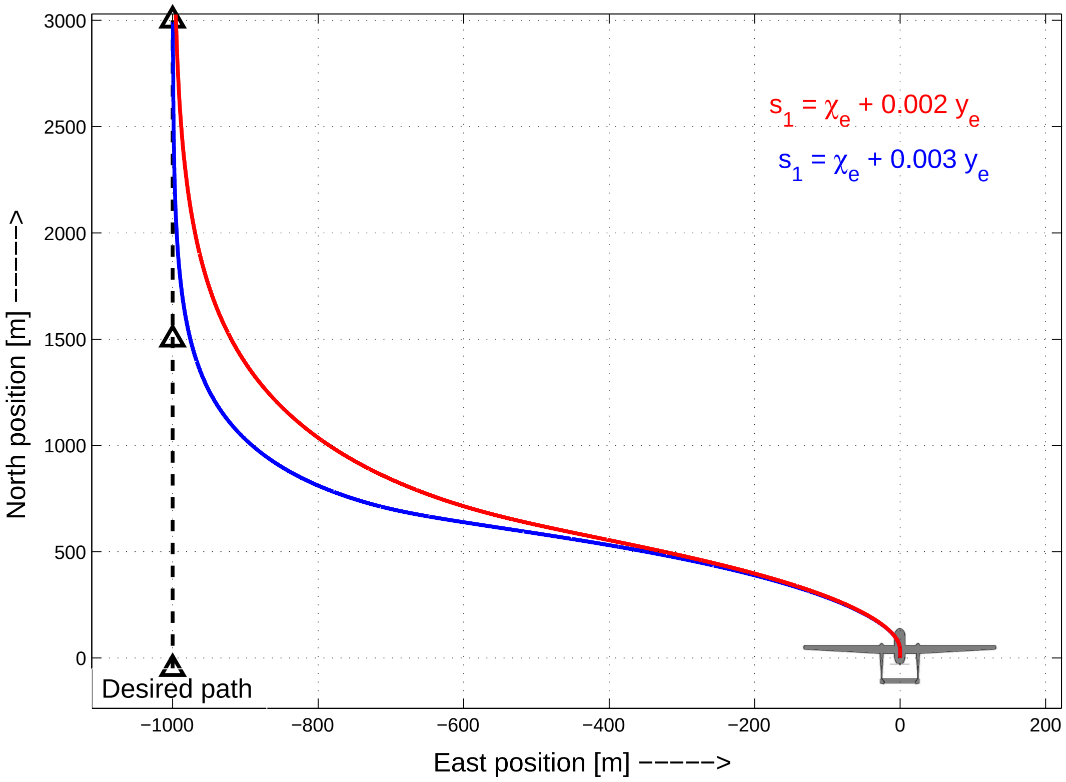


FIGURE 4.2: Linear SS: Actual path of UAV for different values of λ_1.

First to discuss the performance of lateral sliding surface analytically, motion on the sliding surface is described by the equation: $s_1 = \chi_e + \lambda_1 y_e = 0$ (or $\chi_e = -\lambda_1 y_e$). After substitution of lateral state equations (3.17) into the sliding motion we get:

$$\dot{y}_e = -V_g \sin(\lambda_1 y_e). \tag{4.1}$$

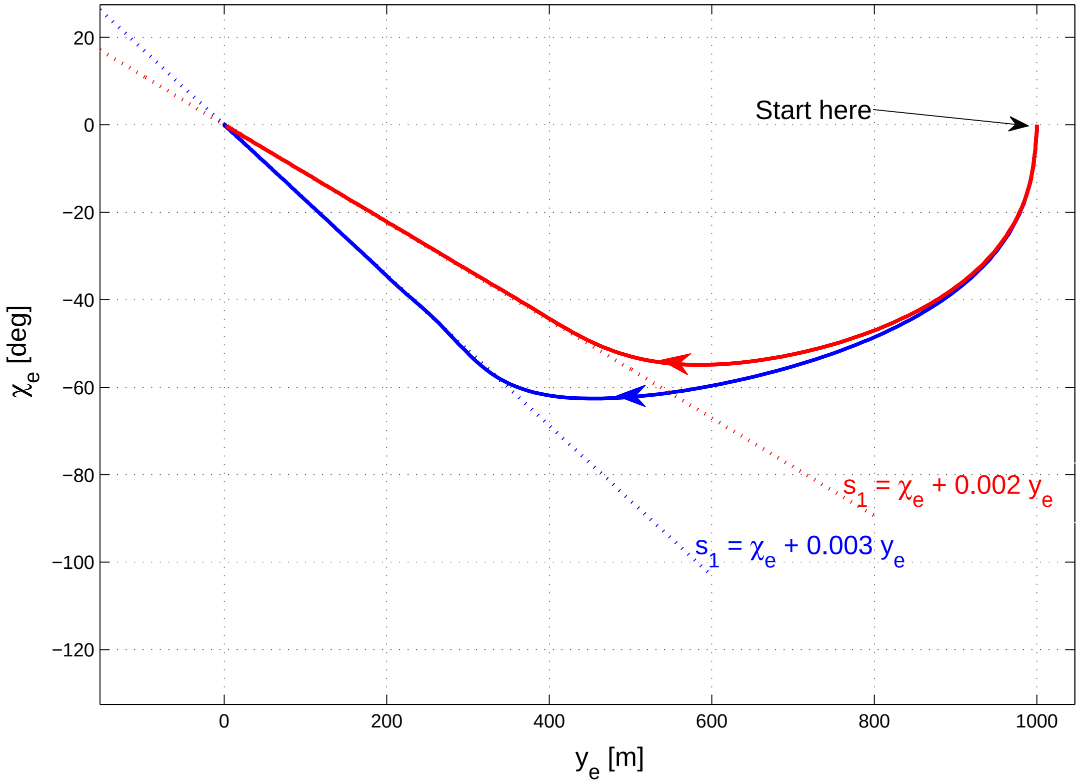


FIGURE 4.3: Linear SS: states trajectory for different values of λ_1.

In case of small cross-track error y_e, equation (4.1) can be approximated by $\dot{y}_e = -V_g \lambda_1 y_e$, which has an analytical solution:

$$y_e = y_{e_i} e^{-V_g \lambda_1 t}, \tag{4.2}$$

where y_{e_i} denotes the initial value of y_e. This shows that the cross-track error decreases exponentially with time, also the larger the value of the parameter λ_1, the more quickly the error magnitude decreases.

We now derive the control law that can achieve the sliding motion given above. First we find the *equivalent control*, i.e., the continuous control which can maintain $\dot{s}_1 = 0$ provided the dynamics are perfectly known (Slotine and Li, 1991). In this case the equivalent control is:

$$u_{eqv} = -\frac{\lambda_1 V_g^2}{g} \sin \chi_e + \dot{\chi}_p \frac{V_g}{g}. \tag{4.3}$$

The total control u is generated by adding the term $-k_1 \operatorname{sgn}(s_1)$ to the equivalent control, where 'sgn' represents the *sign* or *signum* function and k_1 is a positive constant. To avoid chattering in the control signal, $k_1 \operatorname{sgn}(s_1)$ can be approximated by $\frac{s_1}{|s_1|+\epsilon}$ (ϵ is a small positive number). The total control can be written as:

$$u = -\frac{\lambda_1 V_g^2}{g} \sin \chi_e + \dot{\chi}_p \frac{V_g}{g} - k_1 \operatorname{sgn}(s_1), \quad \phi_{req} = \arctan(u). \tag{4.4}$$

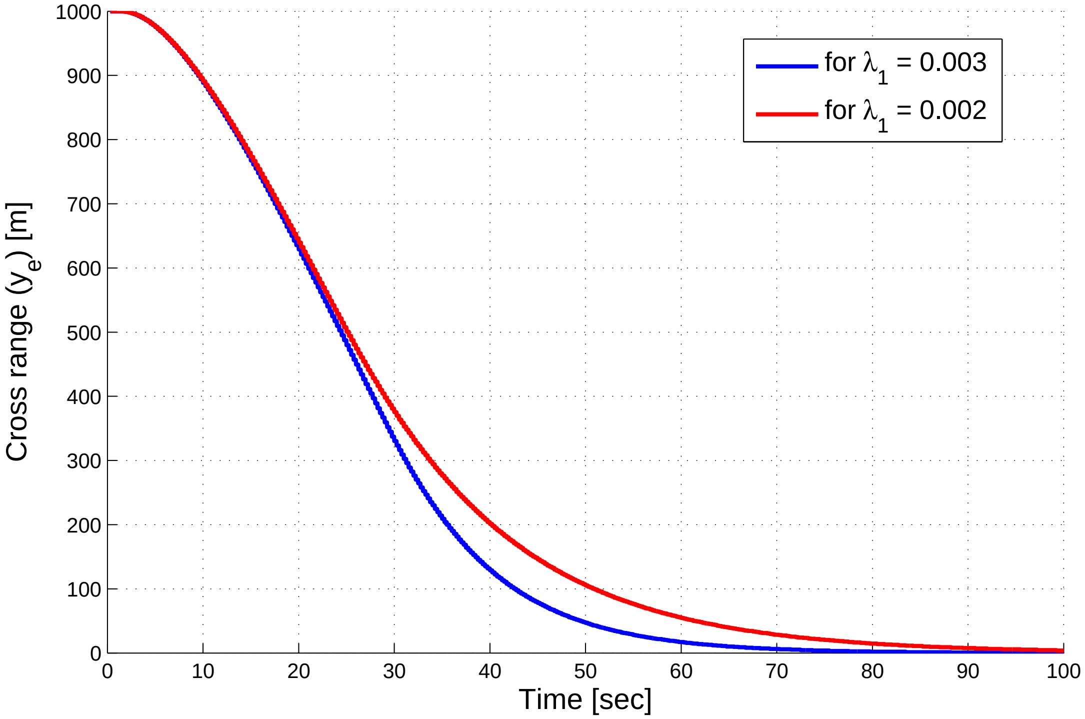


FIGURE 4.4: Linear SS: Cross track error versus time for different values of λ_1.

Simulations are performed using this guidance law ($k_1 = 0.7$) for an initial track error of 1000 meters with different values of λ_1 and results are shown in Figures

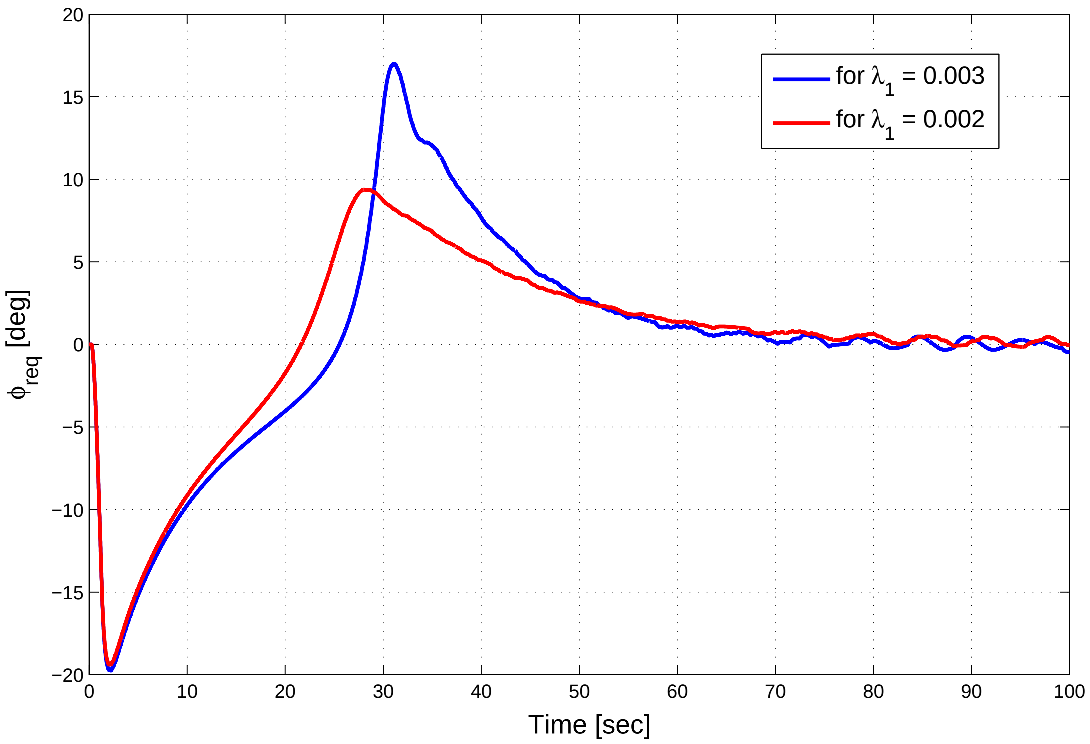


FIGURE 4.5: Linear SS: Roll required generated for different values of λ_1.

4.2 - 4.5. The proposed guidance law is implemented in 6-dof nonlinear simulation and the basic structure of blocks/sub-blocks is shown in Annex-1. Results for $\lambda_1 = 0.003$ and $\lambda_1 = 0.002$ are shown with BLUE and RED colors, respectively. Figure 4.3 shows the trajectory of the state variables in phase portrait form for different values of λ_1. The actual path of UAV on ground (top view) is shown in Figure 4.2. It is evident from these figures that larger $\lambda_1(0.003)$ gives faster convergence towards the desired path. The cross track error versus time is shown in Figure 4.4. The error decreases to about 10 meters in 65 seconds in the case of larger λ_1 (=0.003), but takes 85 seconds for the same reduction for the smaller λ_1 (=0.002) case. A bigger λ_1 therefore gives better performance, i.e., faster convergence to zero. The corresponding roll angle generated by the lateral guidance scheme is shown in Figure 4.5. Greater control effort is required in case of larger λ_1.

Conclusively, a bigger value of λ_1 is desired for good performance i.e., faster convergence towards the desired path. However a value of λ_1 selected for good performance for a particular cross track error y_e may not give good performance for all scenarios, as shown in the next subsection. For a smaller y_e the performance may be degraded, and for a larger y_e control saturation and instability may result.

4.2.2 Stability & control effort issues with lateral SS

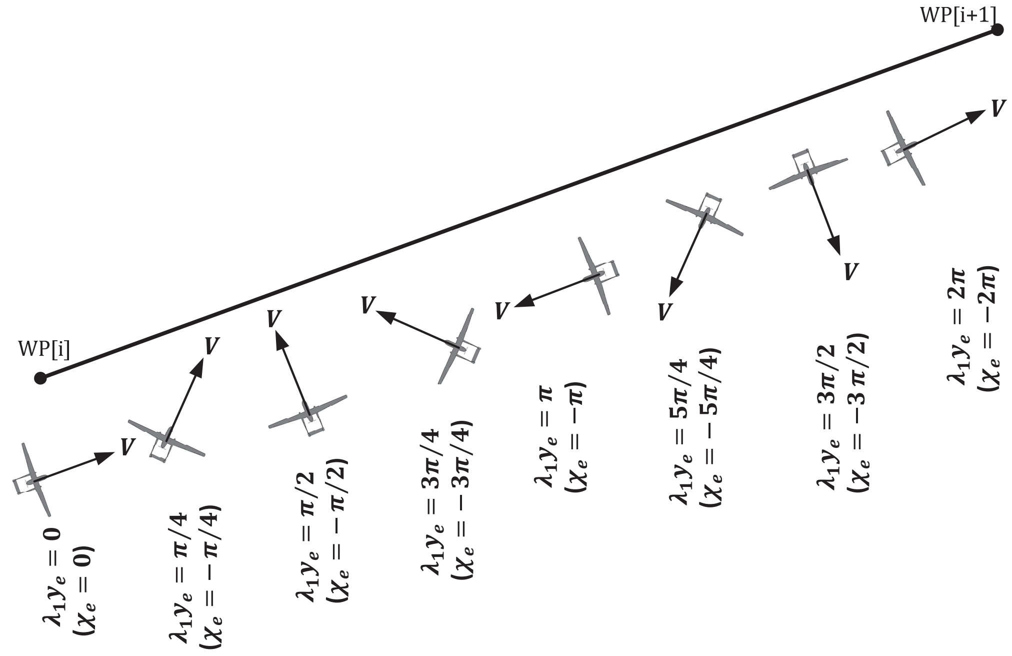


FIGURE 4.6: Linear SS: Physical interpretation of $\lambda_1 y_e$.

The above approach does not work if the initial error is large enough, in fact the error may not even converge. This is seen from the equation of motion on the sliding surface (4.1): $\dot{y}_e = -V_g \sin(\lambda_1 y_e) = -V_g \sin(-\chi_e)$. Suppose we start with a positive cross-track error y_e, then:

$$\dot{y}_e = \begin{cases} -ve, & \text{for } 0 < \lambda_1 y_e < \pi \quad (4.5a) \\ +ve, & \text{for } \pi < \lambda_1 y_e < 2\pi \quad (4.5b) \end{cases}$$

TABLE 4.1: Region of applicability of the linear sliding surface for different values of λ_1.

S. no.	λ_1	Region of applicability (km)
1	0.0002	$-7.854 < y_e < 7.854$
2	0.0004	$-3.927 < y_e < 3.927$
3	0.0007	$-2.243 < y_e < 2.243$
4	0.001	$-1.571 < y_e < 1.571$

The physical interpretation of $\lambda_1 y_e$ $(= -\chi_e)$ for a positive initial cross-track error is explained in Figure 4.6; for $\pi < \lambda_1 y_e < 2\pi$, the error diverges. In case $\frac{\pi}{2} < \lambda_1 y_e < \pi$, the cross-track error converges, but not along the desired direction of motion $WP[i] \rightarrow WP[i+1]$. So the general limitation of the linear sliding surface is its region of applicability, which is: $-\frac{\pi}{2} \leq \lambda_1 y_e \leq \frac{\pi}{2}$ or $-\frac{\pi}{2\lambda_1} \leq y_e \leq \frac{\pi}{2\lambda_1}$ (see Table 4.1).

Thus we see that in the design of the linear sliding surface, there has to be a compromise between the *'region of applicability'* and the *'rate of convergence'*, and there exists a trade-off between the two. If we strive for higher performance (bigger λ_1), the region of applicability gets limited, and if on the other hand we increase the region of applicability by reducing λ_1, the performance gets degraded. Therefore it is not possible to control large cross-track deviations with good performance (fast convergence) and guaranteed stability.

Simulations are performed using linear lateral sliding surface ($\lambda_1 = 0.0035$) for different cases of cross track error (y_e) as shown in Figures 4.7 - 4.10. The desired mission in north-east coordinate system is shown in Figure 4.7, the mission is to follow a straight line at east position of $800m$ heading towards north. In this case, the cross track error (y_e) is the eastward distance from the line $East = 800m$. Three scenarios of y_e (= -800 m, -1250 m & -1500 m) are generated in simulation and results are shown here with BLUE, RED and GREEN colors, respectively. Figure 4.7 shows the actual path of the UAV on ground (top view) for the three

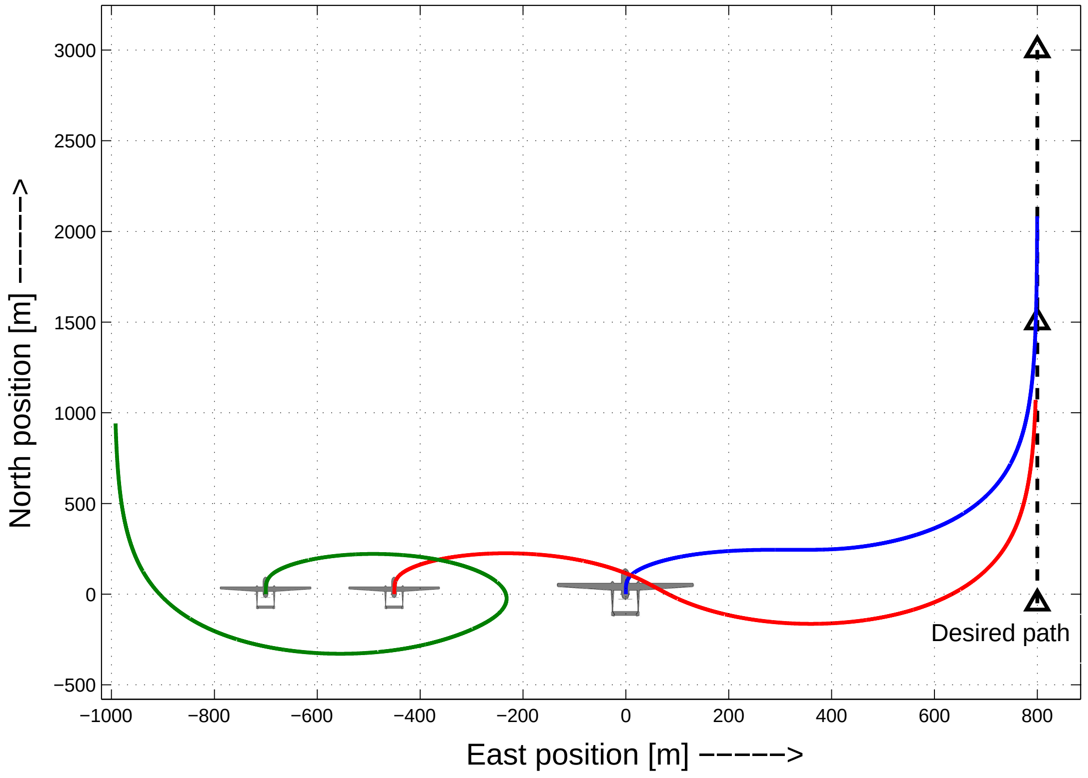


Figure 4.7: Linear SS: actual path of UAV for different values of y_e.

cases, and the corresponding trajectories of state variables are shown in Figure 4.8. The three cases are selected in such a way that state trajectory hits the sliding surface in three different regions i.e., $(0 < \chi_e < \frac{\pi}{2})$, $(\frac{\pi}{2} < \chi_e < \pi)$ and $(\pi < \chi_e < 2\pi)$. Figure 4.9 shows the cross track error versus time for the three cases; the corresponding roll angle generated by the guidance loop is shown in Figure 4.10.

In case of y_e= -800m (BLUE), it takes $\approx 15 seconds$ to reach the sliding surface and sliding starts from the position $(y_e \approx -450m, \chi_e \approx 89°)$ towards the origin. Cross track error converges to zero exponentially after start of the sliding motion as is evident from Figure 4.9. In the second case of y_e= -1250m (RED), although the cross track error (y_e) reduces to zero ultimately but not as efficiently as evident from the actual path followed by the UAV in Figure 4.7. In this case, the state

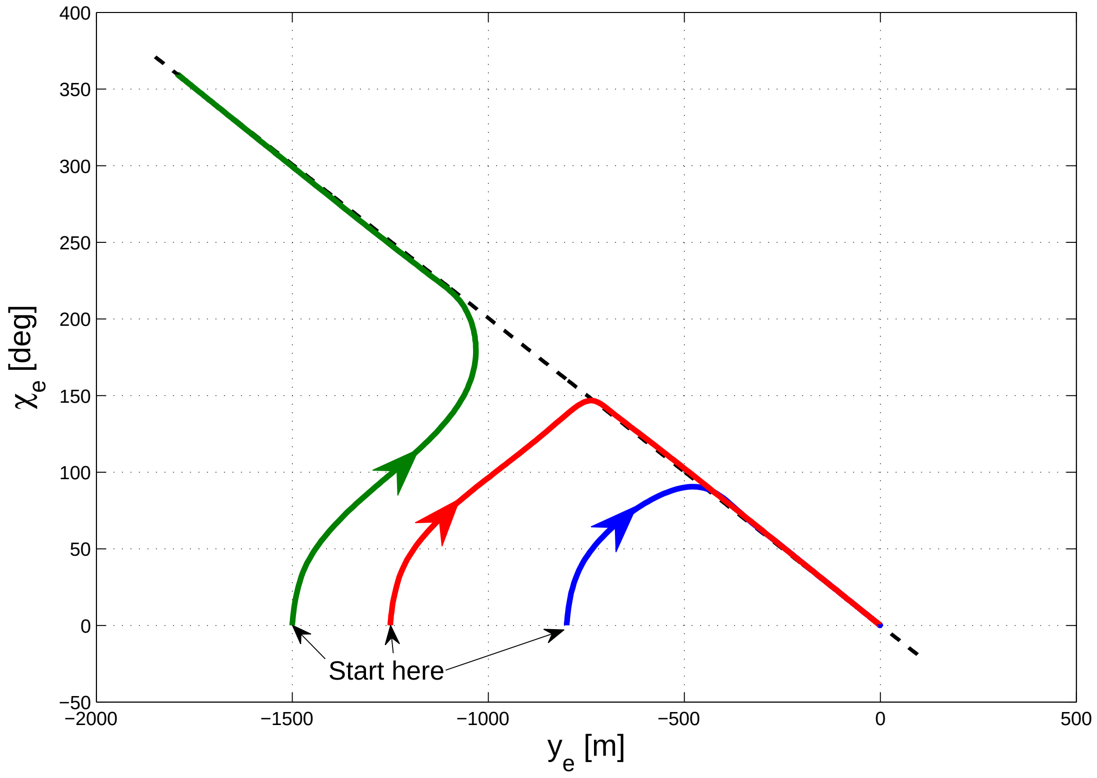


FIGURE 4.8: Linear SS: State variables trajectory for different values of y_e.

trajectory (Figure 4.8) takes $\approx 21 seconds$ to reach the sliding surface and starts sliding towards the origin from the position ($y_e \approx -750m, \chi_e \approx 146°$). The direction of motion is opposite to the desired direction $WP[i] \rightarrow WP[i+1]$ until χ_e reaches 90°. As expected, the degraded performance of linear sliding surface is observed in this case. In third case of y_e= -1500m (GREEN), unacceptable performance of the linear sliding surface is evident from the UAV's actual path (Figure 4.7) and also from y_e versus time (Figure 4.9). Trajectory of states is attracted towards the sliding surface in this case too and reaches the sliding surface at the position ($y_e \approx -1065m, \chi_e \approx 209°$), and thereafter unstable sliding started (in the opposite direction). If we reduce the sliding surface parameter (λ_1) magnitude so that the states trajectory hits the sliding surface at $\chi_e < 90°$ then definitely good performance in this case is possible, but performance will be degraded (slow convergence) for $|y_e| < 1500m$ and unstable behavior will be

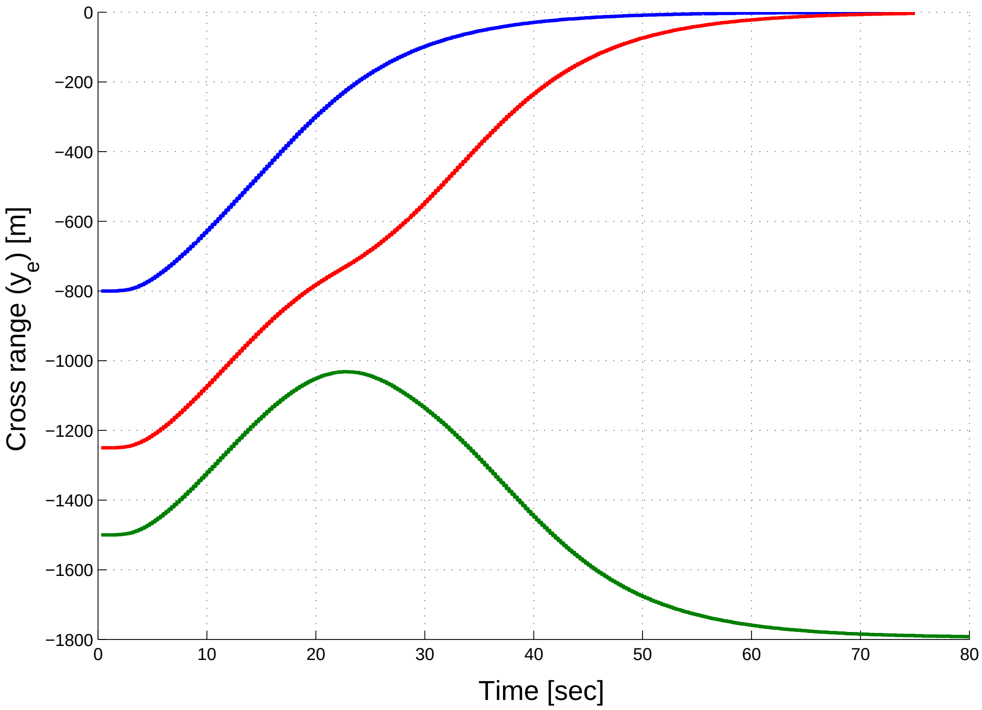


FIGURE 4.9: Linear SS: cross track error verus time for different values of y_e.

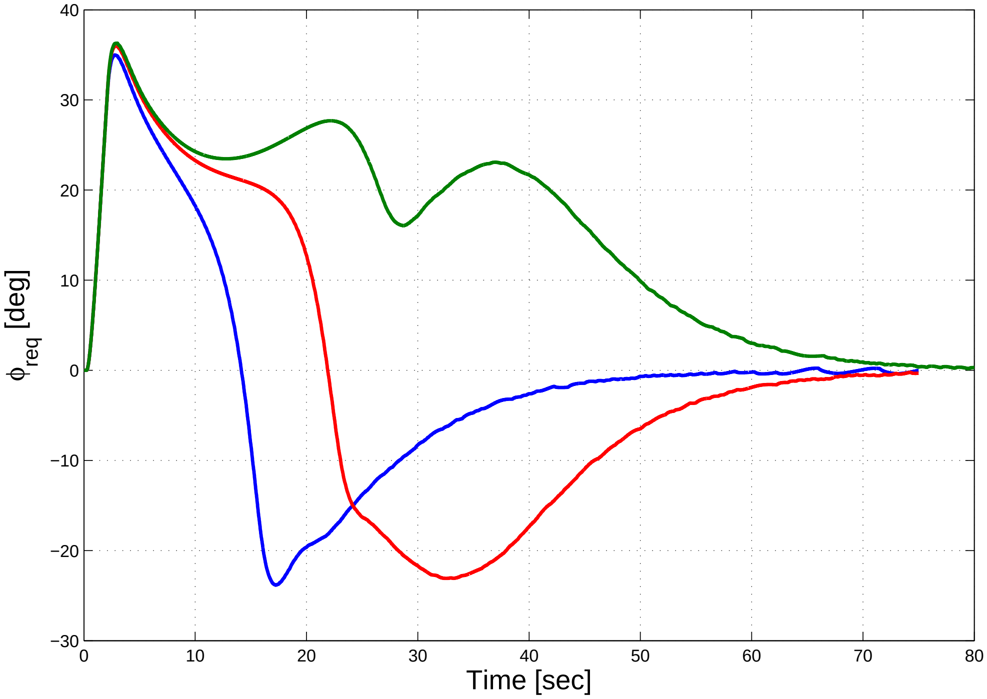


FIGURE 4.10: Linear SS: roll required verus time for different values of y_e.

TABLE 4.2: Control effort ϕ_{req} (in degrees) for different values of λ_1 and χ_e.

	$\lambda_1 = 0.001$	$\lambda_1 = 0.002$	$\lambda_1 = 0.003$	$\lambda_1 = 0.004$
$\chi_e = \pi/12$	-3.7750	-7.5175	-11.1966	-14.7847
$\chi_e = \pi/6$	-7.2641	-14.3019	-20.9268	-27.0155
$\chi_e = \pi/4$	-10.2187	-19.8257	-28.4043	-35.7938
$\chi_e = \pi/3$	-12.4500	-23.8242	-33.5179	-41.4482
$\chi_e = 5\pi/12$	-13.8336	-26.2199	-36.4547	-44.5666
$\chi_e = \pi/2$	-14.3019	-27.0155	-37.4087	-45.5597

seen for some $|y_e| > 1500m$. So conclusively a single linear sliding surface cannot guarantee performance and stability in all scenarios for the UAV guidance problem.

Another problem with the linear sliding surface is that of control saturation. This happens when the magnitude of $\lambda_1 y_e$ (or χ_e) is close to $\frac{\pi}{2}$ radians, as is evident from expression (4.3). Table 4.2 shows values of ϕ_{req} for different values of λ_1 and χ_e, for $V = 50$m/s and $g = 9.8065$ m/s^2. For higher values of λ_1, i.e., better performance, the control $\phi_{req} = \arctan(u_{eq})$ exceeds the maximum limit ($\phi_{max} = 35°$) for larger cross-track errors (larger χ_e).

4.2.3 Performance and stability of vertical SS

Similarly, the performance and stability of vertical sliding surface is described by the equation $s_2 = \gamma_e + \lambda_2 h_e = 0$ (or $\gamma_e = -\lambda_2 h_e$). After substitution of $\gamma_e = -\lambda_2 h_e$ in the state equations (3.25), we get:

$$\dot{h}_e = -V_g \cos\gamma_p \sin(\lambda_2 h_e). \tag{4.6}$$

For stable sliding motion, the sign of $\dot{h}_e$ should be negative for positive h_e, and vice-versa. However, negative $\dot{h}_e$ is not guaranteed for positive h_e as is evident from equation (4.6). As V_g and $\cos\gamma_p$ are positive parameters, so the sign of $\dot{h}_e$ depends on the sign of $\sin(\lambda_2 h_e)$. For positive h_e, negative $\dot{h}_e$ is guaranteed if

$-\frac{\pi}{2} \leq \lambda_2 h_e \leq \frac{\pi}{2}$. Hence the linear sliding surface is stable for the limited region of $-\frac{\pi}{2\lambda_2} \leq h_e \leq \frac{\pi}{2\lambda_2}$ only.

Another important conclusion regarding rate of convergence can also be drawn directly from equation (4.6). For stable region of $-\frac{\pi}{2\lambda_2} \leq h_e \leq \frac{\pi}{2\lambda_2}$, the rate of convergence (magnitude of $\dot{h}_e$) is directly related with the selection of λ_2. Bigger value of λ_2 implies bigger value of $\dot{h}_e$, and hence fast convergence of h_e towards zero. So for high performance we have to choose bigger value of λ_2, however bigger value of it will reduce the region of stability (or region of applicability) $-\frac{\pi}{2\lambda_2} \leq h_e \leq \frac{\pi}{2\lambda_2}$. Conclusively, performance and stability is not possible at the same time with a single linear sliding surface.

Remarks: If we choose $s_2 = \gamma_e - \lambda_2 z_e = 0$ (or $\gamma_e = \lambda_2 z_e$) then $\dot{z}_e = -V_g \sin(\lambda_2 z_e)$ and all above comments are also valid for this too.

4.3 Adhoc Solution: Piece-wise Linear SS for Lateral Guidance

If we choose a linear sliding surface for lateral guidance problem of UAVs as shown in Figure 4.1 then the slope of linear sliding surface is directly related with the performance during cross track errors. Large slope of linear sliding surface implies big χ_e even for small track errors, and hence the track error decrease at faster rate. On the other hand, this linear sliding surface implies a large $\chi_e (> \frac{\pi}{2})$ for a large track error (unstable sliding surface in this case). So good performance in both cases (large and small track errors) is not possible with a single linear sliding surface. The performance in sliding mode control depends on design of the sliding surface, in our case objective is stability and performance in case of both small and large track errors. To avoid instability in case of large track errors, a piece-wise linear sliding surface for lateral plane guidance is proposed here to meet the small & large cross track requirements.

4.3.1 Proposed sliding surface

The instability and degraded performance problems in linear sliding surface arise due to the magnitude of χ_e greater than $\frac{\pi}{2}$ for large values of y_e. The idea is to propose a sliding surface that has $|\chi_e| < \frac{\pi}{2}$ for entire range of y_e. Instead of extending the line linearly for large values of y_e, the sliding surface is extended horizontally after reaching $\chi_{e_{lim}}$ ($< \frac{\pi}{2}$) as shown in Figure 4.11. Equation of the proposed sliding surface for lateral plane is:

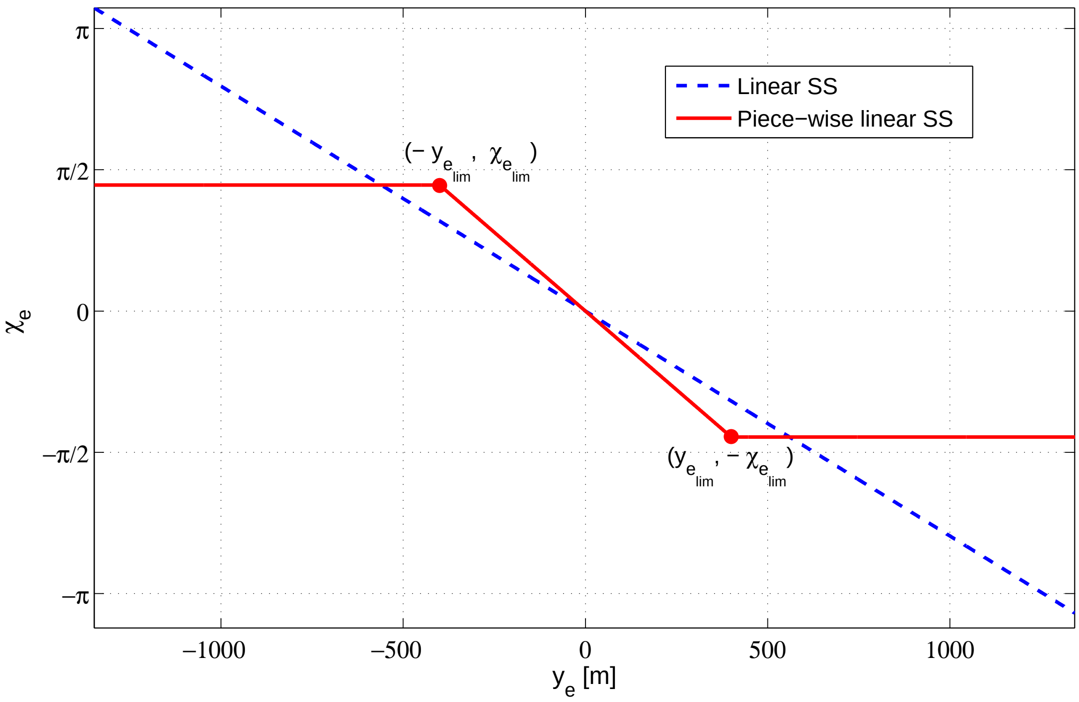


FIGURE 4.11: Proposed piece-wise linear sliding surface for lateral plane.

$$
s_1 = \begin{cases} \chi_e - \chi_{e_{lim}}, & \text{for } y_e < -y_{e_{lim}} \qquad (4.7a) \\ \chi_e + \lambda y_e, & \text{for } -y_{e_{lim}} \le y_e \le y_{e_{lim}} \qquad (4.7b) \\ \chi_e + \chi_{e_{lim}}, & \text{for } y_e > y_{e_{lim}} \qquad (4.7c) \end{cases}
$$

Here $y_{e_{lim}}$ and $\chi_{e_{lim}}$ ($< \frac{\pi}{2}$) are positive numbers to be selected. Basically this piece-wise linear sliding surface has three parts: the two horizontal lines ensure the performance as well as the stability during large track errors by keeping $|\chi_e| < \frac{\pi}{2}$, while the performance in case of small track errors ($-y_{e_{lim}} \leq y_e \leq y_{e_{lim}}$) is related with the central linear part of the sliding surface.

4.3.2 Stability of sliding surface

Stability of the sliding surface can be ensured by finding the analytical solution of the proposed sliding surface. Overall phase portrait is divided into three parts for stability analysis: ($y_e < -y_{e_{lim}}$) , ($-y_{e_{lim}} \leq y_e \leq y_{e_{lim}}$) and ($y_e > y_{e_{lim}}$).

In case-1 ($y_e < -y_{e_{lim}}$), motion on the proposed sliding surface is described by $s_1 = 0$ implying:

$$\chi_e = \chi_{e_{lim}}. \tag{4.8}$$

Substituting this value of χ_e in state equations (3.17), we have following equation for sliding motion:

$$\dot{y}_e = V_g \sin \chi_{e_{lim}}, \tag{4.9}$$

where V_g and $\chi_{e_{lim}}$ ($< \frac{\pi}{2}$) are positive constants. Analytical solution of above equation (4.9) is:

$$y_e = V_g \sin \chi_{e_{lim}} t + y_{e_{initial}}. \tag{4.10}$$

Equation (4.10) shows that y_e will increase from initial value ($y_{e_{initial}}$) as time (t) increase, and will reach '$-y_{e_{lim}}$' in finite time. So in this case ($y < -y_{lim}$), it is guaranteed that (y_e, χ_e) will reach the point ($-y_{e_{lim}}, \chi_{e_{lim}}$) while sliding.

Similarly **in case-3** $(y > y_{_{lim}})$, motion on sliding surface is described by:

$$\dot{y}_e = -V_g \sin \chi_{e_{lim}}. \tag{4.11}$$

Analytical solution of equation (4.11) is:

$$y_e = -V_g \sin \chi_{e_{lim}} t + y_{e_{initial}}. \tag{4.12}$$

Equation (4.12) shows y_e will decrease from initial $y_{initial}$ as time increase and eventually will arrive at the point $(y_{e_{lim}}, -\chi_{e_{lim}})$ in finite time.

In case-2 $(-y_{_{lim}} \leq y_e \leq y_{_{lim}})$, motion on the sliding surface is described by:

$$s_1 = \chi_e + \lambda_1 y_e = 0, \tag{4.13}$$

where λ_1 is a positive number. After substitution of equation (4.13) in state equations (3.17), we have

$$\dot{y}_e = -V_g \sin(\lambda_1 y_e). \tag{4.14}$$

As $-y_{_{lim}} \leq y_e \leq y_{_{lim}}$ in this case and $-\frac{\pi}{2} \leq \lambda_1 y_e \leq \frac{\pi}{2}$, and hence $\sin(\lambda_1 y_e)$ will have same sign as y_e. Conclusively, $\dot{y}_e$ is a negative number for a positive y_e, and vice-versa. So the cross track error y_e will converge towards origin in either case.

In summary, the state trajectory will first slide along the horizontal part of the surface to reach the sliding surface $s_1 = \chi_e + \lambda_1 y_e$ and afterward start sliding along that to reach the origin. In case of a small initial y_e, the state trajectory will get attracted towards the central part of the surface directly.

4.3.3 Lateral guidance law

Sliding mode based guidance law for cross track control is the sum of equivalent control (continuous control law that would maintain $\dot{s} = 0$ if the dynamics were exactly known) and the discontinuous control '$-k\,\mathrm{sgn}(s)$' to cater for parametric uncertainties and disturbances. In case of piece-wise linear sliding surface, we have different equivalent controls for three different regions. Using equation (4.7), we have

$$\dot{s}_1 = \begin{cases} \dot{\chi}_e, & \text{for } y_e < -y_{e_{lim}} \quad (4.15a) \\ \dot{\chi}_e + \lambda_1 \dot{y}_e, & \text{for } -y_{e_{lim}} \le y_e \le y_{e_{lim}} \quad (4.15b) \\ \dot{\chi}_e, & \text{for } y_e > y_{e_{lim}} \quad (4.15c) \end{cases}$$

and hence the expression for equivalent control is:

$$u_{eqv} = \begin{cases} \dfrac{\hat{V}_g}{\hat{g}} \dot{\chi}_p, & \text{for } y_e < -y_{e_{lim}} \quad (4.16a) \\ -\dfrac{\lambda_1 \hat{V}_g^2}{\hat{g}} \sin \chi_e + \dfrac{\hat{V}_g}{\hat{g}} \dot{\chi}_p, & \text{for } -y_{e_{lim}} \le y_e \le y_{e_{lim}} \quad (4.16b) \\ \dfrac{\hat{V}_g}{\hat{g}} \dot{\chi}_p, & \text{for } y_e > y_{e_{lim}} \quad (4.16c) \end{cases}$$

where $\hat{g}$ and $\hat{V}_g$ are the measured (or estimated) values of gravity and ground velocity, respectively.

The total control u is generated by adding the term '$-k\,\mathrm{sgn}(s)$' to the equivalent control, where 'sgn' represents the *sign* or *signum* function and k is a positive constant. The total control now becomes

$$u = \begin{cases} \dfrac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - k\,\mathrm{sgn}(s_1), & \text{for } y_e < -y_{e_{lim}} \qquad (4.17a) \\ -\dfrac{\lambda_1 \hat{V}_g^2}{\hat{g}}\sin\chi_e + \dfrac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - k\,\mathrm{sgn}(s_1), & \text{for } -y_{e_{lim}} \le y_e \le y_{e_{lim}} \qquad (4.17b) \\ \dfrac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - k\,\mathrm{sgn}(s_1), & \text{for } y_e > y_{e_{lim}} \qquad (4.17c) \end{cases}$$

and

$$\phi_{req} = \arctan(u). \tag{4.18}$$

To see the **reachability condition** ($s_1\dot{s}_1 < 0$), the Lyapunov candidate function $\mathbf{W} = \frac{1}{2}s_1^2$ is selected. The derivative of Lyapunov function $\mathbf{W}$ is

$$\dot{\mathbf{W}} = s_1\dot{s}_1 = s_1 \begin{cases} \left(\dfrac{g}{V_g}u - \dot{\chi}_p\right), & y_e < -y_{e_{lim}} \qquad (4.19a) \\ \left(\dfrac{g}{V_g}u + \lambda_1 V_g \sin\chi_e - \dot{\chi}_p\right), & -y_{e_{lim}} \le y_e \le y_{e_{lim}} \qquad (4.19b) \\ \left(\dfrac{g}{V_g}u - \dot{\chi}_p\right), & y_e > y_{e_{lim}} \qquad (4.19c) \end{cases}$$

Replacing the value of u in equation (4.19) from equation (4.17) and neglecting the measurement error in the gravity term ($g \approx \hat{g}$), we have

$\dot{\mathbf{W}} =$

$$-s_1 \begin{cases} \left(\dot{\chi}_p\left(1 - \dfrac{\hat{V}_g}{V_g}\right) + \dfrac{gk}{V_g}\mathrm{sgn}(s_1)\right), & \text{for } y_e < -y_{e_{lim}} \\ \left(\dot{\chi}_p\left(1 - \dfrac{\hat{V}_g}{V_g}\right) + \lambda_1 V_g \sin\chi_e\left(\dfrac{\hat{V}_g^2}{V_g^2} - 1\right) + \dfrac{gk}{V_g}\mathrm{sgn}(s_1)\right), & \text{for } -y_{e_{lim}} \le y_e \le y_{e_{lim}} \\ \left(\dot{\chi}_p\left(1 - \dfrac{\hat{V}_g}{V_g}\right) + \dfrac{gk}{V_g}\mathrm{sgn}(s_1)\right), & \text{for } y_e > y_{e_{lim}} \end{cases}$$

Assume a worst case error of 10% in the measurement of velocity (i.e., $0.9V \leq \hat{V} \leq 1.1V$), we have

$\dot{\mathbf{W}} =$

$$-s_1 \begin{cases} \left(0.1\dot{\chi}_p + \dfrac{gk}{V_g}\operatorname{sgn}(s_1)\right), & \text{for } y_e < -y_{e_{lim}} \qquad (4.21a) \\ \left(0.1\dot{\chi}_p + 0.21\lambda_1 V_g \sin\chi_e + \dfrac{gk}{V_g}\operatorname{sgn}(s_1)\right), & \text{for } -y_{e_{lim}} \leq y_e \leq y_{e_{lim}} \qquad (4.21b) \\ \left(0.1\dot{\chi}_p + \dfrac{gk}{V_g}\operatorname{sgn}(s_1)\right), & \text{for } y_e > y_{e_{lim}} \qquad (4.21c) \end{cases}$$

In case of (4.21a) and (4.21c), $\dot{\mathbf{W}}$ will be negative definite if $\left|\frac{gk}{V_g}\right| > |0.1\dot{\chi}_p|$. However, in worst case of (4.21b), $\left|\frac{gk}{V_g}\right|$ should be greater than $|0.1\dot{\chi}_p + 0.21\lambda_1 V_g \sin\chi_e|$ for $\dot{\mathbf{W}}$ to be negative definite. Hence the gain k selected to make $\dot{\mathbf{W}} < 0$ in case of (4.21b) will also ensure negative definiteness of $\dot{\mathbf{W}}$ for the other two cases. Taking the maximum value of $|\sin(\chi_e)| = 1$, existence of sliding mode is guaranteed if

$$k > \frac{0.21\lambda_1 V_g^2}{g} + \frac{0.1V_g}{g}\left|\dot{\chi}_p\right|. \tag{4.22}$$

In order to bound the roll angle (ϕ), it is necessary to bound the reference roll angle (ϕ_{req}) which is the output of the guidance loop and depends on the gain k. As is evident from equation (4.22), the control gain k is directly proportional to the sliding surface coefficient λ_1 for $-y_{e_{lim}} \leq y_e \leq y_{e_{lim}}$. Hence any arbitrarily large value of λ_1 cannot be chosen. Now we will derive a condition on λ_1 to keep $|\phi_{req}| < \phi_{max}$ (or $u < \tan(\phi_{max})$). In order to bound the control effort, its maximum value should be bounded, i.e.,

$$\left|\frac{\lambda_1 \hat{V}_g^2}{\hat{g}}\sin\chi_e\right| + \left|\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p\right| + k \leq \tan(\phi_{max}), \tag{4.23}$$

or in extreme case $(|\sin(\chi_e)| = 1)$

$$\frac{\lambda_1 \hat{V}_g^2}{\hat{g}} + \frac{\hat{V}_g}{\hat{g}} |\dot{\chi}_p| + k \leq \tan(\phi_{max}). \tag{4.24}$$

After simplification, we have

$$\lambda_1 \leq \left(\tan \phi_{max} - k - \frac{\hat{V}_g}{\hat{g}} |\dot{\chi}_p| \right) \frac{\hat{g}}{\hat{V}_g^2}. \tag{4.25}$$

Therefore, the values of λ_1 and k should be selected such that both equations (4.22) and (4.25) are satisfied to ensure reachability and to avoid control saturation.

4.3.4 Simulation results

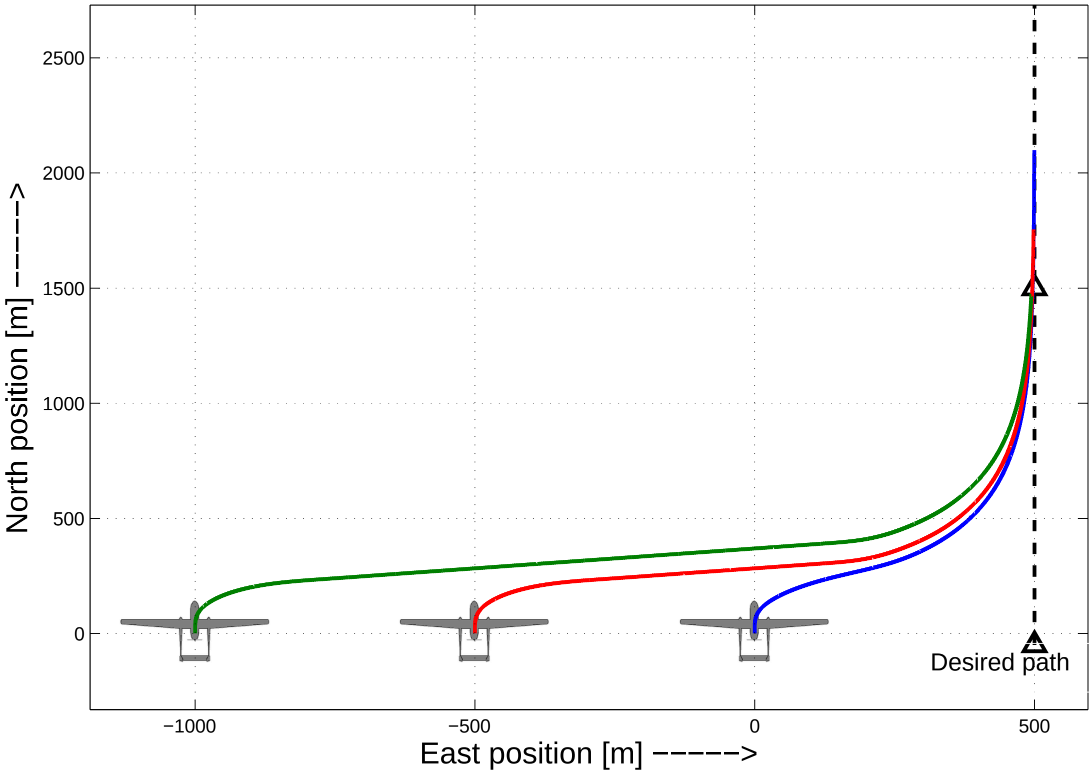


FIGURE 4.12: Piece-wise linear SS: Actual path of UAV for different values of y_e.

Simulations are conducted with the proposed piece-wise linear sliding surface to show the guidance logic performance for different initial cross track errors. The desired path is a straight line directed towards North in all these cases. The sliding surface used in simulation is:

$$s_1 = \begin{cases} \chi_e - 1.4, & \text{for } y_e < -400 \qquad (4.26a) \\ \chi_e + 0.0035 y_e, & \text{for } -400 \leq y_e \leq 400 \qquad (4.26b) \\ \chi_e + 1.4, & \text{for } y_e > 400 \qquad (4.26c) \end{cases}$$

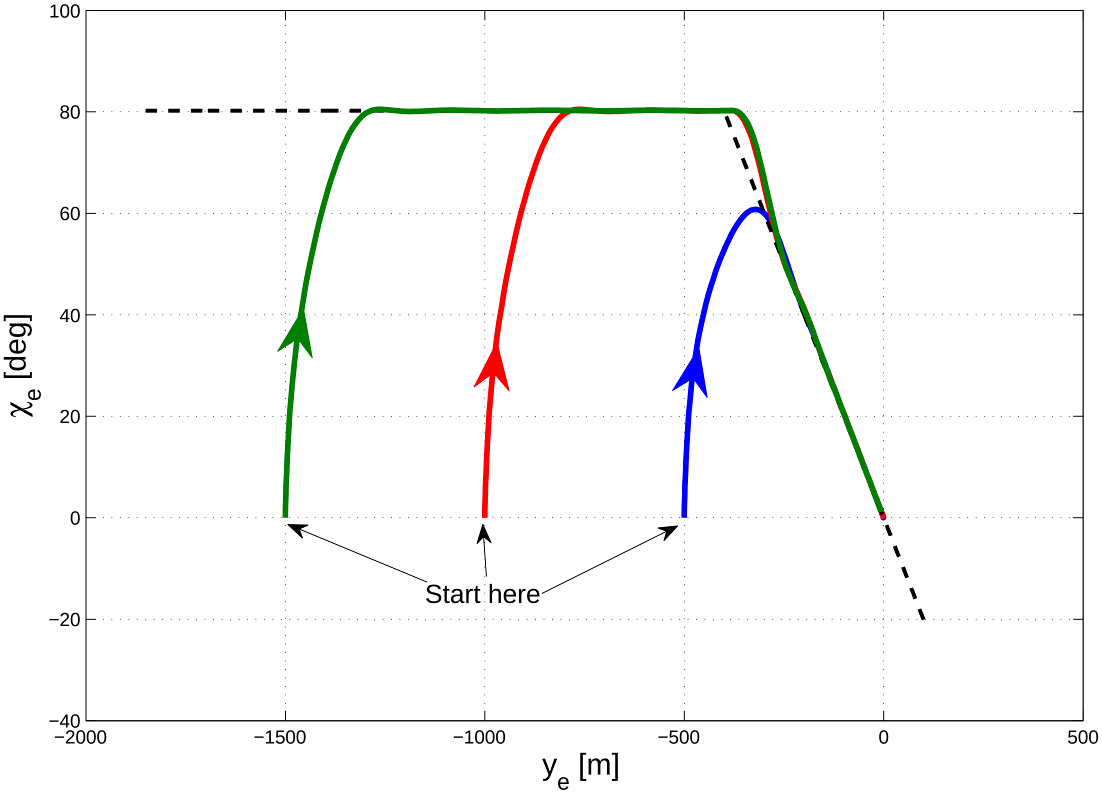


FIGURE 4.13: Piece-wise linear SS: States trajectory for different values of y_e.

The selected value of gain 'k' is 0.6 that satisfies both reachability and control boundedness conditions for $-400 \leq y_e \leq 400$. A boundary layer approximation $\text{sgn}(s_1) \approx \frac{s_1}{|s_1|+\epsilon}$ is used to avoid chattering in the computation of ϕ_{req}, and hence the total guidance law used for simulations is:

$$u = \begin{cases} -0.75\dfrac{s_1}{|s_1| + 0.3}, & \text{for } y_e < -400 \quad (4.27a) \\ -\dfrac{0.0035\hat{V}_g^2}{\hat{g}}\sin\chi_e - 0.6\dfrac{s_1}{|s_1| + 0.3}, & \text{for } -400 \leq y_e \leq 400 \quad (4.27b) \\ -0.75\dfrac{s_1}{|s_1| + 0.3}, & \text{for } y_e > 400 \quad (4.27c) \end{cases}$$

and

$$\phi_{req} = \arctan(u). \tag{4.28}$$

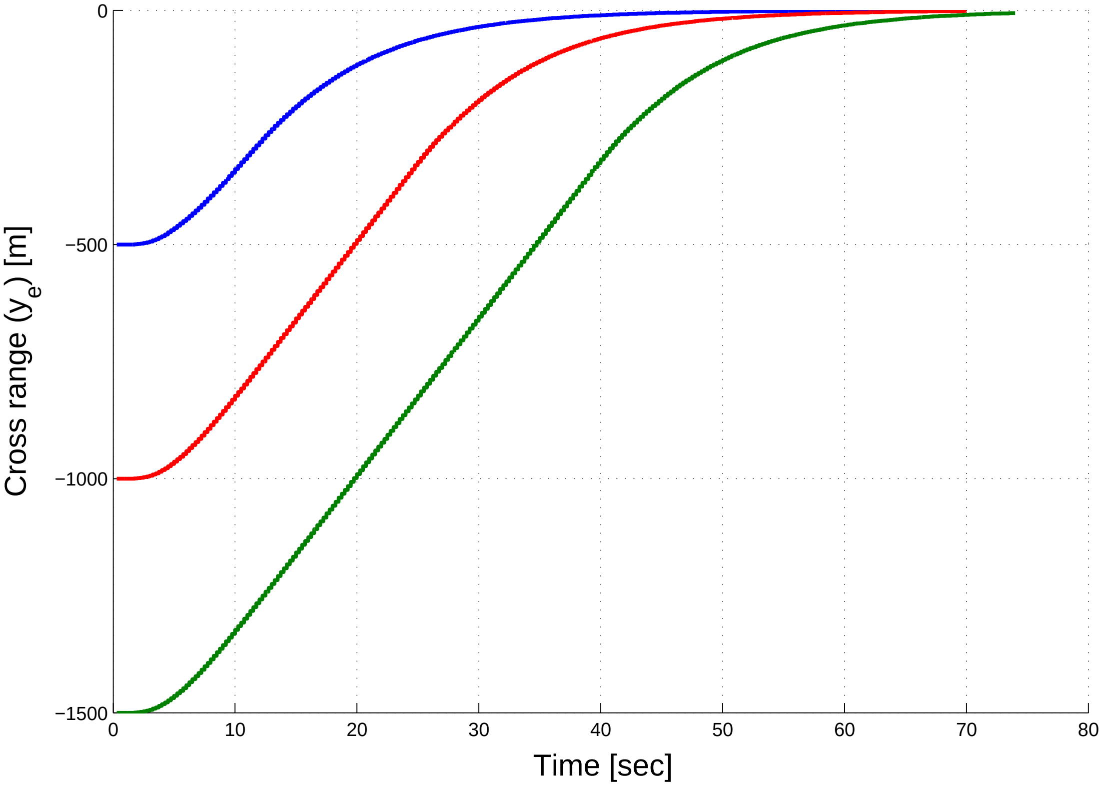


FIGURE 4.14: Piece-wise linear SS: Cross track error versus time for different values of y_e.

Three scenarios with initial cross track errors y_e = -500 m, -1000 m & -1500 m are generated in the 6-dof nonlinear simulation (see Annex-1 for simulation structure) and results are shown here with BLUE, RED and GREEN colors, respectively. Figure 4.12 shows the actual ground track of the UAV (top view) for the three cases, and the corresponding state trajectories along-with the sliding surface are shown in Figure 4.13. Cross track error and corresponding roll angle generated by the guidance loop versus time for the three cases are shown in Figures 4.14 and 4.15, respectively.

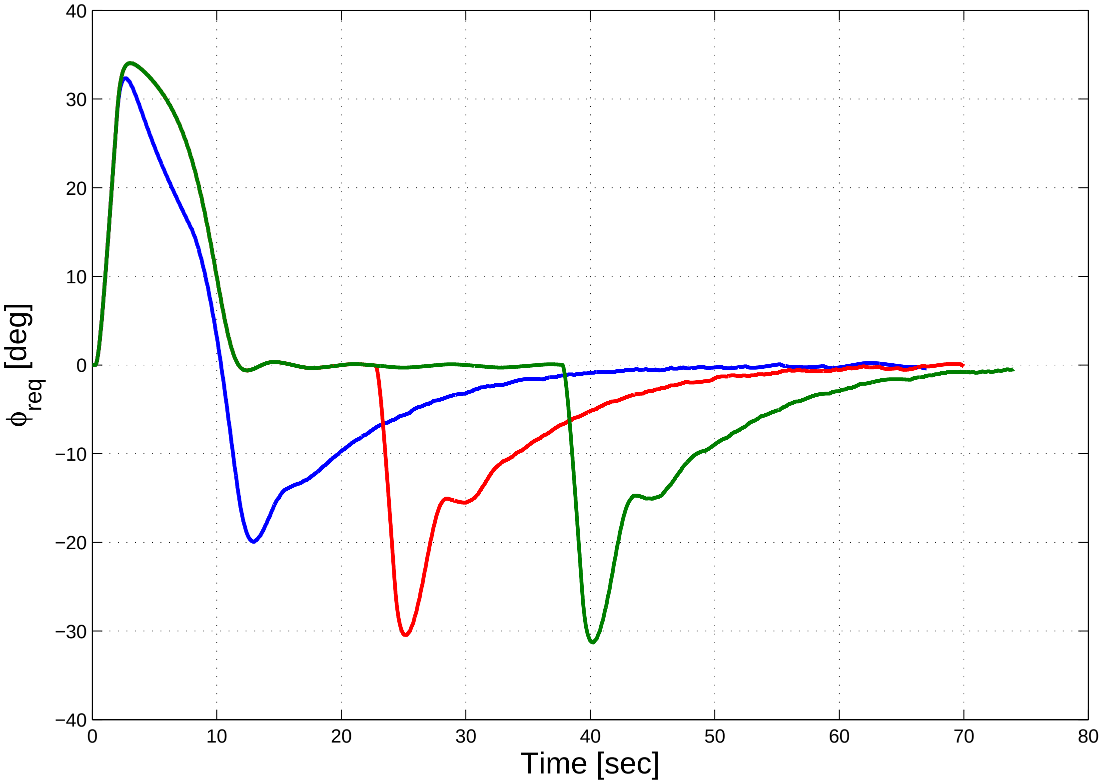


FIGURE 4.15: Piece-wise linear SS: Roll required generated for different values of y_e.

In case of y_e= -500m (BLUE), the state trajectory is attracted towards the central linear part of the sliding surface and the sliding motion is similar to that of section 4.2. On the other hand, the state trajectories behave differently in case of y_e= -1000m (RED) and y_e= -1500m (GREEN). In both cases, the trajectory is first attracted towards the horizontal part of the sliding surface and slides along that to arrive at the point $(y_e, \chi_e) = (-400m, 80.2°)$ as shown in Figure 4.13. In all scenarios, χ_e remains less than $\frac{\pi}{2}$ and thus there are no issues of stability as is evident from Figure 4.12. However, there is a discontinuity from horizontal part of the sliding surface to the central linear part which is hard to follow in practice.

4.4 Summary of the Chapter

Design of a stable sliding surface is an essential requirement for sliding mode based control design. Linear combination of states is a natural and commonly used option for sliding surface design. Here in this chapter, it is first shown that linear sliding surface is not a feasible solution for the UAV guidance problem because of stability and performance issues (for large initial errors). An ad-hoc solution in terms of a combination of different linear sliding surfaces (piece-wise linear) is proposed to cater for these problems. However, good performance for all scenarios is still not possible with the proposed discontinuous and piece-wise linear sliding surface. A high performance stable sliding surface is therefore required that ensures good performance in all scenarios.

Chapter 5

HIGH PERFORMANCE SLIDING MANIFOLDS

5.1 Introduction

As discussed earlier in Chapter 2, sliding mode based control law design can be divided into two parts, i.e., the design of stable sliding manifolds (number of sliding manifolds equal to the number of control inputs) followed by the design of a control law to force the system states onto the chosen manifolds in finite time despite the presence of model imprecision and disturbances (Edwards and Spurgeon, 1998; Utkin, 1977). The design of manifolds (sliding surfaces/switching surfaces) should address all constraints and required specifications, therefore they should be designed optimally to meet all the requirements. Design of the switching surface is vital because the closed loop dynamics is governed by the parameters of the switching surface. Therefore, the sliding surface should be designed to meet the closed loop specifications. Linear combinations of the states are commonly used as sliding manifolds (known as linear sliding surfaces), but it is not feasible in every case as shown in Chapter 4. Many systems demand high performance with robustness and stability in all scenarios. To achieve high performance in all scenarios, we propose nonlinear manifolds in this chapter for the outer guidance loop design for UAVs. In the subsequent sections, nonlinear sliding manifolds are presented for lateral and longitudinal planes, followed by the derivation of criterion for optimum sliding surface coefficients for the proposed nonlinear manifolds.

5.2 Nonlinear Sliding Surface for the Lateral Plane

For good performance in case of both small and large cross track errors, and to keep the magnitude of χ_e less than $\frac{\pi}{2}$, we now propose a high performance nonlinear sliding manifold for the lateral plane. The proposed sliding surface is:

$$s_1 = \chi_e + c_1 \arctan(c_2 y_e) = 0, \tag{5.1}$$

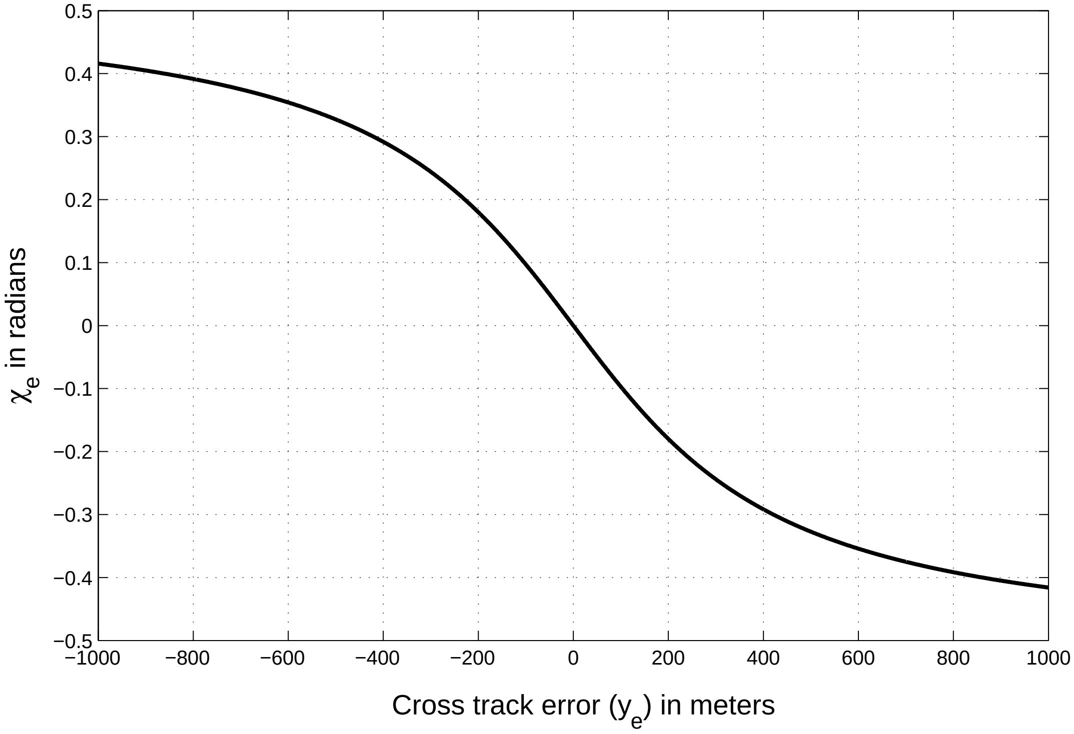
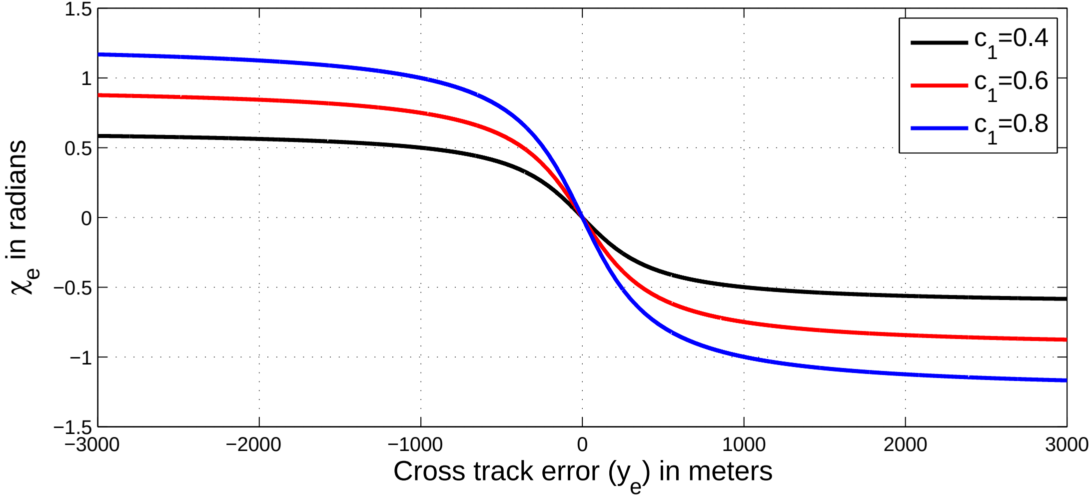


Figure 5.1: The proposed nonlinear sliding surface for lateral plane.

where the constants $c_1, c_2 \in \Re$ (the set of real numbers); later it will be shown that for sliding surface stability we need $c_1 c_2 > 0$. Figure 5.1 shows a plot of the nonlinear surface for particular values of c_1 and c_2. It is clear from equation (5.1) that we need $|c_1| \leq 1$ while sliding ($s_1 = 0$) in order to ensure $|\chi_e| \leq \frac{\pi}{2}$. Compared to the linear sliding surface where we had a single tunable parameter λ_1, we now have two adjustable parameters c_1 and c_2 for performance tuning for large and small track errors, respectively. Both these parameters can be adjusted independently to give acceptable performance for large and small y_e. Also the constraint $|\chi_e| \leq \frac{\pi}{2}$ can be met if $|c_1| \leq 1$. The proposed nonlinear sliding surface therefore overcomes the limitations of the linear sliding surface presented earlier.


(a): c_2=0.003 and different values of c_1

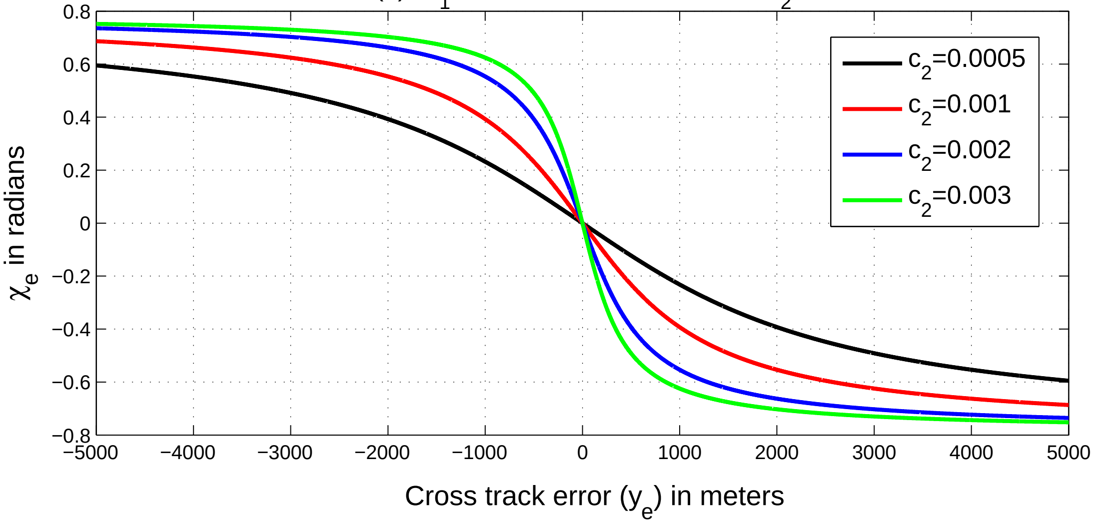

(b): c_1=0.5 and different values of c_2

FIGURE 5.2: The nonlinear sliding surface for different values of c_1 and c_2.

Motion on the sliding surface is represented by the expression (5.1), i.e., $s_1 = \chi_e + c_1 \arctan(c_2 y_e) = 0$, or $\chi_e = -c_1 \arctan(c_2 y_e)$. For large cross-track errors, $\pm \arctan(c_2 y_e) \approx \pm\frac{\pi}{2}$, and hence $\chi_e \approx \mp|c_1|\frac{\pi}{2}$ for $\pm y_e$, whenever the magnitude of y_e is large. Thus we can say that χ_e is directly proportional to $|c_1|$ for large track errors, and a large χ_e implies fast convergence of the error to zero. The parameter c_1 can therefore be adjusted to provide good performance during large track errors. Figure 5.2 shows plots of the nonlinear manifold for different values of c_1 and c_2. In Figure 5.2(a) the effect of varying c_1 is shown with c_2 held constant; it is clear that $\chi_e \approx -c_1\frac{\pi}{2}$ for large track errors ($|y| > 1500$ m). In Figure 5.2(b), the effect of varying c_2 is shown for constant c_1. The parameter c_2 defines the curvature of the curve and determines how quickly the vehicle would turn and reduce the track error to zero (reach the origin). Small values of c_2 indicate that an appreciable χ_e and hence turning would only come about for a relatively large y_e, thus the system dynamics would be sluggish. Large c_2 means an appreciable χ_e for a relatively small y_e, and hence fast dynamics so that the system zeros the error quickly. Performance in case of small cross track errors (small y_e) is thus directly related to c_2; a large c_2 however implies a bigger control effort, and hence the value of c_2 cannot be kept arbitrarily large. The issue of control effort boundedness is discussed in more detail later.

5.2.1 Stability of sliding surface

Stability of the (nonlinear) sliding motion can be proved with the help of Lyapunov theory. A candidate Lyapunov function $\mathbf{W}$ for proving stability of the sliding motion ($s_1 = 0$) is given by:

$$\mathbf{W} = \frac{1}{2}(y_e^2 + \chi_e^2) = \frac{1}{2}\left(y_e^2 + c_1^2 \arctan^2(c_2 y_e)\right). \tag{5.2}$$

Taking the time derivative and substituting state equations (3.17), we get:

$$\dot{\mathbf{W}} = -V_g y_e \sin\left(c_1 \arctan(c_2 y_e)\right) - \frac{c_1 c_2 V_g}{1 + c_2^2 y_e^2} c_1 \arctan(c_2 y_e) \sin(c_1 \arctan(c_2 y_e)). \tag{5.3}$$

Here V_g and $\frac{c_1 c_2 V_g}{1+c_2^2 y_e^2}$ are positive definite, and for the negative definiteness of $\dot{\mathbf{W}}$ we have to show that $y_e \sin\left(c_1 \arctan(c_2 y_e)\right)$ and $c_1 \arctan(c_2 y_e) \sin(c_1 \arctan(c_2 y_e))$ are also positive definite. In other words, we have to show that y_e, $c_1 \arctan(c_2 y_e)$ and $\sin(c_1 \arctan(c_2 y_e))$ have the same sign. Since 'arctan' is an odd function so the sign of $\arctan(c_2 y_e)$ will be the same as that of $c_2 y_e$, and the sign of $c_1 \arctan(c_2 y_e)$ will be the same as that of y_e provided $c_1 c_2 > 0$. Also since $|c_1| \leq 1$, the range of the function $c_1 \arctan(c_2 y_e)$ is $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$; and $\sin(c_1 \arctan(c_2 y_e))$ has the same sign as $c_1 \arctan(c_2 y_e)$. Hence the sign of $\sin(c_1 \arctan(c_2 y_e))$ will be the same as that of y_e, provided $c_1 c_2 > 0$ and $|c_1| \leq 1$.

The first term on the right hand side in (5.3) is negative definite since V_g (the magnitude of the velocity) is always positive, and the sign of $\sin(c_1 \arctan(c_2 y_e))$ is the same as that of y_e. For the second term on the right hand side, $\frac{c_1 c_2 V_g}{1+c_2^2 y_e^2}$ is positive definite and $c_1 \arctan(c_2 y_e)$ and $\sin(c_1 \arctan(c_2 y_e))$ have the same signs. The second term is therefore also negative definite, and so $\dot{\mathbf{W}}$ is negative definite. Thus we see that the proposed positive definite Lyapunov function has a negative definite time derivative and hence the proposed sliding surface is stable, provided $c_1 c_2 > 0$ and $|c_1| \leq 1$.

Alternatively, stability of the sliding surface can also be guaranteed by analyzing the sliding motion. Motion on the sliding surface is described by $s_1 = 0$, implying $\chi_e = -c_1 \arctan(c_2 y_e)$. Using state equations (3.17), equation for sliding motion becomes:

$$\dot{y}_e = -V_g \sin\left(c_1 \arctan\left(c_2 y_e\right)\right). \tag{5.4}$$

Here we have two cases for positive and negative values of y_e, with $c_1 c_2 > 0$ and $|c_1| \leq 1$:

1. Case-1 (y_e positive): As $c_1 \leq 1$ and both c_1 & c_2 are either positive or negative, hence for any positive value of y_e,

$c_1 \arctan(c_2 y_e) \in \left(0, \frac{\pi}{2}\right)$

$\Rightarrow \sin(c_1 \arctan(c_2 y_e))$ is a positive number

$\Rightarrow \dot{y}_e < 0$

$\Rightarrow y_e$ will decrease.

2. Case-2 (y_e negative): In this case,

   $c_1 \arctan(c_2 y_e) \in \left(-\frac{\pi}{2}, 0\right)$

   $\Rightarrow \dot{y}_e > 0$

   $\Rightarrow y_e$ will increase.

So in either of the two cases, y_e will converge to zero and therefore $\chi_e = -c_1 \arctan(c_2 y_e)$ will also converge to zero. Hence we have a stable sliding surface and both state variables will converge to zero by sliding on the proposed nonlinear surface.

5.3 Nonlinear Sliding Surface for Vertical Plane

For good performance in all scenarios and to avoid instability during sliding motion, we now propose a nonlinear sliding surface for vertical plane guidance (see Figure 5.3):

$$s_2 = \gamma_e + c_3 \arctan(c_4 h_e) = 0 \quad \text{or} \quad s_2 = \gamma_e - c_3 \arctan(c_4 z_e) = 0, \tag{5.5}$$

where the constants c_3 and c_4 are real numbers. For stability, both coefficients should have the same sign (let both be positive real numbers) and $c_3 \leq 1$. Although the longitudinal (vertical) sliding surface is similar to the lateral one, the difference is in the selection of coefficients. In the vertical plane, gravity plays an important role in the design of guidance logic; more thrust is required to balance gravity as the flight path angle is increased. Maximum flight path angle demand is generally linked to the magnitude of c_3 which is kept well below 1 due to limited thrust in UAVs. Compared to the traditional linear sliding surface function shown in section 4.2.3, here we have two tunable sliding surface parameters. Performance for small

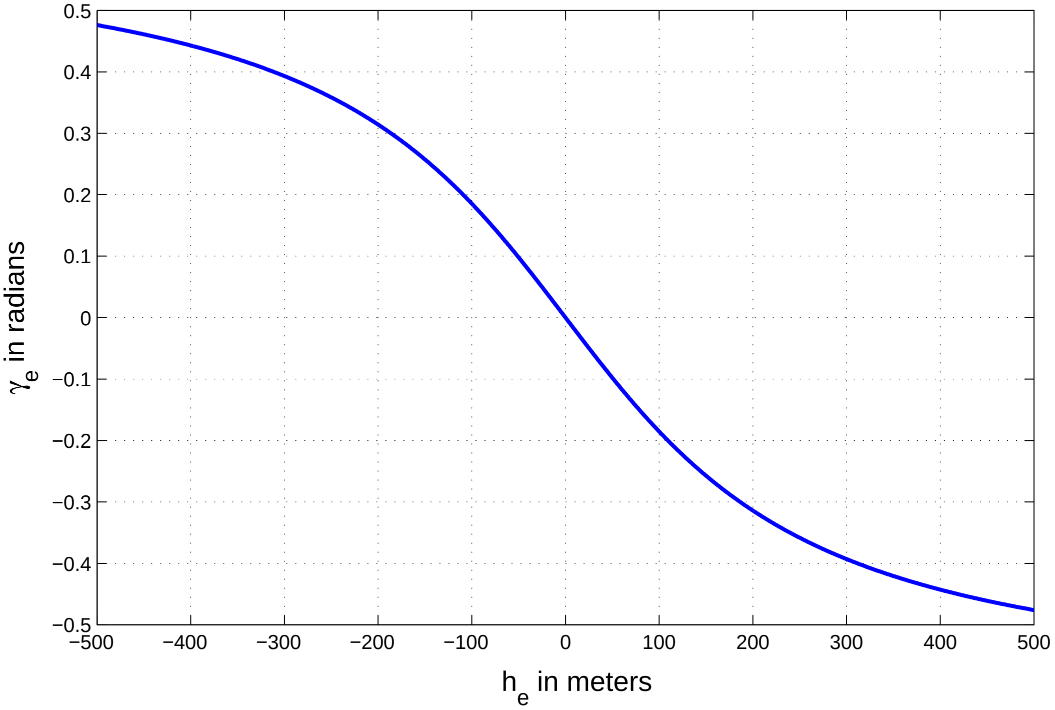


FIGURE 5.3: The proposed nonlinear sliding surface for vetical plane.

errors in altitude can be adjusted using c_4, as in the lateral sliding surface shown in Figure 5.2. The parameter c_3 can be used to adjust performance for large errors in altitude as well as to limit the maximum flight path angle (keeping in view the vehicle's maneuvering capabilities, such as flight loads and climb/decent rate).

Stability of the sliding surface can be proved as follows: sliding motion is represented by $s_2 = 0$ or $\gamma_e = -c_3 \arctan(c_4 h_e)$. After substitution in the state equations (3.25), motion on the sliding surface becomes:

$$\dot{h}_e = -V_g \cos \gamma_p \sin \left(c_3 \arctan \left(c_4 h_e \right) \right). \tag{5.6}$$

To prove stability of the sliding surface, we have to show that h_e converges to zero for both positive and negative initial h_e. Assuming both c_3 and c_4 are positive real numbers, and $c_3 \leq 1$, we have $c_3 \arctan \left(c_4 h_e \right) \in \left(0, \frac{\pi}{2} \right)$ for positive h_e, implying a positive value of the term $\sin \left(c_3 \arctan \left(c_4 h_e \right) \right)$. As V_g and $\cos \gamma_p$ are positive, we have negative $\dot{h}_e$ for a positive h_e, this implies convergence of h_e towards zero. Similarly in case of negative h_e, the term $c_3 \arctan(c_4 h_e) \in \left(-\frac{\pi}{2}, 0\right)$, and hence a positive $\dot{h}_e$, which again implies convergence of h_e towards zero.

Conclusively h_e will converge to zero for both positive and negative errors while sliding on the proposed nonlinear surface, and γ_e will also converge to zero once h_e converges to zero. Hence we have a stable sliding surface and both state variables converge to zero while sliding on this surface.

Remarks: If we choose $s_2 = \gamma_e - c_3 \arctan(c_4 z_e) = 0$ (or $\gamma_e = c_3 \arctan(c_4 z_e) = 0$) then $\dot{z}_e = -V_g \sin(c_3 \arctan(c_4 z_e))$ and all above comments are also valid for this too.

5.4 Optimized Sliding Surface Coefficients

After proposing nonlinear sliding surfaces for lateral and longitudinal planes, a natural question arises about selection of coefficients for these surfaces. Path planning is not the topic of this thesis, the objective here is to see how to reach and maintain the planned mission (which may be changed online anytime). To reach the planned mission through sliding, the coefficients of the sliding surfaces should be optimized for high performance and minimum use of energy. The principle of work and energy is used here to find optimal coefficients. This allows us to find the change in energy (work done) due to an external force (the thrust force here) to reach the desired path, and hence a minimum energy path can be selected by a suitable choice of sliding coefficients. During sliding motion, the state variables are linked to each other by the equation $s = 0$, i.e., it relates the variables through a state trajectory (independent of time), and hence the principle of work and energy is applicable here. The objective is to reach the planned mission by sliding with minimum work done, and also not exceeding the UAV maneuverability capabilities. In this section optimization of these variables is discussed in detail for both lateral (straight and circular paths) and longitudinal planes.

5.4.1 Lateral plane coefficients

To find optimum values of lateral sliding surface coefficients, the shape of the desired path (straight or circular) also needs to be considered. In case of straight path, the problem is relatively simple as the cross-track error in this case is computed with respect to a fixed straight line and depends only on the position of the UAV. However, the problem becomes more complicated for circular paths as the cross track error is measured from the closest point on the desired path which changes at every instant. In subsequent sections, selection of coefficients is discussed for both straight and circular paths.

5.4.1.1 Straight path following

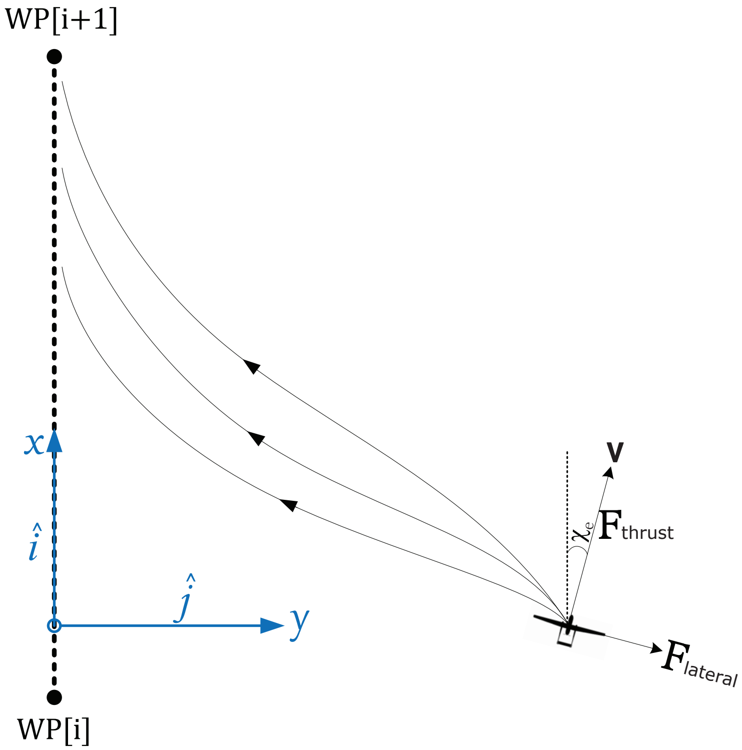


FIGURE 5.4: Conventions for lateral sliding coefficients optimization (straight path).

In order to find optimal sliding coefficients for straight path following, we have to find the coefficients that correspond to minimum work done. Let $WP[i]$-$WP[i+1]$

be the desired straight path; initially the UAV is y_0 meters away from the desired path as shown in Figure 5.4. A local reference frame is introduced on the desired path at the closest point to the initial position of the UAV such that the initial position of the UAV is $(x, y) = (0, y_0)$. The y-component in this frame represents the cross track error (y_e). Different curved paths are possible to bring back the UAV to the desired path as shown in Figure 5.4; the objective is to choose a path (C) that is possible to follow (within the UAV maneuvering capability) and also corresponds to minimum work done (or minimum change in energy). Let $\mathbf{F}(x, y)$ be a continuous force field and C be a curve defined by the position vector $\mathbf{P}(x, y)$, then the work done by $\mathbf{F}(x, y)$ on the UAV moving along C is given by:

$$WD = \int_C \mathbf{F}(x, y) \cdot d\mathbf{P}. \tag{5.7}$$

The force field $\mathbf{F}(x, y)$ is the sum of the thrust force in the lateral plane ($\mathbf{F}_{thrust}$) and the lift vector component in the lateral plane ($\mathbf{F}_{lateral}$), therefore we get:

$$WD = \int_C \mathbf{F}_{lateral} \cdot d\mathbf{P} + \int_C \mathbf{F}_{thrust} \cdot d\mathbf{P}. \tag{5.8}$$

As the lateral force ($\mathbf{F}_{lateral}$) is perpendicular to the path, so the first term in the above equation is zero, hence the total work done is:

$$WD = \int_C \mathbf{F}_{thrust} \cdot d\mathbf{P}. \tag{5.9}$$

Resolving the above vectors in the fixed Cartesian-coordinates as shown in Figure 5.4, $\mathbf{F}_{thrust} = F_{thrust} \cos \chi_e \hat{\mathbf{i}} + F_{thrust} \sin \chi_e \hat{\mathbf{j}}$ and $d\mathbf{P} = dx\hat{\mathbf{i}} + dy\hat{\mathbf{j}}$, we get:

$$WD = \int_C \left(F_{thrust} \cos \chi_e dx + F_{thrust} \sin \chi_e dy \right). \tag{5.10}$$

In the lateral plane, the position of the UAV in xy-coordinate system can be computed using $\dot{x} = V \cos \chi_e$ and $\dot{y} = V \sin \chi_e$. Dividing these equations, we have the relation $dx = \cot\chi_e dy$, and the expression for work done becomes:

$$WD = \int_{y=y_0}^{0} F_{thrust} \left(\cos\chi_e \cot\chi_e + \sin\chi_e\right) dy \tag{5.11}$$

As the objective is to bring the UAV back to the desired path using the proposed sliding surface (equation 5.1), during sliding $s_1 = 0$ which implies $\chi_e = -c_1 \arctan(c_2 y)$ (as mentioned earlier $y_e = y$). Assuming constant thrust and using the expression $\chi_e = -c_1 \arctan(c_2 y)$, we have the following expression after simplification:

$$WD = -F_{thrust} \int_{y=y_0}^{0} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy. \tag{5.12}$$

Alternatively, the expression (5.12) can be derived by finding the arc-length using calculus theory and then multiplying it by the force. According to calculus theory, the arc length of a curve in Cartesian coordinates $x - y$ is defined as:

$$\text{Arc Length} = \int_{a}^{b} \sqrt{1 + \left(\frac{dx}{dy}\right)^2} dy, \qquad \text{where } a < b. \tag{5.13}$$

In our case, $\dot{x} = V\cos\chi_e$ and $\dot{y} = V\sin\chi_e$ implies $\frac{dx}{dy} = \cot\chi_e$. Using this expression and integrating from $y = y_0$ to zero, we have:

$$\text{Arc Length} = \int_{y=y_0}^{0} \sqrt{1 + (\cot\chi_e)^2} dy = \int_{y=y_0}^{0} \frac{1}{\sin\left(-c_1 \arctan(c_2 y)\right)} dy. \tag{5.14}$$

After multiplication with Force, the expression for work done becomes:

$$WD = -F_{thrust} \int_{y=y_0}^{0} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy. \tag{5.15}$$

This expression for work done is same as (5.12). This integral has no analytic solution but can easily be integrated using numerical techniques to find the work done for specific values of c_1 and c_2. While sliding $y \to 0$ as $t \to \infty$, so there should be a lower limit on the magnitude of y (say 1 m) to stop the integration instead of integrating to $y = 0$. For a boundary width of 1 m and a positive y_0, the work done expression takes the form:

$$WD = -F_{thrust} \int\limits_{y=y_0}^{1} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy. \tag{5.16}$$

The negative sign here is due to the reference frame convention for y; for positive

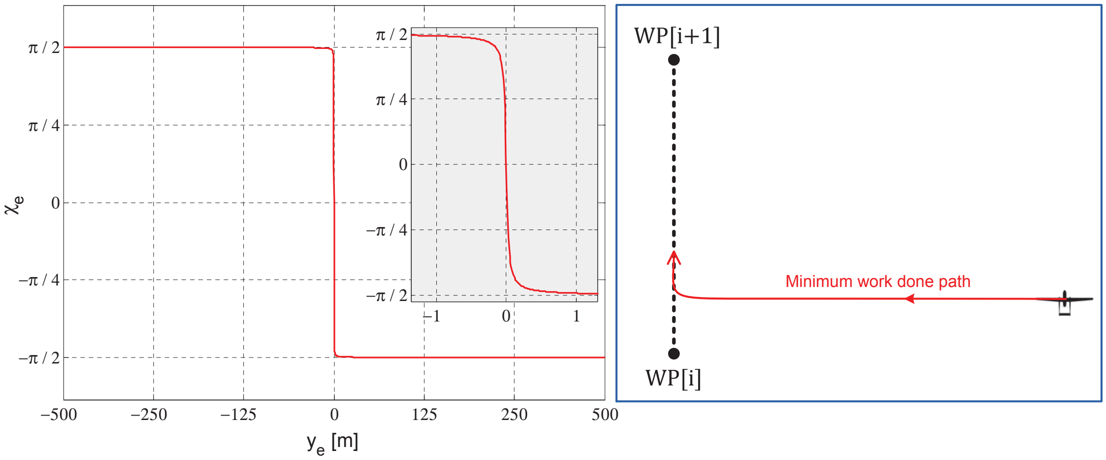


FIGURE 5.5: Unconstraint optimized sliding surface and path for lateral plane (straight path).

y_0 we have to move in the negative y-direction and hence the work done is negative, and vice-versa. Without imposing any constraint, this integral can be analyzed to find values of c_1 and c_2 that correspond to minimum work done. For positive values of c_1 (≤ 1) and c_2, the integrand $\frac{1}{\sin(c_1 \arctan(c_2 y))}$ is positive at every instant of integration from $y = y_0$ to $y = 1$, so minimizing the integrand at every point will result in minimum work done. Minimizing the integrand means maximizing the term $\sin\left(c_1 \arctan(c_2 y)\right)$ at every instant, this is possible if we choose $c_1 = 1$ and a large value of c_2. The resulting lateral sliding surface and UAV trajectory for optimum sliding coefficients are shown in Figure 5.5. Practically it is impossible to follow that trajectory with a limited lateral force $F_{lateral}$, hence the availability of lateral force at every instant should also be considered while minimizing the above integral. The criterion therefore should be:

$$\left.\begin{array}{l} \text{minimize} \quad WD = \begin{cases} -F_{thrust} \displaystyle\int_{y=y_0}^{-1} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy & \text{for -ve } y_0 \\ -F_{thrust} \displaystyle\int_{y=y_0}^{1} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy & \text{for +ve } y_0 \end{cases} \\ \text{subject to} \quad F_{lateral} \leq F_{lateral_{max}} \text{ at every point of integration} \end{array}\right\} \tag{5.17}$$

In order to find optimum values of the coefficients c_1 and c_2 for a specific y_0, the following procedure may be followed:

1. Choose a positive y_0.
2. Select an acceptable range of the coefficients $0 \leq c_1 \leq 1$ and $0 \leq c_2$. Initially a wide range with a large step size may be selected, this may be refined to a narrow range with finer resolution in the next iteration. c_1 is usually given a range from 0.0001 to 1 and c_2 from 0.0001 to 5 with a step size of 0.01 for both.
3. Run a nested loop for c_1=0.0001:0.01:1 and c_2=0.0001:0.01:5.
4. In each run first check the $F_{lateral} \leq F_{lateral_{max}}$ constraint by varying y from 1 to y_0 with a reasonable step size (say 1m). If the constraint fails at any point, that combination of c_1 and c_2 is dropped. Calculate the work done for each successful combination of c_1 and c_2, and store it.
5. Generate a grid of all work done values and choose the combination of c_1 and c_2 which corresponds to the least work done.

6. Repeat the process for a different value of y_0.
7. Analyze the computed values of work done, and either choose the best single combination of c_1 and c_2 that works well for various value of y_0, or make these sliding parameters adaptive.

5.4.1.2 Circular path following

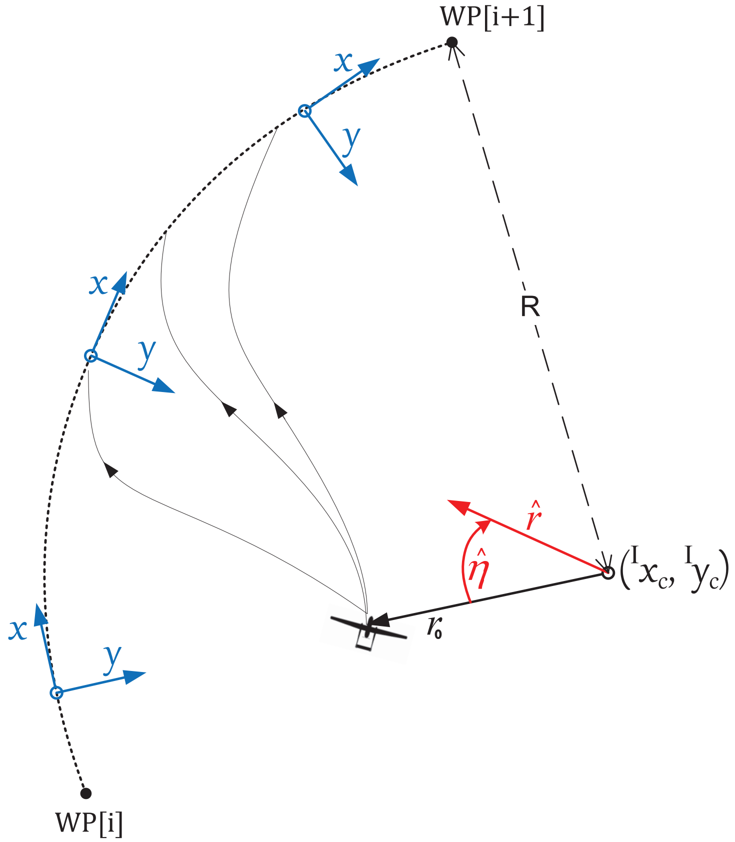


FIGURE 5.6: Conventions for lateral sliding coefficients optimization (circular path).

In this section we derive an expression for work done in case of circular path following to find optimum values of lateral sliding coefficients c_1 and c_2. Let $WP[i]$ and $WP[i+1]$ be two consecutive way-points in the lateral plane connected by a circular arc of radius R centered at $({}^{I}x_c, {}^{I}y_c)$ in an inertial frame as shown in Figure 5.6. In this case, if we choose a fixed Cartesian coordinate system on the desired path nearest to the position of the UAV (like in the straight path case) then this reference frame will rotate and translate on the desired path as the UAV moves. For a rotating and translating coordinate system, the change in position $d\mathbf{P} \neq dx\hat{\mathbf{i}} + dy\hat{\mathbf{j}}$, and hence the calculation of work done is not straightforward. To handle this complexity, we introduce a new polar coordinate reference frame $(\hat{\mathbf{r}}, \hat{\eta})$ centered at $({}^{I}x_c, {}^{I}y_c)$ to describe the position of the UAV at any time t. Let the initial deviation of the UAV from the desired circular path be y_0 $(= R - r_0)$, then the guidance objective is to make $y = 0$ (or $r = R$) by choosing an optimal path from the many available paths as shown in Figure 5.6. Let $\mathbf{F}(r, \eta)$ be a continuous force field and C be a curve (path) defined by the position vector $\mathbf{P}(r, \eta)$, then the work done by $\mathbf{F}(r, \eta)$ on the UAV moving along C is given by:

$$WD = \int_C \mathbf{F}(r, \eta) \cdot d\mathbf{P}, \tag{5.18}$$

where $\mathbf{F}(r, \eta)$ is the sum of the thrust force in the lateral plane ($\mathbf{F}_{thrust}$) and the lift vector component in the lateral plane ($\mathbf{F}_{lateral}$). The part ($\mathbf{F}_{lateral} \cdot d\mathbf{P}$) is zero as both vectors are perpendicular to each other, and hence the expression for work done becomes:

$$WD = \int_C \mathbf{F}_{thrust} \cdot d\mathbf{P}. \tag{5.19}$$

Position of the UAV in polar coordinates $(\hat{\mathbf{r}}, \hat{\eta})$ is represented by $\mathbf{P} = r\hat{\mathbf{r}}$, and its rate of change by:

$$\dot{\mathbf{P}} = \dot{r}\hat{\mathbf{r}} + r\dot{\eta}\hat{\eta}, \tag{5.20}$$

or in alternative form:

$$d\mathbf{P} = dr\hat{\mathbf{r}} + rd\eta\hat{\eta}. \tag{5.21}$$

Similarly $\mathbf{F}_{thrust}$ can be expressed in polar coordinates as (see Figure 5.7):

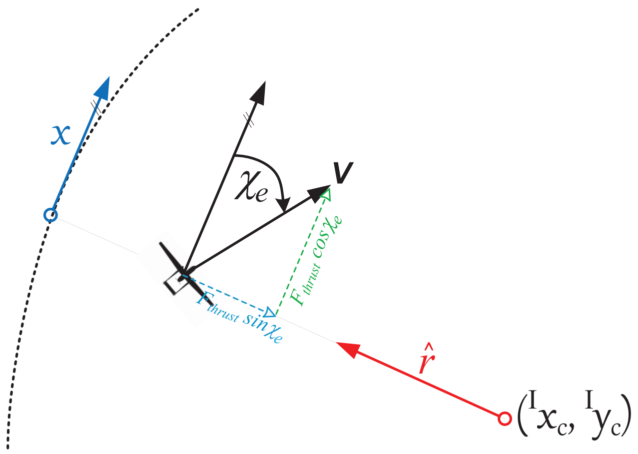


FIGURE 5.7: Thrust force in polar coordinates (circular path).

$$\mathbf{F}_{thrust} = -F_{thrust}\sin\chi_e\hat{\mathbf{r}} + F_{thrust}\cos\chi_e\hat{\eta}. \tag{5.22}$$

Using equations (5.21) and (5.22) in the expression for work done (5.19), and assuming a constant thrust force, we get:

$$WD = F_{thrust}\int_C \left(-\sin\chi_e dr + r\cos\chi_e d\eta\right). \tag{5.23}$$

While sliding ($s_1 = 0$), $\chi_e = -c_1\arctan(c_2 y_e)$, which can be expressed in polar coordinates as $\chi_e = -c_1\arctan(c_2(R-r))$; this implies:

$$WD = F_{thrust}\int_C \left(\sin(c_1\arctan(c_2(R-r)))dr + r\cos(c_1\arctan(c_2(R-r)))d\eta\right). \tag{5.24}$$

This expression has two terms; the first term corresponds to work done due to change in r while the second term corresponds to the work done due to change in angle η. Also we get positive work done for a positive y_0 from equation (5.24) as we have to move in the direction of $\hat{\mathbf{r}}$ to bring the cross track error to zero.

Let us have an initial cross track error $r = r_0 (= R - y_0)$ expressed in polar coordinates, the guidance aim is to make $r = R$. The optimal path (minimum work done) in this case is to change only r from '$R - y_0$' to R, without change in angle η (similar to the straight path case as shown in Figure 5.5). This is possible if we choose the sliding coefficients $c_1 = 1$ and a large c_2. However this trajectory is not possible with a limited lateral force, therefore selection of the sliding coefficients should be done by checking for the maximum lateral force available at each instant. Hence, the criterion should be:

$$\left.\begin{aligned} &\text{minimize} \\ &WD = F_{thrust} \int_C (\sin(c_1 \arctan(c_2(R - r)))dr + r\cos(c_1 \arctan(c_2(R - r)))d\eta) \\ &\qquad\quad \text{subject to} \quad F_{lateral} \leq F_{lateral_{max}} \end{aligned}\right\} \tag{5.25}$$

To find optimum values of coefficients c_1 and c_2 for a specific y_0 and R, a procedure similar to the straight path case can be followed. First a range of the two coefficients should be selected and then work done can be calculated for each combination that does not violate the $F_{lateral} \leq F_{lateral_{max}}$ condition.

5.4.2 Vertical plane coefficients

The maximum available thrust and gravity are the two important factors to be considered for selection of longitudinal sliding coefficients. The combination of these two factors dictates the climb/decent rate and the achievable flight path angle capabilities of the UAV, which is important for guidance logic design. Unlike the lateral plane, here the coefficient selection is different for positive and negative altitude errors as gravity supports in one case and opposes in the other. Generally way-points in the vertical plane are connected by a straight line to maintain constant rates of climb or decent, and only local treatment at the corners is applied

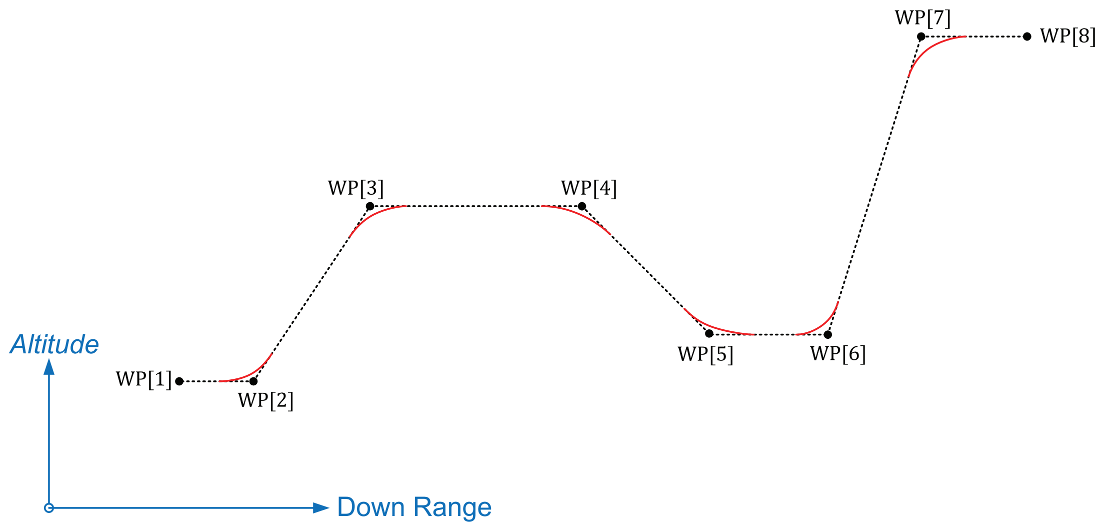


FIGURE 5.8: A sample mission plan for Longitudinal plane.

for smoothing. A sample longitudinal mission plan is shown in Figure 5.8 with corner treatments for a UAV to follow. Generally the longitudinal guidance scheme is designed to follow straight connected way-points, and the same is considered here.

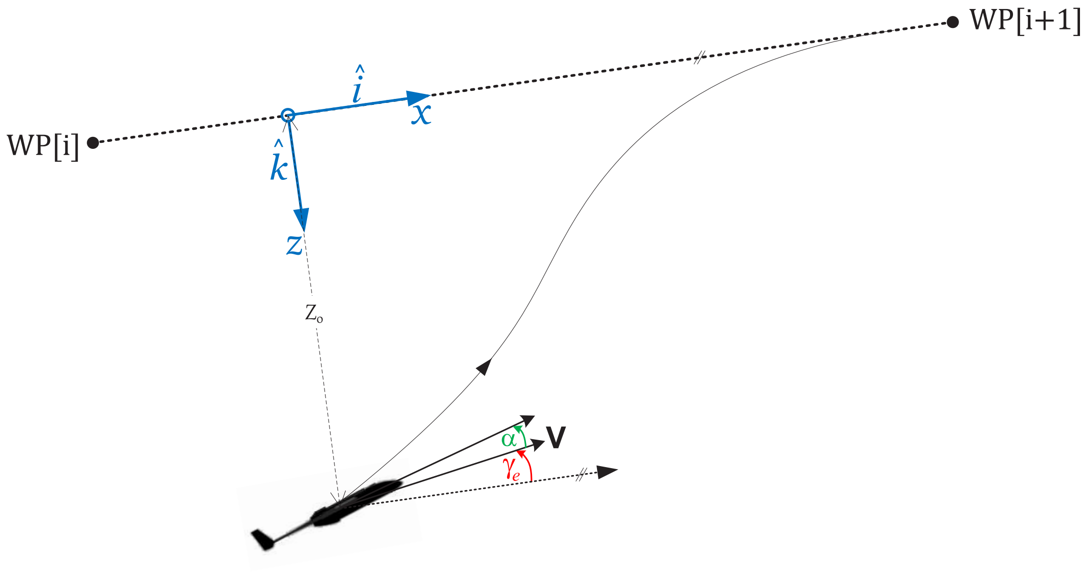


FIGURE 5.9: Conventions for longitudinal sliding coefficients optimization.

For the longitudinal sliding coefficients, the convention and definition of different variables is shown in Figure 5.9. Let $WP[i]$-$WP[i+1]$ be the desired straight path (making an angle γ_p with the local horizontal) in the vertical plane and let initially the UAV be z_0 meters away from the desired path as shown. A fixed Cartesian coordinated system $(x - z)$ is introduced on the desired path at the closest point. The thrust force acts along the body central line and the difference between the angles of body central line and the velocity vector is the angle of attack (α). Here the variable z denotes deviation from the desired path and is our main variable of interest. Different geometric paths are possible to bring the deviation to zero by using different sliding coefficients c_3 and c_4, here the objective is to find the combination that corresponds to minimum work done (or minimum change in energy) and does not exceed the flying capability of the UAV. By definition of work done for a continuous force field $\mathbf{F}(x, z)$ over a curved path C, we have:

$$WD = \int_C \mathbf{F}(x, z) \cdot d\mathbf{P}. \tag{5.26}$$

Here the force field of interest is the applied external force $\mathbf{F}_{thrust}$ to bring the UAV back on the desired path. Gravity is a conservative force and is not of our interest for work done calculation, it will however be taken as a constraint. The work done due to thrust force over C is:

$$WD = \int_C \mathbf{F}_{thrust}(x, z) \cdot d\mathbf{P}. \tag{5.27}$$

Resolving the above vectors in the fixed Cartesian-coordinates as shown in Figure 5.9: $\mathbf{F}_{thrust} = F_{thrust}\cos(\alpha + \gamma_e)\hat{\mathbf{i}} - F_{thrust}\sin(\alpha + \gamma_e)\hat{\mathbf{k}}$ and $d\mathbf{P} = dx\hat{\mathbf{i}} + dz\hat{\mathbf{k}}$, we get:

$$WD = \int_C F_{thrust}\cos(\alpha + \gamma_e)dx - F_{thrust}\sin(\alpha + \gamma_e)dz. \tag{5.28}$$

Position of the UAV in the x-z coordinate system can be computed using the expressions $\dot{x} = V\cos\gamma_e$ and $\dot{z} = -V\sin\gamma_e$, dividing these we have the geometric relation $dx = -\cot\gamma_e dz$. Assuming thrust of a constant magnitude, and using $dx = -\cot\gamma_e dz$, we have the following expression for work done:

$$WD = -F_{thrust} \int\limits_{z=z_0}^{0} \left(\cos(\alpha+\gamma_e)\cot\gamma_e + \sin(\alpha+\gamma_e)\right) dz, \tag{5.29}$$

which after simplification becomes:

$$WD = -F_{thrust} \int\limits_{z=z_0}^{0} \left(\cos\alpha\cos\gamma_e\cot\gamma_e + \cos\alpha\sin\gamma_e\right) dz. \tag{5.30}$$

To reduce the height error to zero by sliding along the surface $s_2 = 0$ (or $\gamma_e = c_3\arctan(c_4 z)$), we have the following simplified expression for work done (assuming constant α):

$$WD = -F_{thrust}\cos\alpha \int\limits_{z=z_0}^{0} \frac{1}{\sin\left(c_3\arctan(c_4 z)\right)} dz. \tag{5.31}$$

For work done computation, the required vertical force and the maximum available vertical force along the sliding trajectory need to be considered at each point for different values of c_3 and c_4. Hence the final criterion for minimization of work done to bring the height error close to zero (say within 1 m) may be written as:

$$\left.\begin{array}{l} \text{Minimize} \quad WD = \begin{cases} -F_{thrust}\cos\alpha \displaystyle\int\limits_{z=z_0}^{-1} \frac{1}{\sin\left(c_3\arctan(c_4 z)\right)} dz & \text{for -ve } z_0 \\ -F_{thrust}\cos\alpha \displaystyle\int\limits_{z=z_0}^{1} \frac{1}{\sin\left(c_3\arctan(c_4 z)\right)} dz & \text{for +ve } z_0 \end{cases} \\ \quad\quad \text{Subject to} \quad F_{longitudinal} \leq F_{longitudinal_{max}} \text{ at each point of integration} \end{array}\right\} \tag{5.32}$$

This expression appears similar to that in the lateral case, the main difference is that $F_{longitudinal}$ force also includes a component of gravity (discussed in detail in the example in section 7.3.1). This gravity term enhances $F_{longitudinal}$ for negative z_0 and reduces it for positive z_0, implying different values of work done and hence sliding coefficients for positive and negative height errors. In order to find the optimum values of c_3 and c_4 for a specific z_0, first a range of both coefficients should be selected, followed by computation of work done for each combination to select the best (minimum work done or minimum change in energy) sliding coefficients. A complete case study is discussed in Chapter 7.

5.5 Summary of the Chapter

Performance of sliding mode based control law depends on the selection of sliding surface and its coefficients. For guaranteed stability and high performance in all scenarios, two nonlinear sliding surfaces are proposed for the lateral and longitudinal planes. The idea is inspired from the good Helmsman criterion in the literature used to guide ships onto the desired paths. After proposing new nonlinear sliding manifolds, selection of optimum sliding coefficients is discussed in detail. The selection is based on the work and energy principle to use minimum energy to bring the UAV onto the desired path. The problem now is to design the guidance logic based on these surfaces and its practical implementation. In the next chapter a sliding mode based lateral guidance scheme is derived for UAVs based on the sliding surface proposed here. The generalized 3-D path following guidance algorithm is discussed in Chapter 7.

Chapter 6

LATERAL GUIDANCE OF UAVS

6.1 Introduction

The main objective of the lateral guidance scheme is to keep the UAV on the desired ground track by controlling its lateral deviations in the presence of disturbances and uncertainties. If a ground cross-track error develops due to any reason (for example when guidance is activated the first time after takeoff, or if an error develops in the navigation solution due to a long GPS outage), the role of the guidance algorithm is to reduce it to zero in a smooth and stable manner without saturating the control input to the vehicle. The lateral guidance problem is about driving the cross-track error and the course angle error to zero – the two degrees of freedom of interest in this case. The guidance law will usually generate roll reference commands for the inner roll control loop to follow; the logic must perform well both for small and large track errors, without saturating the roll angle control input, and guarantee closed-loop stability.

As discussed earlier, sliding mode based design can be divided into two parts; the design of a stable sliding surface followed by the design of a control law to force the system states trajectory onto the chosen sliding surface in finite time, and maintain motion on it for subsequent time. In this chapter, we discuss a novel nonlinear guidance law based on sliding mode control theory using the high performance nonlinear sliding manifold proposed in section 5.2. A nonlinear guidance law is derived for reaching and maintaining the sliding motion. It is shown that no control saturation occurs during the sliding motion for either small or large track errors. The proposed scheme is implemented in the flight control computer of a small research UAV, and flight results for different scenarios are presented.

6.2 Lateral Guidance Law Design

Sliding mode control design is essentially a two step process: sliding surface design, and derivation of a control law to ensure that the phase trajectory is attracted towards the sliding surface. For sliding mode lateral guidance law design, a high performance nonlinear stable sliding surface is presented in the section 5.2, i.e.,

$$s_1 = \chi_e + c_1 \arctan(c_2 y_e) = 0. \tag{6.1}$$

Here in this section, we use this sliding surface and derive a sliding mode based guidance law; conditions to ensure sliding and control boundedness are also derived and discussed in detail. Sliding mode control (Utkin, 1977) is fundamentally discontinuous control containing terms like '$-k\,\mathrm{sgn}(s)$' to slide system state trajectories on a sliding surface. Later developments in the area of sliding mode theory gave birth to new ideas like the equivalent control method in addition to the discontinuous control term (Utkin and Shi, 1999). Discontinuous control changes at high (theoretically infinite) frequency such that the state vector moves precisely along the sliding surface, however various imperfections make the state oscillate in the vicinity of the sliding surface. These oscillations have a high frequency component and a low frequency component; the plant motion during sliding is determined by the low frequency component. The low frequency component is approximately equal to the equivalent control that can be found by using the equation $\dot{s} = 0$ (Utkin, 1977; Edwards and Spurgeon, 1998). From a geometrical point of view the equivalent control method means replacement of discontinuous control in the intersection of switching surfaces by a continuous control such that the state vector lies tangential to the sliding surface.

The lateral guidance law based on sliding mode theory is the sum of two terms: the equivalent control term ($\dot{s}_1 = 0$) and a discontinuous term '$-k\,\mathrm{sgn}(s_1)$' to cater for parametric uncertainties and disturbances. In subsequent subsections, first the guidance law is derived, followed by derivation of reachability and control boundedness conditions.

6.2.1 Equivalent lateral control

Equivalent control is the control input which when applied to the system, enables the system to continue sliding, once it is on the sliding surface (Bandyopadhyay and Janardhanan, 2006). The equivalent control that maintains the sliding mode is therefore the input u_{eqv} satisfying $\dot{s}_1 = 0$. In our case:

$$\dot{s}_1 = \dot{\chi}_e + \frac{c_1 c_2}{1 + c_2^2 y_e^2} \dot{y}_e. \tag{6.2}$$

Using state equations (3.16), $\dot{s}_1 = 0$ can be written as:

$$0 = \frac{\hat{g} u_{eqv}}{\hat{V}_g} \cos\left(\sin^{-1}\left(\frac{1}{\hat{V}_a}\left(-\hat{V}_{w_n}\sin\chi + \hat{V}_{w_e}\cos\chi\right)\right)\right) + \frac{c_1 c_2}{1 + c_2^2 y_e^2}\hat{V}_g \sin\chi_e - \dot{\chi}_p, \tag{6.3}$$

where $\hat{g}$ and $\hat{V}$ are the measured values of gravity and velocity, respectively. After simplification, the expression for equivalent control u_{eqv} becomes:

$$u_{eqv} = \frac{1}{\cos\left(\sin^{-1}\left(\frac{1}{\hat{V}_a}\left(-\hat{V}_{w_n}\sin\chi + \hat{V}_{w_e}\cos\chi\right)\right)\right)}\left(\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - \frac{c_1 c_2}{1 + c_2^2 y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e\right). \tag{6.4}$$

This control expression has two terms: a feedback term and an anticipatory feed-forward term ($\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p$) which is dependent on the rate of change of the desired course angle. During a steady turn, the feed-forward control component generates acceleration in the lateral plane to balance the centripetal acceleration (equal but opposite in direction), provided V_g is exactly known. As discussed in detail in Chapter 2, sliding motion can be maintained with this equivalent control in the absence of uncertainties and disturbances (if the vehicle dynamics are exactly known). But generally this is not the case and a formal control law (for example variable structure based) has to be formulated to maintain the sliding motion even in the presence of parametric uncertainties and un-modeled dynamics.

6.2.2 The complete guidance law

The guidance logic must be robust and able to achieve and maintain the sliding motion from any arbitrary initial condition in the presence of input disturbances and parametric uncertainties, in finite time. Different reaching laws are discussed in Chapter 2 to ensure sliding motion in the presence of uncertainties, traditionally the discontinuous control term ($-k\,\mathrm{sgn}(s)$) is added to the equivalent control term to cater for uncertainties. The control gain k mainly depends on the magnitude of uncertainties and is selected using Lyapunov theory. The expression for the lateral guidance law is:

$$\begin{aligned} u &= \frac{1}{\cos\left(\sin^{-1}\left(\frac{1}{\hat{V}_a}\left(-\hat{V}_{w_n}\sin\chi + \hat{V}_{w_e}\cos\chi\right)\right)\right)}\left(\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - \frac{c_1c_2}{1+c_2^2y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e\right) \\ &\qquad - k\,\mathrm{sgn}(s_1), \\ \phi_{req} &= tan^{-1}(u), \end{aligned} \tag{6.5}$$

where $c_1c_2 > 0$ and $|c_1| \leq 1$. The term $\frac{1}{\cos\left(\sin^{-1}\left(\frac{1}{\hat{V}_a}\left(-\hat{V}_{w_n}\sin\chi + \hat{V}_{w_e}\cos\chi\right)\right)\right)}$ is for compensation of wind; in the absence of wind it is equal to 1. Generally wind measurement is not available onboard UAVs, and hence this term has to be estimated. The above guidance law (6.5) can be written in the form:

$$\begin{aligned} u &= \Gamma\left(\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - \frac{c_1c_2}{1+c_2^2y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e\right) - k\,\mathrm{sgn}(s_1), \\ \phi_{req} &= tan^{-1}(u), \end{aligned} \tag{6.6}$$

where $\Gamma = \frac{1}{\cos\left(\sin^{-1}\left(\frac{1}{\hat{V}_a}\left(-\hat{V}_{w_n}\sin\chi+\hat{V}_{w_e}\cos\chi\right)\right)\right)}$. Here we assume accurate measurements of the state variables: the cross-track error (y_e) and the intercept course (χ_e). To cater for parametric uncertainties in other variables including Γ, the gain k required is worked out in the following sections.

Wind disturbances can be handled in one of two ways. In the first approach, wind is considered as a disturbance and its maximum value is analyzed during guidance logic design and the gain k is adjusted accordingly. In the second approach, the wind speed and direction is estimated online using an observer and Γ is computed in real time for use in the guidance equation (6.6). In this case the estimation error should be taken care of during the design and selection of gain k. For simplicity, here we opt for the first approach and take Γ equal to 1. The resultant simplified lateral guidance law is:

$$\begin{aligned} u &= \frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - \frac{c_1 c_2}{1 + c_2^2 y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e - k\,\mathrm{sgn}(s_1), \\ \phi_{req} &= tan^{-1}(u). \end{aligned} \tag{6.7}$$

6.2.3 Reachability condition

In sliding mode control, one must ensure that the sliding motion starts in finite time from any arbitrary initial condition (Perruquetti and Barbot, 2002; Bandyopadhyay et al., 2009). The control must be designed such that it drives the state trajectories towards the sliding surface, and once achieved, maintains that sliding motion. The *attractivity* of the sliding surface can be expressed by the condition:

$$\lim_{s\to 0^+} \dot{s} < 0, \qquad \lim_{s\to 0^-} \dot{s} > 0. \tag{6.8}$$

The above conditions ensure that s and $\dot{s}$ have opposite signs, i.e.,

$$s\dot{s} < 0. \tag{6.9}$$

The above condition is referred to as the *reachability condition.* To check for the reachability condition here, let us take $\mathbf{W} = \frac{1}{2}s_1^2$ as the Lyapunov candidate function. The derivative of $\mathbf{W}$ is:

$$\begin{aligned}\dot{\mathbf{W}} &= s_1 \dot{s}_1 \\ &= s_1 \left(\frac{gu}{V_g} \cos \left(\sin^{-1} \left(\frac{1}{V_a} \left(-V_{w_n} \sin\chi + V_{w_e} \cos\chi \right) \right) \right) - \dot{\chi}_p + \frac{c_1 c_2}{1 + c_2^2 y_e^2} V_g \sin\chi_e \right).\end{aligned} \tag{6.10}$$

Substituting for the control input u from equation (6.7), we get:

$$\begin{aligned}\dot{\mathbf{W}} = s_1 \frac{g}{V_g} \cos \left(\sin^{-1} \left(\frac{1}{V_a} \left(-V_{w_n} \sin\chi + V_{w_e} \cos\chi \right) \right) \right) & \left[\frac{\hat{V}_g}{\hat{g}} \dot{\chi}_p \right. \\ \left. - \frac{c_1 c_2}{1 + c_2^2 y_e^2} \frac{\hat{V}_g^2}{\hat{g}} \sin\chi_e - k \operatorname{sgn}(s_1) \right] & + s_1 \left(-\dot{\chi}_p + \frac{c_1 c_2}{1 + c_2^2 y_e^2} V_g \sin\chi_e \right),\end{aligned} \tag{6.11}$$

or

$$\begin{aligned}\dot{\mathbf{W}} = s_1 \frac{g}{V_g} \tau & \left[\frac{\hat{V}_g}{\hat{g}} \dot{\chi}_p - \frac{c_1 c_2}{1 + c_2^2 y_e^2} \frac{\hat{V}_g^2}{\hat{g}} \sin\chi_e - k \operatorname{sgn}(s_1) \right] \\ & + s_1 \left(-\dot{\chi}_p + \frac{c_1 c_2}{1 + c_2^2 y_e^2} V_g \sin\chi_e \right),\end{aligned} \tag{6.12}$$

where $\tau = \cos \left(\sin^{-1} \left(\frac{1}{V_a} \left(-V_{w_n} \sin\chi + V_{w_e} \cos\chi \right) \right) \right)$. The uncertain variables $\hat{V}_g$ and $\hat{g}$ (with some measurement error) appear from substitution of the control input u above. Neglecting the error in the gravity term ($g \approx \hat{g}$), we have:

$$\dot{\mathbf{W}} = -\frac{s_1}{V_g} \left(\frac{c_1 c_2}{1 + c_2^2 y_e^2} \sin\chi_e \left(\tau \hat{V}_g^2 - V_g^2 \right) - \dot{\chi}_p \big(\tau \hat{V}_g - V_g \big) + gk\tau \operatorname{sgn}(s_1) \right). \tag{6.13}$$

$\dot{\mathbf{W}}$ will be negative definite if

$$|gk\tau| > \left| \frac{c_1 c_2}{1 + c_2^2 y_e^2} \sin\chi_e \left(\tau \hat{V}_g^2 - V_g^2 \right) \right| + \left| \dot{\chi}_p \big(\tau \hat{V}_g - V_g \big) \right|, \tag{6.14}$$

or

$$k > \frac{1}{g\tau}\frac{c_1c_2}{1+c_2^2y_e^2}\left|\sin\chi_e\right|\left|\tau\hat{V}_g^2 - V_g^2\right| + \frac{1}{g\tau}\left|\dot{\chi}_p\right|\left|\tau\hat{V}_g - V_g\right|. \tag{6.15}$$

To find the value of 'k' which satisfies the above for the entire flight envelope, i.e., to ensure reachability from any initial condition, we find the maximum of the right hand side in the above inequality. The maximum value of $\frac{c_1c_2}{1+c_2^2y_e^2}$ is c_1c_2 at $y_e = 0$, and the maximum value of $|\sin\chi_e|$ is 1 at $\chi_e = \pm\frac{\pi}{2}$. Hence, the maximum value in the entire phase portrait occurs at the point: $(\chi_e, y_e) = (\pm\frac{\pi}{2}, 0)$. Let us also assume a worst case error of 10% in the measurement of ground velocity (i.e., $\hat{V}_g \approx 1.1V_g$); now $\dot{\mathbf{W}}$ will be negative definite, if

$$k > \frac{V_g^2c_1c_2}{g}\left(1.21 - \frac{1}{\tau}\right) + \frac{V_g}{g}\left|\dot{\chi}_p\right|\left(1.1 - \frac{1}{\tau}\right). \tag{6.16}$$

Using the minimum value of $\frac{1}{\tau} = 1$ and the relation $V_g = \dot{\chi}_pR$, we have:

$$k > 0.21\frac{V_g^2c_1c_2}{g} + \frac{0.1}{g}\frac{V_g^2}{R}. \tag{6.17}$$

With this selection of gain k, the state trajectory will reach the sliding surface from any point in the phase portrait. The value of k mainly depends on velocity, sliding surface parameters and the radius of turn. Selection of a large value of the product c_1c_2 gives quick convergence of cross-track error vis-à-vis the requirement of a larger value of gain k (and larger control input u). Similarly more control is required for smaller values of turn radius (R). In the above derivation, it is assumed that a maximum error of 10% is expected in velocity measurement, however expression (6.17) may be modified accordingly in case of a different error magnitude for a specific application.

6.2.4 Control boundedness

A typical guidance and control system will have an outer guidance loop which generates roll reference commands for the inner control loop to follow. The inner loop will in practice have limiter and saturation blocks to avoid excessive bank maneuvers. For example, there could be a saturation check on the roll reference command being generated by the guidance logic, and also similar checks on the actuators driving the (aileron) control surfaces. Stability of the two interconnected loops in the presence of nonlinearities such as these is usually not formally proved; most practitioners rely on extensive numerical simulations to generate confidence before going to flight. Our objective here is for the outer loop to generate commands that do not cause saturation of signals in the inner loop, thus resulting in graceful maneuvers with guaranteed stability margins.

As discussed earlier, larger values of c_1 and c_2 are desirable for good performance and quick response in terms of driving the cross-track error to zero. We have also derived the reachability condition above and shown that the control gain k is related to the product c_1c_2. If an arbitrarily large k satisfying condition (6.17) is chosen, it may result in saturation of some signals in the inner loop, and thus the chosen gain k will not get actually applied to the system. In other words, saturation may result in a lower *effective* gain being applied to the system, which in an extreme case may even violate the reachability condition (6.17). In such a case, sliding motion will not be guaranteed. In order to avoid saturation therefore, it is necessary to bound the roll command (6.7) away from the maximum allowed value ϕ_{max}.

The output of the guidance block ($\phi_{req} = \arctan u$) is to be bounded by ϕ_{max} during both the reaching and sliding phases. Here we derive conditions on the sliding surface parameters c_1 and c_2 so that $|\phi_{req}| \leq \phi_{max}$ throughout the reaching and sliding phases. During reaching phase, the objective is to attract the state trajectory towards the sliding surface, any effective gain satisfying the reachability condition (6.17) can accomplish this. If the maximum allowed value of control input (ϕ_{max}) satisfies the condition (6.17), then the state trajectory will be attracted towards the sliding surface even in case of saturation $|\phi_{req}| = \phi_{max}$. However, control boundedness ($|\phi_{req}| \leq \phi_{max}$) is important during sliding motion to slide the states along the desired trajectory (sliding surface) with limited control input. Motion on the sliding surface is described by the equation (6.1) and the corresponding equivalent control is derived in section 6.2.1. In order to avoid control saturation, the total control magnitude should be bounded by ϕ_{max}, i.e.,

$$\left| -\frac{\hat{V}_g^2}{\hat{g}} \frac{c_1 c_2}{1 + c_2^2 y_e^2} \sin \chi_e + \frac{\hat{V}_g}{\hat{g}} \dot{\chi}_p - k \operatorname{sgn}(s_1) \right| \leq \tan \phi_{max},$$

or in the worst case:

$$\left| \frac{\hat{V}_g^2}{\hat{g}} \frac{c_1 c_2}{1 + c_2^2 y_e^2} \sin \chi_e \right| + \left| \frac{\hat{V}_g}{\hat{g}} \dot{\chi}_p \right| + |k \operatorname{sgn}(s_1)| \leq \tan \phi_{max}. \tag{6.18}$$

After simplification, we have:

$$\frac{c_1 c_2}{1 + c_2^2 y_e^2} |\sin \chi_e| \leq \frac{\hat{g}}{\hat{V}_g^2} \left(\tan \phi_{max} - k - \frac{\hat{V}_g}{\hat{g}} |\dot{\chi}_p| \right). \tag{6.19}$$

Substituting $\chi_e = -c_1 \arctan(c_2 y_e)$ from equation (6.1) into the above equation, we have:

$$\frac{c_1 c_2}{1 + c_2^2 y_e^2} |\sin (-c_1 \arctan(c_2 y_e))| \leq \frac{\hat{g}}{\hat{V}_g^2} \left(\tan \phi_{max} - k - \frac{\hat{V}_g}{\hat{g}} |\dot{\chi}_p| \right). \tag{6.20}$$

The parameters c_1 and c_2 and hence the control gain k should be chosen so that the above inequality is satisfied, and thus control saturation is avoided during motion on the sliding surface for all values of the track error y_e. Different approaches are possible to check the control boundedness condition (6.20).

Approach-1: One way to check for control boundedness is to first fix the parameters c_1, c_2 and k and then plot $\frac{c_1 c_2}{1+c_2^2 y_e^2} \left|\sin\left(-c_1 \arctan(c_2 y_e)\right)\right|$ versus y_e, and to check if this is less than $\frac{\hat{g}}{\hat{V}_g^2}\left(\tan\phi_{max} - k - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right|\right)$. If the check fails, c_1 and c_2 have to be readjusted and the check repeated again.

Approach-2: Another approach is to find the maximum point $y_{e_{max}}$ for the function $\frac{c_1 c_2}{1+c_2^2 y_e^2} \sin\left(-c_1 \arctan(c_2 y_e)\right)$ using calculus theory as explained below. Let us define the function $h(y_e)$ as follows:

$$h(y_e) = \frac{c_1 c_2}{1+c_2^2 y_e^2} \sin\left(-c_1 \arctan(c_2 y_e)\right).$$

In order to find the maximum, we first find the critical points by solving $\frac{dh}{dy_e} = 0$, i.e.,

$$\frac{d}{dy_e}\left\{\frac{c_1 c_2}{1+c_2^2 y_e^2} \sin\left(-c_1 \arctan(c_2 y_e)\right)\right\} = 0,$$

which becomes:

$$\frac{-2c_1 c_2^3 y_e}{(1+c_2^2 y_e^2)^2} \sin\left(-c_1 \arctan c_2 y_e\right) - \frac{c_1^2 c_2^2}{(1+c_2^2 y_e^2)^2} \cos\left(-c_1 \arctan c_2 y_e\right) = 0. \quad (6.21)$$

After simplification, we get:

$$\frac{-(c_1 c_2) c_2^2}{(1+c_2^2 y_e^2)^2} \sin\left\{-c_1 \arctan c_2 y_e + \arctan\left(\frac{c_1}{2c_2 y_e}\right)\right\} = 0. \quad (6.22)$$

Since $\frac{-(c_1 c_2) c_2^2}{(1+c_2^2 y_e^2)^2} \neq 0$ for any value of y_e, hence the critical points are the solutions to the equation:

$$\sin\left\{-c_1 \arctan c_2 y_e + \arctan\left(\frac{c_1}{2c_2 y_e}\right)\right\} = 0, \quad (6.23)$$

which may be written as:

$$c_1 \arctan c_2 y_e = \arctan\left(\frac{c_1}{2c_2 y_e}\right). \quad (6.24)$$

The above equation may be solved either analytically or numerically to get the critical points. For example, if we take $c_1 = 1$ (maximum performance for large cross-track errors), we can write the above equation as:

$$c_2 y_e = \frac{1}{2 c_2 y_e}, \tag{6.25}$$

which yields the two critical points $\pm\frac{1}{c_2\sqrt{2}}$ for the function $h(y_e)$. The point $y_{e_{max}} = \frac{1}{c_2\sqrt{2}}$ gives the maximum of the function, whereas $y_{min} = -\frac{1}{c_2\sqrt{2}}$ yields the minimum. Substituting in equation (6.20), we get:

$$\begin{aligned}
0.3849 c_2 &\leq \frac{\hat{g}}{\hat{V}_g^2}\left(\tan\phi_{max} - k - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right|\right) \\
\Rightarrow c_2 &\leq \frac{1}{0.3849}\frac{\hat{g}}{\hat{V}_g^2}\left(\tan\phi_{max} - k - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right|\right) \\
\Leftrightarrow k &\leq \tan\phi_{max} - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right| - 0.3849\frac{\hat{V}^2 c_2}{\hat{g}} \qquad (6.26)
\end{aligned}$$

Neglecting the error in the gravity term ($g \approx \hat{g}$) and using the assumption of maximum 10% uncertainty in velocity measurement ($\hat{V}_g \approx 1.1 V_g$), we may write the above as:

$$k \leq \tan\phi_{max} - \frac{V_g}{g}\left|\dot{\chi}_p\right| - 0.4657\frac{V_g^2 c_2}{g} \tag{6.27}$$

The above inequality along with the reachability condition (6.17) should be satisfied to ensure both control boundedness and reachability for the $c_1 = 1$ case.

Approach-3: A simple approach to check the control boundedness (6.20) is to write this inequality for the worst case scenario. This approach is conservative but simple. The maximum value of the term $\frac{c_1 c_2}{1+c_2^2 y_e^2}\left|\sin\left(-c_1 \arctan(c_2 y_e)\right)\right|$ cannot exceed $c_1 c_2$ in the entire phase plane (for all values of y_e and χ_e). So in order to bound the control input for all conditions, we must have:

$$c_1c_2 \leq \frac{\hat{g}}{\hat{V}_g^2}\left(\tan\phi_{max} - k - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right|\right), \tag{6.28}$$

or

$$k \leq \tan\phi_{max} - \frac{\hat{V}_g}{\hat{g}}\left|\dot{\chi}_p\right| - \frac{\hat{V}_g^2}{\hat{g}}c_1c_2. \tag{6.29}$$

The gain k and the sliding surface parameters c_1 and c_2 should be selected considering the UAV flight conditions (V_g, g), the maximum allowable roll angle (ϕ_{max}), and the mission requirement (R). First the sliding parameters c_1 and c_2 may be chosen, followed by the selection of the gain k that satisfies equations (6.17) and (6.29). If no feasible value of k is found, the above procedure may be reiterated with comparatively smaller values of the sliding parameters c_1 and c_2.

6.2.5 Implementation issues

To implement the SMC based scheme in real systems, different methods are suggested in the literature to avoid chattering in the control variable because of the *signum* function. One commonly used approach is the 'boundary layer approach': a continuous *sigmoid* approximation $\frac{s_1}{|s_1|+\epsilon}$ of the discontinuous function $\text{sgn}(s_1)$ (Burton and Zinober, 1986; Boiko, 2013). Due to this approximation, the control magnitude decreases as the system trajectory approaches the sliding surface. As a result the sliding surface is not exactly zero (ideal sliding mode) and the system state trajectories are restricted to a small vicinity of the sliding surface (real sliding mode).

The sigmoid approximation ($\frac{s_1}{|s_1|+\epsilon}$) of $\text{sgn}(s_1)$ is plotted versus s_1 for different ϵ in Figure 6.1. In the horizontal axis as s_1 approaches zero, magnitude of the effective gain (vertical axis) also reduces, hence resulting in reduced chattering in

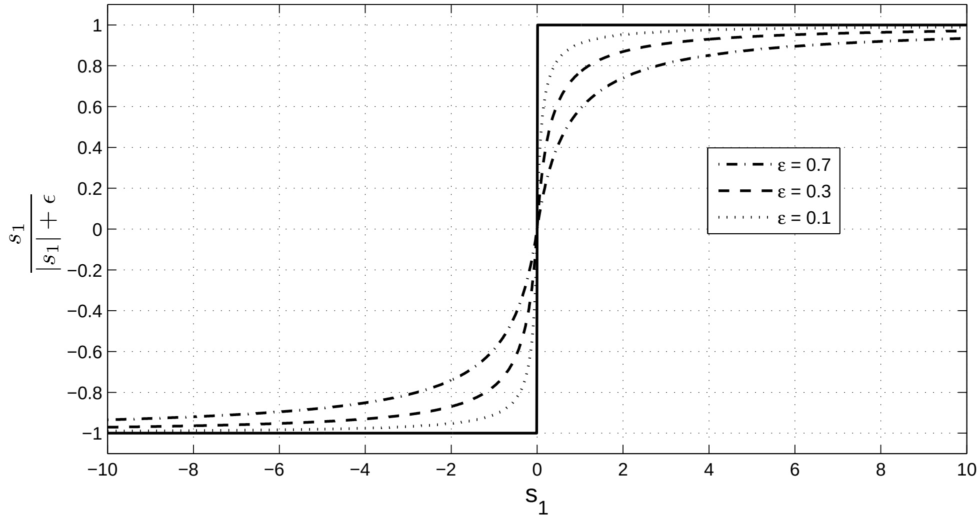


FIGURE 6.1: Approximation of sgn(s_1) function.

the control signal. As the gain reduces due to this approximation, hence control boundedness (6.20) is not a problem here.

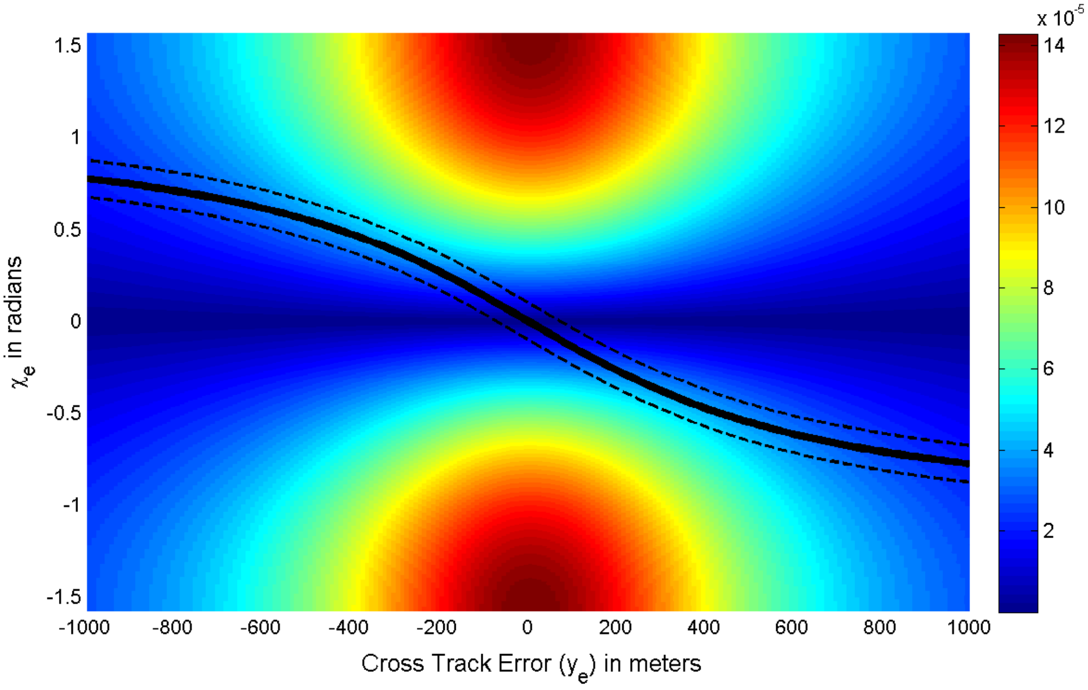


FIGURE 6.2: Gain k required for reachability over the phase plane.

As the effective gain reduces in the vicinity of the sliding surface due to this sigmoid approximation, hence its effect on the reachability condition (6.17) should also be analyzed. This condition (6.17) gives the gain required to attract the states towards the sliding surface from the worst point in the phase plane (the required worst maximum gain), the gain requirement from other points in the phase plane is relatively lower. The worst point in the phase plane is $(\chi_e, y) = (\pm\frac{\pi}{2}, 0)$. To analyze the gain requirement at other points of the phase portrait, the first term in the generalized reachability condition (6.15) is dominant, and this has a variable factor $\frac{1}{g}\frac{c_1 c_2}{1+c_2^2 y_e^2}|\sin\chi_e|$ depending on the states. This factor is shown by a colored mesh in Figure 6.2 for the entire phase plane (from this figure it is also clear that the worst point is $(\pm\frac{\pi}{2}, 0)$). Gain requirement at other points of the phase portrait can also be worked out using this plot in terms of its ratio to the maximum required gain. The sliding surface is also shown in black solid line for $c_1 = 0.7, c_2 = 0.002$ to see the gain requirement in its neighborhood. It is clear from the figure that the requirement of gain in locality of the sliding surface is not as much as compared to the maximum, the requirement reduces to one fifteenth of the maximum near the origin.

The effect of sigmoid approximation in terms of gain reduction and the resultant corresponding boundary in the vicinity of the sliding surface can be analyzed for each specific case. As an example let us assume that we want a sliding accuracy of 0.1 ($|s_1| < 0.1$), this implies a boundary layer width of 0.2. For $\epsilon = 0.3$ we have $\frac{s_1}{|s_1|+\epsilon} = \pm 0.25$ at $s_1 = \pm 0.1$, which indicates that one fourth of the selected gain will effectively be applied to the system at the edges of the boundary layer. Figure 6.2 shows the boundary layer with black dashed lines, it is clear that the gain requirement is approximately one fourth of the maximum on the edges of this boundary layer, implying we can achieve an accuracy of $|s_1| < 0.1$ with $\epsilon = 0.3$.

FIGURE 6.3: A photograph of the test vehicle during landing.

6.3 Case Study: Application on actual UAV

The proposed guidance law is programmed in the flight control computer of a scaled YAK-54 UAV (Figure 6.3) to demonstrate its effectiveness; comparisons are also made with other guidance laws. The UAV under consideration is a research vehicle (a scaled version of the YAK-54) already used by many researchers to demonstrate their work (Jager, 2008; Shahriar keshmiri and Hale, 2008). It is propeller driven in a tractor configuration. Pitch control is provided by a set of elevators located on the horizontal tail, roll control is provided by ailerons on the main wing, and the vertical tail employs a rudder. The cruising speed is in the range of 30–40 m/sec. Basic data of the said vehicle is listed in Table 6.1.

Before presenting flight test results, first the complete cycle of design process is explained here in this section. The design process is explained for both cases of straight and circular path following. A complete cycle includes: selection of optimized sliding surface parameters (c_1 and c_2), finding the gain k satisfying both reachability and control boundedness conditions, and finally a complete guidance

Table 6.1: Geometrical and mass properties of the scaled YAK-54 UAV.

#	Parameter	Value
1	Take-off mass	11 kg
2	Length	2080 mm
3	Wing span	2235 mm
4	Wing area	943869 mm^2
5	Wing root/tip airfoil	NACA 0016 / 0017
6	Vertical tail root/tip airfoil	NACA 0009 / 0010
7	Horizontal tail root/tip airfoil	NACA 0015 / 0012
8	Moments of inertia: Ixx, Iyy, Izz	$1.36, 2.848, 4.07$ kg-m^2

law expression for implementation in the flight control computer. All these steps are covered in the subsequent subsections.

6.3.1 Optimized sliding surface parameters

As discussed in detail in Chapter 5, the selection criterion for sliding surface parameters is different for straight and circular path following. Here the optimum sliding parameters are worked out for both cases.

6.3.1.1 Straight path case

A criterion is derived in section 5.4.1 for the selection of sliding parameters using the work and energy principle. For constant force, the work done expression mainly depends on the travel distance of UAV. Hence minimizing the travel path/distance will minimize the work done value. It may also be mentioned here that path length is the same for both positive and negative value of y_0 (for the straight path case), and hence only positive y_0 can be considered for selection of sliding parameters. From equation (5.17), we have the following expression for path length calculation:

$$\text{Path Length} = \int\limits_{y=y_0}^{1} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy \quad \text{subject to} \quad F_{lateral} \leq F_{lateral_{max}}. \tag{6.30}$$

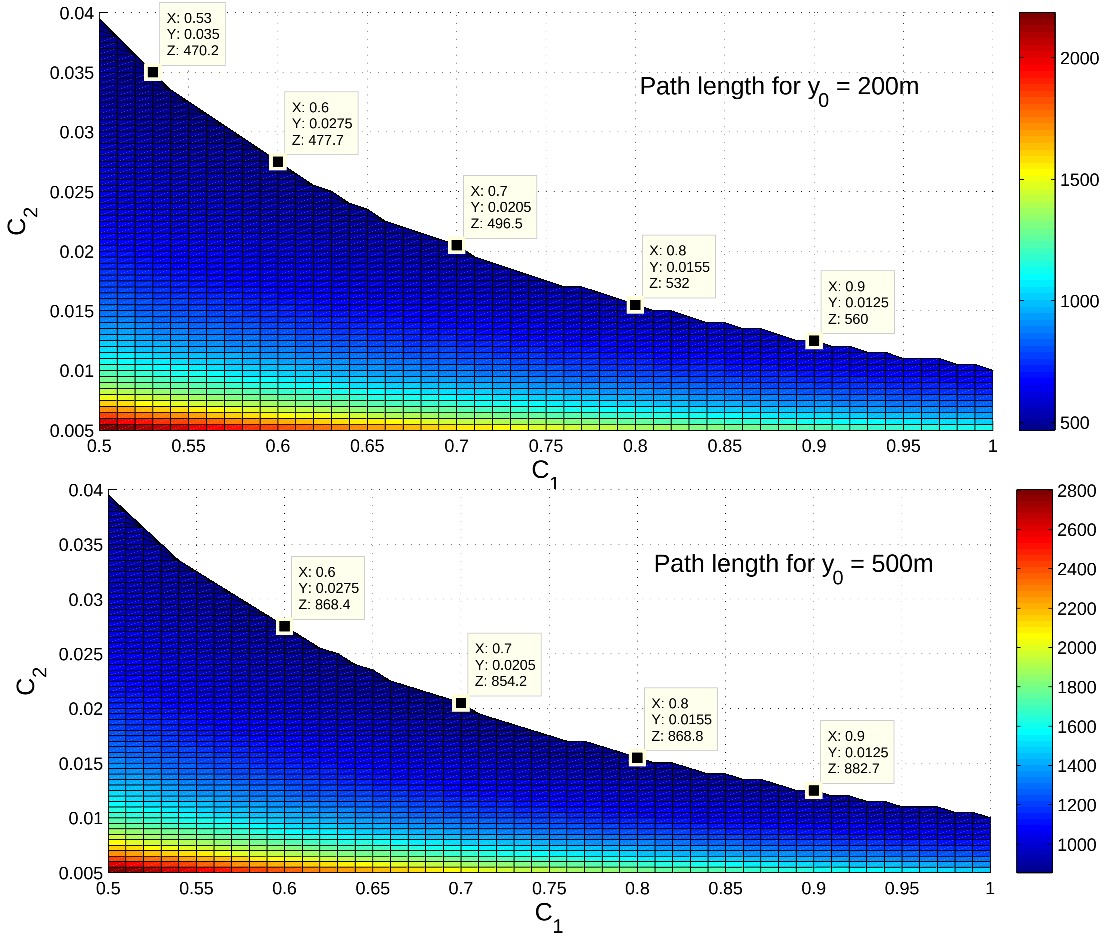


FIGURE 6.4: Path length for different values of sliding parameters (cases of small initial cross track error).

During sliding, the constraint equation $F_{lateral} \leq F_{lateral_{max}}$ can be derived from the equivalent control expression (6.4). Assuming a maximum bank/roll angle of 27 degrees (with some margins for uncertainty), velocity of 35 m/s, gravitational acceleration equal to $9.8m/s^2$ and neglecting the wind, the above equation (6.30) becomes:

$$\left.\begin{aligned} \text{Path Length} &= \int_{y=y_0}^{1} \frac{1}{\sin\left(c_1 \arctan(c_2 y)\right)} dy \\ \text{subject to } & \frac{c_1 c_2}{1+c_2^2 y^2}\frac{35^2}{9.8}\sin\left(c_1 \arctan(c_2 y)\right) < 0.5 \end{aligned}\right\} \qquad (6.31)$$

The path length can be calculated for specific y_0 by numerical integration of the above expression from $y = y_0$ to $y = 1$ for different values of c_1 and c_2. During integration, the constraint $(\frac{c_1 c_2}{1+c_2^2 y^2}\frac{35^2}{9.8}\sin(c_1 \arctan(c_2 y)) < 0.5)$ should be checked at every point of integration and if violated then that combination of c_1 and c_2 should be dropped out.

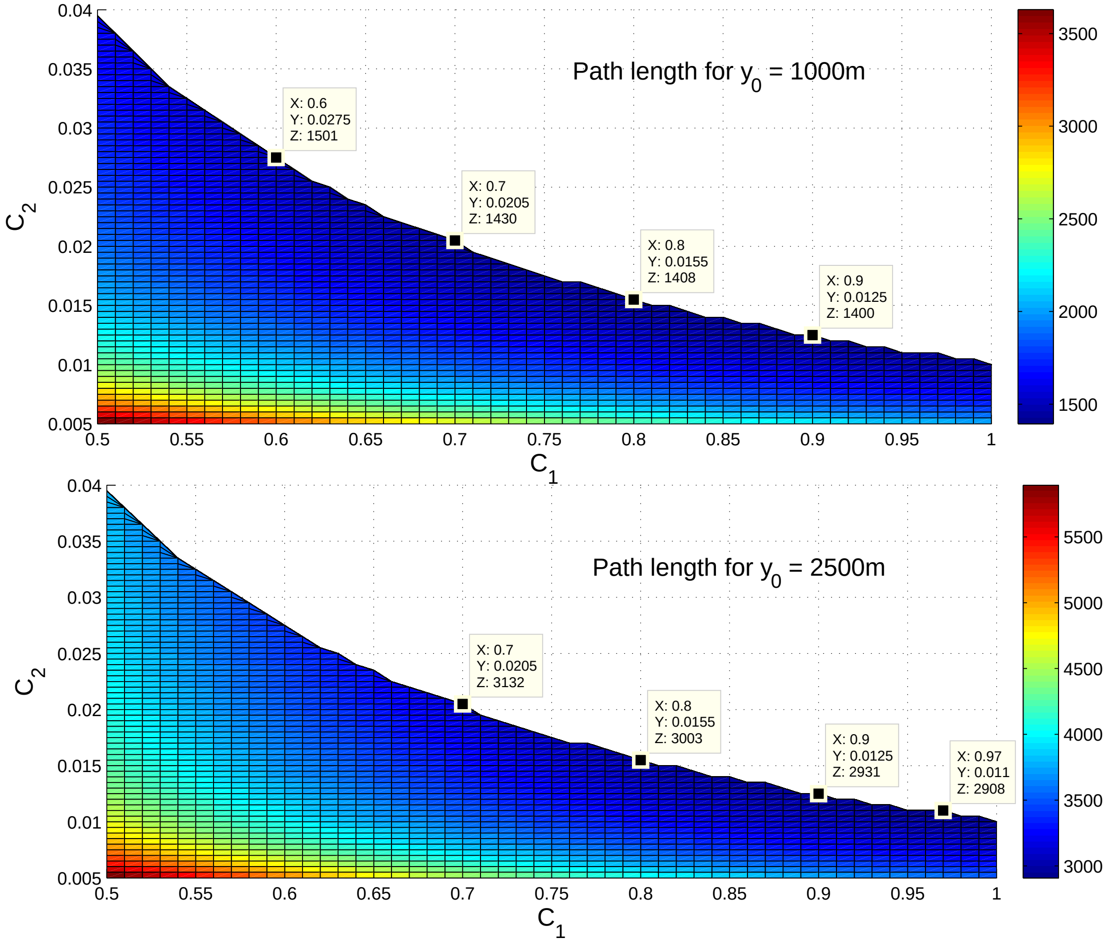


FIGURE 6.5: Path length for different values of sliding parameters (cases of large initial cross track error).

In Figure 6.4, the resultant values of traveled path are shown for initial small cross track error (y_0) of 200 and 500 meters. The path length is worked out for different values of c_1 (0.5 to 1.0) and c_2 (0.005 to 0.04), and some combinations are naturally dropped out because of the constraint violation. In case of $y_0 = 200$, the

TABLE 6.2: Summary of travel path length for different y_0 (straight path case).

$y_0[m]$	Path length [m]	Best combination (c_1, c_2)	Near optimum range (c_1, c_2)
200	470 - 2300	(0.53, 0.0350)	(0.5-0.7, 0.035-0.020)
500	800 - 2800	(0.70, 0.0200)	(0.6-0.9, 0.027-0.013)
1000	1390 - 3700	(0.90, 0.0125)	(0.7-1.0, 0.020-0.011)
2500	2900 - 5900	(0.97, 0.0110)	(0.8-1.0, 0.015-0.011)

length of traveled path varies from $470m$ to $2300m$ depending on the combination of c_1 and c_2. Similarly, the path length varies between $800 - 2800m$ in case of $y_0 = 500m$. The traveled path length for large initial cross track errors (1000 and $2500m$) are shown in Figure 6.5. All these results are summarized in table 6.2. To avoid complexity, a single combination of c_1 and c_2 can be chosen that gives near to optimum performance for all cases of y_0. For flight tests, we have chosen $c_1 = 0.9$ and $c_2 = 0.012$ for straight path following. The expected UAV ground trajectory with the selected sliding surface ($s_1 = \chi_e + 0.9 \arctan(0.012 y_e)$) for different initial cross track errors is shown in Figure 6.6. First the trajectory is directed towards the desired path almost straight, and once in the neighborhood of the desired path, it turns along the path utilizing the bank angle within the allowable limit.

6.3.1.2 Circular path case

Following a similar procedure, optimized sliding coefficients can be found for the circular path following case using equation (5.25). However, path lengths would be different for positive and negative values of initial cross track error of same magnitude, and both cases would be considered here. Similarly, the constraint $F_{lateral} \leq F_{lateral_{max}}$ can be derived from the equivalent control expression (6.4) in this case too. This constraint in the worst case can be written as (neglecting the

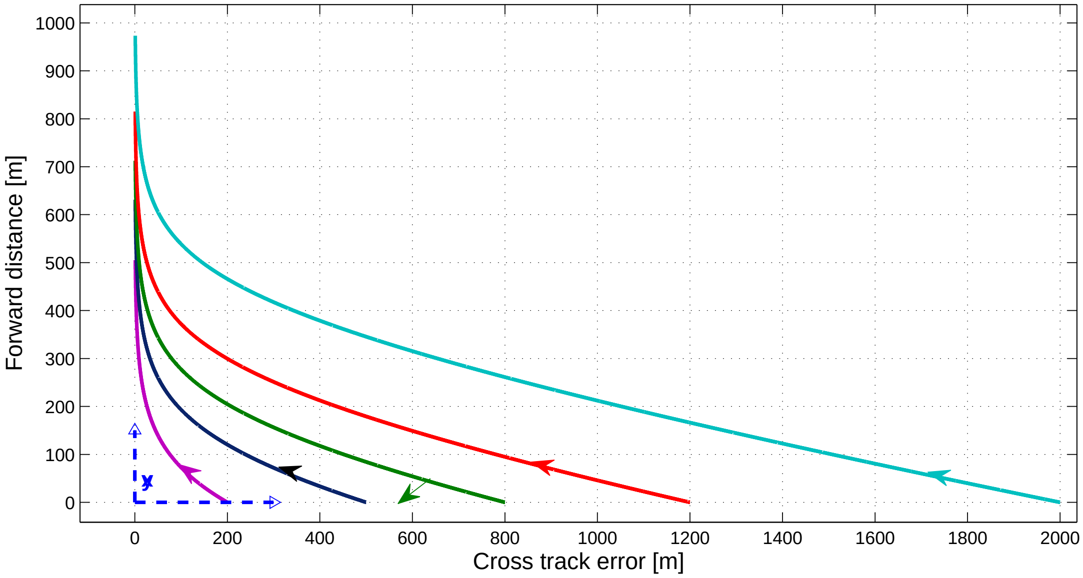


FIGURE 6.6: Straight path case: UAV trajectory on ground with selected sliding surface for different cross-track errors.

wind):

$$\left|\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p\right| + \left|\frac{c_1 c_2}{1 + c_2^2 y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e\right| \leq tan(\phi_{max}). \tag{6.32}$$

In case of circular path, this constraint also depends on the rate of change of reference course angle ($\dot{\chi}_p$), and a more strict constraint in case of circular arc of small radius. So keeping in view the worst case scenario, here we have derived sliding parameters for the smallest turn radius case (R=400m), the sliding parameters for any other turn radius can be derived using the same procedure. For turn radius of 400m, approximately 20deg roll angle is required to follow the curved path and hence we have a very small budget for any cross track error correction after keeping some budget for uncertainties. However, we have sufficient lateral force (or roll angle) for relatively large turn radii and the straight path (which can be considered as an infinity turn radius case with more room to correct errors). Using the equivalent control expression (6.32) and work done equation (5.25), we

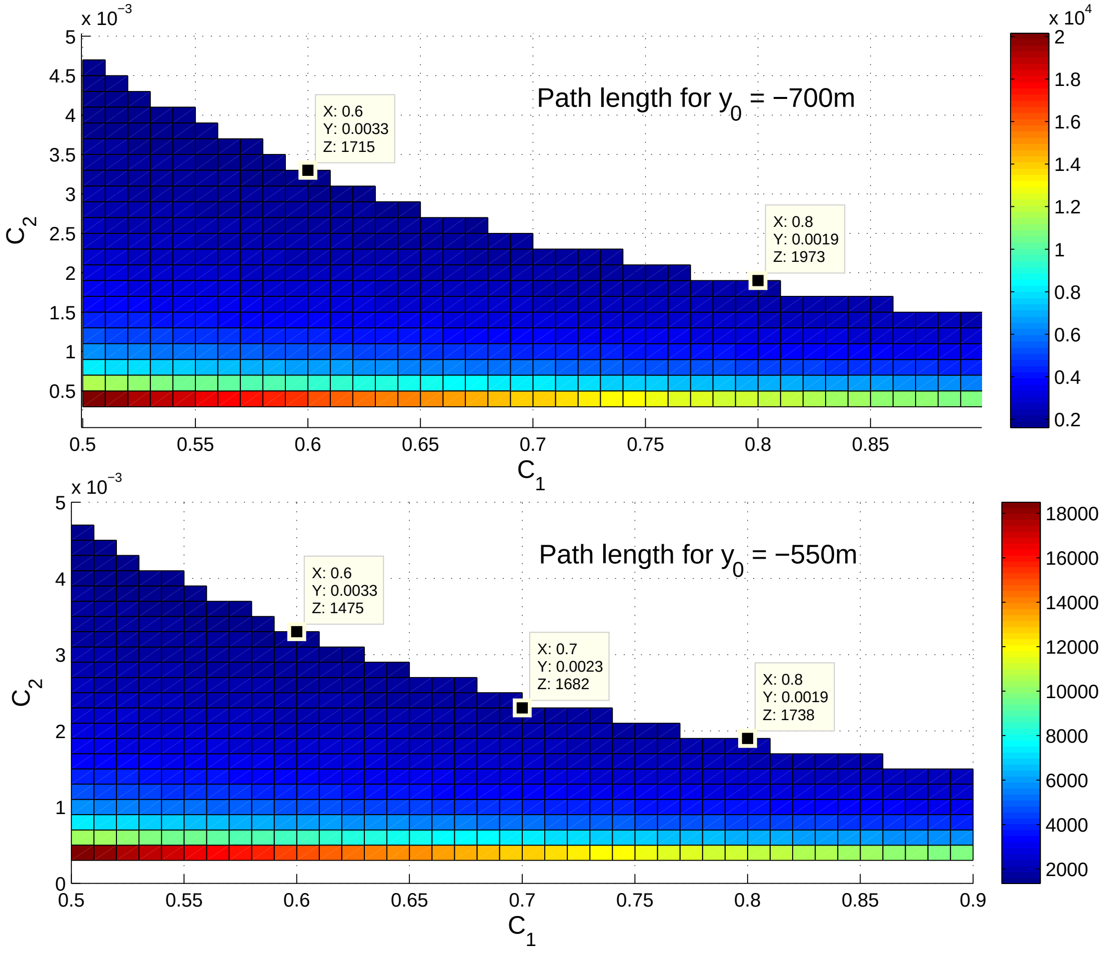


FIGURE 6.7: Circular path case: Path length for different values of sliding parameters (cases of large negative initial cross track error).

have the following expression for path length calculation in case of desired circular paths:

$$\left.\begin{aligned}\text{Path Length} &= \int_{r=r_0}^{\Delta} (\sin(c_1 \arctan(c_2(R-r)))dr + r\cos(c_1 \arctan(c_2(R-r)))d\eta)\\ &\text{subject to } \frac{c_1 c_2}{1+c_2^2(R-r)^2}\frac{35^2}{9.8}\left|\sin\left(c_1 \arctan(c_2(R-r))\right)\right| < 0.1\end{aligned}\right\} \tag{6.33}$$

As the objective here is to find the sliding parameters by comparing the path length for different combinations of c_1 and c_2, hence Δ can be chosen to be any number

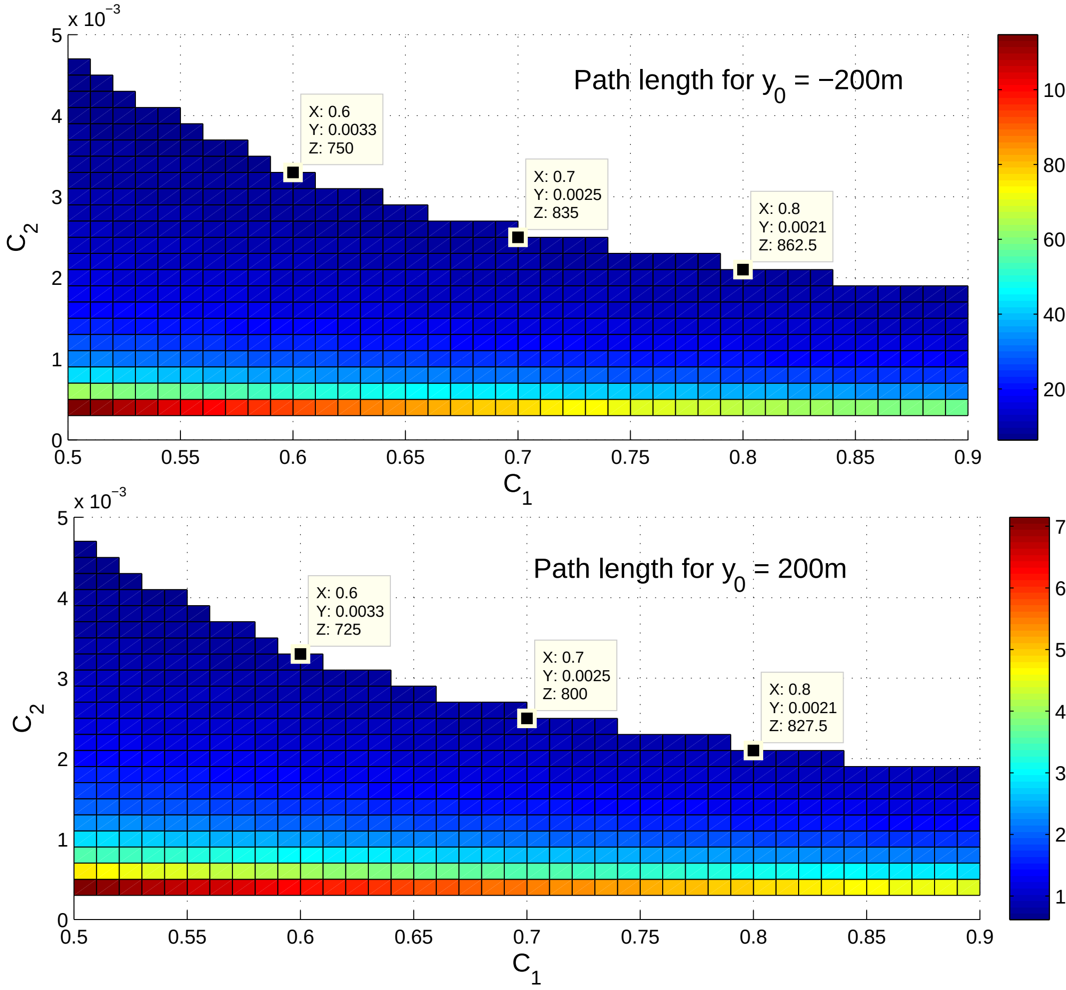


FIGURE 6.8: Circular path case: Path length for different values of sliding parameters (cases of small initial cross track error).

that defines the acceptable error to stop the integration. In case of a circular path, it takes a long time for the error to reduce, so we stop the integration when the cross track error reduces to $20m$. In Figure 6.7, path length calculations are shown for initial cross track error (y_0) of -700 and $-550m$. Similarly, the worked out travel path values for the cases of y_0 of $-200m$ and $200m$ are shown in Figure 6.8. In all these four cases, the path length for different combinations of c_1 and c_2 is calculated and any combination violating the constraint is dropped out. By analyzing these graphs similar to the previous straight path following case, we select $c_1 = 0.7$ and $c_2 = 0.002$ for implementation in the flight control computer. With these selected sliding parameters, the trajectory converges towards the desired path for different cases of initial cross track error as shown in Figure 6.9. In all cases, the error converges smoothly towards the desired path without any overshoot.

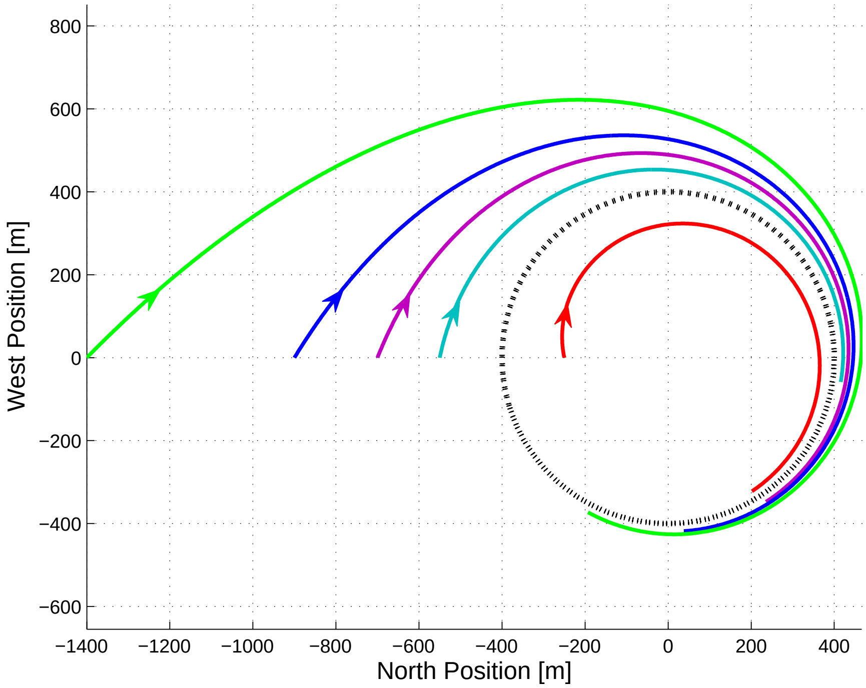


FIGURE 6.9: Circular path case: UAV trajectory on ground with selected sliding surface for different cross-track errors.

6.3.2 Reachability and control boundedness

After the selection of sliding surface parameters, the next step is to derive the reachability condition (6.17) and hence find the feasible value of control gain 'k' and also to check the control boundedness with the selected gain. The selection of gain is different for the straight and circular path cases.

For straight path following case, the reachability condition (6.17) with selected sliding parameters ($c_1 = 0.9$ and $c_2 = 0.012$) is ensured for $k > 0.238$. Finally, a control gain (k) value of 0.24 is selected for flight. To ensure control boundedness

(6.20), we used the approach-1 as explained in detail above. After fixing the values of parameters c_1, c_2 and k; the right hand side of inequality (6.20) is a constant number and left hand side depends on y_e. Both sides of the inequality are plotted against the track error y_e in Figure 6.10. It is clearly seen that the condition (6.20) is satisfied, thus ensuring $|\phi_{req}| \leq \phi_{max}$ (we choose $\phi_{max} = 35^\circ$ for our application). The figure also shows that the maximum control effort during sliding occurs for a track error of ~ 55 m; this is confirmed from flight results.

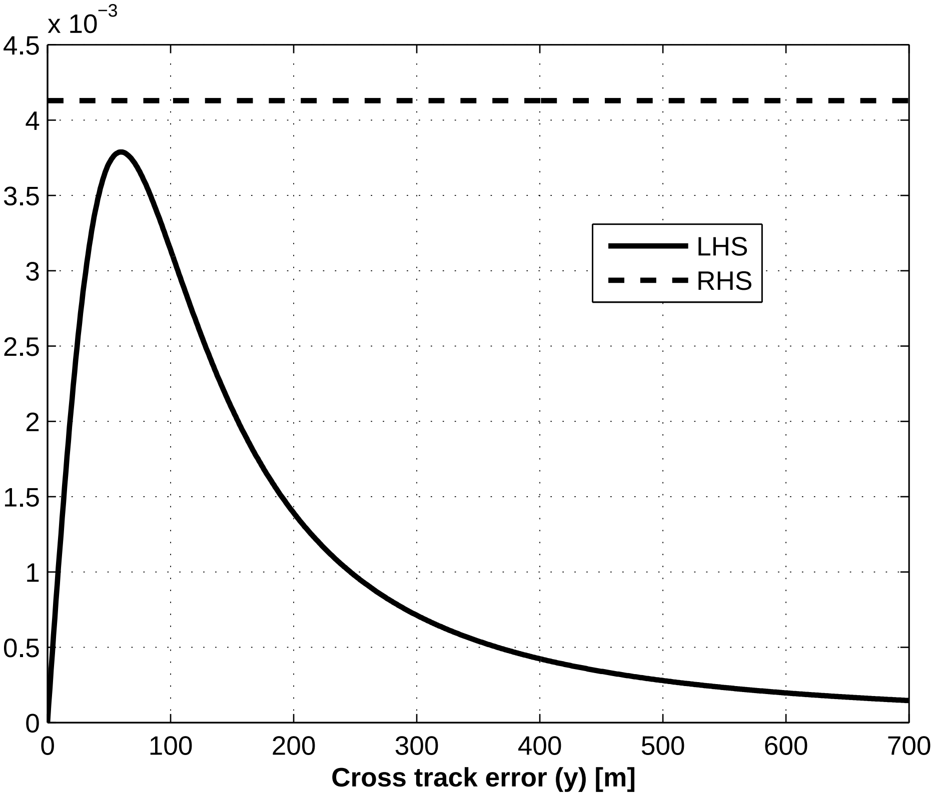


FIGURE 6.10: Verification of the control effort boundedness for straight path case (Condition 6.20).

For circular path following case, the sliding parameters (c_1 and c_2) and turn radius (R) dictates the minimum requirement of gain k, and can be workout using the reachability condition (6.17). To check the control boundedness, the easiest way is to follow the approach-3 (conservative approach) mentioned above. In this case, the reachability condition (6.17) and control boundedness (6.29) will impose minimum and maximum control gain requirements, respectively. With the selected sliding parameter values ($c_1 = 0.7$ and $c_2 = 0.002$) and $\dot{\chi}_p = 0.0875$, conditions for reachability and control boundedness become: $0.06 < k < 0.212$. We choose a larger gain of 0.21 for flight implementation.

6.3.3 Final lateral guidance law

A boundary layer approximation $\text{sgn}(s) \approx \frac{s}{|s|+\epsilon}$ is used (section 6.2.5) to avoid chattering in the computation of ϕ_{req}. For flight we want to achieve a sliding accuracy of 0.1 (i.e., $|s| < 0.1$) and this can be achieved with $\epsilon = 0.3$ for $k = 0.06$ as discussed in section 6.2.5. Lot of simulations are performed for different scenarios by varying different parameters, and simulation runs suggest that $0.5 \sim 0.7$ is a good range of ϵ for implementation.

In case of circular path following, we can still achieve accuracy of $|s| < 0.1$ with $\epsilon = 0.7$ as we have already chosen a larger gain of 0.21 for implementation. With $\epsilon = 0.7$, the effective gain reduced to one eighth of the selected gain of 0.21 near the boundary layer limits, but this effective gain ($0.21 * 0.125$) is sufficient to meet the reachability condition around the limits of boundary layer ($|s| < 0.1$). With this larger gain we will have greater disturbance rejection outside the boundary layer, and can achieve $|s| < 0.1$ even with $\epsilon = 0.7$, which gives a much smoother control signal. In case of straight path following with $\epsilon = 0.7$, the sliding accuracy of $|s| < 0.1$ is not possible throughout the sliding phase. However, as the control gain requirement reduces significantly near the origin (see Figure 6.2) so $|s| < 0.1$ accuracy near the origin is achievable with $\epsilon = 0.7$ (and this is our region of interest). Maximum deviation of 0.2 from the sliding surface is expected near the turning region of sliding surface.

Finally, the implemented lateral guidance law for straight path following is:

$$\begin{aligned} u &= -\frac{0.0108}{1+0.000144y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e - 0.24\frac{(\chi_e + 0.9\arctan(0.012y_e))}{|\chi_e + 0.9\arctan(0.012y_e)| + 0.7}\ , \\ \phi_{req} &= tan^{-1}(u). \end{aligned} \tag{6.34}$$

Similarly, the lateral guidance scheme for circular path following is:

$$\begin{aligned} u &= \frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p - \frac{0.0014}{1+0.000004y_e^2}\frac{\hat{V}_g^2}{\hat{g}}\sin\chi_e - 0.21\frac{(\chi_e + 0.7\arctan(0.002y_e))}{|\chi_e + 0.7\arctan(0.002y_e)| + 0.7}\ , \\ \phi_{req} &= tan^{-1}(u). \end{aligned} \tag{6.35}$$

As compared to expression (6.34), here we also have a feed-forward term $\frac{\hat{V}_g}{\hat{g}}\dot{\chi}_p$ to keep the UAV on a circular track. To show the effectiveness of this feed-forward term, we test this algorithm (6.35) with and without the feed-forward term. It may be noted that $\hat{V}_g$ is available from GPS, $\hat{g}$ is a known function of measured altitude, state variable (y_e and χ_e) measurement is also available and $\dot{\chi}_p$ is available from the pre-planned mission.

6.4 Experimental Results

Before implementation in the flight control computer, the proposed guidance law is simulated in the 6-dof nonlinear simulation to see the performance (see Annex-A for complete structure of the nonlinear simulation). After extensive testing in simulation, the proposed guidance law is programmed in the flight control computer of the test vehicle to demonstrate its effectiveness. The 30% scaled YAK-54 model is available ARF (almost ready to fly), manufactured by EG-Aircraft for remote controlled flying and is powered by a DLE-55 engine. The structure of the UAV is modified to house sensors and a flight computer for autopilot controlled flight; the layout of the flight control computer is shown in

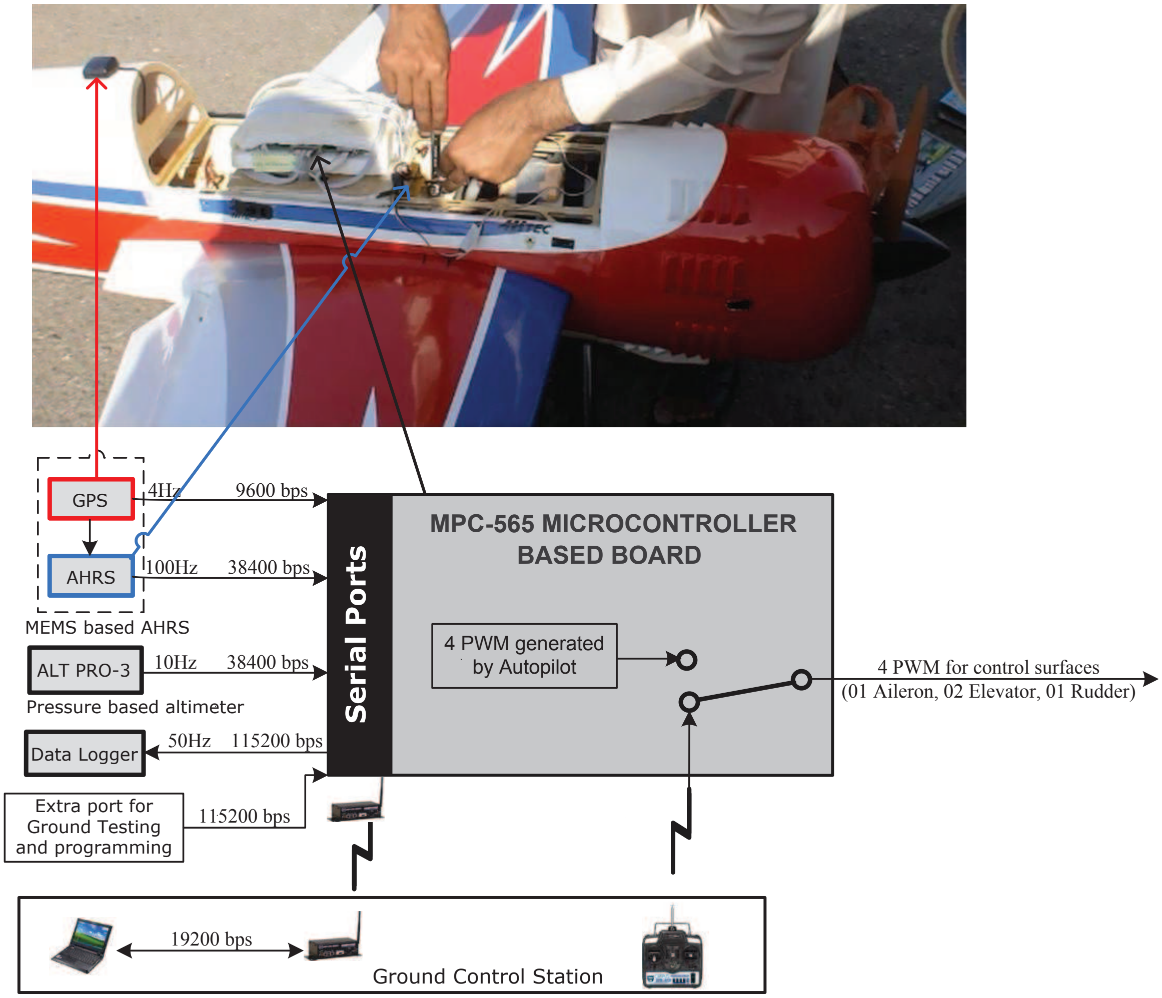


FIGURE 6.11: Main interfaces of the flight control computer.

Figure 6.11. Normally take-off and landing are performed manually, and the rest of the flight is conducted in autopilot mode. The heart of the flight computer is a generic MPC-565 microcontroller based board with 6 serial ports for communication with different on-board sensors and the ground terminal. RC (remote control) receiver channel-7 is used for switching between manual and autopilot modes. For attitude measurement and navigation, a MEMS based AHRS (attitude and heading reference system) with an integrated 4Hz GPS is employed. It outputs attitude and velocity information at an update rate of 100 Hz, whereas the position information is updated at 4 Hz.

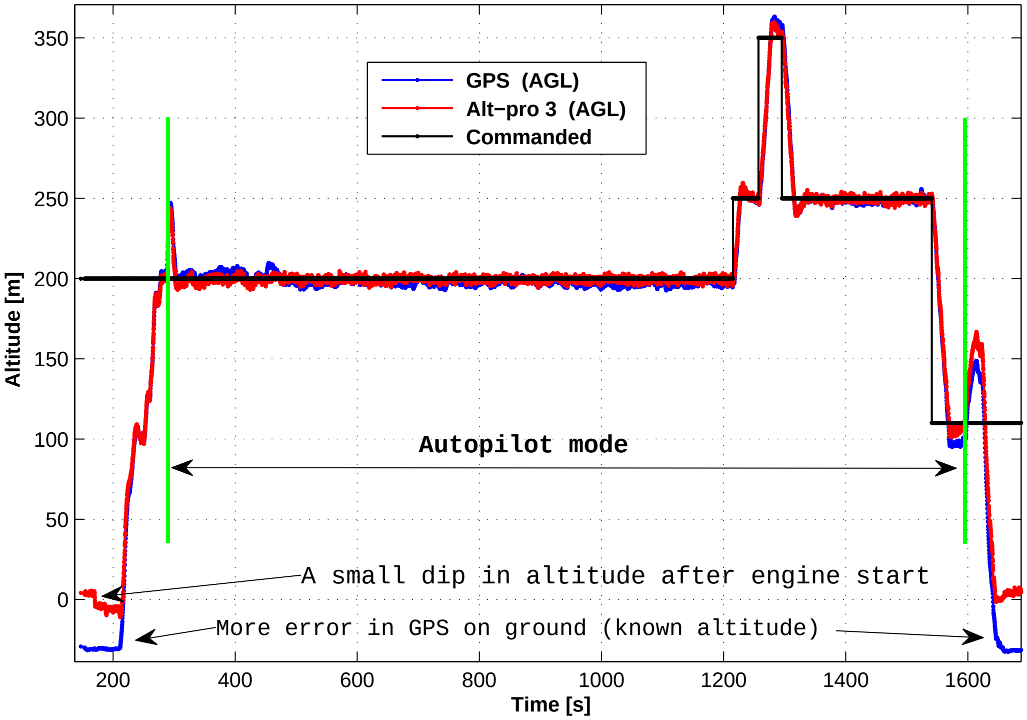


FIGURE 6.12: Altitude Tracking of the YAK-54.

For altitude control, a low-cost pressure altitude sensor RC-AltPro-3 (www.rc-electronics.org) is used that outputs mean sea level altitude, along with the rate of climb/descent at an update rate of 10 Hz. The altimeter gives acceptable performance; comparison with GPS altitude measurements is shown in Figure 6.12. Detailed flight results of altitude control are discussed in Annex-C. Communication with the ground terminal is through a 19.2 kbps RF modem which talks to the flight computer through a serial port. Layout of the ground station is shown in Annex-C. Telemetry of important flight parameters is made via the RF modem at 4 Hz, however detailed flight data is stored in a data recorder at 115.2 kbps. With the available fuel capacity, the UAV can fly autonomously for about one hour but most flights are of approximately 30 minutes duration. Since we are focusing here on the design of the guidance loop, therefore an existing PID-based roll control law is used for the inner loop as discussed in detailed in Annex-A. Two flights are conducted with the proposed lateral guidance scheme and results are shown in the subsequent subsections.

6.4.1 Straight path cases

Different scenarios of small, large and sharp turns are generated in the flight and results are shown here. A complete mission is also demonstrated here.

6.4.1.1 Small cross-track error case

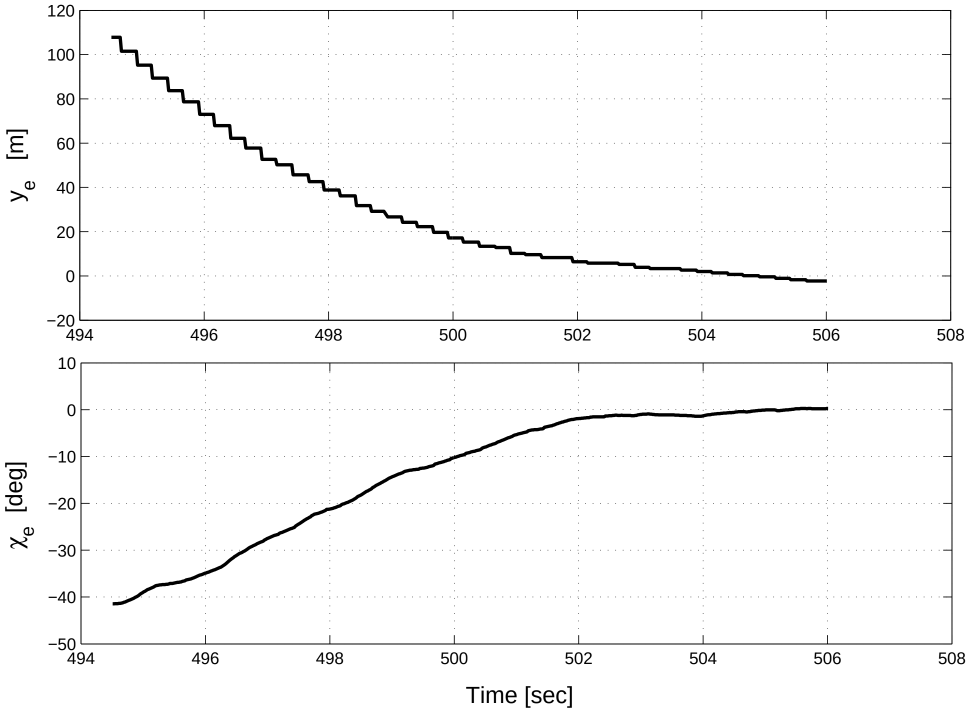


FIGURE 6.13: Flight test results for a small initial cross-track error (y_e and χ_e versus time).

Figures 6.13–6.15 show flight results for an initial cross-track error of 108 m during a straight leg between two waypoints. An initial track error may be generated in several ways: the guidance algorithm may be enabled some time after take-off, or the GPS may become available after a period of outage, or there may be a change in the mission during flight. Figure 6.13 shows the cross-track error y_e, and intercept course χ_e versus time, the track error reduces to within 10 m in $\sim$ 6.5

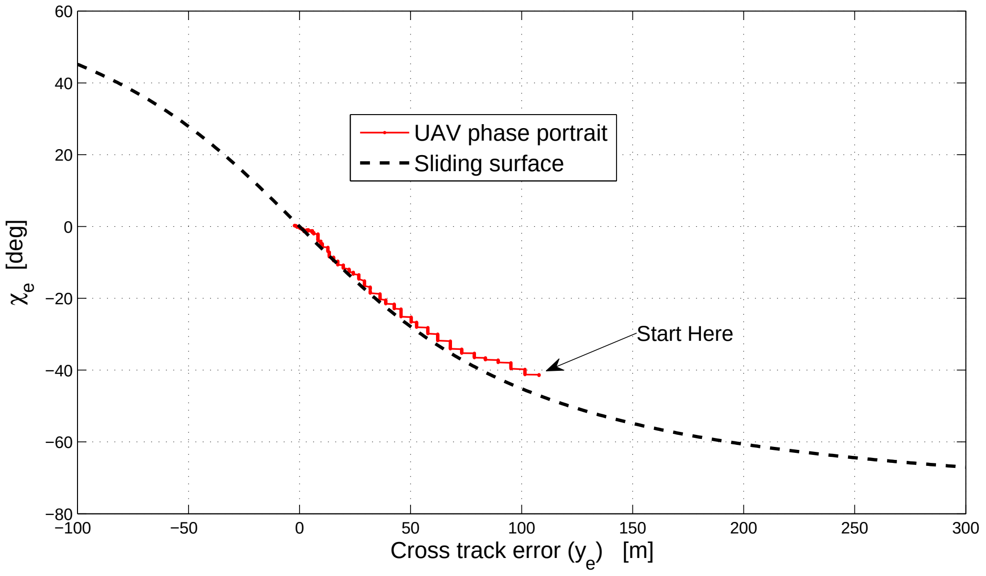


FIGURE 6.14: Phase portrait for the small track error case(χ_e versus y_e).

sec. In Figure 6.14, the UAV state trajectory (χ_e vs y_e) is shown in phase portrait form along with the sliding surface. Starting from the initial point, the state trajectory is first attracted towards the sliding surface, and thereafter remains on it. The commanded (reference) roll angle, actual roll angle and aileron deflection are shown in Figure 6.15. The maximum control effort (roll reference) is 25° at $\sim$ 497 sec when the track error is approximately 55 m, this proves our earlier claim from Figure 6.10.

6.4.1.2 Large track error case

Flight results for a 500 m initial cross-track error are shown in Figures 6.16–6.18. The track error y_e and χ_e versus time are shown in Figure 6.16; it takes $\sim$ 19 sec to reduce the error from 500 m to 10 m. The track error reduces to zero smoothly with no overshoot. The phase portrait is shown in Figure 6.17 along with the sliding surface. The state trajectory converges to the origin on the sliding surface, the trajectory follows the sliding surface quite closely. In the initial few seconds, χ_e is big and almost constant, and reduces the cross-track error quickly, achieving good performance. In Figure 6.18, the reference (commanded) roll angle along

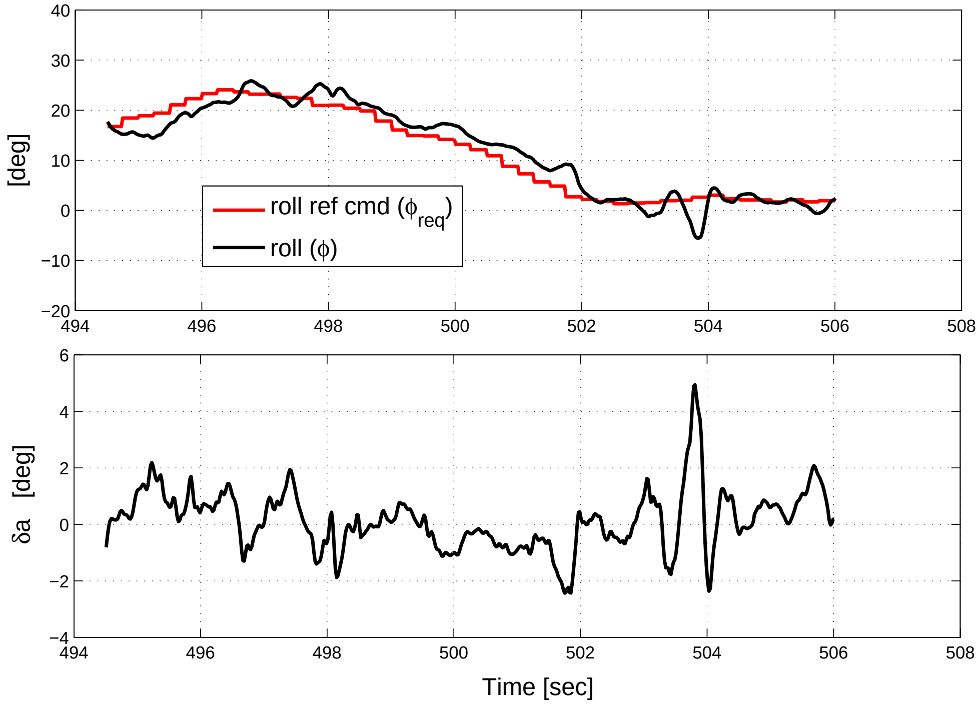


FIGURE 6.15: Roll angle and aileron deflection for the small cross-track error scenario.

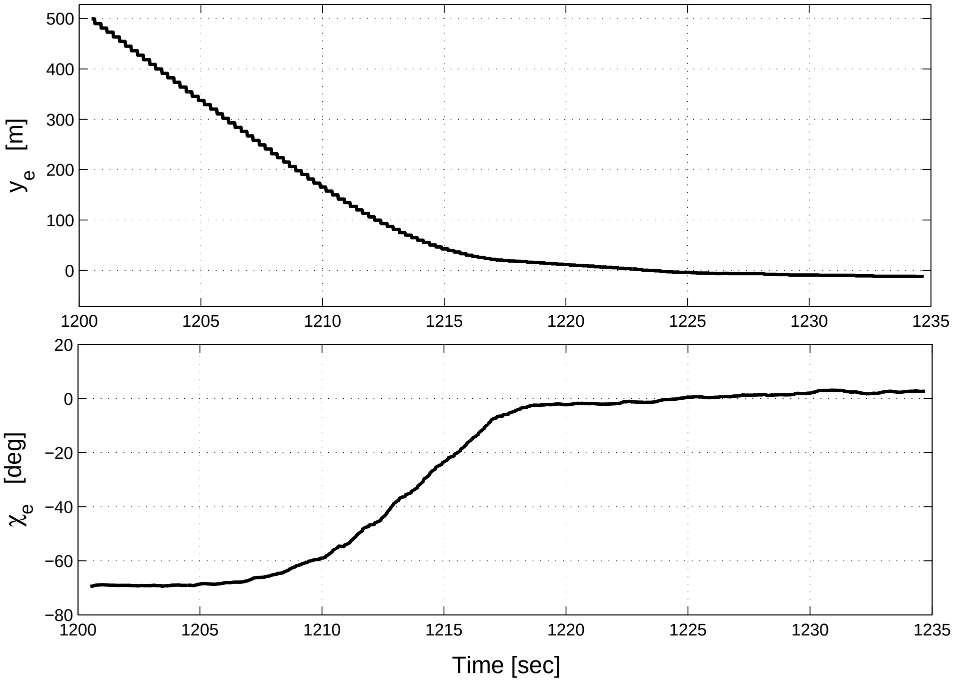


FIGURE 6.16: Flight test results for a large initial cross-track error (y_e and χ_e versus time).

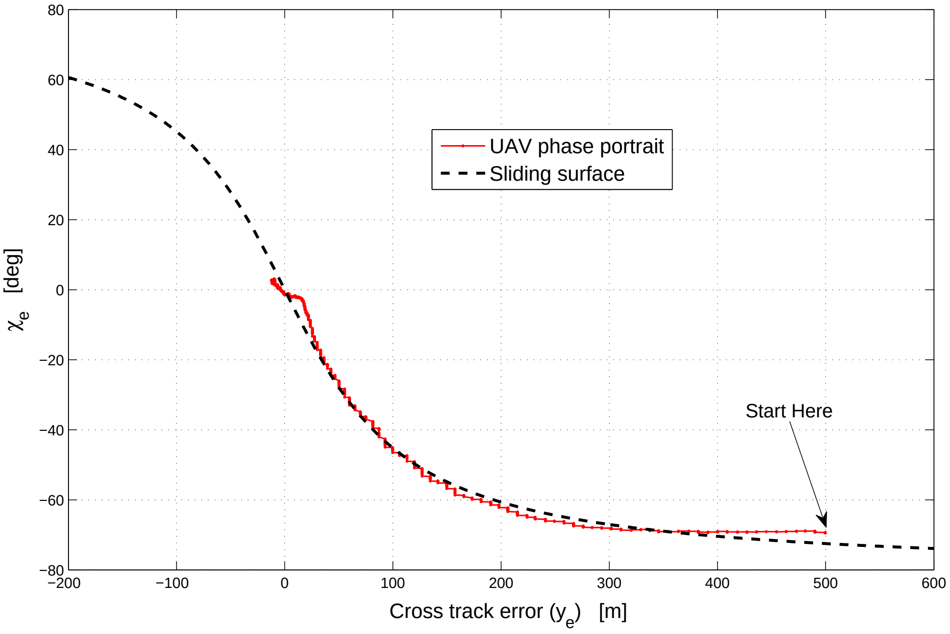


Figure 6.17: Phase portrait for the large track error case(χ_e versus y_e).

with the actual roll angle and aileron deflection are shown. As expected, the maximum commanded roll angle is $\sim 35^\circ$ at 1214 sec, when the track error is approximately 55 m. The control surface (aileron) deflection is well behaved and generally less than 3°.

6.4.1.3 Loiter pattern

Flight results for a loiter pattern are shown in Figures 6.19–6.20. Figure 6.19 shows the desired mission and the actual path flown by the vehicle. The take-off point is taken as the origin, northward distance travelled is denoted by 'x' (or Pos_x), while eastward distance travelled is denoted by 'y' (or Pos_y). For smooth transition at the corners, distance to the next leg is continuously monitored, and when it reduces to 300 m, the 'leg shift' command is issued. The cross-track error is thereafter computed with reference to the next leg, and so a smooth curved path is traced out inside of the corner point. Good performance of the proposed algorithm is observed in this case also. The commanded roll angle, actual roll

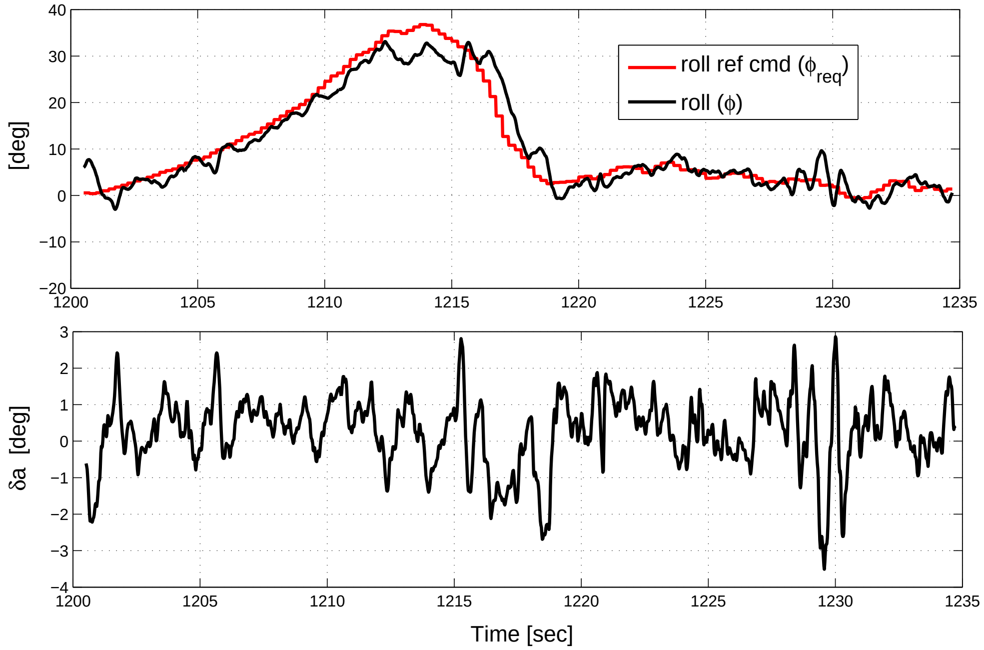


FIGURE 6.18: Roll angle and aileron deflection for the large cross-track error scenario.

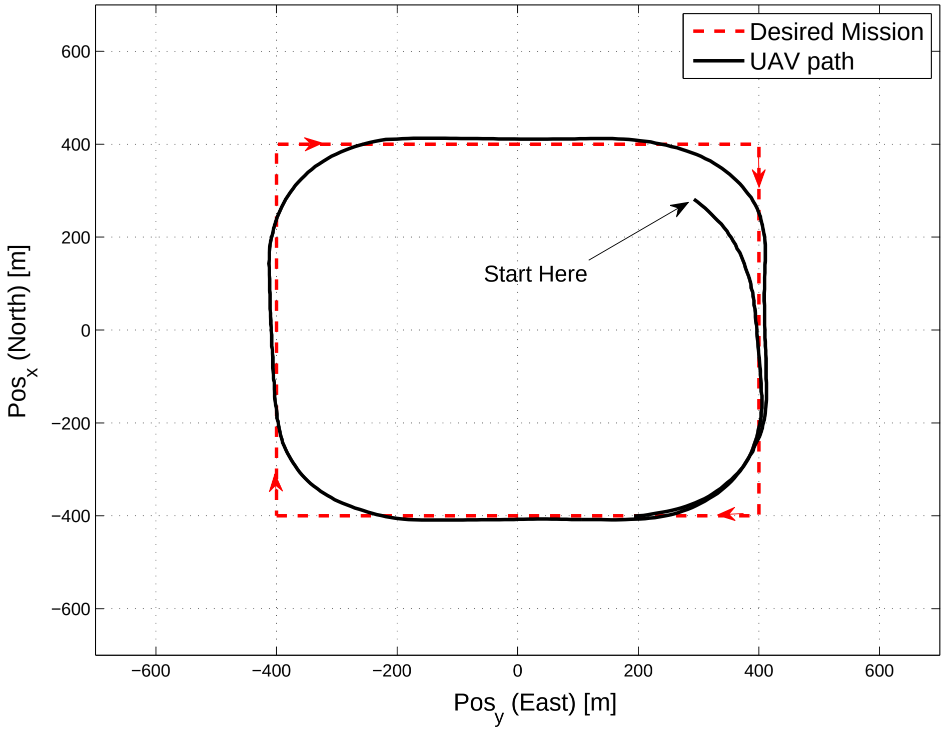


FIGURE 6.19: Flight test results for the loiter pattern (desired and actual flight paths).

angle and aileron deflection are shown in Figure 6.20. The maximum commanded (reference) roll angle generated by the algorithm is approximately 35°.

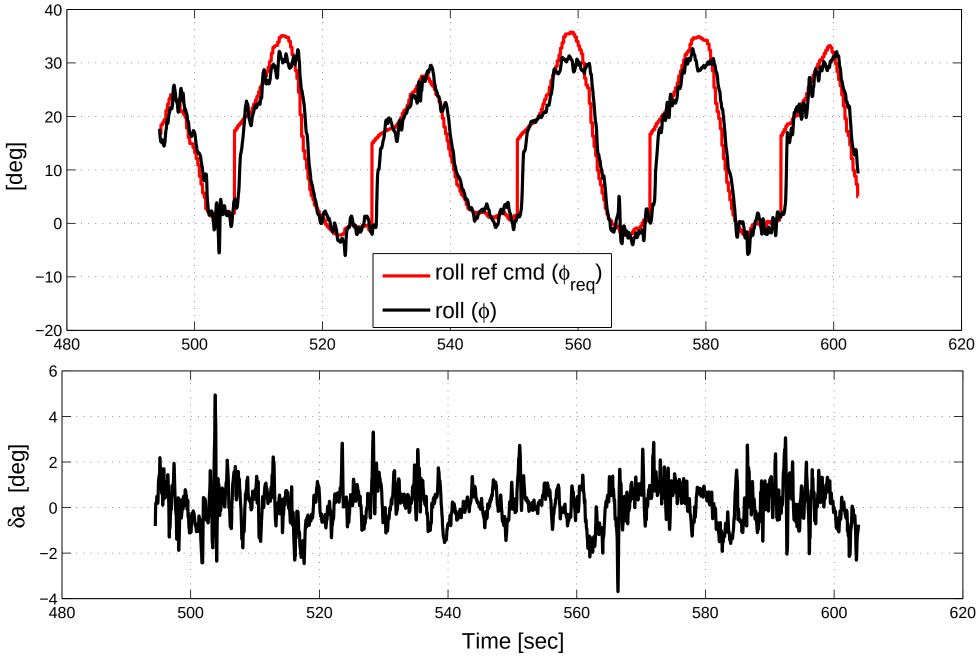


FIGURE 6.20: Roll angle and aileron deflection for the loiter pattern.

6.4.1.4 Sharp heading change

Flight results of the proposed algorithm for a straight path followed by a sharp heading change of approximately 135 degrees are shown in Figures 6.21–6.23. Figure 6.21 shows the desired mission plan along with the actual trajectory flown by the UAV. The mission consists of three parts: a straight path $WP1$-$WP2$, a sharp turn (heading change of $\sim$ 135 deg), followed by another straight segment $WP2$-$WP3$. To see robustness of the proposed algorithm, the speed of the UAV is varied during the mission and ascend/descend commands are also given, as shown in Figure 6.22. For flight safety at such a low altitude, the roll reference command is saturated at 30 deg for this maneuver. Figure 6.23 shows the reference roll angle, the actual roll angle, and the aileron deflections. With the proposed guidance algorithm, the aircraft follows the entire mission successfully. In the last

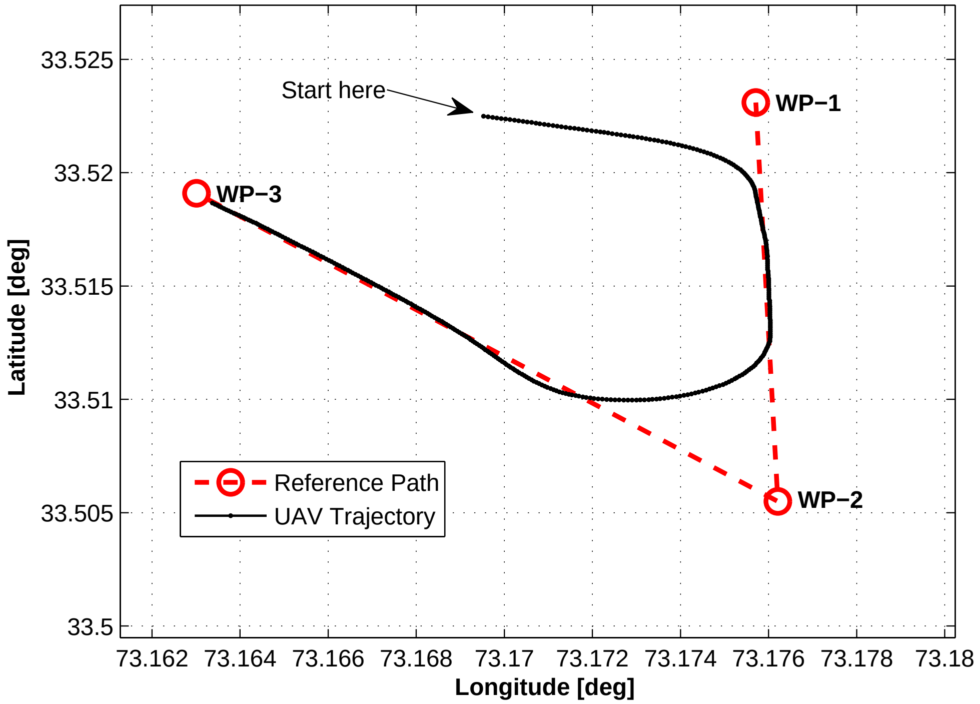


FIGURE 6.21: Trajectory following for a straight path followed by a sharp turn.

segment of the mission (straight path $WP2$-$WP3$), the following remains very good despite ascend/descend maneuvers and variations in speed.

6.4.2 Circular path cases

Flight results for two different circular loiters are discussed here. The takeoff point is taken as the origin, northward distance travelled is denoted by x (or Pos_x), while eastward distance is denoted by y (or Pos_y). The UAV has an open loop speed control which is adjustable from ground. Initially a circular loiter is performed with a fixed throttle setting; ground speed V_g versus course angle χ is plotted in Figure 6.24. A nearly constant airspeed is maintained with the fixed throttle, variation in ground speed is seen due to wind. From Figure 6.24 it is estimated that there was a wind of $\sim$ 5 m/s during the flight; all subsequent flight results show the performance of the guidance algorithm in the presence of wind of approximately 5 m/s.

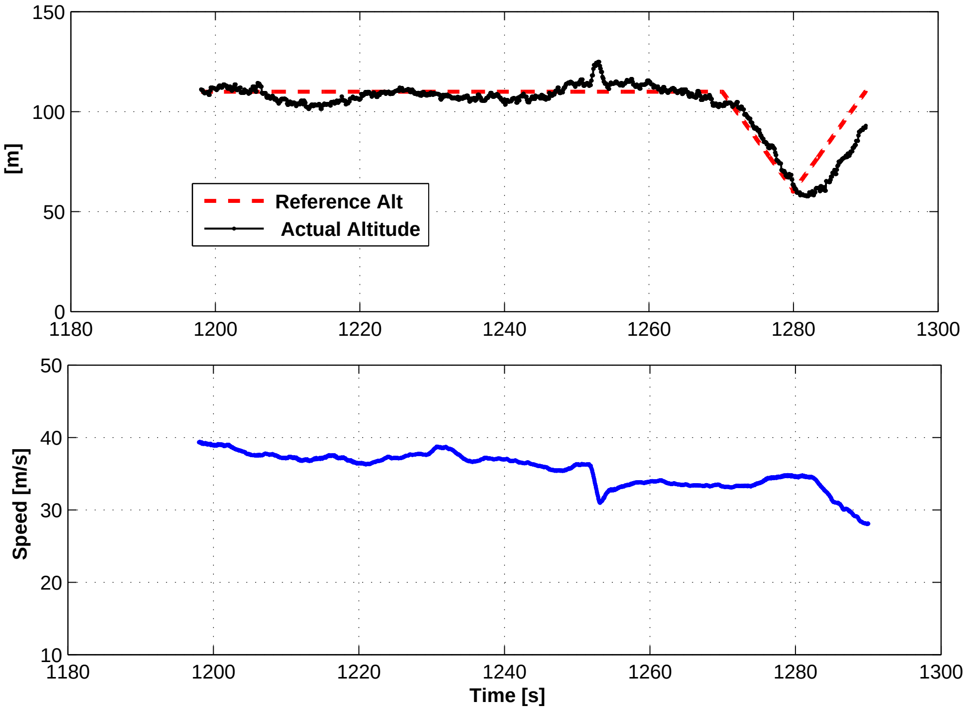


FIGURE 6.22: Altitude and speed for straight path and sharp turn.

Flight results with different lateral guidance schemes are presented in Figures 6.25–6.27 for a loiter mission of 800 m radius (clockwise direction). In Figure 6.25, the desired mission and the actual trajectory flown by the UAV are shown with dashed and solid lines, respectively. Initially the proposed algorithm without the feed-forward term (6.34) is activated, path following is not accurate in this case. After some time (transition point marked in the figure), the guidance logic is switched to the proposed algorithm (6.35) and thereafter substantial improvement in tracking accuracy can be seen in Figure 6.25. The last segment of this flight is executed with Park's algorithm (Park et al., 2004). The corresponding lateral cross track error (y_e) and the intercept course (χ_e) versus time are shown in Figure 6.26. In the initial part of the flight (1340–1412.5 seconds), the cross-track error is $\sim$ 55 m with the guidance law (6.34). Later (1412.5–1494 seconds), this error improves to $\sim$ 2 m when the proposed guidance law (6.35) is brought online. In the last segment (1494–1527 seconds), the cross-track error remains within a 3 m

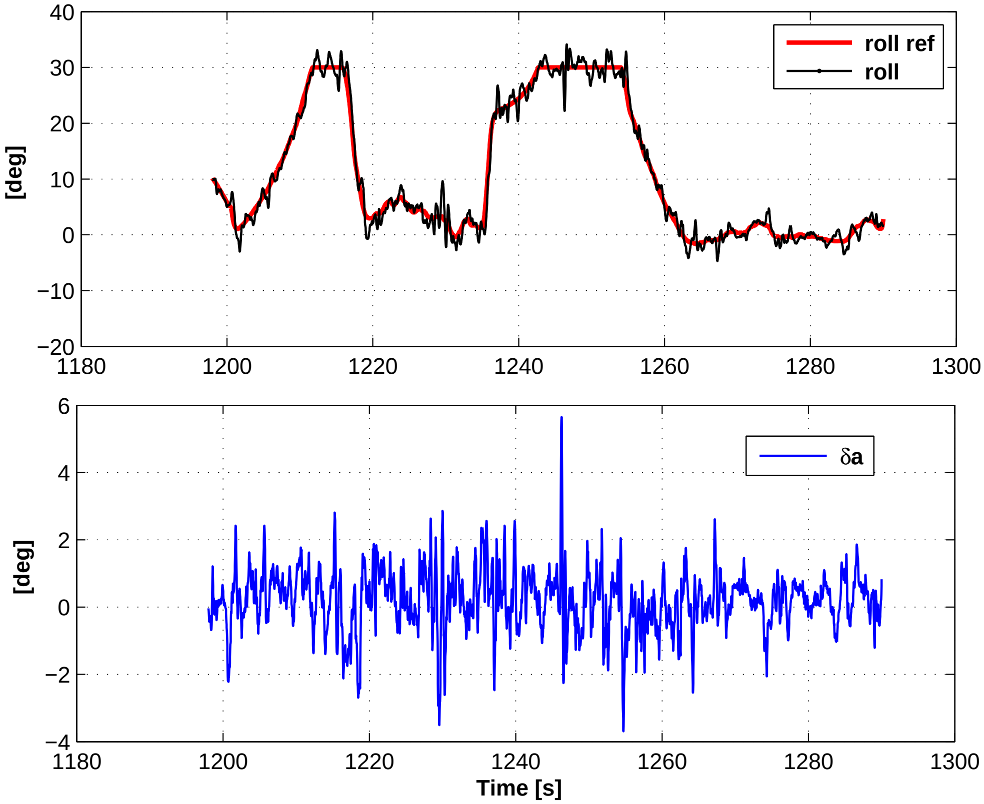


Figure 6.23: Roll angle and aileron deflection for straight path and sharp turn.

band with Park's algorithm. The commanded roll angle generated by these three guidance schemes along with the actual roll angle and aileron deflection are shown in Figure 6.27. There is no control saturation and the aileron deflection is normal.

Figures 6.28–6.30 describe flight results for a circular mission of radius 400 m (close to the UAV's maximum capability). Initially (337–413 seconds), the desired mission is executed with our algorithm without the feed-forward term (6.34), followed by Park's algorithm from 413–461.5 seconds. Finally the algorithm proposed in expression (6.35) is activated to follow the desired mission from 461.5–495 seconds. The tracking performance of the algorithm (6.34) further degrades in this case and a cross-track error of $\sim$ 80 m is seen in Figure 6.29. This accuracy is improved to $\sim$ 20 m by Park's algorithm. With the proposed lateral guidance

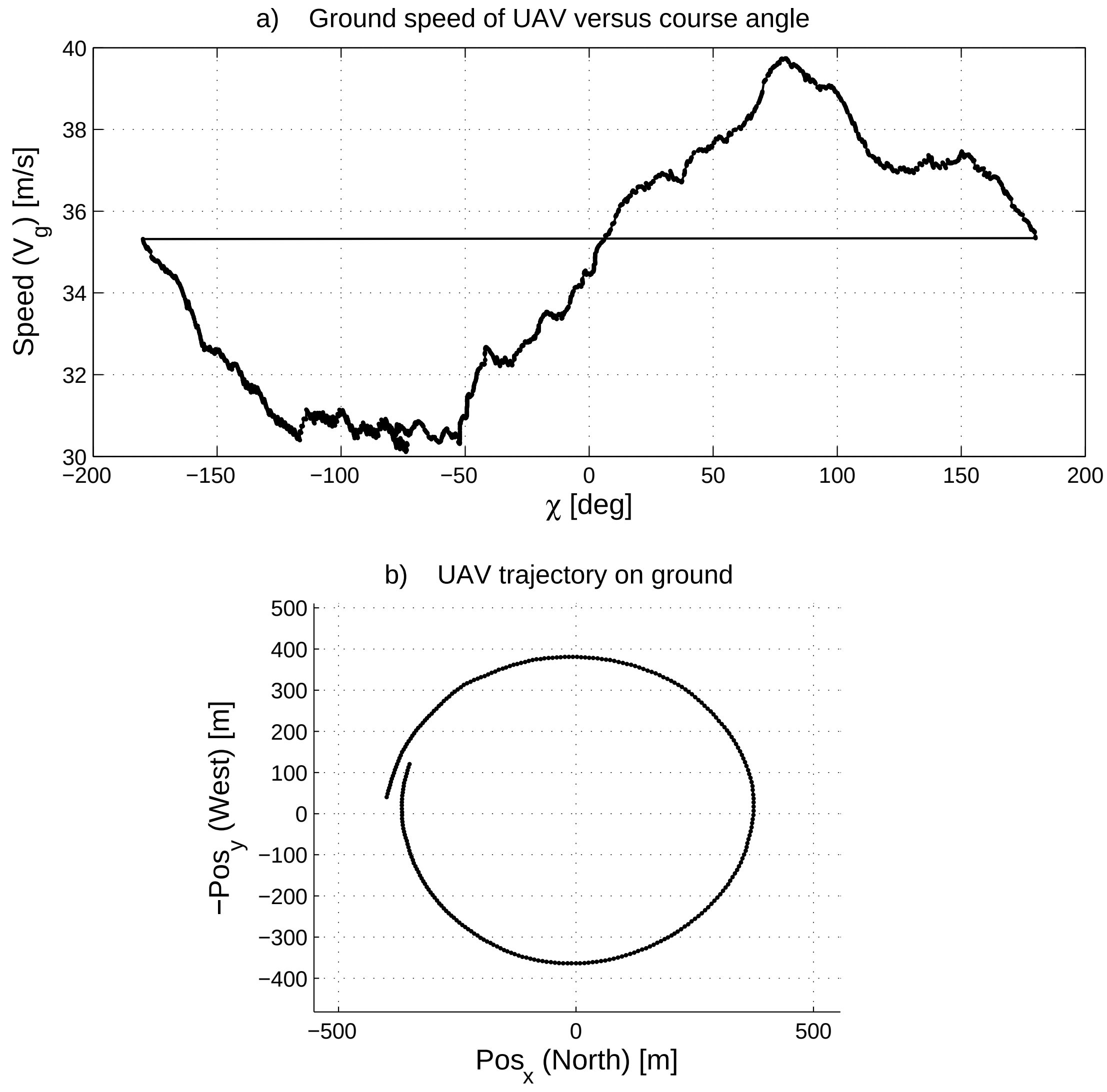


FIGURE 6.24: Estimated wind on flight day.

scheme (6.35), the path following accuracy improves to ~ 5 m. Figure 6.30 shows the commanded and actual roll angles along with aileron deflections.

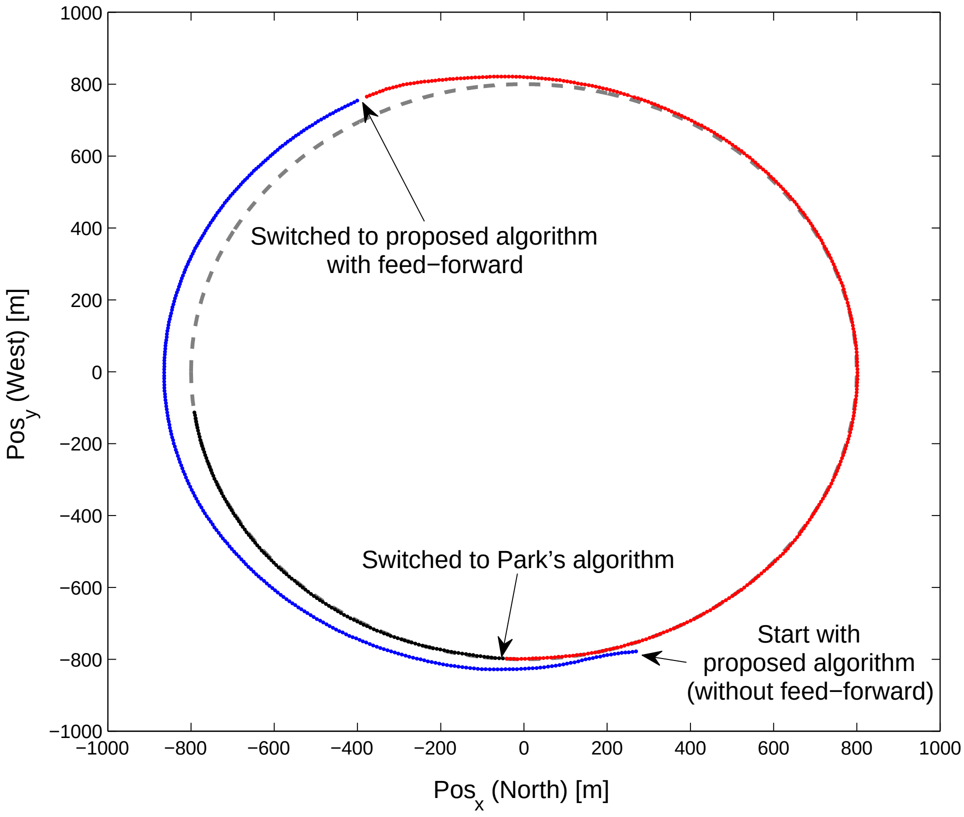


FIGURE 6.25: Trajectory following for a circular loiter of radius 800 m.

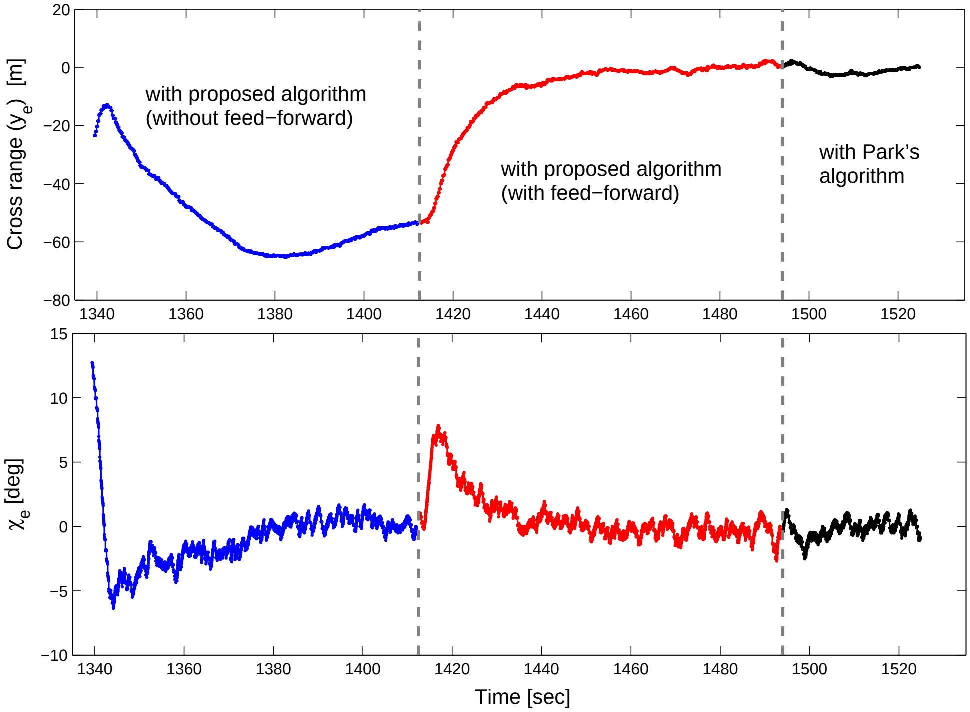


FIGURE 6.26: Cross-track error y_e and heading error χ_e vs time for the circular loiter of radius 800 m.

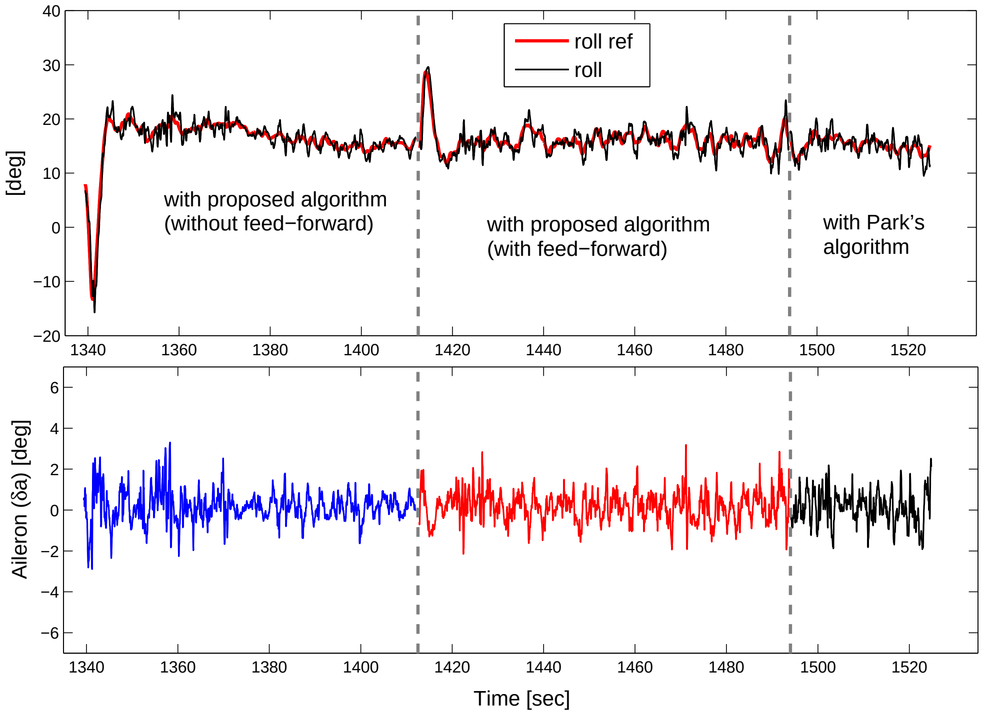


FIGURE 6.27: Roll angle and aileron deflection (δ_a) vs time for the circular loiter of radius 800 m.

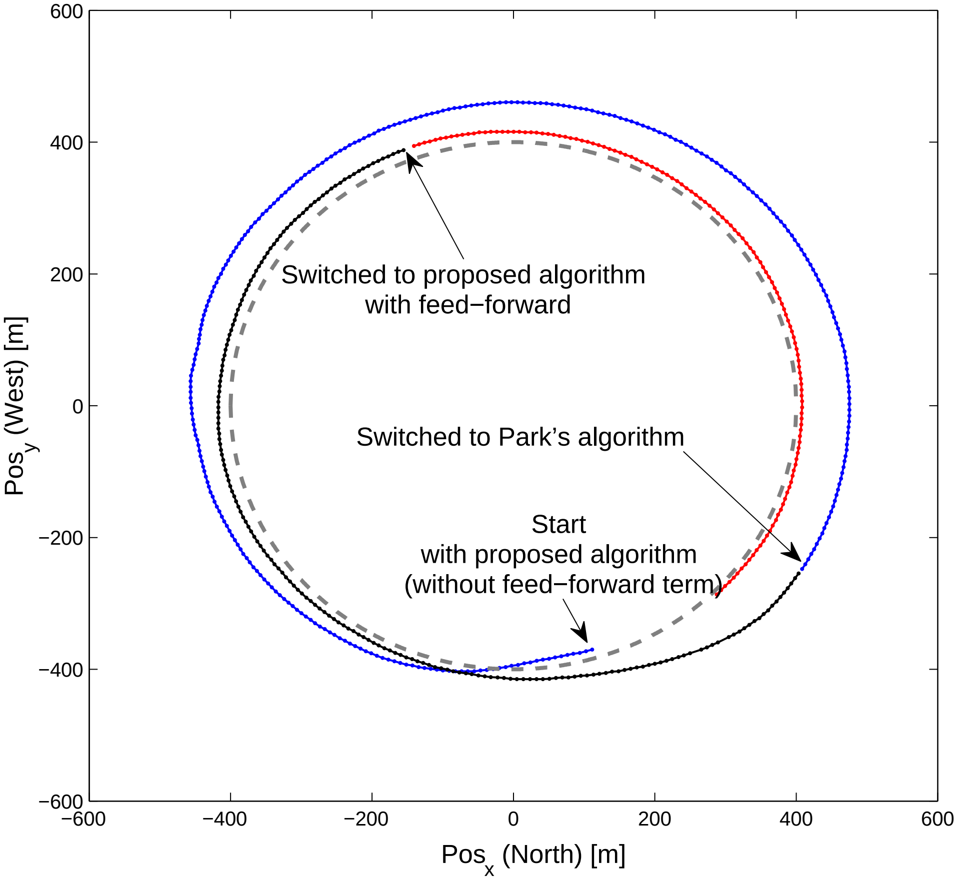


Figure 6.28: Trajectory following for a circular loiter of radius 400 m.

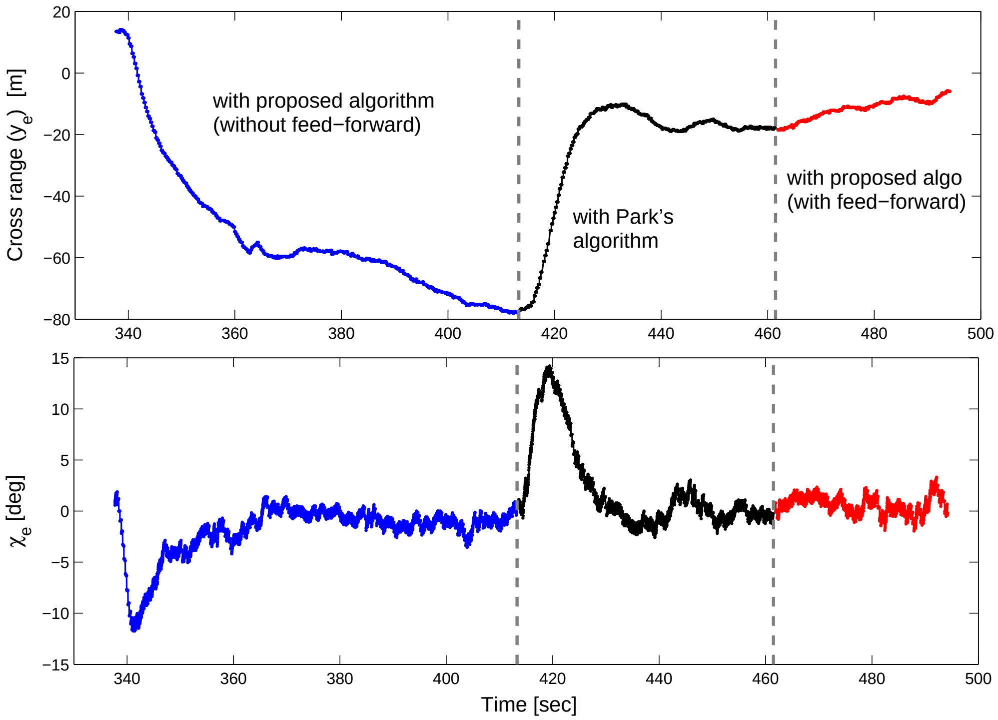


FIGURE 6.29: Cross-track error y_e and heading error χ_e vs time for the circular loiter of radius 400 m.

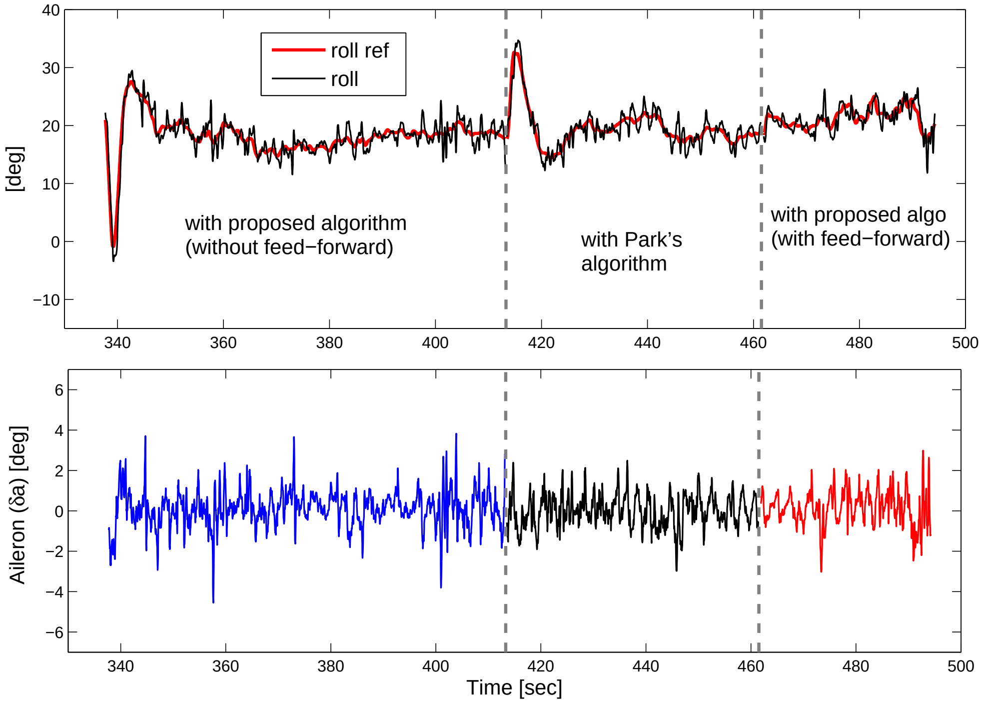


FIGURE 6.30: Roll angle and aileron deflection (δ_a) vs time for the circular loiter of radius 400 m.

6.5 Comparison of Experimental Results with Simulation

A natural progression from design to flight testing for any algorithm development includes an intermediate step of computer simulation. A number of simulations are carried out before flight testing to see robustness and performance of the proposed algorithm in the presence of disturbances (like wind, gusts etc), delays and noise in sensor data, and parametric uncertainties in aerodynamic and structural coefficients of the UAV. To simulate different scenarios, a detailed 6-dof nonlinear simulation is developed including UAV dynamics, sensors modeling, actuators dynamics and the flight control computer. Details of the non-linear simulation are given in Annex-B.

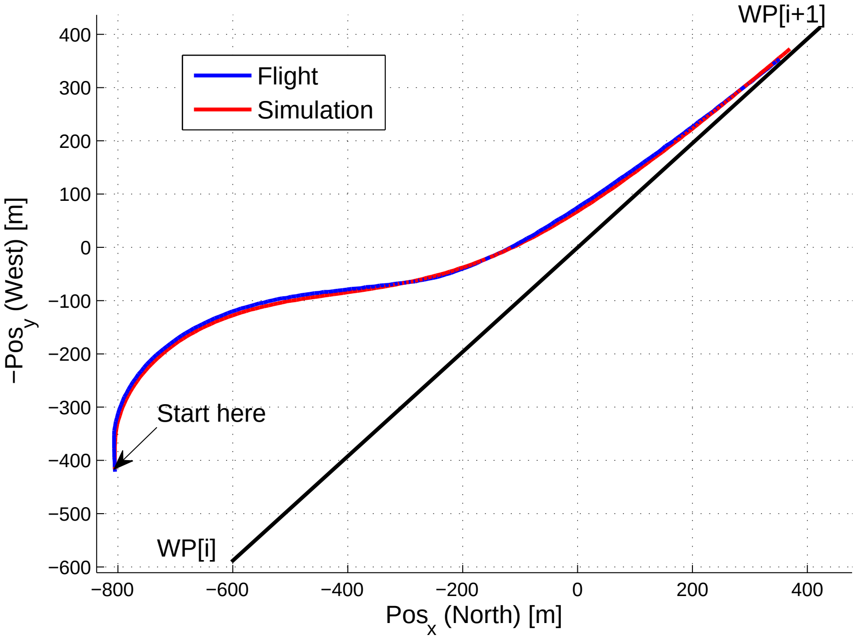


FIGURE 6.31: Comparison of Flight & Simulation results: UAV position on ground for straight path case.

After a flight test, validation/calibration of the simulation is necessary for further algorithm testing. In the presence of a good simulation which matches the flight results, the development and validation time for algorithm design reduces significantly.

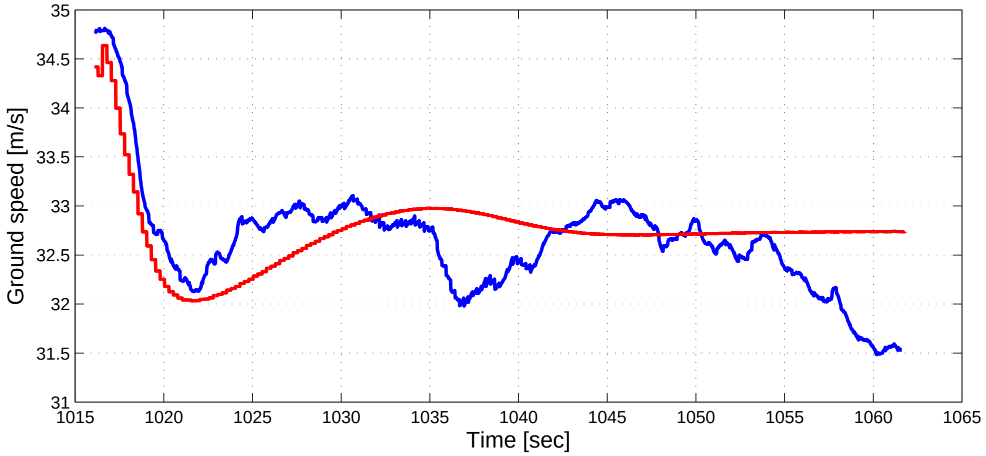


FIGURE 6.32: Comparison of Flight & Simulation results: Ground speed for straight path case.

The purpose of this section is to compare flight test results with simulated results to develop confidence on the simulation environment. As discussed earlier, two flight tests are conducted with the proposed algorithm and different scenarios are generated in flight to see effectiveness of the proposed algorithm. It is difficult to cover every scenario in a single chapter and hence some important results are discussed here. For comparison with simulation, two flight scenarios (straight and circular path) are selected and the comparison is shown. It is worthwhile to mention here that we change an aerodynamic coefficient Cy_0 in simulation from zero to -0.013 to match the results. Generally, propeller driven tractor configurations have a down-wash effect on the vertical tail due to flow from the propeller that generates a side force in flight. This down-wash effect in terms of Cy_0 is generally modeled in the simulation and then vertical rudder is engaged by a few degrees to counter this effect thus resulting in zero Cy_0. In our case, we have not modeled this down wash effect in our simulation and as result some side force is observed in our flights. As an approximation of this down wash effect, the flight results match with the simulation when $Cy_0 = -0.013$ is used in simulation.

In Figures 6.31 - 6.34, comparison is shown for a straight path following case.

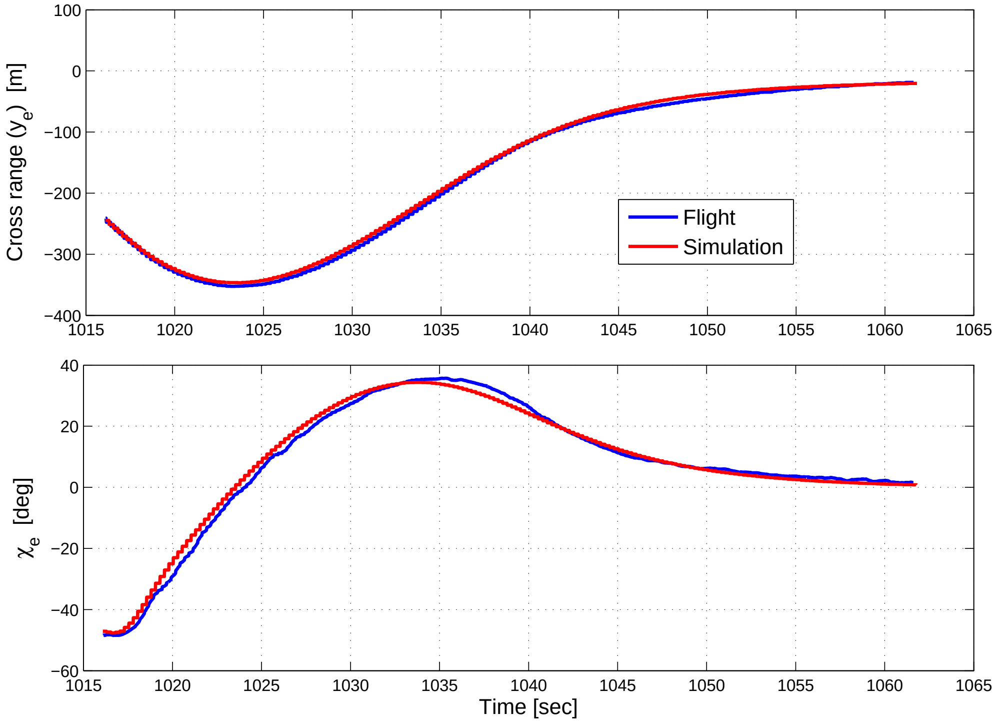


FIGURE 6.33: Comparison of Flight & Simulation results: cross track & course angle errors vs time for straight path case.

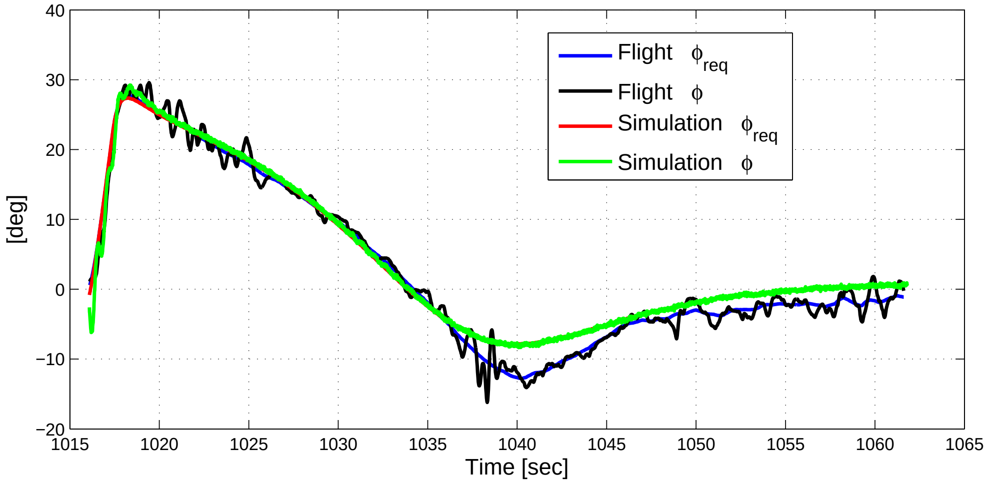


FIGURE 6.34: Comparison of Flight & Simulation results: Reference & actual roll angle vs time for straight path case.

These flight results are not covered in the above section. In Figure 6.31, the desired ground path along with the UAV trajectory in flight and simulation are shown. The UAV is initially going away from the desired path when the guidance algorithm is activated. The flight and simulation ground trajectories match well in this case. The UAV ground speed is in the range of 31−35 m/s and the comparison is shown in Figure 6.32. The ground trajectory comparison is further elaborated in Figure 6.33; the state variables, i.e., y_e and χ_e from both flight and simulation are plotted versus time. The difference of simulation and flight is negligible. The guidance logic generated the signal (ϕ_{req}), the actual roll angle (ϕ) comparison is shown in Figure 6.34. The matching in these variables is good, however more noise/disturbance in the flight roll angle is evident. There is approximately a 5° difference in the flight and simulation guidance loop control variable ϕ_{req}.

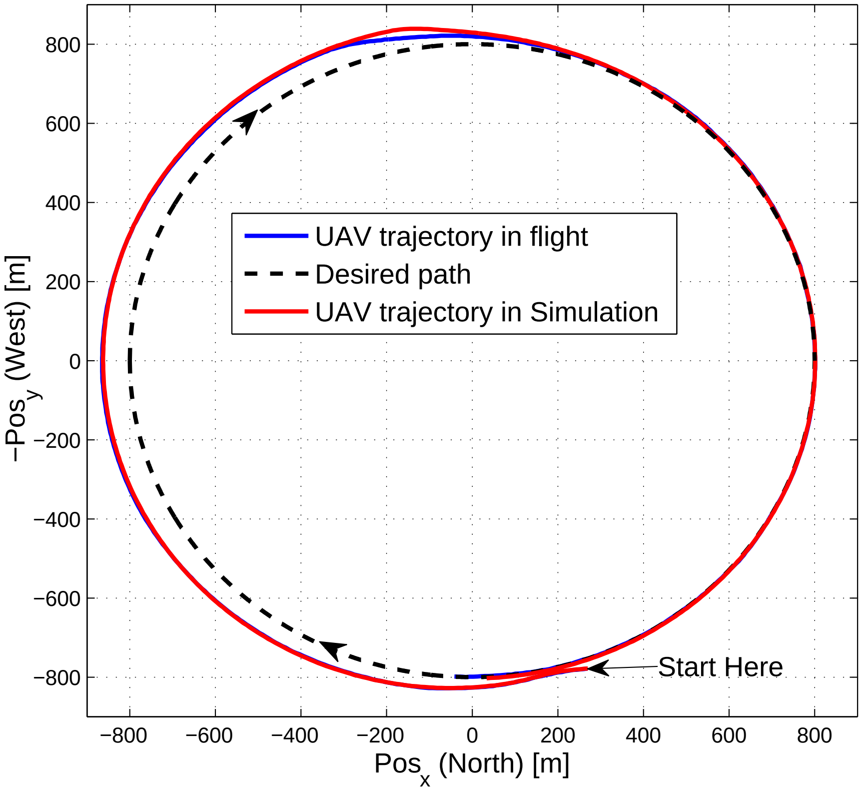


FIGURE 6.35: Comparison of Flight & Simulation results: UAV position on ground for circular path case.

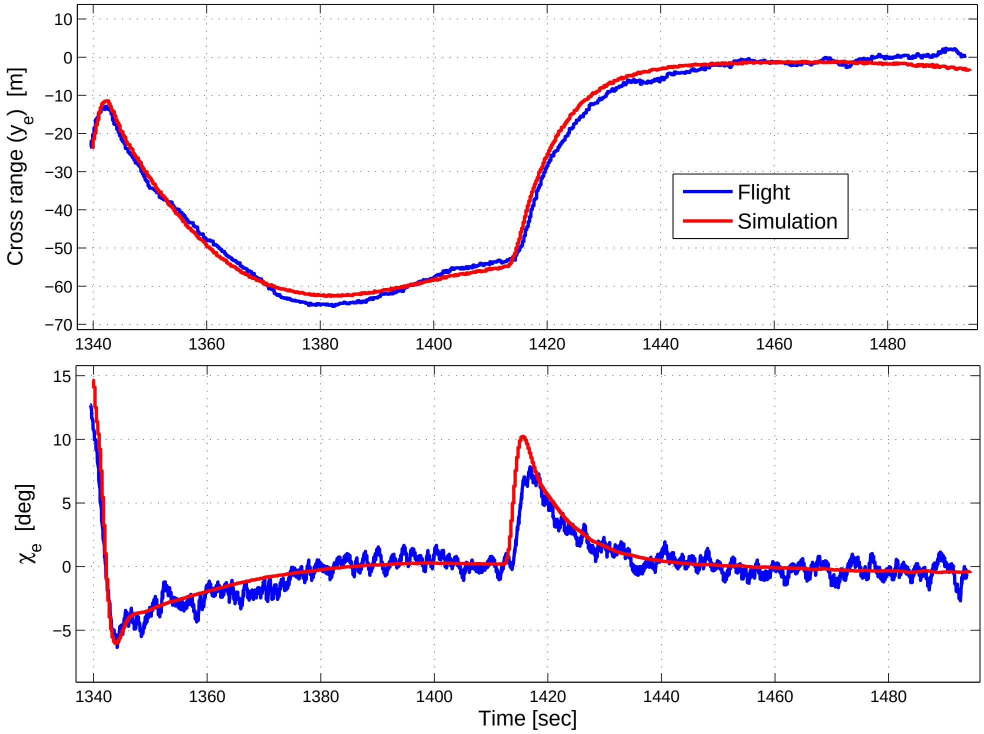


FIGURE 6.36: Comparison of Flight & Simulation results: cross track & course angle errors vs time for circular path case.

Flight results for the circular path following case are compared with simulation results in Figures 6.35 - 6.38. The UAV trajectory comparison is shown in Figure 6.35 along with the desired path. As discussed earlier in section 6.4.2, initially the results are without the feed-forward term and later the feed-forward term is activated at 1412.5 sec. Figure 6.36 shows the state variables (y_e and χ_e) comparison versus time for this case. The simulation results are in good agreement with the flight results in this case also, the maximum difference in y_e and χ_e is $5m$ and $3°$, respectively. Another important parameter is the sliding surface in SMC based guidance scheme and its comparison versus time is shown in Figure 6.37. Good matching of this variable is seen and no significant departure in its magnitude and trend is observed. Comparison of reference and actual roll angle is shown in Figure 6.38, these variables are also in good agreement, except for a small shift in time due to difference in ground velocity and noise/disturbance in flight roll angle.

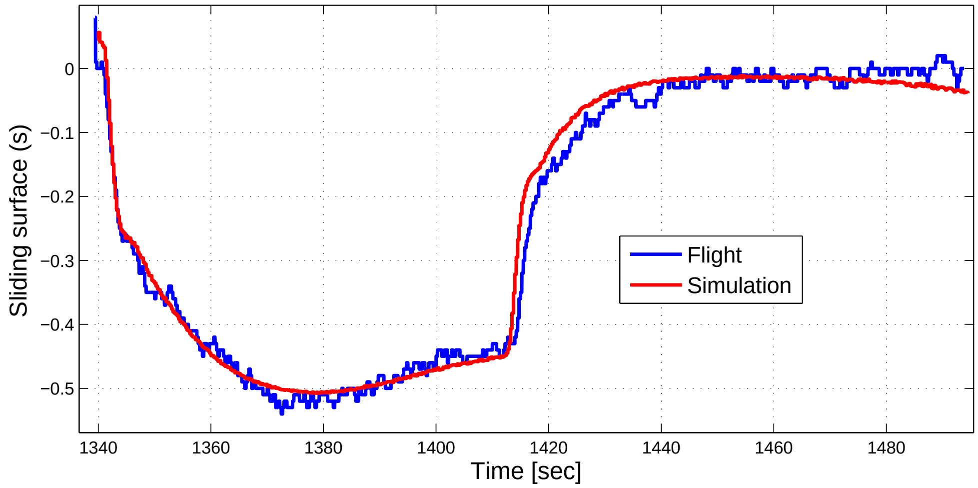


FIGURE 6.37: Comparison of Flight & Simulation results: Sliding surface (s_1) vs time for circular path case.

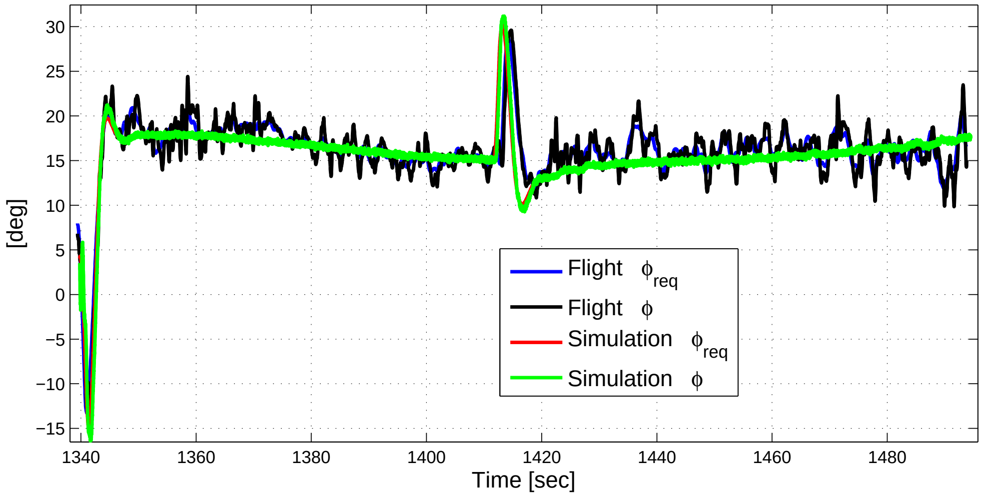


FIGURE 6.38: Comparison of Flight & Simulation results: Reference & actual roll angle vs time for circular path case.

Conclusively, simulation results are in good agreement with flight results, the simulation platform is suitable for demonstration of new ideas related to guidance and control of UAVs. One extension of the work presented in this chapter is its generalization for 3-D path following, and the simulation environment is good enough to show its effectiveness.

6.6 Summary of the Chapter

Sliding mode lateral guidance logic for UAVs is derived in this chapter for both cases of straight and circular path following. High performance non-linear sliding surface (proposed in the last chapter) is used in the SMC based guidance logic design. To see effectiveness of the proposed scheme, the logic is programmed in the flight control computer of a scaled YAK-54 research UAV and different scenarios are generated during various flights. Flight test results prove robustness and performance of the proposed scheme. Flight results are also compared with computer simulation results, and both are in good agreement. Natural extension of this SMC based lateral guidance scheme is its generalization to 3-D trajectory following. In the next chapter, sliding mode based 3-D guidance logic is derived for UAVs.

Chapter 7

3-D GUIDANCE OF UAVS

7.1 Introduction

As discussed in detail in Chapter 3, the desired path is generally expressed in the form of lines and arcs in 3-D space and then this path following problem is solved by two independent lateral and longitudinal guidance logics. The lateral and longitudinal dynamics are decoupled during guidance design, and the coupling between these two planes is usually ignored for simplicity. In the last chapter, a decoupled lateral guidance scheme based on sliding mode theory is proposed. Generalizing our previous work, here we propose a sliding mode based guidance logic for 3-D path following in a single framework, considering both the coupling effects and the parametric/input disturbances. In this case, the vehicle dynamics become more complex and the degrees of freedom that are not directly actuated increase, thus making the control design more involved.

In case of 3-D path following, the problem is formulated in Section 3.5.2 with two control variables, i.e., required longitudinal and lateral forces to keep the vehicle on the desired 3-D path. As the proposed 3-D guidance scheme presented here is based on the sliding mode theory, so naturally two sliding surfaces are required in this case, these have been proposed in Chapter 5. Based on the proposed nonlinear sliding manifolds, a nonlinear guidance scheme is derived for the MIMO system under consideration here. In order to reduce chattering in the control signal, the bang-bang control terms are accompanied by proportional terms in the reaching phase. The proposed guidance is implemented on a 6-dof nonlinear simulation and different flight scenarios are simulated. Simulation results in the absence and presence of disturbances are presented here to show robustness of the proposed guidance scheme.

7.2 3-D Guidance Logic Design

For 3-D guidance logic design, we have derived the generalized kinematics equations in section 3.4, considering coupling between the lateral and longitudinal planes. The equations are:

$$\left.\begin{aligned}
\dot{z}_e &= -V_g \sin\gamma_e \\
\dot{y}_e &= V_g \cos\gamma \sin\chi_e \\
\dot{\gamma}_e &= \frac{g}{V_g}\left(\frac{L_{req}\cos\phi_{req}}{mg} - \cos\gamma\right) - \dot{\gamma}_p \\
\dot{\chi}_e &= \frac{L_{req}\sin\phi_{req}}{mV_g\cos\gamma} - \dot{\chi}_p,
\end{aligned}\right\} \tag{7.1}$$

where z_e and y_e are the position errors in the vertical (normal to the reference path) and lateral planes, respectively. The other two state variables γ_e and χ_e are the errors in flight path angle and course angle, respectively. The two control variables are $F_{req_{vert}} = L_{req}\cos\phi_{req}$ (required force in the vertical plane) and $F_{req_{lat}} = L_{req}\sin\phi_{req}$ (required lateral force), that the guidance loop has to generate for 3-D mission following. Based on these control variables, we can compute L_{req} and ϕ_{req} as reference signals for the inner control loops.

Sliding mode theory based control law design is a two step process, i.e., first the selection of sliding manifolds (number of manifolds equal to number of control inputs), followed by the derivation of a control law for reaching and maintaining the sliding motion. In the 3-D path-following case (a MIMO system), we require two sliding manifolds. We have proposed two independent sliding manifolds for longitudinal and lateral planes in Chapter 5, the sliding surfaces are:

$$\left.\begin{aligned} s_1 = &\quad \chi_e + c_1 \arctan(c_2 y_e) = 0 \\ s_2 = &\quad \gamma_e - c_3 \arctan(c_4 z_e) = 0, \end{aligned}\right\} \tag{7.2}$$

where the constants c_1, c_2, c_3 and c_4 are real positive numbers and $c_1, c_3 \leq 1$. Stability of these sliding manifolds is already proved in Chapter 5. Based on these sliding manifolds, we can now proceed for derivation of the 3-D guidance law.

7.2.1 3-D Guidance logic

After the selection of stable sliding manifolds, the next step is the derivation of control law to reach and maintain motion on these manifolds. Generally the control law is derived using the Lyapunov function $\mathbf{W} = \frac{1}{2}s^2$. A sliding phase control is derived from $\dot{s} = 0$ and then '$-k\,\mathrm{sgn}(s)$' is added to cater for uncertainties and unmodeled dynamics. In Chapter 2, different reaching laws to ensure sliding motion are discussed in detail. To reduce chattering in the control signal, here we derive control law using the 'constant plus proportional rate reaching law', i.e., using $\dot{s} = -k_\triangle\,\mathrm{sgn}(s) - ks$. This yields:

$$\left.\begin{aligned} \dot{\chi}_e + \frac{c_1 c_2}{1 + c_2^2 y_e^2}\dot{y} = -k_{\triangle_1}\,\mathrm{sgn}(s_1) - k_1 s_1 \\ \dot{\gamma}_e - \frac{c_3 c_4}{1 + c_4^2 z_e^2}\dot{z}_e = -k_{\triangle_2}\,\mathrm{sgn}(s_2) - k_2 s_2, \end{aligned}\right\} \tag{7.3}$$

or

$$\left.\begin{aligned} \frac{L_{req}\sin\phi_{req}}{m\hat{V}\cos\gamma} - \dot{\chi}_p + \frac{c_1 c_2}{1 + c_2^2 y_e^2}\hat{V}\cos\gamma\sin\chi_e = -k_{\triangle_1}\,\mathrm{sgn}(s_1) - k_1 s_1 \\ \frac{L_{req}\cos\phi_{req}}{m\hat{V}} - \frac{g\cos\gamma}{\hat{V}} + \frac{c_3 c_4}{1 + c_4^2 z_e^2}\hat{V}\sin\gamma_e - \dot{\gamma}_p = -k_{\triangle_2}\,\mathrm{sgn}(s_2) - k_2 s_2, \end{aligned}\right\} \tag{7.4}$$

where $\hat{V}$ is the measured/estimated value of ground velocity V; it is assumed here that all other parameters have negligible uncertainty (so they are accurately measurable or known beforehand). After rearranging the terms, we have the following expression for control variables $F_{req_{lat}} = L_{req} \sin \phi_{req}$ and $F_{req_{vert}} = L_{req} \cos \phi_{req}$:

$$\left.\begin{aligned} L_{req} \sin \phi_{req} &= m\hat{V} \cos \gamma \Big(\frac{-c_1 c_2}{1 + c_2^2 y_e^2} \hat{V} \cos \gamma \sin \chi_e + \dot{\chi}_p - k_{\triangle_1} \operatorname{sgn}(s_1) - k_1 s_1 \Big) \\ L_{req} \cos \phi_{req} &= m\hat{V} \Big(\frac{g \cos \gamma}{\hat{V}} - \frac{c_3 c_4}{1 + c_4^2 z_e^2} \hat{V} \sin \gamma_e + \dot{\gamma}_p - k_{\triangle_2} \operatorname{sgn}(s_2) - k_2 s_2 \Big). \end{aligned}\right\} \tag{7.5}$$

These give the required control force in the lateral and longitudinal inertial planes to keep the vehicle on track in 3-D space. These can easily be converted to L_{req} (required lift force) and ϕ_{req} (required roll angle) to follow the desired mission. Also, L_{req} can be written in the form of α_{req} or θ_{req} as per the structure of the inner control loop.

7.2.2 Reachability condition

Using the global positive definite Lyapunov function $\mathbf{W} = \frac{1}{2} s^2$, condition on control variables $(k_{\Delta_1}, k_1, k_{\Delta_2}, k_2)$ can be derived to ensure reachability despite parametric uncertainties and disturbances. These feedback gains can be chosen so that $\dot{\mathbf{W}} = s\dot{s} < 0$ in the domain of attraction (Edwards and Spurgeon, 1998). As we have decoupled sliding surfaces here for our MIMO system, so the reachability conditions are:

$$s_1 \dot{s_1} < 0 \quad \text{and} \quad s_2 \dot{s_2} < 0. \tag{7.6}$$

These inequalities imply

$$\left.\begin{aligned} & s_1 \left[\frac{L_{req} \sin \phi_{req}}{mV \cos \gamma} - \dot{\chi}_p + \frac{c_1 c_2}{1 + c_2^2 y_e^2} V \cos \gamma \sin \chi_e \right] < 0 \\ & s_2 \left[\frac{L_{req} \cos \phi_{req}}{mV} - \frac{g \cos \gamma}{V} + \frac{c_3 c_4}{1 + c_4^2 z_e^2} V \sin \gamma_e - \dot{\gamma}_p \right] < 0. \end{aligned}\right\} \tag{7.7}$$

Substituting values of control variables $L_{req}\sin\phi_{req}$ and $L_{req}\cos\phi_{req}$ from (7.5), we have:

$$\left.\begin{aligned} &s_1\left[\frac{\hat{V}}{V}\left(\frac{-c_1c_2}{1+c_2^2y_e^2}\hat{V}\cos\gamma\sin\chi_e+\dot{\chi}_p\right)-\dot{\chi}_p+\frac{c_1c_2}{1+c_2^2y_e^2}V\cos\gamma\sin\chi_e\right.\\ &\qquad\qquad\left.-\frac{\hat{V}}{V}k_{\triangle_1}\operatorname{sgn}(s_1)-\frac{\hat{V}}{V}k_1s_1\right]<0\\ &s_2\left[\frac{\hat{V}}{V}\left(\frac{-c_3c_4}{1+c_4^2z_e^2}\hat{V}\sin\gamma_e+\dot{\gamma}_p\right)-\dot{\gamma}_p+\frac{c_3c_4}{1+c_4^2z_e^2}V\sin\gamma_e\right.\\ &\qquad\qquad\left.-\frac{\hat{V}}{V}k_{\triangle_2}\operatorname{sgn}(s_2)-\frac{\hat{V}}{V}k_2s_2\right]<0. \end{aligned}\right\}\quad(7.8)$$

Assuming a maximum measurement uncertainty in $\hat{V}$ of $\tilde{V}=\hat{V}-V$, i.e. $\hat{V}=V+\tilde{V}$, (7.8) becomes:

$$\left.\begin{aligned} &s_1\left[(2V+\tilde{V})\left(\frac{-c_1c_2}{1+c_2^2y_e^2}\tilde{V}\cos\gamma\sin\chi_e\right)+\tilde{V}\dot{\chi}_p-\hat{V}k_{\triangle_1}\operatorname{sgn}(s_1)-\hat{V}k_1s_1\right]<0\\ &s_2\left[(2V+\tilde{V})\left(\frac{-c_3c_4}{1+c_4^2z_e^2}\tilde{V}\sin\gamma_e\right)+\tilde{V}\dot{\gamma}_p-\hat{V}k_{\triangle_2}\operatorname{sgn}(s_2)-\hat{V}k_2s_2\right]<0. \end{aligned}\right\}\quad(7.9)$$

The control gains $k_{\triangle_1}$ and $k_{\triangle_2}$ are designed to cater for parametric uncertainties and keep the state trajectory on the sliding surface. The other two control gains k_1 and k_2 play their role during the reaching phase, values of these gains can be adjusted to reduce the reaching time. Now the left hand sides of (7.9) are negative definite, if

$$\left.\begin{aligned} &\hat{V}k_{\triangle_1}>(2V+\tilde{V})\frac{c_1c_2}{1+c_2^2y_e^2}\tilde{V}\cos\gamma|\sin\chi_e|+\tilde{V}|\dot{\chi}_p|\\ &\hat{V}k_{\triangle_2}>(2V+\tilde{V})\frac{c_3c_4}{1+c_4^2z_e^2}\tilde{V}|\sin\gamma_e|+\tilde{V}|\dot{\gamma}_p|. \end{aligned}\right\}\quad(7.10)$$

Now two options are possible: the simpler option is to find the maximum value of these control gains for the extreme case and use these maximum control values in the guidance logic. The other option is to keep the control gains adaptive and select their values depending on the states at every step. Here we use the first strategy and in the extreme case, we have:

$$\left.\begin{aligned} k_{\triangle_1} >& 0.707\frac{(2V_{max}+\tilde{V})\tilde{V}}{V_{min}}c_1c_2 + \frac{\tilde{V}}{V_{min}}|\dot{\chi}_p| \\ k_{\triangle_2} >& 0.707\frac{(2V_{max}+\tilde{V})\tilde{V}}{V_{min}}c_3c_4 + \frac{\tilde{V}}{V_{min}}|\dot{\gamma}_p|. \end{aligned}\right\} \tag{7.11}$$

From (7.11) it is clear that the control gains $k_{\triangle_1}$ and $k_{\triangle_2}$ are directly proportional to the uncertainty $\tilde{V}$ in velocity. The other control variables $k_1, k_2 > 0$ can be used for fast convergence towards the sliding surface, and a smoother control signal with less chattering.

7.3 Optimized Sliding Parameters

As discussed in detail in Chapter 5, the selection criterion for sliding surface parameters is derived for longitudinal and lateral sliding manifolds. In case of 3-D path following, we need two sliding surfaces as proposed in (7.2). Now we have to optimize the coefficients of both the lateral sliding surface (s_1) and the longitudinal (vertical) sliding surface (s_2).

7.3.1 Longitudinal surface parameters

For selection of optimized longitudinal surface parameters, a criterion is derived in Section 5.4.2 using work and energy principle. The work done expression mainly depends on the distance traversed to bring the UAV back on the desired path, and hence minimizing the distance will minimize the work done. From (5.32), we have the following expression for the path length (distance traversed to come back to the reference path) calculation:

$$\left.\begin{array}{l} \text{Path length} = \begin{cases} \displaystyle\int\limits_{z=z_0}^{-1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for -ve } z_0 \\ \displaystyle\int\limits_{z=z_0}^{1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for +ve } z_0 \end{cases} \\ \text{Subject to} \quad F_{longitudinal} \leq F_{longitudinal_{max}} \text{ at each point of integration} \end{array}\right\} \quad (7.12)$$

As compared to the lateral sliding coefficient optimization, here gravity plays an important role in selection of the sliding parameters. Structural loading ($\frac{L}{mg}$) imposes restrictions on the maximum lift force. Another restriction arises from maximum and minimum flight path angles (or rate of climb/decent) to maintain the desired speed.

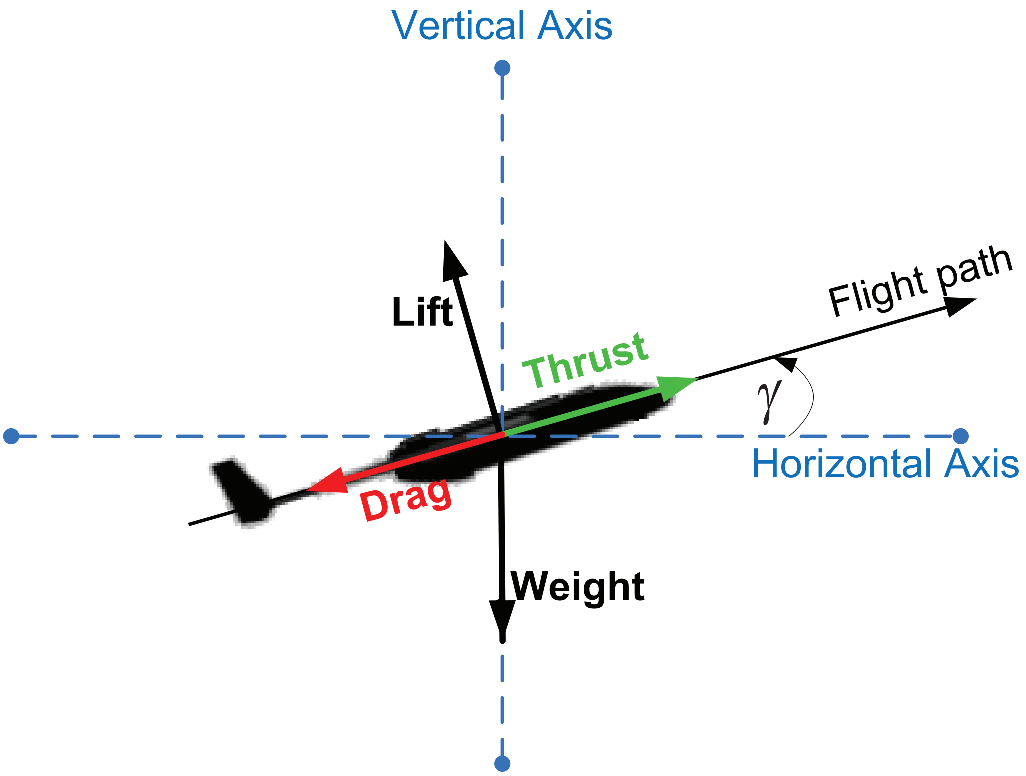


FIGURE 7.1: Longitudinal forces acting on a UAV during non-accelerating climb/decent.

Before proceeding to the derivation, understanding of some basic flight mechanics terms is necessary. In steady (non-accelerating) level cruising flight, the drag is balanced by the engine thrust and the UAV weight is balanced by the lift force. However, the force balance equations change during the climb/decent phases as shown in Figure 7.1. In this case, the forces are approximately balanced as follows:

$$F_{thrust} - Drag - mg\sin\gamma = 0, \qquad Lift - mg\cos\gamma = 0. \tag{7.13}$$

From this it can be concluded that the lift force is approximately equal to the weight for a small flight path angle γ. Now looking at the first equation above, the lift to drag ratio for UAVs generally varies between 10-20, and hence even the minimum ratio ($Lift = 10Drag$ or $mg = 10Drag$) can disturb the balance in the first equation in (7.13) for even small values of γ. For positive γ (during climb), additional thrust will be required to balance the term $mg\sin\gamma$. Generally UAVs have limited extra power and hence accordingly the flight path angle γ should be restricted to γ_{max} (see Figure 7.2). During decent (negative γ), the situation reverses and the term $mg\sin\gamma$ supports engine thrust. In this case, if we decrease γ then we should also decrease engine thrust to maintain balance in the equation (non-accelerating), otherwise the speed of UAV will increase. Hence the flight path angle can be decreased to the point where the engine reaches its idle RPM (minimum), this defines γ_{min}. Conclusively, we have a constraint $\gamma_{min} \leq \gamma \leq \gamma_{max}$ in the longitudinal plane.

Another constraint in the longitudinal plane is to keep the structural loading ($\frac{L}{mg}$) within limits. As shown above, the lift force (L) is approximately equal to the weight during level and climb/decent phases. However, additional lift force is required to turn the flight path angle γ (or ground velocity vector V_g) as shown in Figure 7.2. To increase the flight path angle, we need additional lift (this should not exceed the maximum allowable limit). To reduce the flight path angle, we decrease the lift force, and this is usually not a problem. The maximum lift force

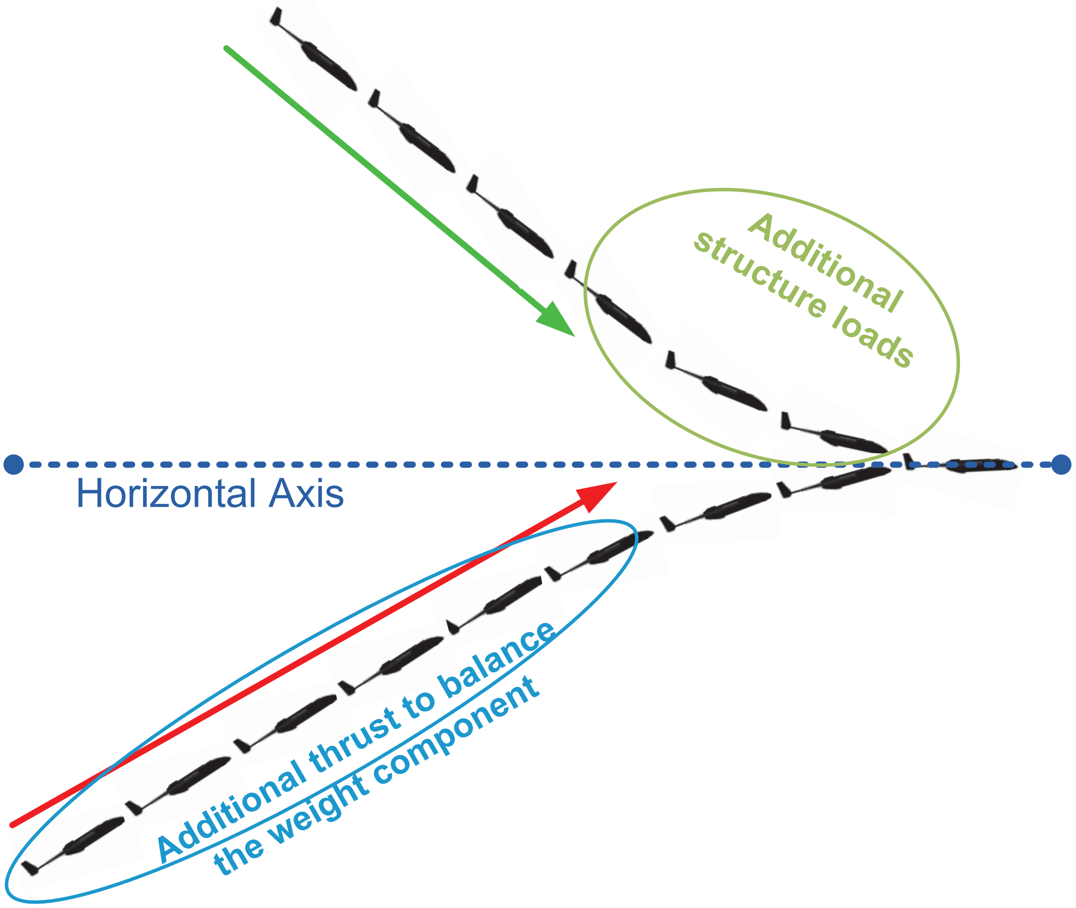


FIGURE 7.2: UAV trajectory in longitudinal plane and associated constraints.

constraint can be derived directly from the (nominal) equivalent control expression in (7.5), i.e.,

$$L_{req}\cos\phi_{req} = m\hat{V}\left(\frac{g\cos\gamma}{\hat{V}} - \frac{c_3c_4}{1+c_4^2z_e^2}\hat{V}\sin\gamma_e + \dot{\gamma}_p\right). \tag{7.14}$$

Assuming $\dot{\gamma}_p = 0$ (as discussed in section 5.4.2), and a maximum allowable structural loading of $2g$ (maximum lift force equal to twice the weight), we have the following constraint on the sliding coefficient (here $z = z_e$):

$$\left|\cos\gamma - \frac{\hat{V}^2}{g}\frac{c_3c_4}{1+c_4^2z^2}\sin\gamma_e\right| \leq 1.6 \tag{7.15}$$

Conclusively, we have the following problem to find the optimum values of the sliding coefficients c_3 and c_4:

$$\left.\begin{array}{r}
\text{Minimize Path length} = \begin{cases} \displaystyle\int\limits_{z=z_0}^{-1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for -ve } z_0 \\ \displaystyle\int\limits_{z=z_0}^{1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for +ve } z_0 \end{cases} \\
\text{Subject to} \quad \gamma_{min} \leq \gamma \leq \gamma_{max} \text{ and } \left| \cos\gamma - \frac{\hat{V}^2}{g} \frac{c_3 c_4}{1 + c_4^2 z^2} \sin\gamma_e \right| \leq 1.6 \\
\text{at each point of integration.}
\end{array}\right\} \tag{7.16}$$

For the case study of the scaled YAK-54 UAV (Figure 6.3), we have chosen $\gamma_{min} = -20°$ and $\gamma_{max} = 30°$ to find optimum sliding coefficients for different scenarios of level and climb/decent flight. Using the relation $\gamma_e = \gamma - \gamma_p$ and the sliding equation $\gamma_e = c_3 \arctan(c_4 z)$, we have the following final form of (7.16) for optimization of the sliding coefficients:

$$\left.\begin{array}{r}
\text{Minimize Path length} = \begin{cases} \displaystyle\int\limits_{z=z_0}^{-1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for -ve } z_0 \\ \displaystyle\int\limits_{z=z_0}^{1} \frac{1}{\sin\left(c_3 \arctan(c_4 z)\right)} dz & \text{for +ve } z_0 \end{cases} \\
\text{Subject to} \quad -20\frac{\pi}{180} + \gamma_p \leq c_3 \arctan(c_4 z) \leq 30\frac{\pi}{180} + \gamma_p \\
\text{and } \left| \cos(c_3 \arctan(c_4 z) + \gamma_p) - \frac{\hat{V}^2}{g} \frac{c_3 c_4}{1 + c_4^2 z^2} \sin(c_3 \arctan(c_4 z)) \right| \leq 1.6 \\
\text{at each point of integration.}
\end{array}\right\} \tag{7.17}$$

Generally the position of the UAV in the vertical plane is measured by GNSS and pressure altimeter. In case of GNSS outage, the altimeter continuously outputs altitude information with reasonable accuracy. Typically, the accuracy of the altimeter is $< 100m$ and it can sustain flight in the vertical plane even in the absence of GNSS. However, a sudden change in mission plan can generate position errors in the longitudinal plane, generally this is of the order of a few hundred meters. Conclusively, we consider small errors of few hundreds meters in the vertical plane for optimization of the longitudinal sliding coefficients.

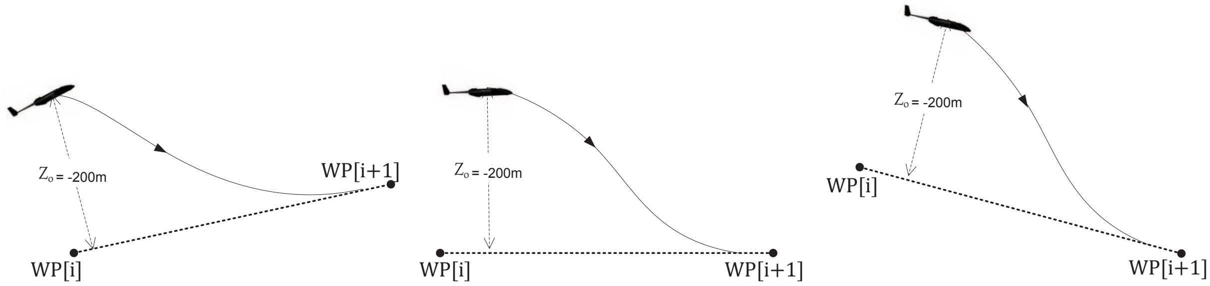


FIGURE 7.3: Trajectories for $z_0 = -200m$ and different γ_p (+ve, 0 and -ve values).

The negative z_0 means that the UAV is above the desired path in the longitudinal plane. In this case, the desired path may be horizontal or at some angle as shown in Figure 7.3. In the three cases, the traveled path calculations for $z_0 = -200m$ are shown in Figure 7.4 for $\gamma_p = 10$, 0 and -5 degrees. In case of $\gamma_p = 10deg$, the UAV has to cover approximately $445m$ distance to return back to the desired path. In other cases of γ_p, the traversed distance increases, it is $608m$ and $789m$ for $\gamma_p = 0deg$ and $\gamma_p = -5deg$, respectively. Similarly, the path calculations for $z_0 = -100m$ are shown in Figure 7.5 for the three different values of γ_p. In this case the travel distance is reduces to $240m$ for $\gamma_p = 10deg$, $313m$ for $\gamma_p = 0deg$ and $399m$ for $\gamma_p = -5deg$. The corresponding optimal values of the sliding coefficients are summarized in Table 7.1.

The positive z_0 is a different case as we have to climb against gravity to achieve the desired path. Similar to Figure 7.3, here we also have three cases of $+ve$, 0 and

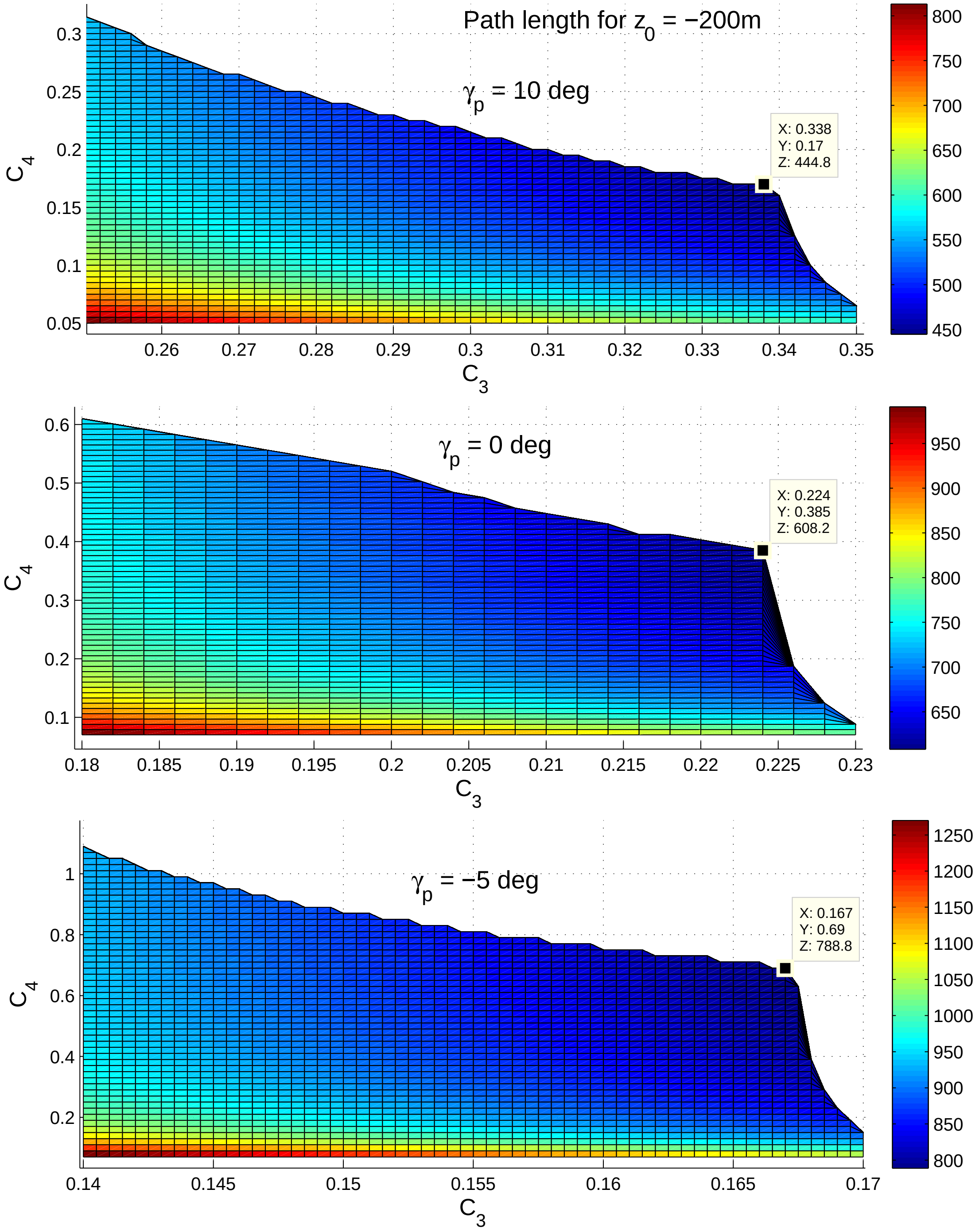


FIGURE 7.4: Long parameter selection $z_0 = -200m$.

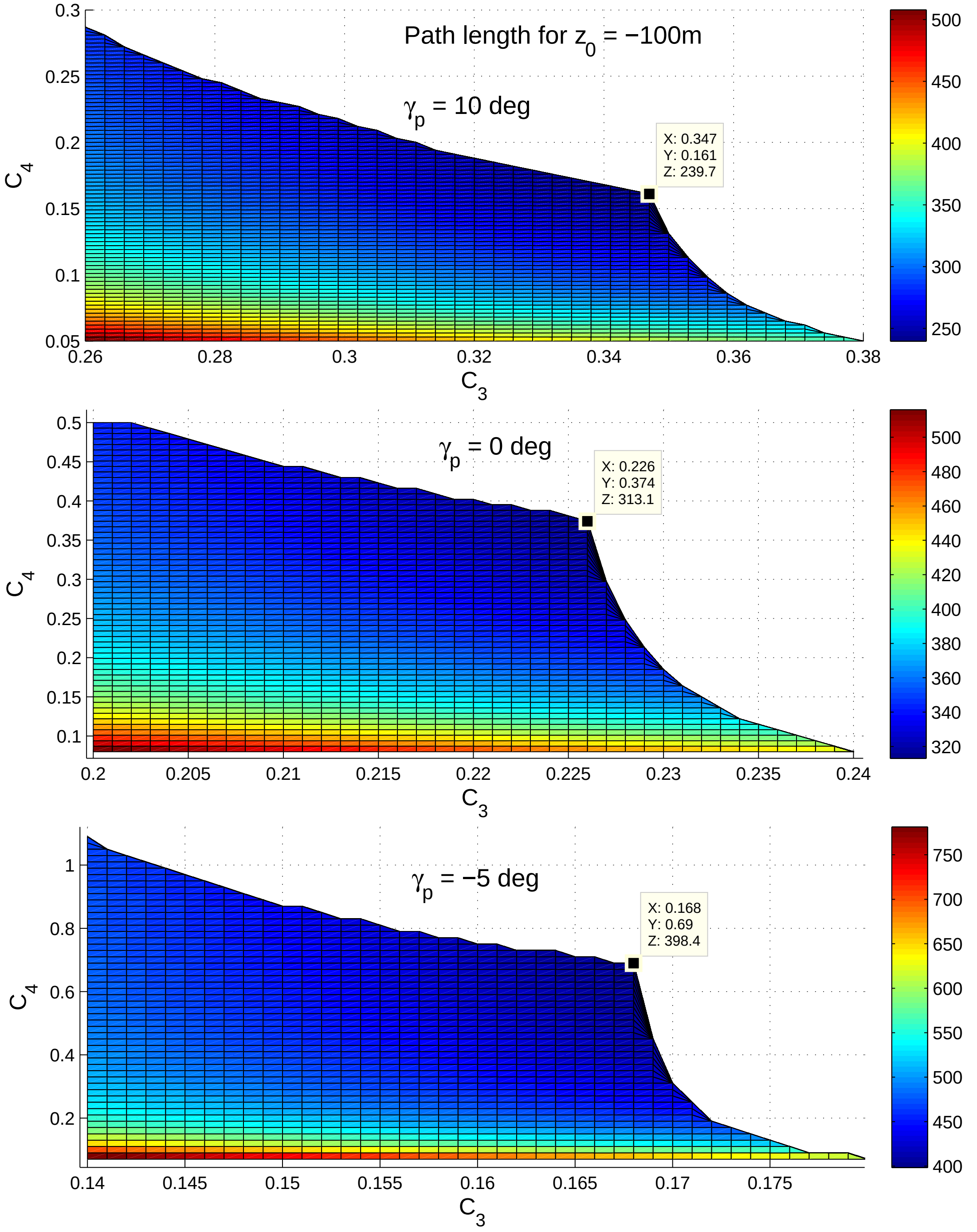


FIGURE 7.5: Long parameter selection $z_0 = -100m$.

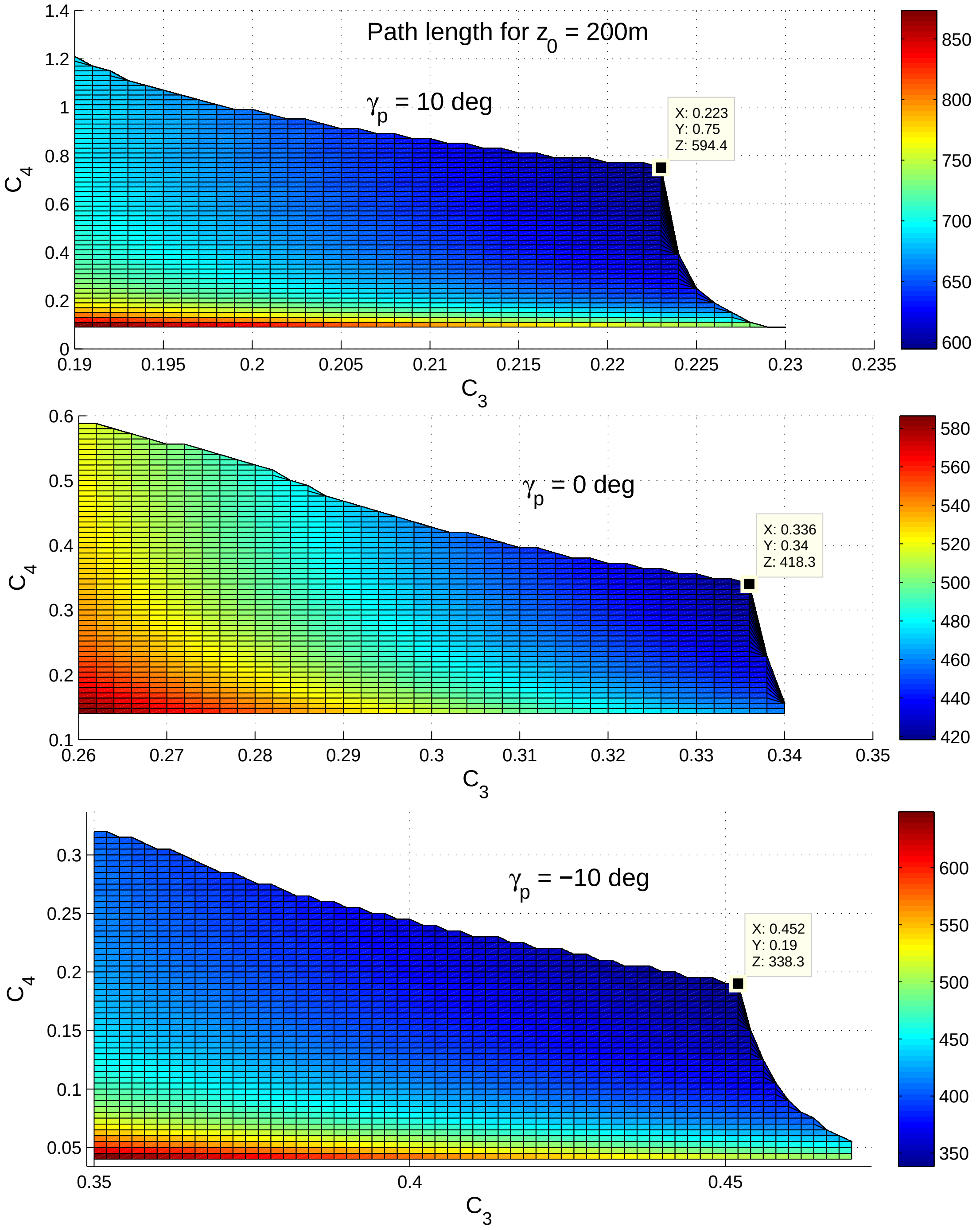


FIGURE 7.6: Long parameter selection $z_0 = 200m$.

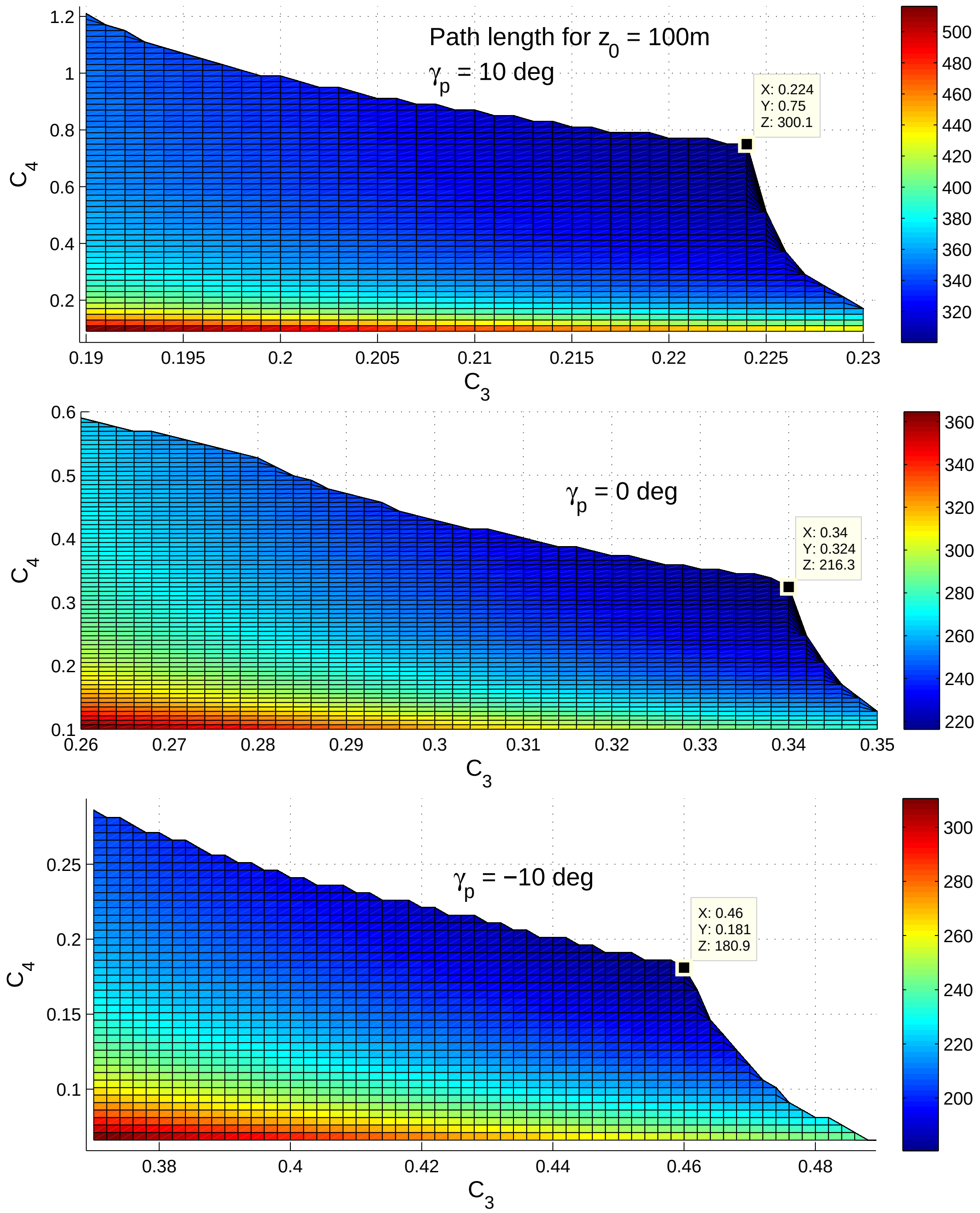


Figure 7.7: Long parameter selection $z_0 = 100m$.

Table 7.1: Summary of travel path length for different z_0 (longitudinal plane).

$z_0[m]$	$\gamma_p[deg]$	Best (c_3, c_4) combination	Path length with best (c_3, c_4) combination [m]	Path length with (c_3, c_4) = (0.17,0.69)	ratio (col-4 ÷ col-5)
-200	10	(0.34, 0.17)	445	789	0.56
-200	0	(0.22, 0.39)	608	789	0.77
-200	-5	(0.17, 0.69)	789	789	1.00
-100	10	(0.35, 0.16)	240	398	0.60
-100	0	(0.23, 0.37)	313	398	0.79
-100	-5	(0.17, 0.69)	398	398	1.00
200	10	(0.22, 0.75)	594	789	0.75
200	0	(0.37, 0.34)	418	789	0.53
200	-10	(0.45, 0.19)	338	789	0.43
100	10	(0.22, 0.75)	300	398	0.75
100	0	(0.34, 0.32)	216	398	0.54
100	-10	(0.46, 0.18)	181	398	0.45
				AVERAGE	0.73

$-ve$ values of γ_p, but in all cases we have to climb to reach the desired path. The traversed path calculations for $z_0 = 200m$ and $z = 100m$ are shown in Figures 7.6 and 7.7, respectively, for different values of γ_p. In case of $\gamma_p = 10deg$, the constraints become more stringent resulting in an increased path length. In case of $z_0 = 200m$, we have to travel at least $338 - 594m$ depending on γ_p to achieve the desired path. This distance in case of $z_0 = 100m$ is $181 - 300m$ depending on γ_p. The optimized sliding coefficient values for all these cases are summarized in Table 7.1.

Another way to deal with selection of sliding parameters is to make them adaptive for high performance in different situations, this is the topic of future research. Another simple way is to choose a single combination of sliding coefficients that gives near optimum performance for all scenarios. This is a simple and quick solution, and is opted here for implementation. Here, $c_3 = 0.17$ and $c_4 = 0.69$ are near optimum for implementation. The corresponding path lengths with this combination of c_3 and c_4 are shown in Table 7.1. In the last column of this table the ratio of 'Path length with $c_3 = 0.17$ and $c_4 = 0.69$' and the 'optimum path lengths with the best combination of c_3 and c_4' are shown. On average the chosen combination of c_3 and c_4 gives $\sim 73\%$ optimum performance for all scenarios.

7.3.2 Lateral surface parameters

Optimized lateral sliding surface parameter selection is deliberated in detail in Section 6.3.1. However the problem addressed there was the lateral path following ignoring the longitudinal dynamics. Here we have to optimize the lateral sliding parameters to follow the 3-D path without neglecting the longitudinal dynamics. In this case the main difference arises in the constraints during optimization. The maximum lateral force constraint can be derived directly from the (nominal) equivalent control expression in (7.5):

$$L_{req} \sin \phi_{req} = m\hat{V} \cos \gamma \Big(\frac{-c_1 c_2}{1 + c_2^2 y_e^2} \hat{V} \cos \gamma \sin \chi_e + \dot{\chi}_p \Big). \tag{7.18}$$

For simplicity of sliding parameter optimization, we assume non-accelerating longitudinal dynamics implying $L \cos \phi = mg \cos \gamma$ or $L_{req} \cos \phi_{req} = mg \cos \gamma$. Using this assumption, we have the following constraint in the lateral plane:

$$\left| m\hat{V} \cos \gamma \Big(\frac{-c_1 c_2}{1 + c_2^2 y_e^2} \hat{V} \cos \gamma \sin \chi_e + \dot{\chi}_p \Big) \right| \leq mg \cos \gamma \tan \phi_{req_{max}} \tag{7.19}$$

After simplification we have:

$$\left| \frac{\hat{V}}{g} \Big(\frac{-c_1 c_2}{1 + c_2^2 y_e^2} \hat{V} \cos \gamma \sin \chi_e + \dot{\chi}_p \Big) \right| \leq \tan \phi_{req_{max}}, \tag{7.20}$$

As compared to the optimization problem in section 6.3.1, this constraint depends on the value of flight path angle γ. In fact $\hat{V}$ is replaced by the velocity component $\hat{V} \cos \gamma$ here. So the velocity is reduced by the factor $\cos \gamma$ during climb/decent.

$\gamma = 0$ yields the lateral guidance optimization problem as discussed in section 6.3.1. For optimization of lateral sliding surface parameters for the 3-D guidance problem, the previous process of section 6.3.1 should be repeated for $\gamma = \gamma_{max}$. It may be mentioned here that as the term $\hat{V}\cos\gamma$ reduces in case of $\gamma = \gamma_{max}$, hence we expect larger values of the sliding parameters (c_1 and c_2). However, the conservative case is $\gamma = 0$. For simplicity we do not repeat the process here and use the same values of c_1 and c_2 as derived in section 6.3.1.

7.4 Limitations in Implementation

To implement the proposed 3-D guidance law in the flight control computer of the test UAV, accurate measurement/estimation of state variables (y_e, z_e, χ_e, γ_e) is required. In the case of the scaled YAK-54 UAV, the difficulty arises in the measurement of the flight path angle γ, and hence to compute γ_e. For the measurement of flight path angle $\gamma = \arctan\left(\frac{-\dot{z}}{\sqrt{\dot{x}^2+\dot{y}^2}}\right)$, rate of change of position (ground velocity components) is required. As this is a low-cost research platform, a MEMS based AHRS with an integrated GPS is used for navigation state measurement. Generally the 3-D position available from GPS has some inaccuracy and fluctuation, this is more prominent in altitude. MEMS based AHRS outputs vertical velocity by integrating the measured acceleration and augmenting this with GPS information. The piston engine in our YAK-54 UAV produces significant vibrations in the vertical plane due to the movement of the piston and spark ignition in this plane. The vibrations are enhanced by the light weight of the UAV, resulting in poor measurement of vertical velocity in this case.

In Figure 7.8 altitude measurement and the corresponding vertical velocity output from the AHRS is shown for a sample flight. Due to vibrations from the engine and noisy measurement of rate of change of altitude from GPS, correct information of vertical velocity is not available from this sensor. Due to this limitation direct

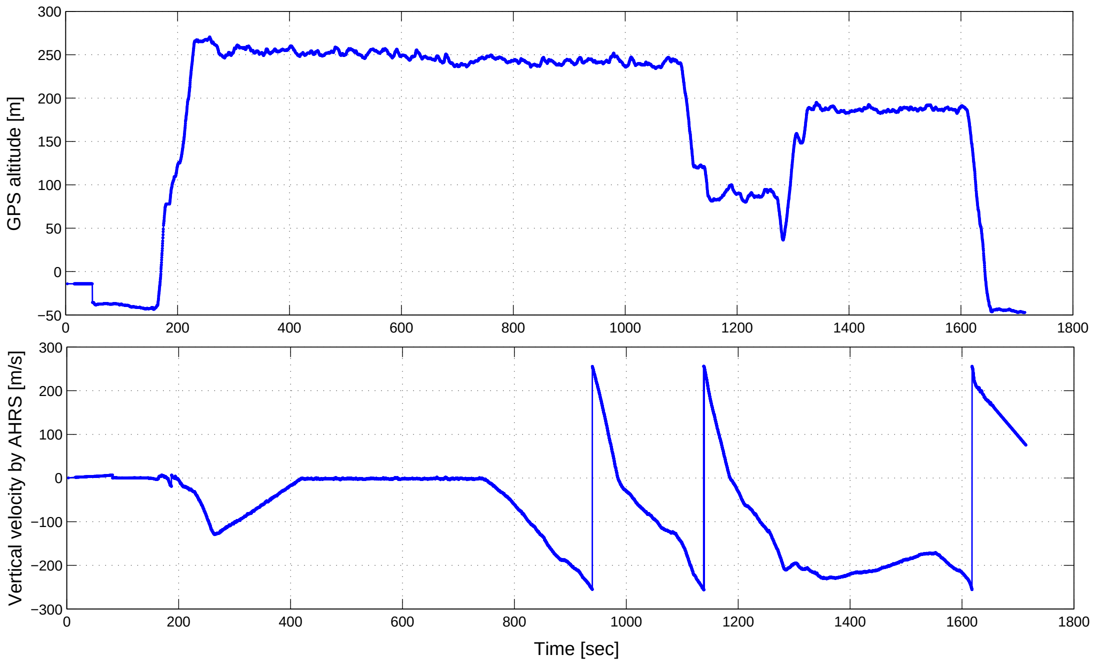


FIGURE 7.8: Vertical velocity measurement during a flight.

application of 3-D guidance is not possible on our test platform. However, this problem can be solved by a better navigation sensor, or by the use of an observer, but this is beyond the scope of this thesis. The 6-dof nonlinear simulation already validated through flights will be used for demonstration of the proposed 3-D guidance scheme.

7.5 Simulation Results

The proposed 3-D guidance law is implemented in the 6-dof nonlinear simulation of the scaled YAK-54 UAV to show its effectiveness. The structure and details of the nonlinear simulation are covered in Annexure-A. In section 6.5, the results of this 6-dof simulation are compared with that of actual flights, and both are found to be in good agreement. The nonlinear 6-dof simulation has a complete nonlinear model of the UAV (Stevens and Lewis, 2003) with options of different input disturbances like winds. Different scenarios of 3-D mission following are simulated and results presented in this section.

The proposed guidance law (7.5) generates $L_{req} \sin \phi_{req}$ and $L_{req} \cos \phi_{req}$ as control inputs. The required roll angle (ϕ_{req}) and the required lift force (L_{req}) can be computed from these terms. As per structure of the inner control loop, θ_r command is generated in our case for the inner pitch control loop using the relation $\theta = \alpha + \gamma$ where $L = C_{L_\alpha} \alpha$.

7.5.1 Parameter selection for the scaled YAK-54 UAV

The proposed 3-D guidance algorithm (7.5) has four gains ($k_{\Delta_1}, k_1, k_{\Delta_2}, k_2$) to be selected for its implementation. Here the two gains k_{Δ_1} and k_1 are to be selected to keep the lateral deviation as small as possible, and the other two gains for controlling the longitudinal track error. The two gains k_{Δ_1} and k_{Δ_2} can be selected directly from the reachability condition (7.11) to ensure sliding motion even in the presence of disturbances. In our case $V_{min} = 30 m/s$, $V_{max} = 40 m/s$, maximum $\dot{\chi}_p = 0.0875$ (for a minimum turn radius of 400m), $\dot{\gamma}_p = 0$, uncertainty in velocity $\tilde{V} = 5 m/s$ and sliding parameters ($c_1 = 0.7$, $c_2 = 0.002$, $c_3 = 0.17$, $c_4 = 0.69$). We have:

$$k_{\triangle_1} > 0.0286, \qquad k_{\triangle_2} > 1.1749. \tag{7.21}$$

We choose $k_{\triangle_1} = 0.05$ and $k_{\triangle_2} = 1.2$ in our case for implementation. The other two gains k_1 and k_2 can be selected for reducing the reaching time to the sliding surface. In fact the effective gains passing to the system due to these control terms are $k_1 s_1$ and $k_2 s_2$ that depends on the value of sliding surfaces too. The objective is to increase the system control input in case of large deviation from the sliding surface. We have targeted atleast 10% contribution from this factor when the deviation from from sliding surface is 0.25 and more contribution in case of large deviations, so selected $k_1 = 0.02$ and $k_2 = 0.5$. The value of k_1 is chosen such that the system effective gain increases $\sim$ 10% for $s_1 = 0.25$. Here $k_1 s_1 = 0.005$ for $s_1 = 0.25$, that is $\sim$ 10% of $k_{\triangle_1} = 0.05$. Similarly, the selection of other gain k_2 is done. A boundary layer approximation $\mathrm{sgn}(s) \approx \frac{s}{|s|+\epsilon}$ is used to avoid chattering in the control signal. The value of ϵ is selected 0.7 for first expression and 0.4 for the other expression in equation (7.5).

7.5.2 Nominal case

Simulation results for a nominal (without disturbance) pre-planned trajectory are shown in Figures 7.9 - 7.12. In Figure 7.9, the desired 3-D path and the actual path flown by the UAV are shown. The desired path has all type of maneuvers including ascend/descend and turning at the same time. Initially the UAV is slightly away from the desired path, and is then guided towards the desired path to follow the mission. The mission following is good in this case as is evident from the lateral and longitudinal errors in Figure 7.10. Since the desired mission is changing continuously every few seconds, hence errors of $\sim 5m$ are generated during the start of a maneuver. The overall steady state errors in this case are generally less than $1m$. Similarly errors in the direction of the velocity vector in the lateral and longitudinal planes (χ_e and γ_e) are also shown. Small errors of a few degrees are seen during the start of a new maneuver, but the steady state errors are less than 0.5 degree.

The corresponding sliding surfaces (s_1 and s_2) versus time are shown in Figure 7.11, their steady state magnitude is less than 0.01. The required pitch angle and the required roll angle generated by the 3-D guidance scheme are shown in Figure 7.12 along with actual angles tracked by the inner control loop. Both control signals are smooth enough and no significant chattering is seen. The pitch angle rises up to 12 degrees during climb, and drops to -4 degrees during descent. The required roll angle is in the range of -20 to 25 degrees. A zoomed view of 318-352 seconds is also shown to indicate the smoothness of the control signal.

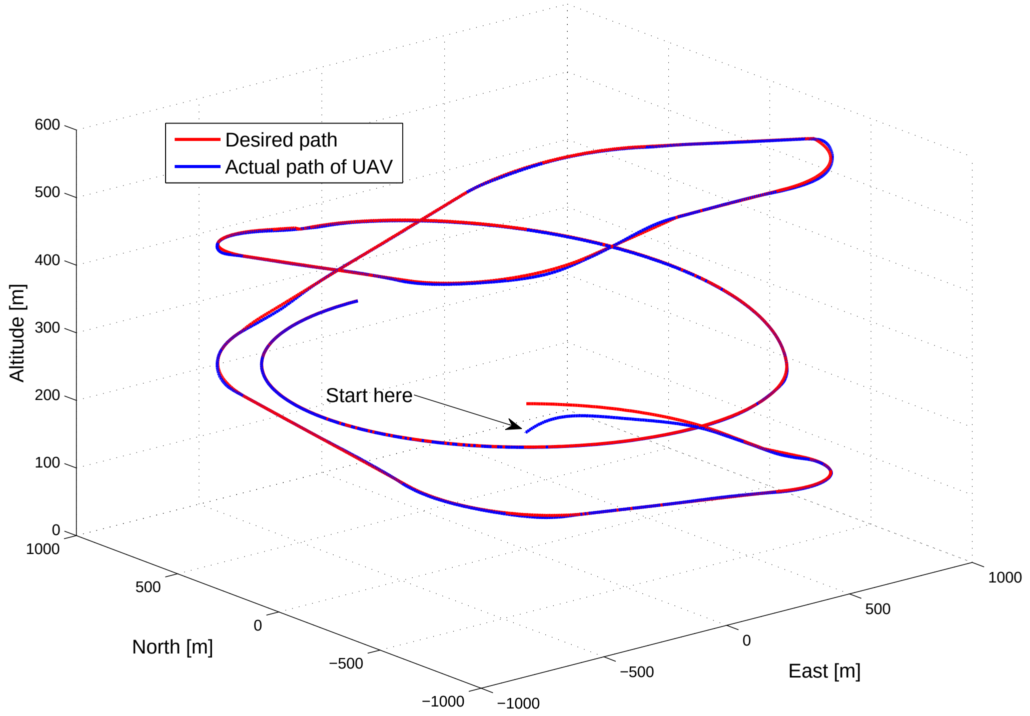


FIGURE 7.9: Desired and actual 3-D paths

7.5.3 With disturbance

In this case, performance of the proposed guidance logic is evaluated in the presence of wind and GPS outage. As an external disturbance, an east wind of $4m/s$ is applied throughout in this case. Also the outage of GPS is simulated, and error in the position of the UAV is fed in to the simulation after few seconds of GPS outage. The position error due to GPS outage is simulated at $200s$, and the 3-D guidance algorithm is required to bring the UAV on to the desired path. In Figure 7.13, the desired 3-D path and the actual UAV position are shown in 3-D space. The sudden jump in the position of the UAV is due to loss of GPS, and shows the UAV in positional error when the GPS becomes available again. The 3-D guidance algorithm brings the vehicle gracefully back on to the desired path even in the presence of wind. The lateral and longitudinal errors in position are shown in Figure 7.14. In this case the steady state error in position is less

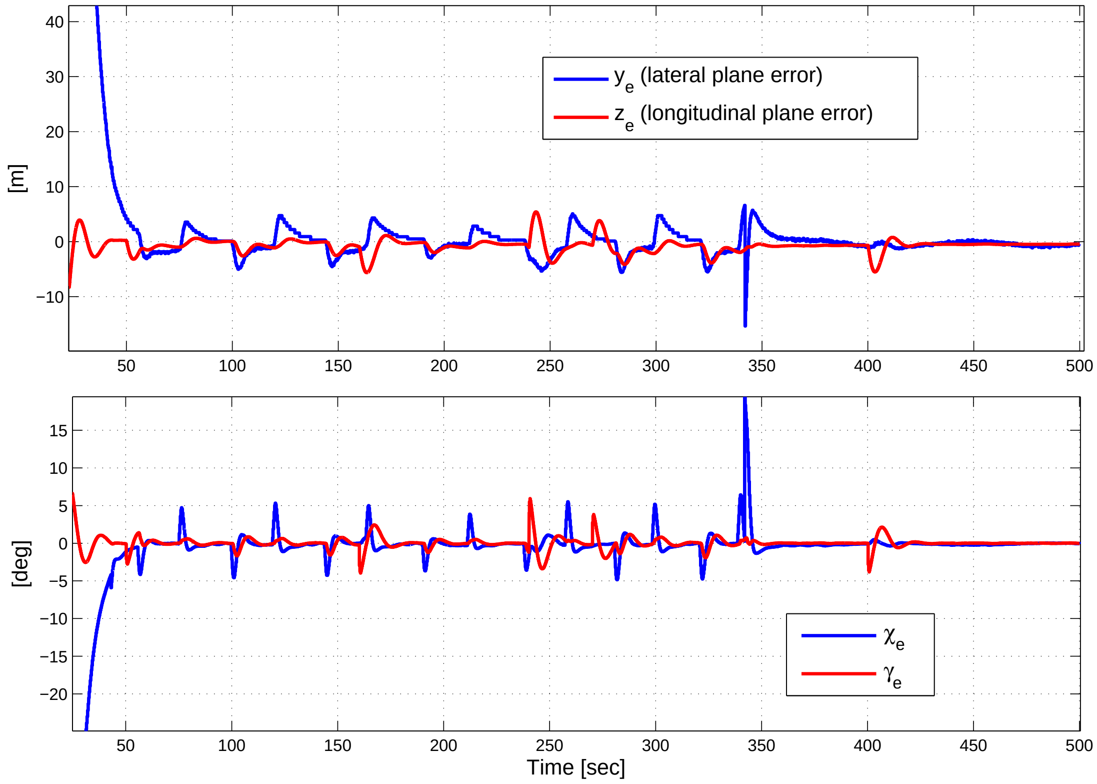


FIGURE 7.10: Nominal errors

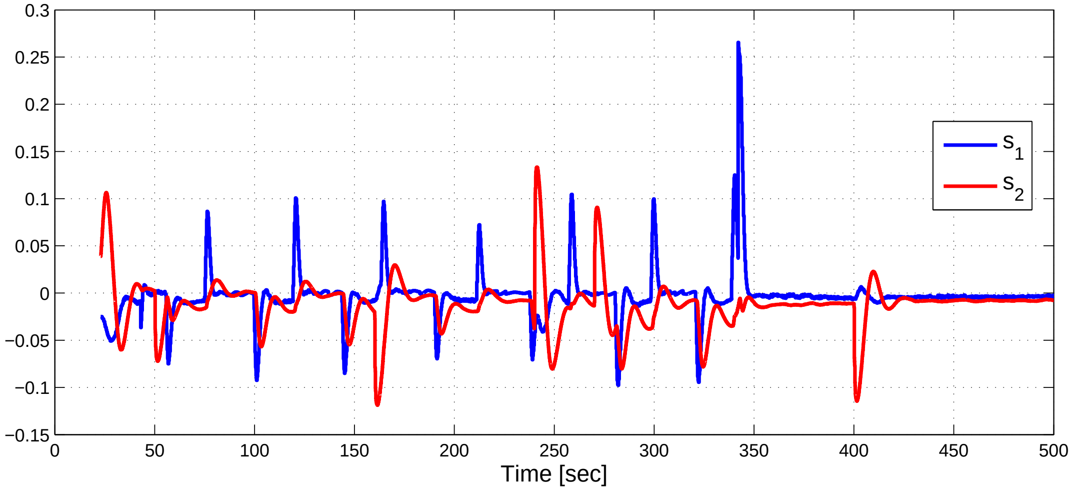


FIGURE 7.11: Nominal ss

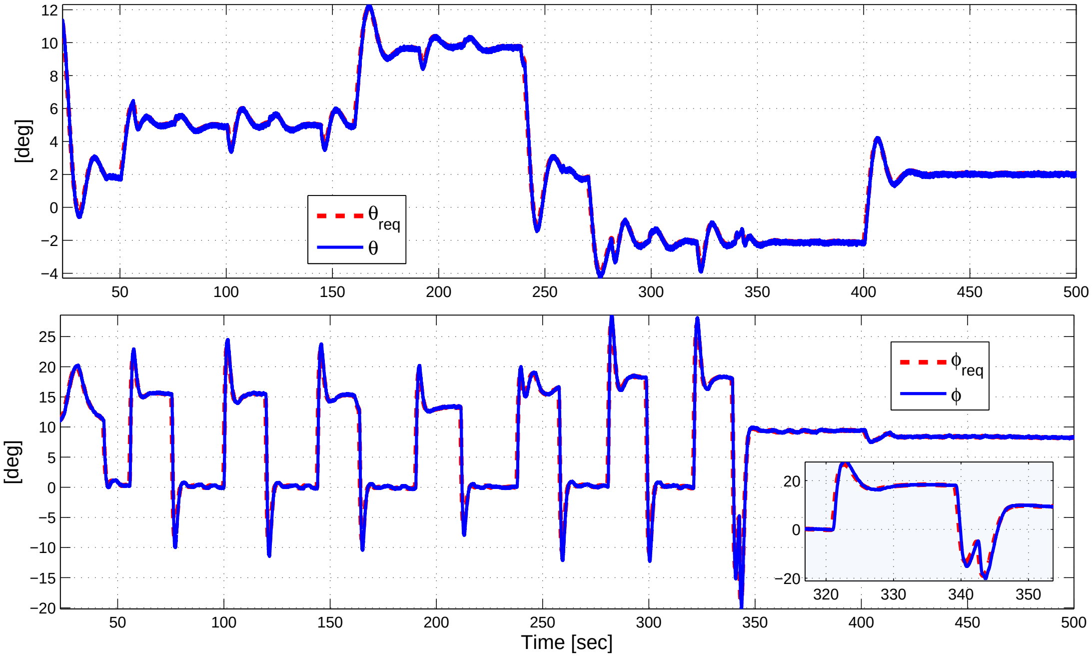


Figure 7.12: Nominal control

than 3m. The 80m vertical error in position is reduced to zero in approximately 12 seconds, however it takes approximately 40 seconds to bring the lateral error of 500m to zero. The velocity direction is also aligned with the desired direction in this case as is evident from the trends of χ_e and γ_e versus time.

The sliding surfaces s_1 and s_2 versus time are shown in Figure 7.15. The steady state magnitude of s_1 and s_2 is less than 0.03 in this case, however some disturbing peaks are visible due to external disturbances. The reference control signals (θ_{req} and ϕ_{req}) generated by the 3-D guidance scheme are shown in the 7.16. In this case, more control effort is required to cancel the effect of disturbances as seen from this figure. The maximum magnitude of θ_{req} and ϕ_{req} in this case is approximately 14 and 35 degrees, respectively. Overall the desired mission is successfully followed even in the presence of disturbances.

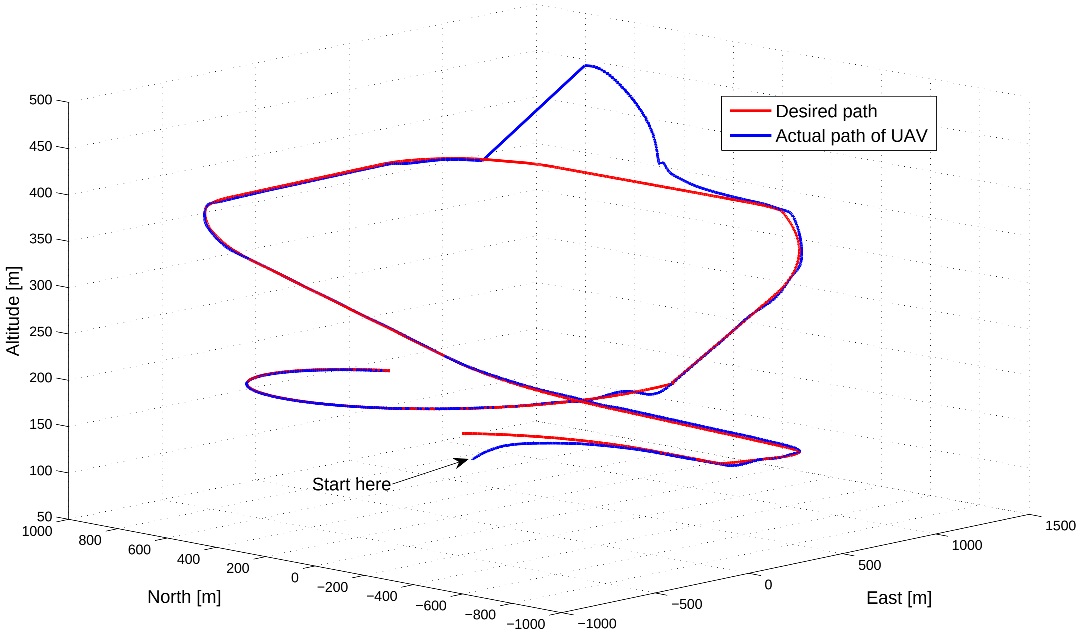


FIGURE 7.13: With disturbance trajectory

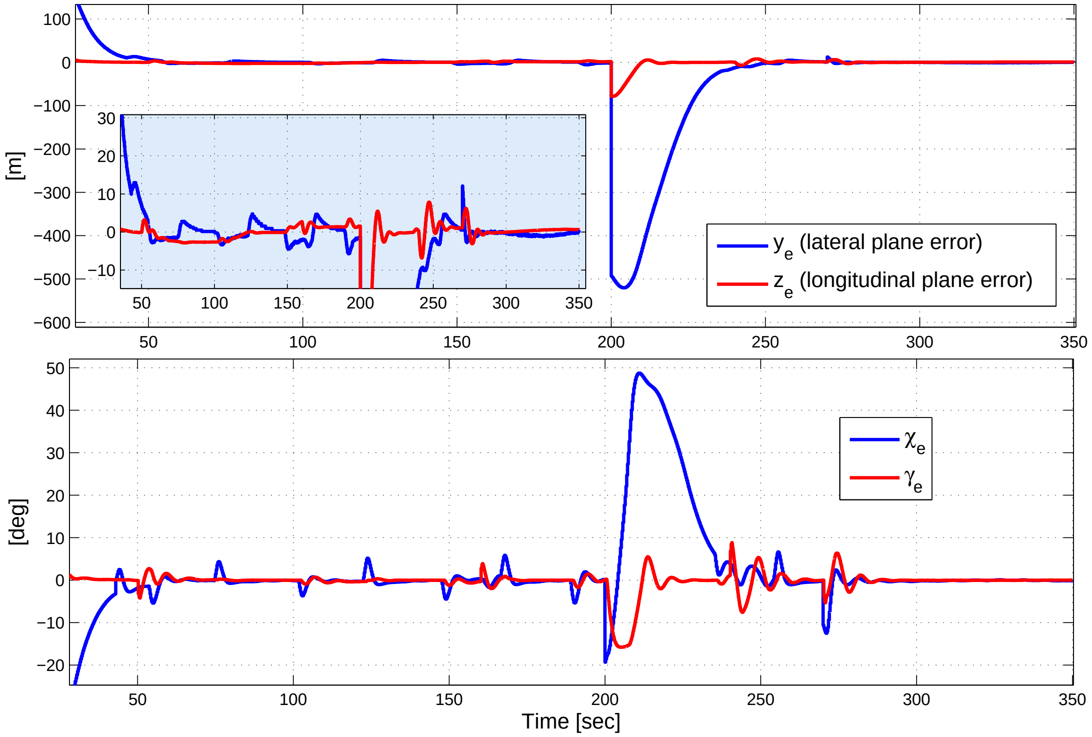


FIGURE 7.14: With disturbance errors

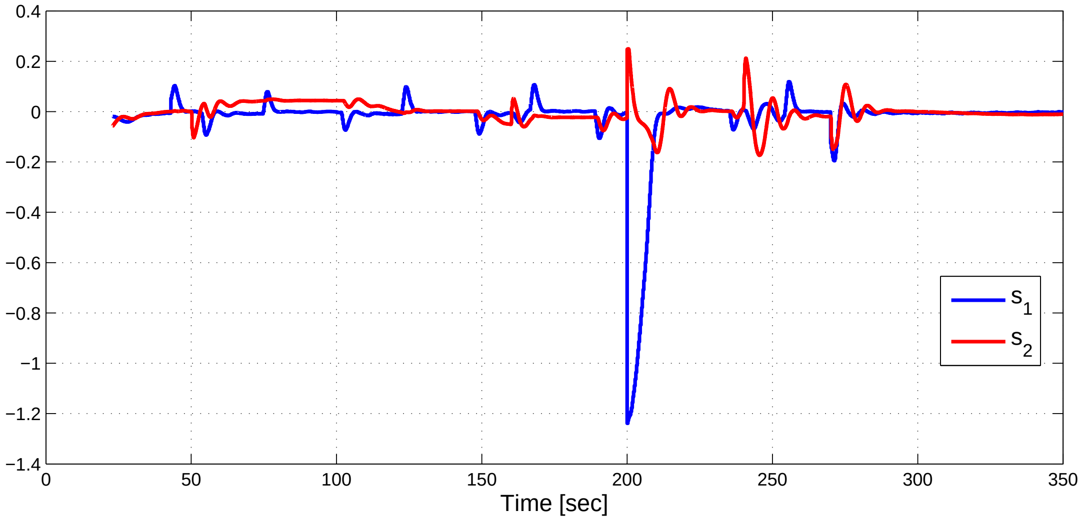


FIGURE 7.15: With disturbance ss

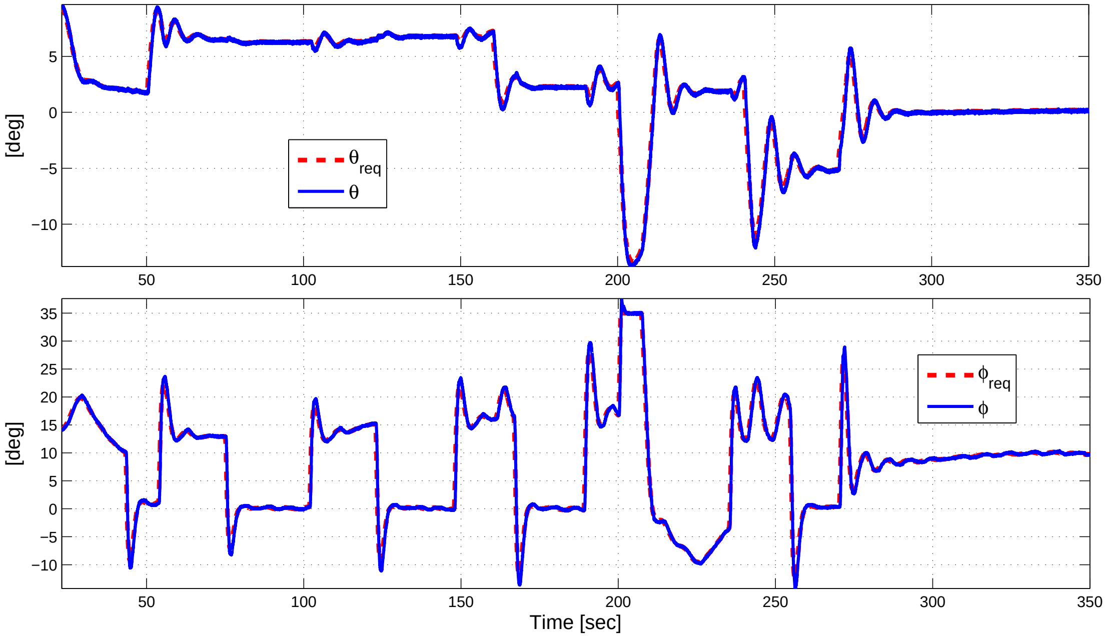


FIGURE 7.16: With disturbance control

7.6 Summary of the Chapter

A generalized 3-D guidance scheme for UAVs based on the sliding mode theory is derived here. The proposed scheme considers the coupling between longitudinal and lateral planes. The scheme utilizes two nonlinear sliding manifolds for high performance in different scenarios. The optimum values of the sliding parameters are computed here for the scaled YAK-54 UAV, and finally a set of parameters is proposed. Based on these manifolds, the guidance logic is derived, ensuring reachability. The proposed 3-D guidance scheme is implemented on a 6-degrees-of-freedom (6-dof) nonlinear simulation of the scaled YAK-54, simulation results are presented for different 3-D trajectories with and without disturbances. It is observed that the accuracy and smoothness requirements are fulfilled satisfactorily.

Chapter 8

CONCLUSIONS AND FUTURE WORK

This thesis presents guidance scheme design for UAVs using sliding mode theory. The main aim of the research is to develop robust 2-D (lateral) and 3-D guidance logic for UAVs so that the desired path can be followed despite the presence of uncertainties and external disturbances.

8.1 Summary

Sliding mode based control law design is a two step process: selection of a stable sliding surface, followed by the design of a discontinuous control law to reach and maintain sliding motion. Most works in sliding mode control applications consider linear sliding surfaces for control law design. In the UAV guidance perspective, we evaluate linear sliding surfaces and show these are not a viable choice here. Linear sliding surfaces and associated stability issues are highlighted in Chapter 4. In this Chapter, we also propose piece-wise linear sliding surfaces for lateral guidance, these overcome the stability issues associated with linear surfaces.

For high performance in all scenarios, nonlinear sliding surfaces for lateral and longitudinal planes are proposed in Chapter 5. Stability of these manifolds is proved using Lyapunov theory. After proposing stable nonlinear sliding surfaces, the selection of sliding coefficients is deliberated in detail. To choose optimum sliding coefficients, work and energy principle is used to derive a criterion for the best selection of these coefficients. The resultant coefficients yield the minimum travel path to return to the desired track in case of any deviation.

Based on the nonlinear sliding surface, lateral guidance logic is derived in Chapter 6 for straight and circular path following. Stability and control boundedness is guaranteed during derivation of the guidance law. To demonstrate effectiveness of the proposed algorithm, a case study of a scaled YAK-54 UAV is undertaken, from design to flight testing. Optimal sliding surface coefficients and guidance law is derived for the said UAV. The proposed algorithm is programmed in the flight control computer of the test UAV and flight results for different scenarios are presented. The flight results are also compared with non-linear 6-dof simulation results, and both are seen to be in good agreement.

Generalized 3-D path following algorithm based on sliding mode theory is derived in Chapter 7. This guidance law is derived based on the generalized kinematic equations that consider coupling between the longitudinal and lateral planes. Two optimized sliding surfaces are used to derive the guidance scheme for this MIMO system. Effectiveness of the proposed algorithm is demonstrated via 6-dof nonlinear simulations under different scenarios.

8.2 Directions for Future Research

There are a number of prospective topics that may extend from this research. Guidelines for future work in this area are provided below.

- Boundary layer approximation is used here to reduce chattering in the control signal. Higher order sliding mode based guidance laws for UAVs is an immediate extension of this work to reduce chattering.
- In the derivation of lateral guidance logic, it is assumed here that the inner loop dynamics are fast enough, i.e., $\phi_{req} \approx \phi$. This assumption may be relaxed and a suitable first or second order dynamical model can be considered for the inner loop. However, this will increase the relative degree of the system and an appropriate guidance law will be required in this case.
- Nonlinear sliding surfaces using the ‘arctan’ function are proposed here. Other nonlinear sliding surfaces are also possible which may be explored.

- Adaptive sliding surface based guidance law design for high performance and reduced reaching time is another extension of this work.

Appendix-A:
6-DoF Nonlinear Simulation

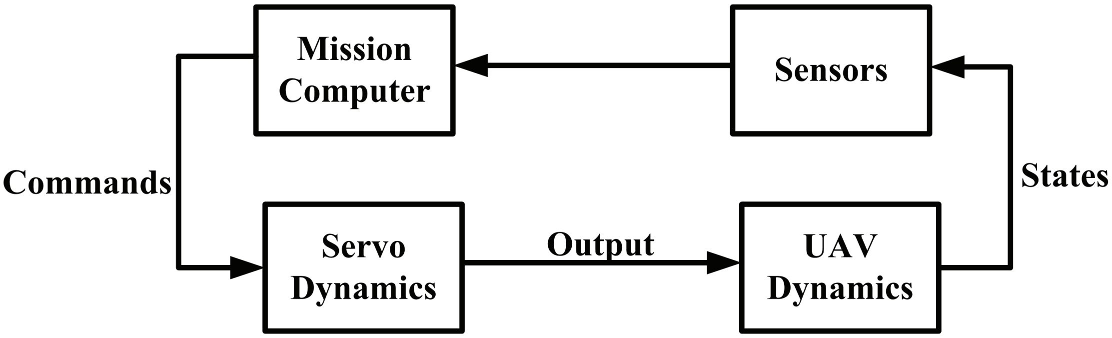


FIGURE 1: Overall simulation structure

UAV Dynamics Block

This block has 12 state equations

- **Force Equations**

$$\begin{aligned} m\left(\dot{u}-vr+wq\right) &= mg_x+F_{A_x}+F_{T_x} \\ m\left(\dot{v}+ur-wp\right) &= mg_y+F_{A_y}+F_{T_y} \\ m\left(\dot{w}-uq+vp\right) &= mg_z+F_{A_z}+F_{T_z} \end{aligned}$$

- **Moment Equations**

$$\begin{aligned} I_{xx}\dot{p}-I_{xz}\dot{r}-I_{xz}pq+\left(I_{zz}-I_{yy}\right)rq &= L_A+L_T \\ I_{yy}\dot{q}+\left(I_{xx}-I_{zz}\right)pr+I_{xz}\left(p^2-r^2\right) &= M_A+M_T \\ I_{zz}\dot{r}-I_{xz}\dot{p}-I_{xz}qr+\left(I_{yy}-I_{xx}\right)pq &= N_A+N_T \end{aligned}$$

- **Euler Rotation Angles (Sequence ψ, θ, ϕ)**

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

- **Position in Earth Fixed Frame**

$$\begin{bmatrix}\dot{x}_I\\ \dot{y}_I\\ \dot{z}_I\end{bmatrix} = \begin{bmatrix}C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi\\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi\\ -S_\psi & S_\phi C_\theta & C_\phi C_\theta\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix}$$

In above state equations

$$\begin{bmatrix}F_{x_a}\\ F_{y_a}\\ F_{z_a}\end{bmatrix} = T_{stab2body}\begin{bmatrix}C_D\\ C_Y\\ C_L\end{bmatrix} q_\infty S_{ref}$$

$$\begin{aligned}
C_D &= C_{D_0} + C_{D_\alpha}\alpha + C_{D_u}\frac{u}{V_\infty} + C_{D_{\delta e}}\delta_e + \frac{C_L^2}{\pi e AR}\\
C_Y &= C_{Y_\beta}\beta + C_{Y_u}\frac{u}{V_\infty} + C_{Y_{\delta a}}\delta_a + C_{Y_{\delta r}}\delta_r + (C_{Y_p}p + C_{Y_r}r)\frac{b}{2V_\infty}\\
C_L &= C_{L_0} + C_{L_\alpha}\alpha + C_{L_u}\frac{u}{V_\infty} + C_{L_{\delta e}}\delta e + (C_{L_q}q + C_{L_{\dot\alpha}}\dot\alpha)\frac{\bar{c}}{2V_\infty}
\end{aligned}$$

q_∞ is the dynamic pressure defined as $\frac{1}{2}\rho V^2$. Reference area $S_{ref} = 1.0126m^2$.

Table 1: Longitudinal Stability Derivatives (rad^{-1})

$C_{T_{xu}}$	-0.1546	C_{D_u}	0.0011	C_{D_a}	0.0863
C_{L_u}	0.0017	C_{L_a}	4.5465	$C_{L_{\dot\alpha}}$	1.8918
C_{L_q}	5.5046	C_{m_u}	0.0002	C_{m_α}	-0.3937
$C_{m_{\dot\alpha}}$	-4.3787	$C_{m_{T_U}}$	0	C_{m_T}	0.0009
C_{m_q}	-16.1064	$C_{m_{T_\alpha}}$	0.0275	$C_{D_{\delta_e}}$	0
$C_{L_{\delta_e}}$	0.3792	$C_{m_{\delta_e}}$	-0.8778	C_{L_0}	0.1470
C_{D_0}	0.0422	C_{m_0}	0.0001	C_{T_x}	0.0515

$$\begin{bmatrix}L_A\\ M_A\\ N_A\end{bmatrix} = \begin{bmatrix}C_l b\\ C_m\bar{c}\\ C_n b\end{bmatrix} q_\infty S_{ref}$$

$$\begin{aligned}
C_l &= C_{l_\beta}\beta + C_{l_{\delta a}}\delta_a + C_{l_{\delta r}}\delta_r + (C_{l_p}p + C_{l_r}r)\frac{b}{2V_\infty}\\
C_m &= C_{m_0} + C_{m_\alpha}\alpha + C_{m_u}\frac{u}{V_\infty} + C_{m_{\delta e}}\delta_e + (C_{m_q}q + C_{m_{\dot\alpha}}\dot\alpha)\frac{\bar{c}}{2V_\infty}\\
C_n &= C_{n_\beta}\beta + C_{n_{\delta a}}\delta_a + C_{n_{\delta r}}\delta_r + (C_{n_p}p + C_{n_r}r)\frac{b}{2V_\infty}
\end{aligned}$$

Wing span $b = 2.4162m$ and chord length $\bar{c} = 0.4408m$.

TABLE 2: Lateral and Directional Stability Derivatives (rad^{-1})

C_{Y_β}	-0.2707	C_{Y_p}	0.0194	C_{Y_r}	0.2531
C_{l_β}	-0.0220	C_{l_p}	-0.5858	C_{l_r}	0.0743
C_{n_β}	0.1052	C_{n_p}	-0.0387	C_{n_r}	-0.2890
$C_{Y_{\delta_a}}$	0.0	$C_{Y_{\delta_r}}$	0.2228	$C_{l_{\delta_a}}$	0.3707
$C_{l_{\delta_r}}$	0.0219	$C_{n_{\delta_a}}$	-0.0088	$C_{n_{\delta_r}}$	-0.1003
$C_{n_{T\beta}}$	-0.0045				

$$\begin{bmatrix} F_{T_x} \\ F_{T_y} \\ F_{T_z} \end{bmatrix} = \text{Engine Thrust Force Components}$$

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = \text{Gravity Force Components}$$

Servo Dynamics Block

For Guidance and Control logic design, servo (actuator) dynamics is very important.

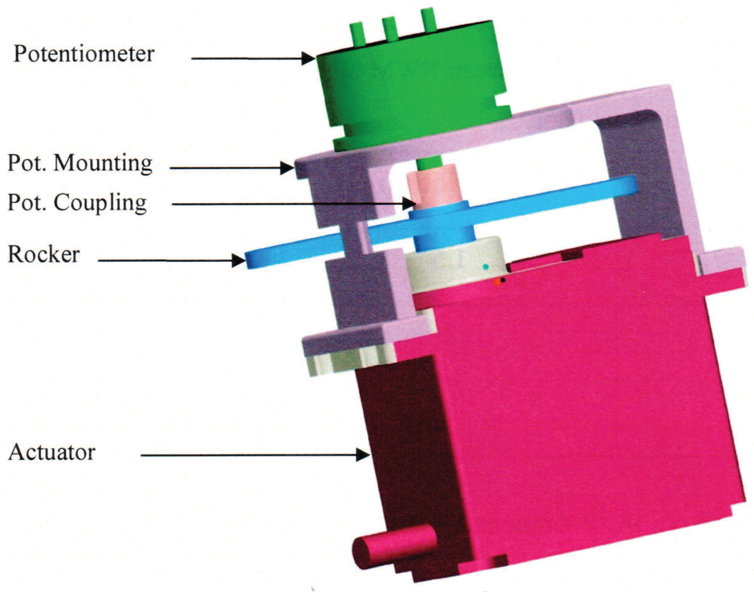


FIGURE 2: HS-5645MG servo with feedback mechanism

In our case, we used HITEC HS-5645MG servo in our UAV as shown in Figure 2. The mathematical model of servo is found experimentally under different load

conditions in a ground testing mechanism for actuator testing.
HITEC HS-5645MG is an electro-mechanical, low cost, commercial standard PWM driven actuator. The specification are as follows:

TABLE 3: HITEC HS-5645MG Servo Specification

Sr.	Parameter	Description
1.	Speed	0.18sec/60° at 6.0V (333°/sec)
2.	Stall Torque	164.2 oz-in at 6.0V (1.16Nm)
3.	Length	1.59 inches (40.6mm)
4.	Width	0.77 inches (19.8mm)
5.	Height	1.48 inches (37.8mm)
6.	Weight	2.11 oz (60g)

- **External Feedback Mechanism:** A special fixture was designed and fabricated with provision of external POT for feedback acquisition mechanism.
  External POT of given parameters was used
  Resistance: 2±0.4KΩ
  Power up voltage: ±15V
  Total Span: 340°±4°
  The fixture along with provision of external POT feedback is shown in the Figure 2.

- **Transfer Function Estimation:**
  For the estimation of T/F, various tests were conducted under no load and load conditions. The said actuator accepts PWM signal as input, however for frequency and time domain analysis it was required that command and feedback are recorded in analog form at one place. For this purpose, an analog to PWM conversion board is used. The Table 4 describes the relation between PWM signals, analog voltage command and the corresponding deflection of actuator.

TABLE 4: HS-5645MG PWM signal v/s analog voltage

Sr.	Voltage	Pulse Width [ms]	Actuator deflection [deg]
1.	0	1.5	0
2.	+10	2	+50
3.	-10	1	-50

Proposed Mathematical Model: The results of extensive testing (no load 15 times, load tests 26 times) were analyzed thoroughly. It was concluded that the system's response best fits the experimental data at 1V (5°) under load and no load conditions using average values of ξ and

Table 5: Test Conditions for Transfer Function Estimation

Sr.	Amplitude (V)	Deflection (°)	Static Load (Kg)	Actual Applied Load (Nm)	% of Maximum Load
1.	0.5	2.5	1	0.19	16.4
2.	0.5	2.5	2	0.38	32.8
3.	1	5	1	0.19	16.4
4.	1	5	2	0.38	32.8
5.	1.5	7.5	1	0.19	16.4
6.	1.5	7.5	2	0.38	32.8

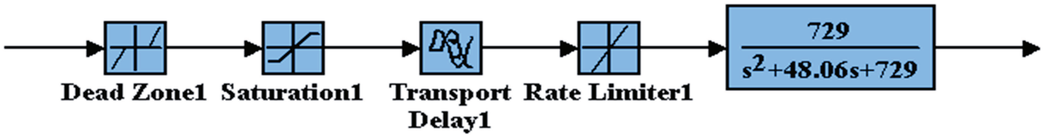


Figure 3: HS-5645MG servo SIMULINK model

w_n. After thorough testing and analysis following mathematical model of HS-5645MG was finalized and implemented in Simulation:
Voltage 2 Deg: 5°/V (5°/0.048ms)
Saturation: ±50°
Delay at 1V & 0.19 Nm (1Kg): 23msec
Dead Zone: ±0.1°
Rate Limit: 500°/sec

$$T/F = \frac{729}{s^2 + 48s + 729}$$

Time domain and frequency domain responses of the actuator system with the proposed transfer function are shown in Figure 4 and Figure 5, respectively.

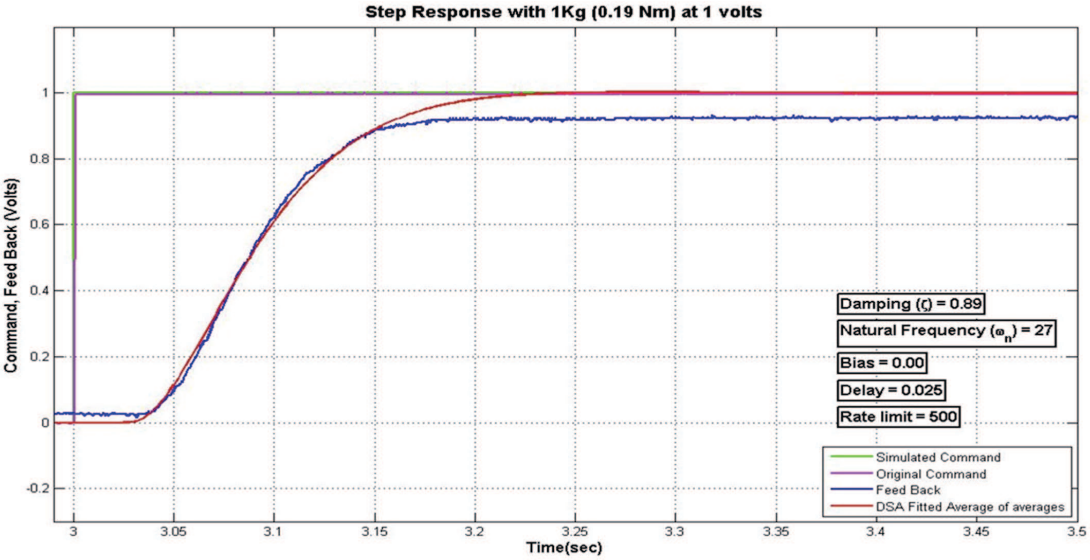


FIGURE 4: Time domain response of actuator with proposed mathematical model.

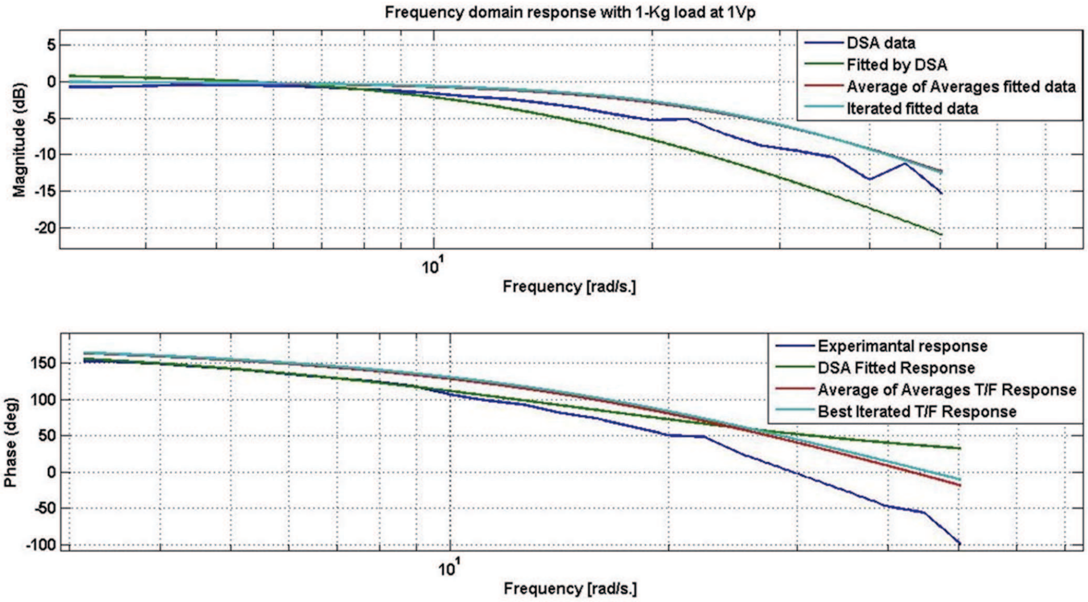


FIGURE 5: Frequency domain response of actuator with proposed mathematical model.

Sensors Block

Two sensors modeled are altimeter (RC-ALTPro3) and AHRS with built in GPS module.

- **Altimeter:** The alt-pro3 outputs data at 10Hz. A sample output is

  $$\$RCALT, +00531, +023, 490, +0351, +04907 * CS$$

  where altitude above ground is 53.1m, ROC 2.3 m/sec, battery voltage is 4.9volts, temperature is 35.1 deg and MSL altitude is 490.7m.

- **AHRS:** The AHRS data is a 36 bytes packet sent every 10msec as shown below in Table below:

Table 6: AHRS Packet Data

Parameters	Size (each)	Update Rate
Roll, Pitch, Yaw angles	2 Bytes	100Hz
Roll, Pitch, Yaw angular rates	2 Bytes	100Hz
North, East, Down Velocity	2 Bytes	100Hz
Latitude, Longitude	4 Bytes	4Hz
Altitude	2 Bytes	4Hz
Header and Checksum	8 Bytes	100Hz

Mission Computer

This block is C++ based s-function, same as programmed in actual flight control computer. The main calling functions are tabulated in Table 7.

Table 7: Mission computer calling functions

Function	Description
void INITIALIZE();	To initialize some parameters.
void GETSENSORDATA();	Check sensor FIFO and get data if available
void MISSION();	Get mission current waypoints
void LATCONT();	Lateral Guidance function, output ϕ_{ref} at every 0.25sec
void LONGCONT();	Longitudinal Guidance function, output θ_{ref} at every 0.1sec
void CONTROL();	Inner loop control, output δ_a, δ_e, δ_r at every 20msec
void PWM();	PWM for actuators
void TELMS();	Send data to data logger at 50Hz and a small packet to ground control station at 4 Hz

Appendix-B: Inner Control Loop Design

Longitudinal Dynamics

$$\begin{aligned}
\dot{u} &= -g\theta\cos\Theta + X_u u + X_{T_u} u + X_\alpha \alpha + X_{\delta_e}\delta_e \\
U_1\dot{\alpha} - U_1\dot{\theta} &= -g\theta\sin\Theta + Z_u u + Z_\alpha\alpha + Z_{\dot{\alpha}}\dot{\alpha} + Z_q\dot{\theta} + Z_{\delta_e}\delta_e \\
\ddot{\theta} &= M_u u + M_{T_u} u + M_\alpha\alpha + M_{\dot{\alpha}}\dot{\alpha} + M_q\dot{\theta} + M_{\delta_e}\delta_e
\end{aligned}$$

For straigh and level flight condition, $q = \dot{\theta}$ and $\dot{q} = \ddot{\theta}$.

$$\begin{aligned}
\dot{u} &= -g\theta\cos\Theta + X_u u + X_{T_u} u + X_\alpha \alpha + X_{\delta_e}\delta_e \\
U_1\dot{\alpha} - Z_{\dot{\alpha}}\dot{\alpha} &= -g\theta\sin\Theta + Z_u u + Z_\alpha\alpha + U_1 q + Z_q q + Z_{\delta_e}\delta_e \\
\dot{q} &= M_u u + M_{T_u} u + M_\alpha\alpha + M_{\dot{\alpha}}\dot{\alpha} + M_q q + M_{\delta_e}\delta_e \\
\dot{\theta} &= q
\end{aligned}$$

In Matrix form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (U_1 - Z_{\dot{\alpha}}) & 0 & 0 \\ 0 & -M_{\dot{\alpha}} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u + X_{T_u} & X_\alpha & 0 & -g\cos\Theta \\ Z_u & Z_\alpha & U_1 + Z_q & -g\sin\Theta \\ M_u + M_{T_u} & M_\alpha + M_{T_\alpha} & M_q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ \alpha \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix} \begin{bmatrix} \delta_e \end{bmatrix}$$

Two more states are introduced to design the outer altitude control i.e., h and $\dot{h} = V_h$.

$$\begin{aligned}
\dot{V}_h &= -Z_u u - Z_\alpha\alpha - Z_q q + g\sin\Theta\theta \\
\dot{h} &= V_h
\end{aligned}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & (U_1 - Z_{\dot{\alpha}}) & 0 & 0 & 0 & 0 \\ 0 & -M_{\dot{\alpha}} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \\ \dot{V}_h \\ \dot{h} \end{bmatrix}$$

$$= \begin{bmatrix} X_u + X_{T_u} & X_\alpha & 0 & -g\cos\Theta & 0 & 0 \\ Z_u & Z_\alpha & U_1 + Z_q & -g\sin\Theta & 0 & 0 \\ M_u + M_{T_u} & M_\alpha + M_{T_\alpha} & M_q & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -Z_u & -Z_\alpha & -Z_q & g\sin\Theta & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ \alpha \\ q \\ \theta \\ V_h \\ h \end{bmatrix} + \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \\ Z_{\delta_e} \\ 0 \end{bmatrix} \begin{bmatrix} \delta_e \end{bmatrix}$$

For YAK-54 (Table 8):

$$\begin{bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \\ \dot{V}_h \\ \dot{h} \end{bmatrix} = \begin{bmatrix} -0.2384 & 12.45 & 0 & -32.13 & 0 & 0 \\ -0.004271 & -7.821 & 0.9228 & -0.009321 & 0 & 0 \\ 0.01634 & -19.18 & -15.14 & 0.03067 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.5142 & 941.5 & 6.921 & 1.122 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ \alpha \\ q \\ \theta \\ V_h \\ h \end{bmatrix} + \begin{bmatrix} 0 \\ -0.01128 \\ -1.842 \\ 0 \\ 1.358 \\ 0 \end{bmatrix} \begin{bmatrix} \delta_e \end{bmatrix}$$

Transfer function from input δ_e [deg] to output θ [deg]:

$$\frac{\theta}{\delta_e} = \frac{-105.5s^2 - 838.2s - 199.6}{s^4 + 23.2s^3 + 141.6s^2 + 33.17s + 6.639}$$

Actuator Transfer Function:

$$\frac{\delta_e}{\delta_{e_c}} = \frac{729}{s^2 + 48s + 729}$$

A filter was designed to remove noise from measured θ:

$$Filter_\theta = \frac{100^2}{s^2 + 120s + 100^2}$$

Designed lead and lag:

$$Klead_\theta = \frac{2.236s + 15}{s + 33.54}, \qquad Klag_\theta = \frac{1.5s + 1.5}{s + 0.05}$$

Table 8: Longitudinal Dimensional Stability Derivatives

X_u	$-qS(C_{D_u}+2C_{D_0}/mU$	-0.1487	(ft/sec^2rad)
X_{T_u}	$qS(C_{T_{x_u}}+2C_{T_x})/mU$	-0.0897	$(ft/secrad)$
X_a	$-qS(C_{D_\alpha}-C_{L_0})/m$	12.4544	$(ft/secrad)$
X_{δ_e}	$-qSC_{D_{\delta_e}}/m$	0	(ft/sec^2rad)
Z_u	$-qS(C_{L_u}+2C_{L_0}/mU$	-0.5142	$1/sec^2$
Z_α	$-qS(C_{L_u}+2C_{L_0}/m$	-941.5051	$1/sec$
$Z_{\dot{\alpha}}$	$qSb^2C_{l_r}/(2I_{xx}U)$	-2.3785	$1/sec$
Z_q	$qSbC_{l_{\delta_a}}/I_{xx}$	-6.9206	$1/sec^2$
Z_{δ_e}	$qSbC_{l_{\delta_r}}/I_{xx}$	-77.8039	$1/sec^2$
M_u	$qSbC_{n_\beta}/I_{zz}$	$4.15e^{-4}$	$1/sec^2$
M_{T_u}	$qSbC_{n_{\delta_a}}/I_{zz}$	0.0019	$1/sec^2$
M_α	$qSb^2C_{n_r}/(2I_{zz}U)$	-48.2893	$1/sec$
M_{T_α}	$qSbC_{n_{T_\beta}}/I_{zz}$	-3.3730	$1/sec^2$
$M_{\dot{\alpha}}$	$qSbC_{n_{T_\beta}}/I_{zz}$	-3.2909	$1/sec^2$
M_q	$qSbC_{n_{T_\beta}}/I_{zz}$	-12.1051	$1/sec^2$
M_{δ_e}	$qSbC_{n_{T_\beta}}/I_{zz}$	-107.6665	$1/sec^2$

Table 9: Eigenvalues of $\frac{\theta}{\delta_e}$

Eigenvalue	Damping	Freq. (rad/s)
$-0.118+0.187i$	0.533	0.221
$-0.118-0.187i$	0.533	0.221
$-11.5+2.08i$	0.984	11.7
$-11.5-2.08i$	0.984	11.7

Basic structure of implementation is shown in Figure 6, $Filter_\theta$ and $Klead_\theta$ are placed in measured θ path.

$$K1 = Klead_\theta \times Filter_\theta = \frac{2.236e004s + 150000}{s^3 + 153.5s^2 + 1.402e004s + 3.354e005}$$

$$K1_z = \frac{0.2675z^3 + 0.2848z^2 - 0.2327z - 0.2501}{z^3 - 1.524z^2 + 0.9293z - 0.2504}$$

$$Klag_z = \frac{1.507z - 1.492}{z - 0.9995}$$

Outer height Controller to generate θ_{ref} is just a lag controller and filter as:

$$Klag_{alt} = \frac{0.6s + 0.06}{s + 0.01} \qquad Filter_{alt} = \frac{8}{s + 8}$$

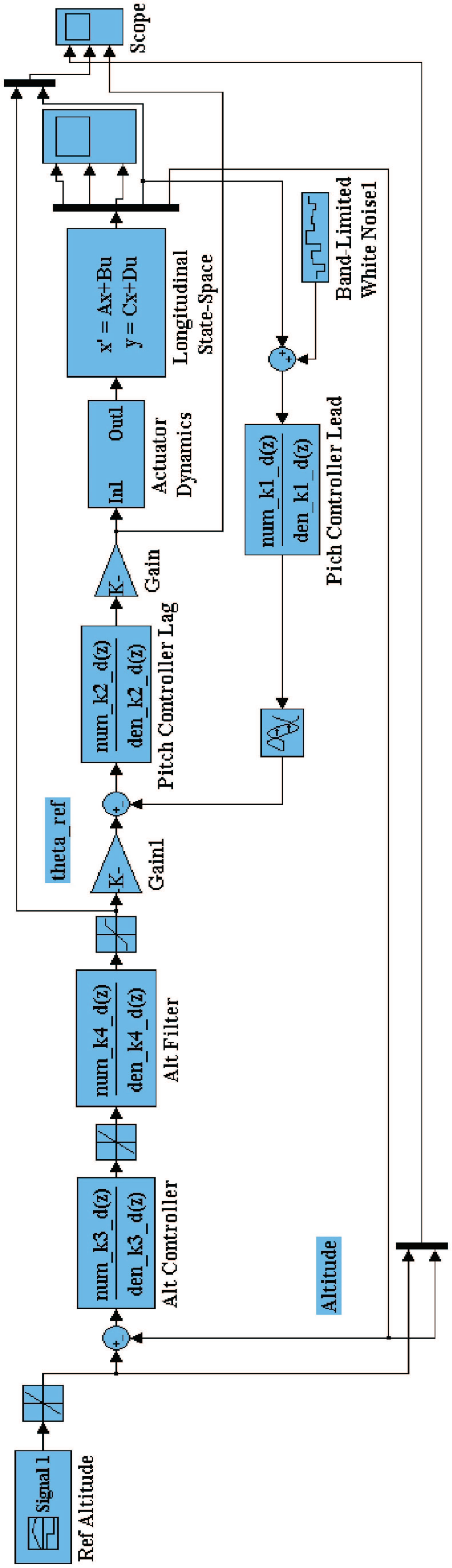


Figure 6: Simulink block diagram for longitudinal control.

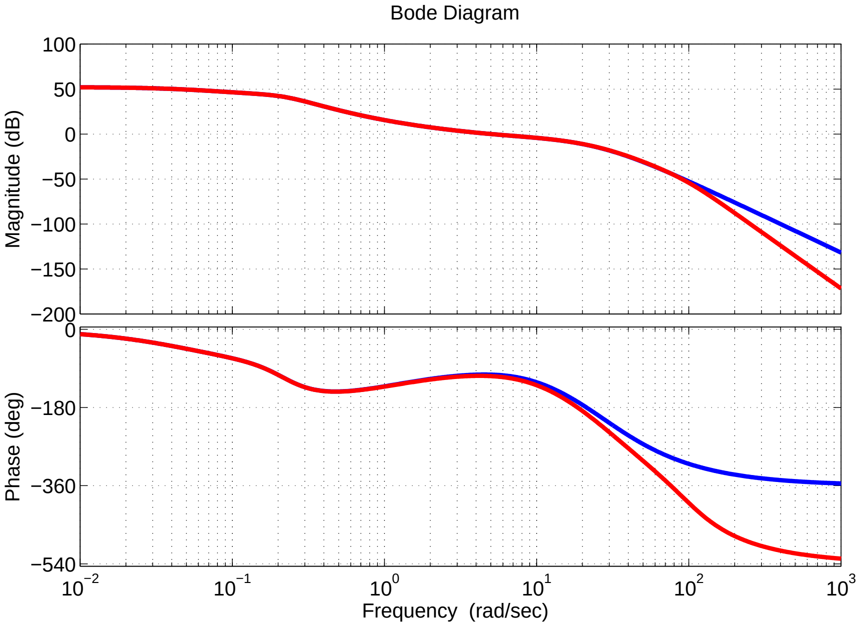


Figure 7: Bode response of $\frac{\theta}{\delta_e}$ with designed Lead-Lag controller with and without filter

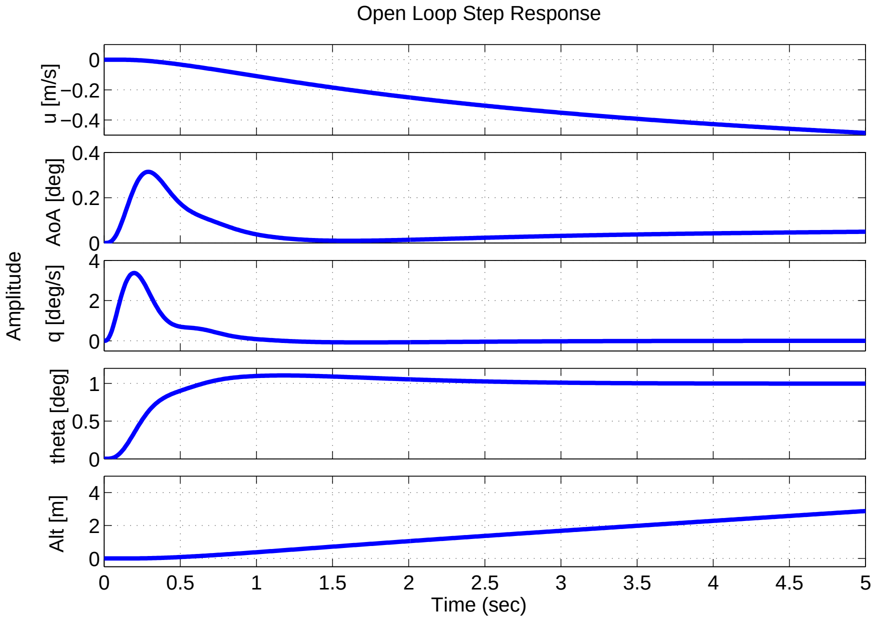


Figure 8: Step response of closed inner loop longitudinal system

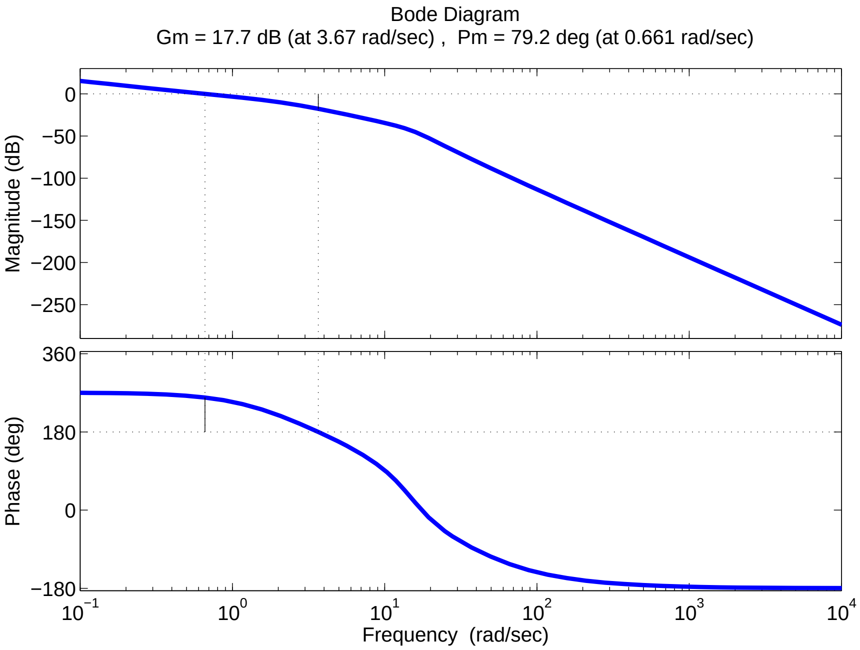


Figure 9: Bode response of $\frac{\theta_r}{h}$ with closed inner loop

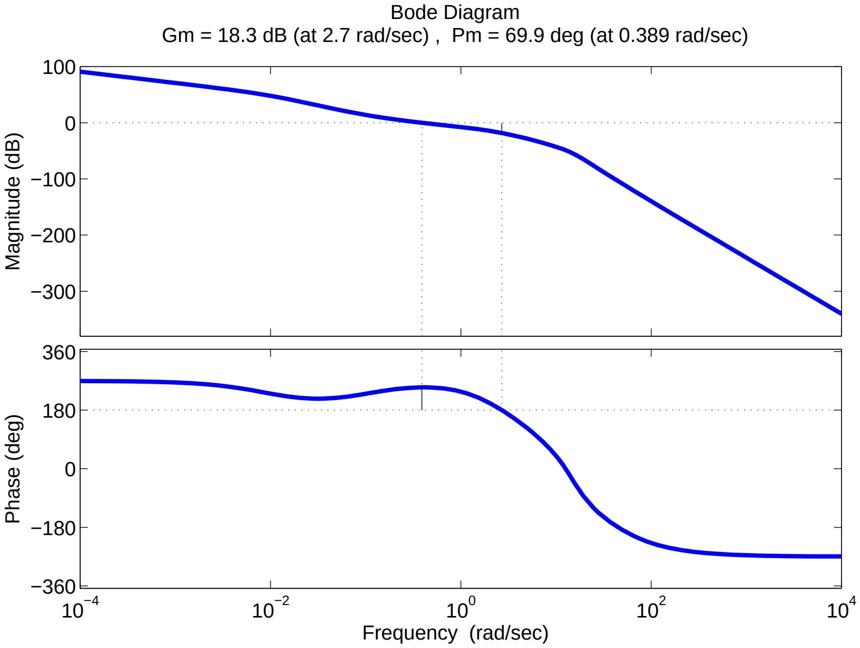


Figure 10: Bode response with designed controller

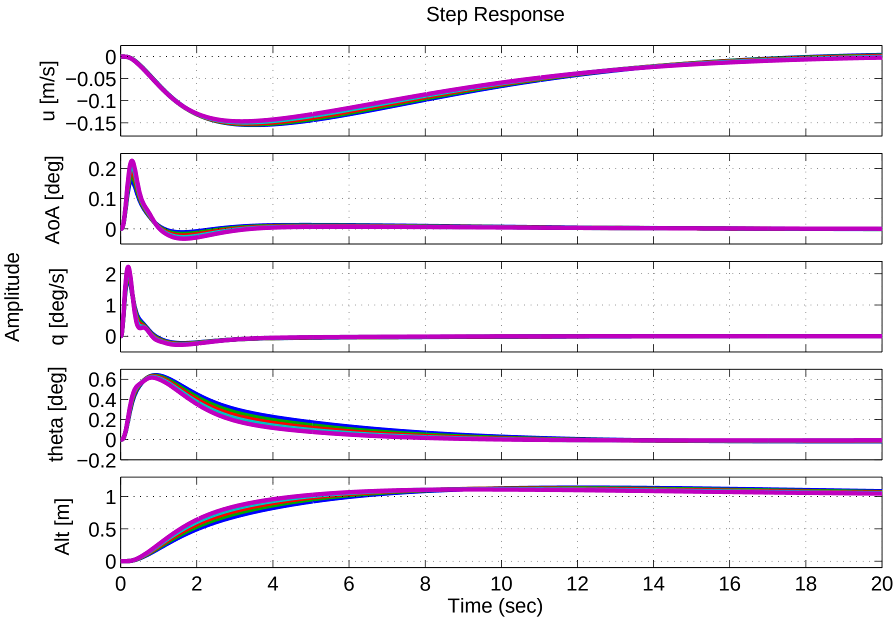


FIGURE 11: Step response of complete close-loop longitudinal system for different velocities

Lateral Dynamics

$$\begin{aligned}
U_1\dot{\beta} + U_1\dot{\psi} &= g\phi\cos\Theta + Y_\beta\beta + Y_p\dot{\phi} + Y_r\dot{\psi} + Y_{\delta_a}\delta_a + Y_{\delta_r}\delta_r \\
\ddot{\phi} - \bar{A}_1\ddot{\psi} &= L_\beta\beta + L_p\dot{\phi} + L_r\dot{\psi} + L_{\delta_a}\delta_a + L_{\delta_r}\delta_r \\
\ddot{\psi} - \bar{B}_1\ddot{\phi} &= N_\beta\beta + N_{T\beta}\beta + N_p\dot{\phi} + N_r\dot{\psi} + N_{\delta_a}\delta_a + N_{\delta_r}\delta_r
\end{aligned}$$

Where

$$\bar{A}_1 = \frac{I_{xz}}{I_{xx}}, \qquad \bar{B}_1 = \frac{I_{xz}}{I_{zz}}$$

Let us take:

$$p = \dot{\phi}, \quad \dot{p} = \ddot{\phi}, \quad r = \dot{\psi}, \quad \dot{r} = \ddot{\psi}$$

$$\begin{aligned}
U_1\dot{\beta} + U_1 r &= g\cos\Theta + Y_\beta\beta + Y_p p + Y_r r + Y_{\delta_a}\delta_a + Y_{\delta_r}\delta_r \\
\dot{p} - \bar{A}_1\dot{r} &= L_\beta\beta + L_p p + L_r r + L_{\delta_a}\delta_a + L_{\delta_r}\delta_r \\
\dot{r} - \bar{B}_1\dot{p} &= N_\beta\beta + N_{T\beta}\beta + N_p p + N_r r + N_{\delta_a}\delta_a + N_{\delta_r}\delta_r
\end{aligned}$$

Rearranging for state space

$$\begin{aligned}
\dot{p} - \bar{A}_1\dot{r} &= L_\beta\beta + L_p p + L_r r + L_{\delta_a}\delta_a + L_{\delta_r}\delta_r \\
\dot{\phi} &= p \\
U_1\dot{\beta} + U_1 r &= g\cos\Theta + Y_\beta\beta + Y_p p + Y_r r + Y_{\delta_a}\delta_a + Y_{\delta_r}\delta_r \\
\dot{r} - \bar{B}_1\dot{p} &= N_\beta\beta + N_{T\beta}\beta + N_p p + N_r r + N_{\delta_a}\delta_a + N_{\delta_r}\delta_r
\end{aligned}$$

In Matrices for it can be written as

$$\begin{bmatrix} 1 & 0 & 0 & -\bar{A}_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & U_1 & 0 \\ -\bar{B}_1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{\phi} \\ \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L_p & 0 & L_\beta & L_r \\ 1 & 0 & 0 & 0 \\ Y_p & g\cos\Theta & Y_\beta & (Y_r - U_1) \\ N_p & 0 & (N_\beta + N_{T\beta}) & N_r \end{bmatrix} \begin{bmatrix} p \\ \phi \\ \beta \\ r \end{bmatrix} + \begin{bmatrix} L_{\delta_a} & L_{\delta_r} \\ 0 & 0 \\ Y_{\delta_a} & Y_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta r \end{bmatrix}$$

Since cross inertial terms are 0 i.e., $I_{xz} = 0$, therefore $\bar{A}_1 = 0$ and $\bar{B}_1 = 0$,

$$\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned}$$

TABLE 10: Lateral-Directional Dimensional Stability Derivatives

Y_β	qSC_{Y_β}/m	354.8161	(ft/sec^2rad)
Y_p	$qSbC_{Y_p}/2mU$	0.1337	$(ft/secrad)$
Y_r	$qSbC_{Y_p}/2mU$	1.7443	$(ft/secrad)$
Y_{δ_a}	$qSC_{Y_{\delta_a}}/m$	0	(ft/sec^2rad)
Y_{δ_r}	$qSC_{Y_{\delta_a}}/m$	45.7139	(ft/sec^2rad)
L_β	$qSbC_{l_\beta}/I_{xx}$	-28.6271	$1/sec^2$
L_p	$qSb^2C_{l_p}/(2I_{xx}U)$	-25.6039	$1/sec$
L_r	$qSb^2C_{l_r}/(2I_{xx}U)$	3.2475	$1/sec$
L_{δ_a}	$qSbC_{l_{\delta_a}}/I_{xx}$	482.3681	$1/sec^2$
L_{δ_r}	$qSbC_{l_{\delta_r}}/I_{xx}$	28.4971	$1/sec^2$
N_β	$qSbC_{n_\beta}/I_{zz}$	49.0483	$1/sec^2$
N_p	$qSb^2C_{n_p}/(2I_{zz}U)$	-0.6061	$1/sec$
N_r	$qSb^2C_{n_r}/(2I_{zz}U)$	-4.5259	$1/sec$
N_{δ_a}	$qSbC_{n_{\delta_a}}/I_{zz}$	-4.1029	$1/sec^2$
N_{δ_r}	$qSbC_{n_{\delta_r}}/I_{zz}$	-46.7637	$1/sec^2$
N_{T_β}	$qSbC_{n_{T_\beta}}/I_{zz}$	-2.0981	$1/sec^2$

where $x = [p, \phi, \beta, r]^t$ defines the states and $u = [\delta_a, \delta_r]^t$ defines the inputs

$$A = \begin{bmatrix} -25.6040 & 0 & -28.6271 & 3.2475 \\ 1 & 0 & 0 & 0 \\ 0.0011 & 0.2718 & 3.0069 & -0.9852 \\ -0.6061 & 0 & 46.9502 & -4.5259 \end{bmatrix}, \quad B = \begin{bmatrix} 482.3682 & 28.4971 \\ 0 & 0 \\ 0 & 0.3874 \\ -4.1029 & -46.7637 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\frac{\phi}{\delta_a} = \frac{482.4s^2 + 719.4s + 1.567e004}{s^4 + 27.12s^3 + 73.54s^2 + 854.8s - 6.226}$$

A filter was designed to remove noise from measured ϕ:

$$Filter_\phi = \frac{4900}{s^2 + 70s + 4900}$$

Designed Lead and Lag Control:

$$Klead_\phi = \frac{2.646s + 20}{s + 52.92}, \qquad Klag_\phi = \frac{s + 1}{s + 0.1}$$

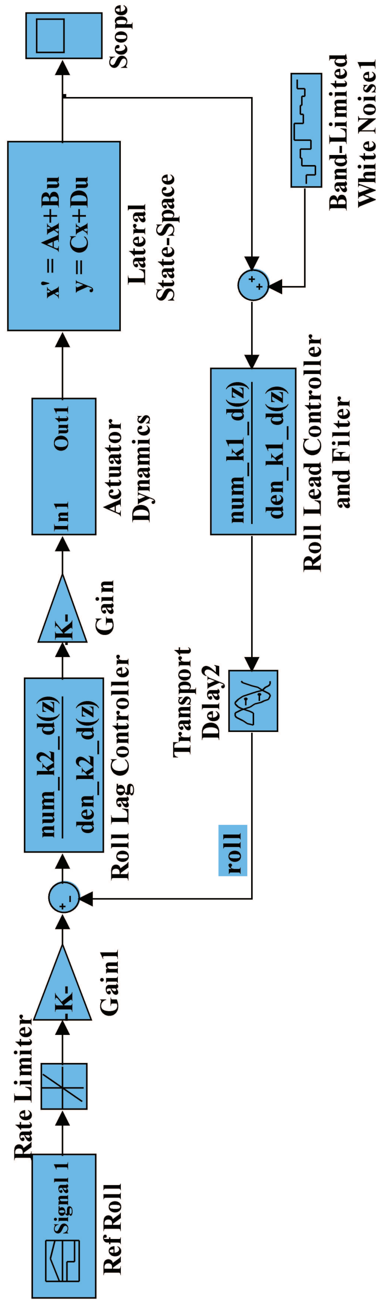


Figure 12: Simulink block diagram for inner roll control

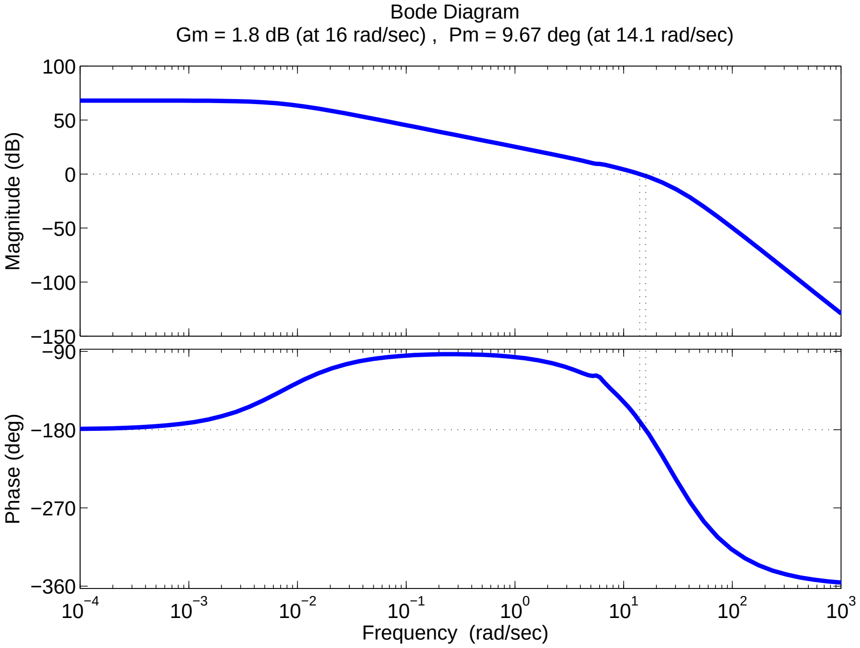


FIGURE 13: Bode response of $\frac{\phi}{\delta_a}$

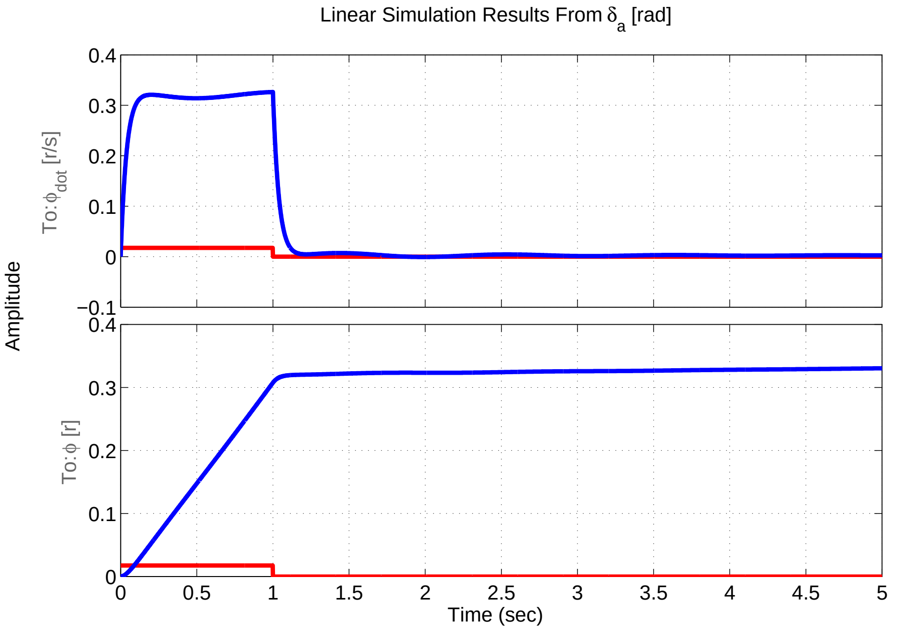


FIGURE 14: Linear simulation results for open loop Lateral Dynamics from Input δ_a

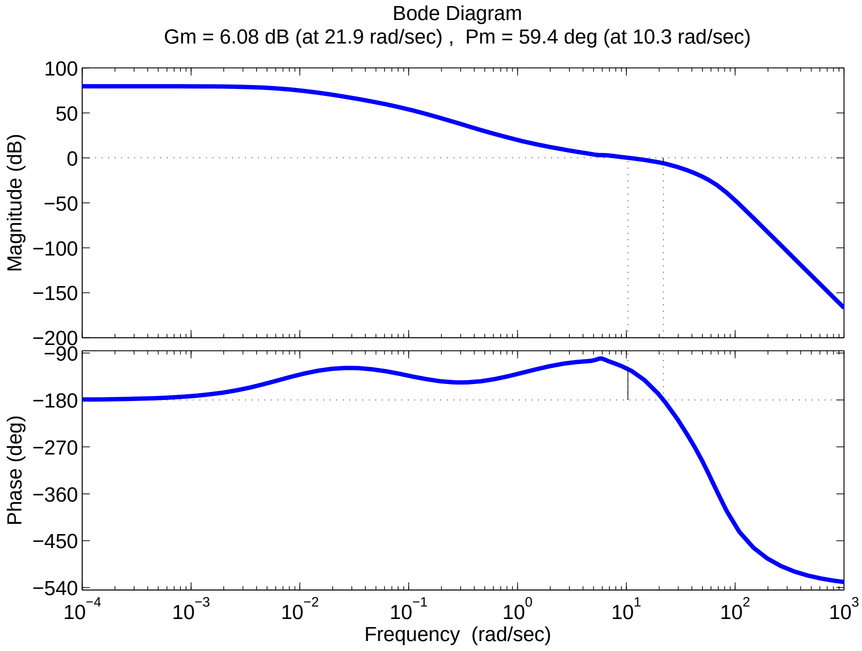


FIGURE 15: Bode Response of $\frac{\phi}{\delta_a}$ with designed Klead-Klag controller along with actuator dynamics and filter

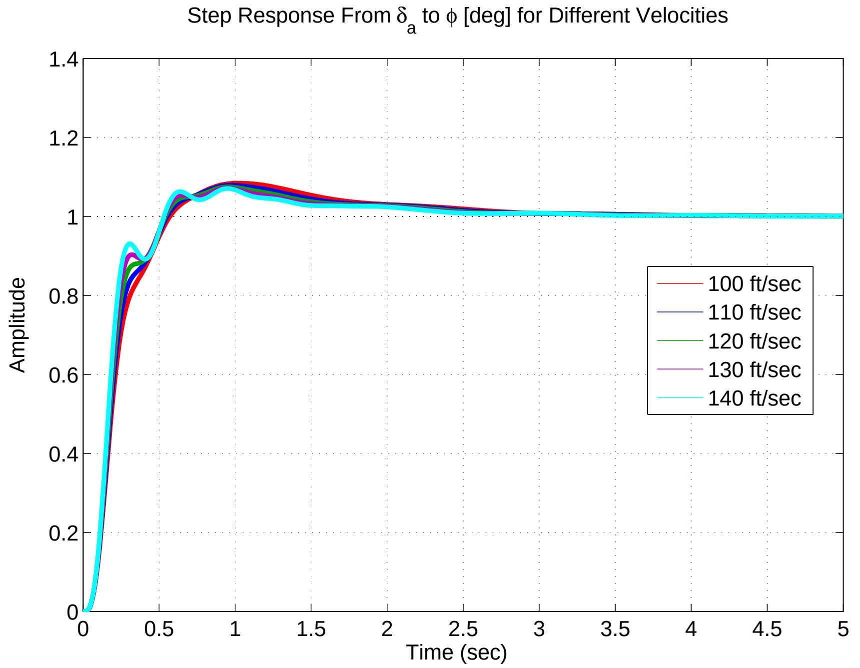
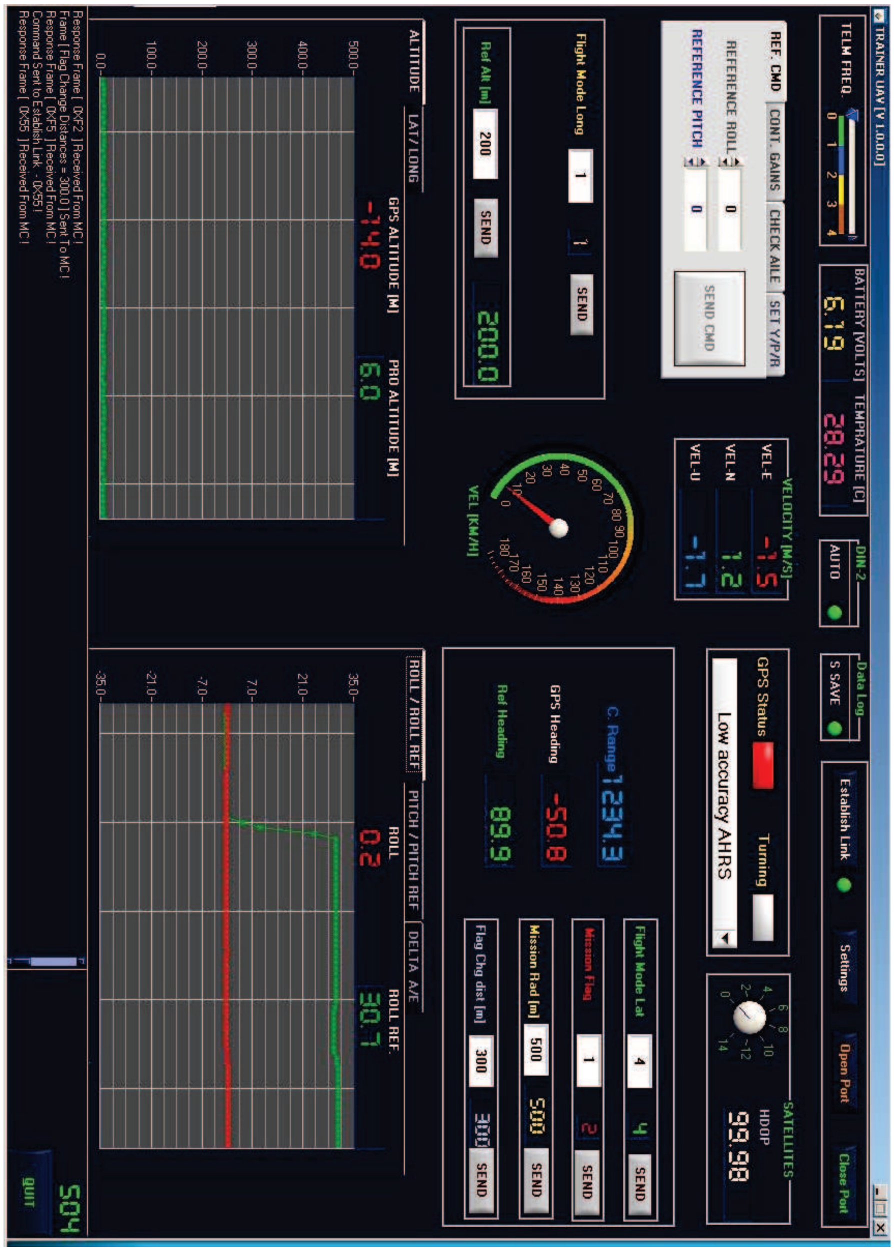


FIGURE 16: Step response for different Velocities

Appendix-C: Flight Results for Altitude Controller



REFERENCE ROLL 0

REFERENCE PITCH 0

Low accuracy AHRS

Flight Mode Long 1 1

Ref Alt [m] 200 200.0

Flight Mode Lat 4 4

Mission Flag 1 2

Mission Rad [m] 500 500

Flag Chg dist [m] 300 300

```
Response Frame [  0XF2  ] Received From MC !
Frame [ Flag Change Distances = 300.0 ] Sent To MC !
Response Frame [  0XF5  ] Received From MC !
Command Sent to Establish Link  - 0X55 !
Response Frame [  0X55  ] Received From MC !
```

Figure 17: Ground control Station

Figures 18-25 show the longitudinal flight test results for the two flight tests.

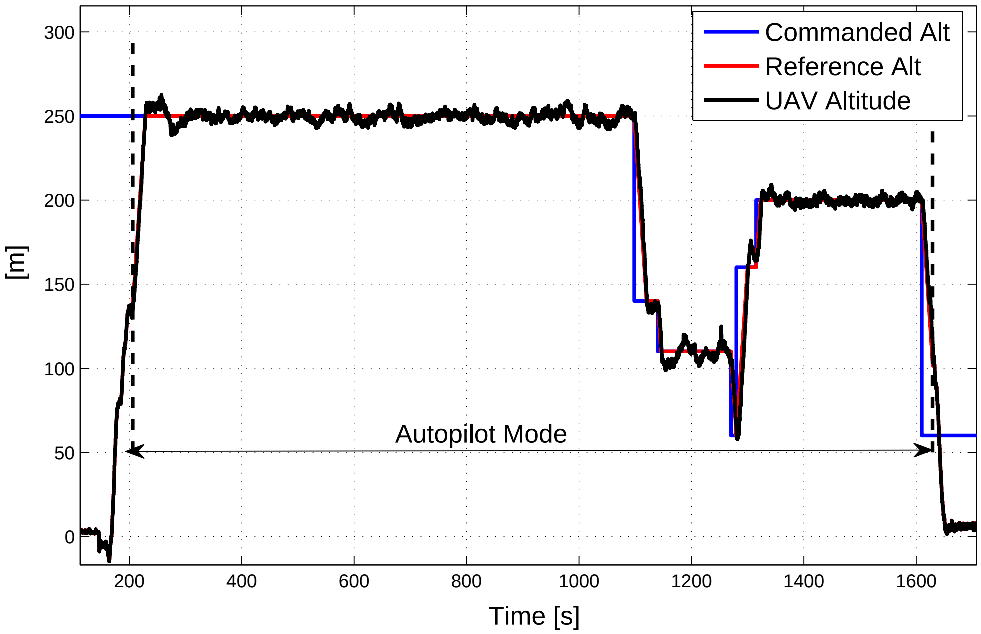


FIGURE 18: Flight-1: Altitude tracking for different commands.

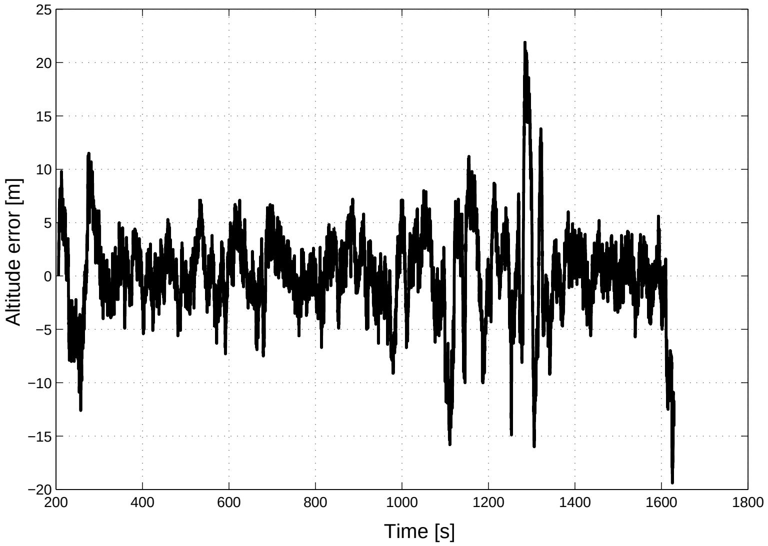


FIGURE 19: Flight-1: Altitude error during flight.

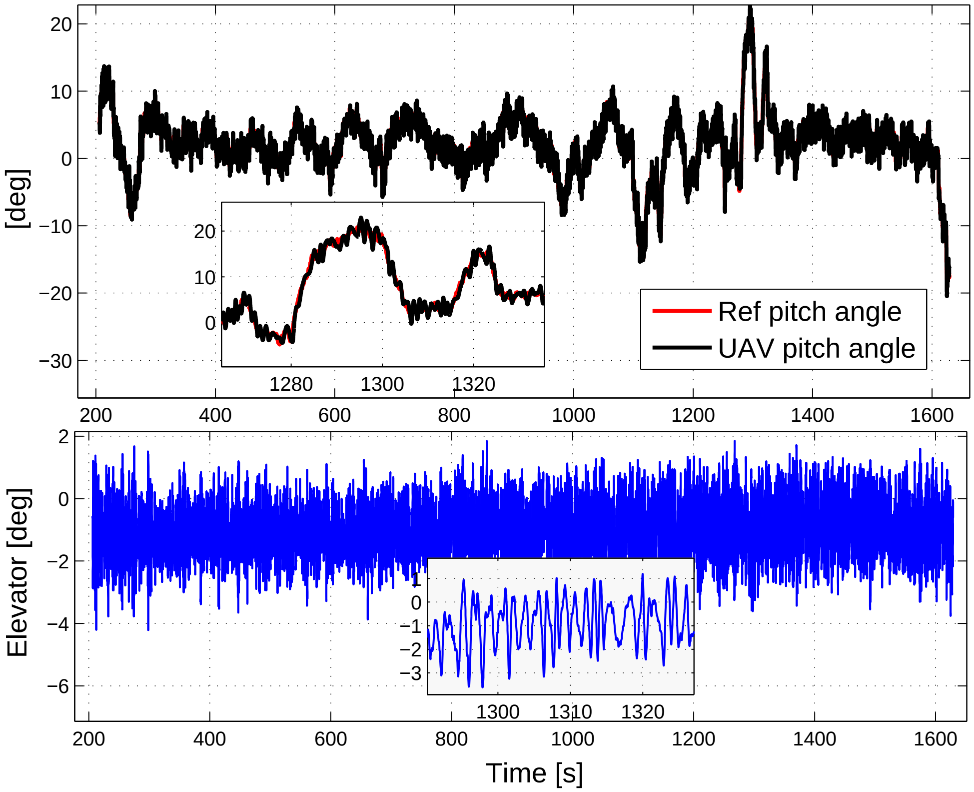


FIGURE 20: Flight-1: Actual and reference pitch angle versus time.

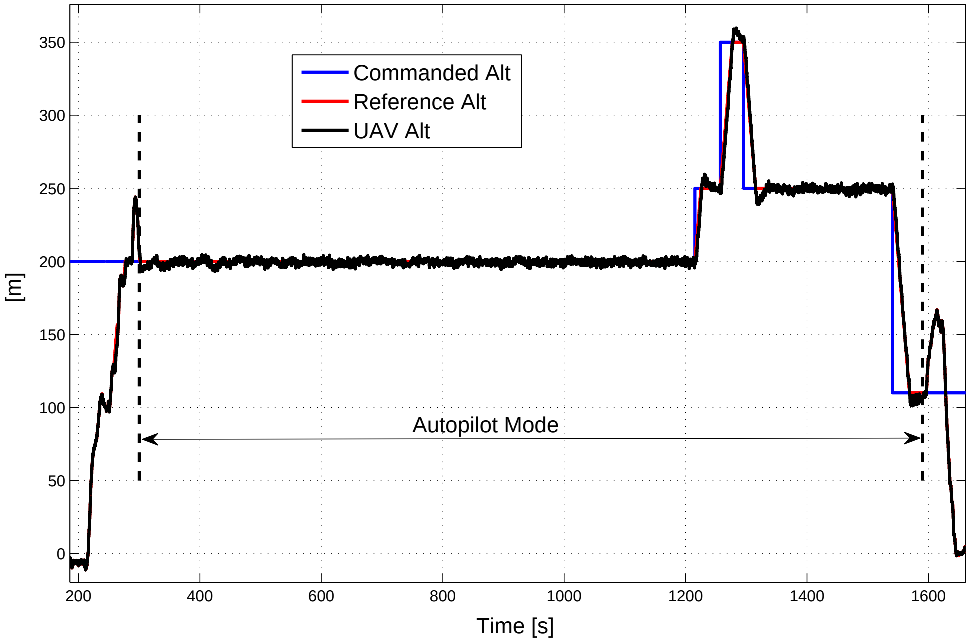


FIGURE 21: Flight-2: Altitude tracking for different commands.

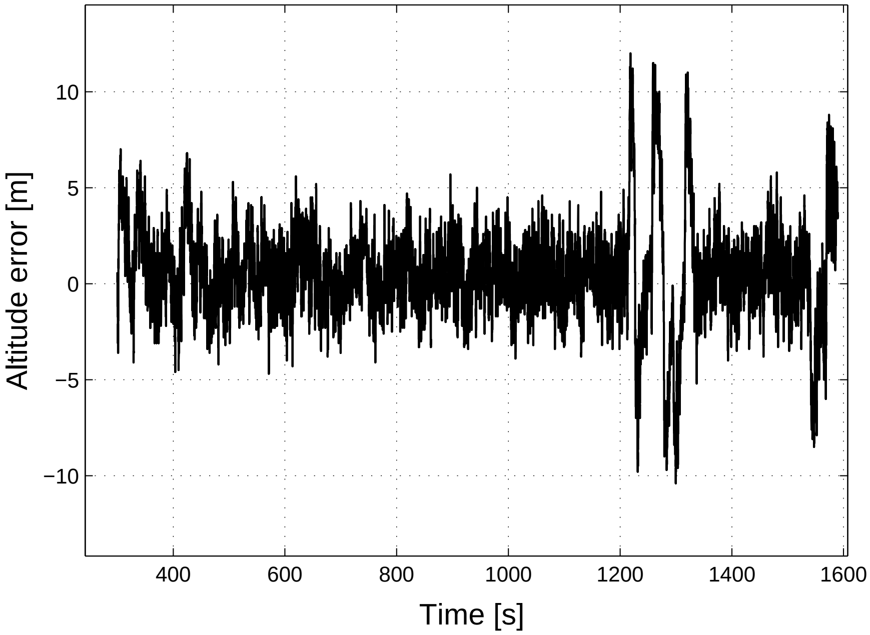


Figure 22: Flight-2: Altitude error during flight.

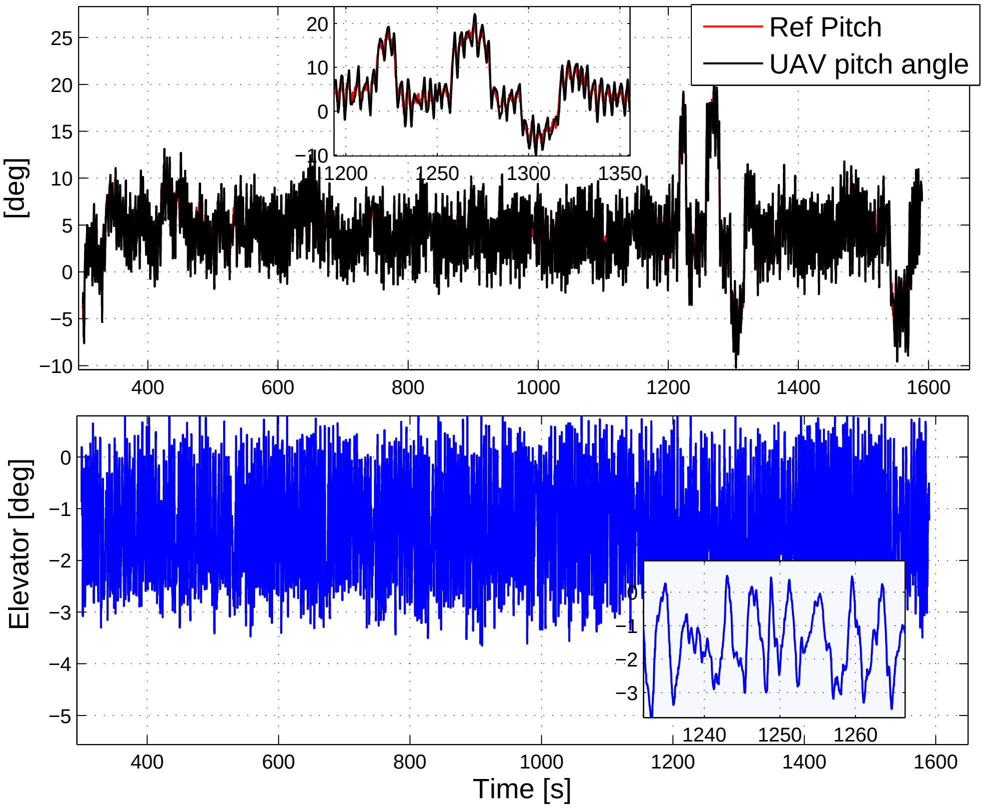


Figure 23: Flight-2: Actual and reference and pitch angle versus time.

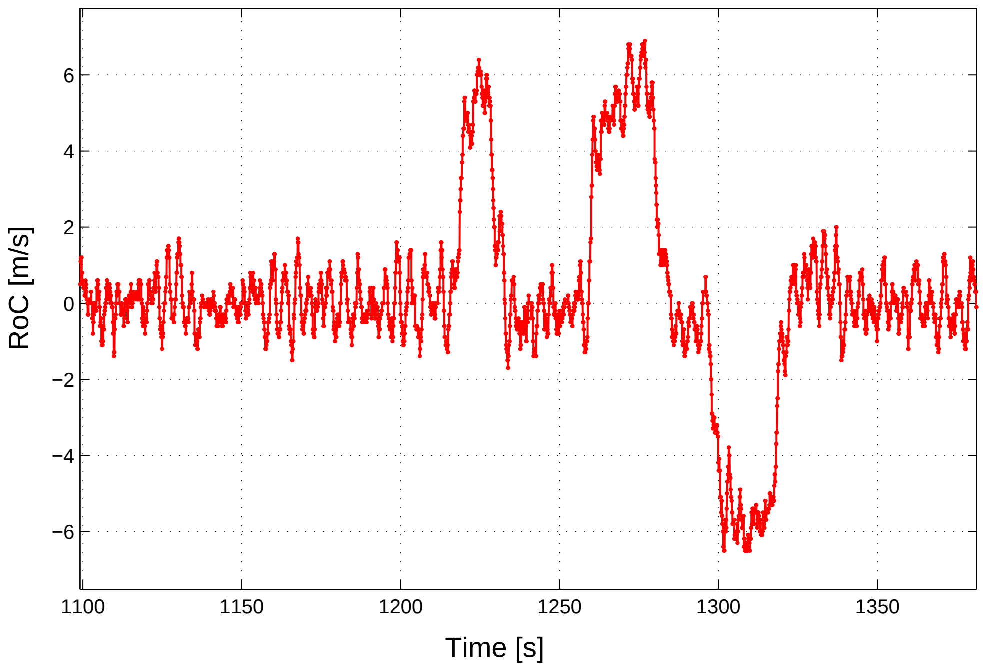


FIGURE 24: Flight-2: Rate of climb/decent during flight versus time.

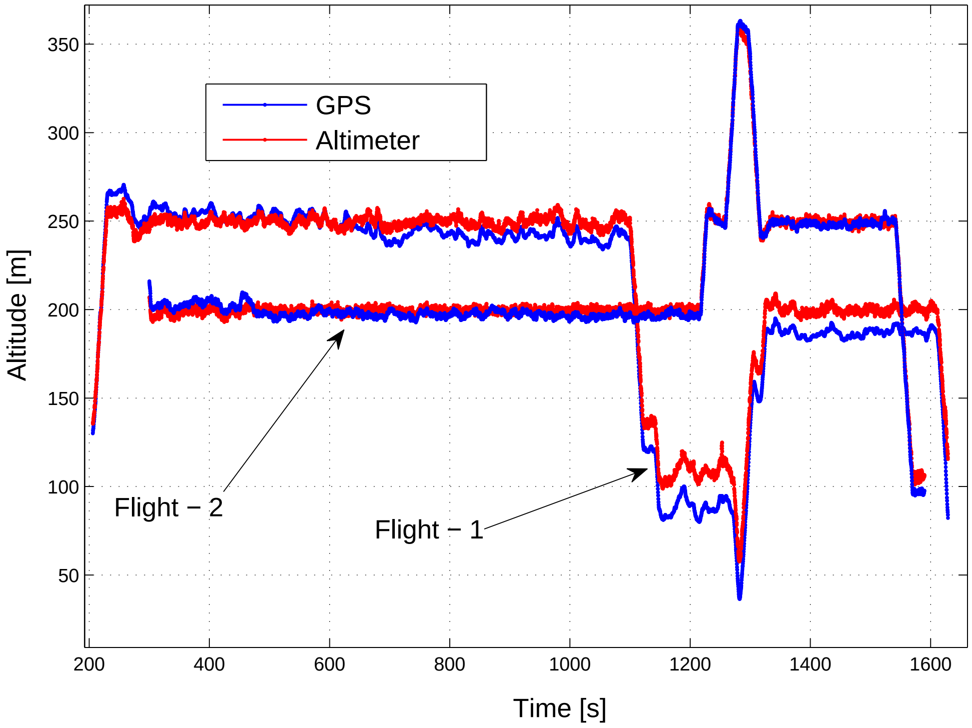


FIGURE 25: GPS and altimeter output (altitude) for Flights-1 & 2.

REFERENCES

Ahmed, M., Subbarao, K., Dec 2010. Nonlinear 3-d trajectory guidance for unmanned aerial vehicles. In: Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on. pp. 1923–1927.

Alessandretti, A., Aguiar, A., Jones, C., July 2013. Trajectory-tracking and path-following controllers for constrained underactuated vehicles using model predictive control. In: Control Conference (ECC), 2013 European. pp. 1371–1376.

Ambrosino, G., Ariola, M., Ciniglio, U., Corraro, F., De Lellis, E., Pironti, A., July 2009. Path generation and tracking in 3-d for UAVs. Control Systems Technology, IEEE Transactions on 17 (4), 980–988.

Bandyopadhyay, B., Deepak, F., Kim, K.-S., 2009. Sliding Mode Control Using Novel Sliding Surfaces. Springer, Springer-Verlag Berlin Heidelberg.

Bandyopadhyay, B., Janardhanan, S., 2006. Discrete-time Sliding Mode Control: A Multirate Output Feedback Approach. Springer, Springer-Verlag Berlin Heidelberg.

Baralli, F., Pollini, L., Innocenti, M., 2002. Waypoint-based fuzzy guidance for unmanned aircraft: a new approach. In: AIAA Guidance, Navigation, and Control Conference and Exhibition.

Beard, R. W., McLain, T. W., 2012. Small Unmanned Aircraft: Theory and Practice. Princeton University Press.

Boiko, I., Jun. 2013. Chattering in sliding mode control systems with boundary layer approximation of discontinuous control. Intern. J. Syst. Sci. 44 (6), 1126–1133.

Breivik, M., Fossen, T., 2005. Principles of guidance-based path following in 2d and 3d. In: Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on. pp. 627–634.

Brezoescu, A., Espinoza, T., Castillo, P., Lozano, R., 2013. Adaptive trajectory following for a fixed-wing UAV in presence of crosswind. Journal of Intelligent & Robotic Systems 69 (1-4), 257–271.

Brhaug, E., Pavlov, A., Panteley, E., Pettersen, K. Y., 2011. Straight line path following for formations of underactuated marine surface vessels. IEEE Transactions on Control Systems Technology 19 (3), 493–506.

Burton, J. A., Zinober, A. S. I., 1986. Continuous approximation of variable structure control. International Journal of Systems Science 17 (6), 875–885.

C., Y., Y., H., 2000. Elimination of reaching phase from variable structure control. Journal of Dynamic Systems, Measurement and Control 122.

Capello, E., Guglieri, G., Marguerettaz, P., Quagliotti, F., 2012. Preliminary assessment of flying and handling qualities of mini-UAVs. Journal of Intelligent and Robotic Systems 65 (1), 43–61.

Ceccarelli, N., Enright, J., Frazzoli, E., Rasmussen, S., Schumacher, C., 2007. Micro UAV path planning for reconnaissance in wind. In: American Control Conference. pp. 5310–5315.

Cesetti, A., Frontoni, E., Mancini, A., Zingaretti, P., Longhi, S., 2009. A vision-based guidance system for UAV navigation and safe landing using natural landmarks. Journal of Intelligent and Robotic Systems 57 (1-4), 233–257.

Chen, Y.-Y., Chen, Y.-L., Zhou, B.-H., June 2014. Robust guidance law design for UAVs. In: Control Automation (ICCA), 11th IEEE International Conference on. pp. 44–49.

Choi S.-B., P. D.-W., S., J., 1994. A time-varying sliding surface for fast and robust tracking control of second order uncertain systems. Automatica 30 (5).

Cunha, R., Silvestre, C., 2005. A 3d path-following velocity-tracking controller for autonomous vehicles. In: Proceedings of the 16th IFAC World Congress, 2005.

D. R. Nelson, D. B. Barber, T. W. M., Beard, R. W., 2007. Vector field path following for miniature air vehicles. IEEE Transactions on Robotics 23 (3).

Dadkhah, N., Mettler, B., 2012. Survey of motion planning literature in the presence of uncertainty: Considerations for UAV guidance. Journal of Intelligent and Robotic Systems 65 (1), 233–246.

De Filippis, L., Guglieri, G., Quagliotti, F., 2012. A novel approach for trajectory tracking of UAVs. In: 2nd EASN workshop on Flight Physics and Propulsion.

Deyst, J., How, J., Park, S., 15-18 August 2005. Lyapunov stability of a nonlinear guidance law for UAVs. In: AIAA Atmospheric Flight Mechanics Conference and Exhibit. San Francisco, CA.

Durmaz, B., Ozgoren, M. K., Salamci, M. U., 2012. Sliding mode control for non-linear systems with adaptive sliding surfaces. Transactions of the Institute of Measurement and Control 34 (-), 56–90.

Edwards, C., Spurgeon, S. K., 1998. Sliding Mode Control: Theory and Applications. Taylor and Francis.

Gao, W., Hung, J. C., 1993. Variable structure control of nonlinear systems: A new approach. IEEE Transactions on Industrial Electronics 40 (1).

Garcia, G., Keshmiri, S., 2011. Nonlinear model predictive controller for navigation, guidance and control of a fixed-wing UAV. In: AIAA Guidance, Navigation, and Control Conference.

Gleason, S., Gebre-Egziabher, D., 2009. GNSS Applications and Methods. ISBN-13: 978-1596933293.

Goerzen, C., Kong, Z., Mettler, B., 2009. A survey of motion planning algorithms from the perspective of autonomous UAV guidance. Journal of Intelligent and Robotic Systems 57 (1-4), 65–100.

Griffiths, S., 21-24 August 2006. Vector-field approach for curved path following for miniature aerial vehicles. In: AIAA Guidance, Navigation, and Control Conference and Exhibit. Keystone, Colorado.

Hull, D. G., 2007. Fundamentals of Airplane Flight Mechanics, 1st Edition. Springer Publishing Company, Incorporated.

Jager, R., 2008. Test and Evaluation of the Piccolo II Autopilot System on a One-third Scale Yak-54. University of Kansas.
URL `https://books.google.com.pk/books?id=ZTp9lQ5cZIcC`

Kaminer, I., Pascoal, A., Hallberg, E., Silvestre, C., 1998. Trajectory tracking for autonomous vehicles: An integrated approach to guidance and control. AIAA journal of Guidance, Control, and Dynamics 21 (1), 29–38.

Kaminer, I., Yakimenko, O., Pascoal, A., Ghabcheloo, R., June 2006. Path generation, path following and coordinated control for timecritical missions of multiple UAVs. In: American Control Conference, 2006. pp. 4906–4913.

Kang, Y., Hedrick, J., 2006. Design of nonlinear model predictive controller for a small fixed-wing unmanned aerial vehicle. In: AIAA Guidance, Navigation, and Control Conference and Exhibit.

Kurowski, M., H. A. K. P. e. a., 2015. Guidance, navigation and control of unmanned surface vehicles. Special Issue: Maritime Systems / Jrgen Beyerer. at - Automatisierungstechnik 63 (5).

Levant, A., 1993. Sliding order and sliding accuracy in sliding mode control. International Journal of Control 58 (6), 1247–1263.

Lim, S., Jung, W., Bang, H., May 2014. Vector field guidance for path following and arrival angle control. In: Unmanned Aircraft Systems (ICUAS), 2014 International Conference on. pp. 329–338.

Liu, C., McAree, O., Chen, W.-H., 2012. Path-following control for small fixed-wing unmanned aerial vehicles under wind disturbances. International Journal of Robust and Nonlinear Control 23 (15), 682–1698.

McGee, T., Hedrick, J., 2006. Path planning and control for multiple point surveillance by an unmanned aircraft in wind. In: American Control Conference, 2006.

Mettler, B., Valenti, M., Schouwenaars, T., Kuwata, Y., How, J., Paunicka, J., Feron, E., 11-14 August 2003. Autonomous UAV guidance build-up: Flight-test demonstration and evaluation plan. In: AIAA Guidance, Navigation, and Control Conference and Exhibit. Austin, Texas.

Miele, A., 1962. Flight Mechanics: Theory of Flight Paths. No. 1. c. in Addison-Wesley Series in the engineering sciences. Elsevier Science & Technology.

M.L., C., G., O., 2007. Linear unstable plants with saturating actuators: Robust stabilization by a time varying sliding surface. Automatica 43.

Mohamed, A., Mamatas, A., 2012. Fundamentals of GNSS-Aided Inertial Navigation. Automatic Flight Control Systems - Latest Developments, InTech, DOI: 10.5772/31093.

Niculescu, M., 8-11 January 2001. Lateral track control law for Aerosonde UAV. In: 39th AIAA Aerospace Sciences Meeting and Exhibit. Reno, NV.

Oland, E., Kristiansen, R., 2014. Trajectory tracking of an underactuated fixed-wing UAV. AIP Conference Proceedings 1637 (1).

Osborne, J., Rysdyk, R., 2005. Waypoint guidance for small UAVs in wind. AIAA Infotech Aerospace, 1–12.
URL `http://www.aa.washington.edu/research/afsl/publications/osborne2005waypoint.`

Pappoullis, F. A., 1994. Cross track errors and proportional tuning rate guidance of marine vehicle. Journal of Ship Research 38, 123–132.

Park, S., 2004. Avionics and control system development for mid-air rendezvous of two unmanned aerial vehicles. Ph.D. thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.

Park, S., Deyst, J., How, J., 16-19 August 2004. A new nonlinear guidance logic for trajectory tracking. In: AIAA Guidance, Navigation, and Control Conference and Exhibit. Providence, Rhode Island.

Perruquetti, W., Barbot, J. P., 2002. Sliding Mode Control in Engineering. Marcel Dekker Inc., New York - Basel.

Pettersen, K., Lefeber, E., December 2001. Way-point tracking control of ships. In: Proc. 40^{th} IEEE Conference on Decision and Control. Orlando, FL.

Pisano, W., Lawrence, D., Gray, P., 7-10 May 2007. Autonomous UAV control using a 3-sensor autopilot. In: AIAA Infotech Aerospace 2007 Conference and Exhibit. Rohnert Park, CA, USA.

Poh, E. K., Wang, J. L., Ling, K. V., 2010. Near optimal tracking solution for input constrained UAV using MPC. In: AIAA Guidance, Navigation, and Control Conference.

Ratnoo, A., 2015. Variable deviated pursuit for rendezvous guidance. Journal of Guidance, Control, and Dynamics 38 (4).

Regina, N., Zanzi, M., March 2011. UAV guidance law for ground-based target trajectory tracking and loitering. In: Proceedings of the IEEE Aerospace Conference.

Ren, W., Beard, R. W., 2004. Trajectory tracking for unmanned air vehicles with velocity and heading rate constraints. IEEE Transactions on Control Systems Technology 12 (5), 706–716.

Rohan C. Shekhar, M. K., Shames, I., 2015. Robust model predictive control of unmanned aerial vehicles using waysets. Journal of Guidance, Control, and Dynamics.

Rysdyk, R., September 2003. UAV path following for constant line of sight. In: In AIAA Unmanned Unlimited Systems, Technologies, and Operations Conference. San Diego, CA.

Rysdyk, R., 2006. Unmanned aerial vehicle path following for target observation in wind. Journal of Guidance, Navigation, and Control 29 (5), 1092–1100.

Samar, R., Ahmed, S., Aftab, M. F., 2007. Lateral control with improved performance for UAVs. In: 17^{th} IFAC Symposium on Automatic Control in Aerospace. Toulouse, France.

Samar, R., Ahmed, S., Nzar, M., 2008. Lateral guidance & control design for an unmanned aerial vehicle. In: 17^{th} IFAC World Congress. Seoul.

Sarras, I., Siguerdidjane, H., Oct 2014. On the guidance of a UAV under unknown wind disturbances. In: Control Applications (CCA), 2014 IEEE Conference on. pp. 820–825.

Sasongko, R., Sembiring, J., Muhammad, H., Mulyanto, T., May 2011. Path following system of small unmanned autonomous vehicle for surveillance application. In: Proceedings of the 8^{th} Asian Control Conference.

Schouwenaars, T., Valenti, M., Feron, E., How, J., March 2005. Implementation and flight test results of MILP-based UAV guidance. In: IEEE Aerospace Conference.

Shahriar keshmiri, Hou In Leong, R. J., Hale, R., August 2008. Modeling and simulation of the yak-54 scaled unmanned aerial vehicle using parameter and system identification. In: AIAA Atmospheric Flight Mechanics Conference and Exhibit.

Shtessel, Y., Shkolnikov, I., Brown, M., 2003. An asymptotic second-order smooth sliding mode control. Asian Journal of Control 4 (5), 959–967.

Shtessel, Y. B., Shkolnikov, I. A., Levant, A., 2007. Smooth second-order sliding modes: Missile guidance application. Automatica 43 (8), 1470 – 1476.

Siouris, G. M., 2004. Missile Guidance and Control Systems. Springer-verlag, USA.

Skogestad, S., Postlethwaite, I., 2005. Multivariable Feedback Control Analysis and Design, 2nd Edition. John Wiley & Sons, West Sussex, England.

Slotine, J.-J. E., Li, W., 1991. Applied Nonlinear Control. Prentice Hall, Englewood Cliffs, New Jersey.

Stevens, B., Lewis, F., 2003. Aircraft Control & Simulation, 2nd Edition. Wiley and Sons, Hoboken, New Jersey, USA.

Sujit, P., Saripalli, S., Borges Sousa, J., Feb 2014. Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicles. Control Systems Magazine, IEEE 34 (1), 42–59.

Sujit, P., Saripalli, S., Sousa, J., July 2013. An evaluation of UAV path following algorithms. In: Control Conference (ECC), 2013 European. pp. 3332–3337.

Utkin, V.I., J. G., Shi, J., 1999. Sliding mode control in electro-mechanical systems. Taylor & Francis, London.

Utkin, V. I., 1977. Variable structure systems with sliding modes: a survey. IEEE Transactions on Automatic Control 22, 212–222.

Valavanis, K. P., Vachtsevanos, G. J., 2015. Handbook of Unmanned Aerial Vehicles. Springer, ISBN: 978-90-481-9706-4.

Wise, R., 2006. UAV control and guidance for autonomous cooperative tracking of a moving target. Ph.D. thesis, Department of Aeronautics and Astronautics, University of Washington.

Yamasaki, T., Balakrishnan, S., Takano, H., June 2012. Integrated guidance and autopilot for a path-following UAV via high-order sliding modes. In: American Control Conference (ACC), 2012. pp. 143 –148.

Yamasaki1, T., Balakrishnan, S. N., 2010. Preliminary assessment of flying and handling qualities of mini-UAVs. Journal of Aerospace Engineering 224 (10), 1057–1067.

Yamasakia, T., Balakrishnanb, S., Hallberg, E., Takanoa, H., 2012. Integrated guidance and autopilot design for a chasing UAV via high-order sliding modes. Journal of the Franklin Institute 349 (2), 531–558.

Yang, K., Sukkarieh, S., Kang, Y., 2009. Adaptive nonlinear model predictive path tracking control for a fixed-wing unmanned aerial vehicle. In: AIAA Guidance, Navigation, and Control Conference.

Young, K. D., Utkin, V. I., Ozguner, U., 1999. A control engineers guide to sliding mode control. IEEE Transactions on Control Systems Technology 7 (3), 328–342.

Yu, J., Xu, Q., Zhi, Y., 11-13 March 2011. A TSM control scheme of integrated guidance/autopilot design for UAV. In: 3rd International Conference on Computer Research and Development (ICCRD). Shanghai, China.

Zexu, Z., etc., 2012. Robust sliding mode guidance and control for soft landing on small bodies. Journal of the Franklin Institute 349 (2), 493–509.

Zhang, Z., Li, S., , Luo, S., 2013. Composite guidance laws based on sliding mode control with impact angle constraint and autopilot lag. Transactions of the Institute of Measurement and Control 35 (-), 764–776.

Zhang Yi, Yang Xiuxia, Z. H., Weiwei, Z., 2015. UAV flyable trajectory generation and its tracking control. International Journal of Control and Automation 8 (4).