SYNTHESIS AND HARDWARE IMPLEMENTATION OF AN UNMANNED AERIAL VEHICLE AUTOMATIC LANDING SYSTEM UTILIZING QUANTITATIVE FEEDBACK THEORY

A Thesis

by

TIMOTHY DANIEL WOODBURY

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	John Valasek
Committee Members,	Suman Chakravorty
	Siavkumar Rathinam
Head of Department,	Rodney Bowersox

August 2014

Major Subject: Aerospace Engineering

ABSTRACT

Approach and landing are among the most difficult flight regimes for automatic control of fixed-wing aircraft. Additional challenges are introduced when working with unmanned aerial vehicles, such as modelling uncertainty and limited gust tolerance. This thesis develops linear discrete-time automatic landing controllers using Quantitative Feedback Theory to ensure control robustness and adequate disturbance rejection. Controllers are developed in simulation and evaluated in flight tests of the low cost Easy Star remote-controlled platform. System identification of the larger Pegasus unmanned aerial vehicle is performed to identify dynamic models from flight data. A full set of controllers are subsequently developed and evaluated in simulation for the Pegasus. The extensive simulation and experimental testing with the Easy Star will reduce the time required to implement the Pegasus control laws, and will reduce the associated risk by validating the core experimental software. It is concluded that the control synthesis process using Quantitative Feedback Theory provides robust controllers with generally adequate performance, based on simulation and hardware results. The Quantitative Feedback Theory framework provides a good method for synthesizing the inner-loop controllers and satisfying performance requirements, but in many of the cases considered here it is found to be impractical for the outer loop designs. The primary recommendations of this work are: perform additional verification flights on the Easy Star; repeat Pegasus system identification for a landing configuration before flight testing the control laws; design and implement a rudder control loop on the Pegasus for control of the vehicle after touchdown.

DEDICATION

To Austin, Dakota, and Grant, for helping me survive everything starting with Brazil.

ACKNOWLEDGEMENTS

Funding was provided by the National Science Foundation through the Graduate Research Fellowship Program; this support is gratefully acknowledged. Particular thanks goes to Dr. John Valasek for his guidance and through both my graduate and undergraduate careers; to my other committee members, Dr. Suman Chakravorty and Dr. Sivakumar Rathinam; to Jim Henrickson, for acting as test pilot and moral support during Easy Star flights; to the entire VSCL flight crew, for their hard work in both early mornings and August afternoons; and to Andrew Beckett, whose advice and example have been invaluable in ongoing flight operations at the Riverside Test Range.

NOMENCLATURE

GPS	Global Positioning System
IMU	Inertial Measurement Unit
QFT	Quantitative Feedback Theory
UAV	Unmanned Aerial Vehicle
SISO	Single-input single-output
MIMO	Multiple-input multiple-output
MAV	Micro air vehicle, unmanned air vehicle massing less than 5 kg
DOF	degree-of-freedom
AHRS	Attitude and heading reference system
APM	Ardupilot Mega
OKID	Observer-Kalman Filter Identification
EKF	Extended Kalman Filter
p	body-axis angular rate about the body 1 axis; roll rate
q	body-axis angular rate about the body 2 axis; pitch rate
r	body-axis angular rate about the body 3 axis; yaw rate
VFR	visual flying rules
i^+	notation for reference frame i
$\mathbf{x}$	vector
$\hat{\mathbf{x}}$	unit vector
$\hat{\mathbf{i}}_j$	generalized reference to the jth axis of reference frame i^+
$[\mathbf{x}]_i$	parameterization of vector $\mathbf{x}$ in reference frame i^+

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

Approach and landing are among the most critical flight regimes for automatic control of fixed-wing aircraft. As in any flight phase, the effects of external disturbances, which appear primarily as air turbulence and static wind, must be mitigated. Landing should not damage the aircraft. Finally, performance of the controller should be repeatable for many flights in potentially different environmental conditions.

Additional challenges are introduced when landing light weight unmanned air vehicles (UAVs). Accurate models are not always available, because traditional design methods do not generally scale down reliably. Parameter estimation may not be usable without wind-tunnel testing or computational fluid dynamics modeling. Further, this scale of aircraft is limited in payload capacity, and therefore in the quality of sensor data available for system identification from flight data. These sensor limitations also constrain any automatic controllers. Each of these challenges must be accounted for in the control design and data management.

1.1 Significance

Implementation of an automatic landing system is of immediate use in ongoing aircraft flight research, and will enable future tests of other control techniques. The Pegasus UAV (Fig. 1.2) is used primarily as a sensor testbed, as it has unusually large internal volume and payload mass capabilities for an academic UAV. The vehicle is generally flown under remote control by a human pilot. Fully automated research flights would reduce operator workload and extend the duration of missions. Because this vehicle is so different in size and mass than the UAVs for which most commercially available low-cost autopilots are designed, an automatic landing controller tailored for this vehicle is desired to ensure reliable and safe performance.

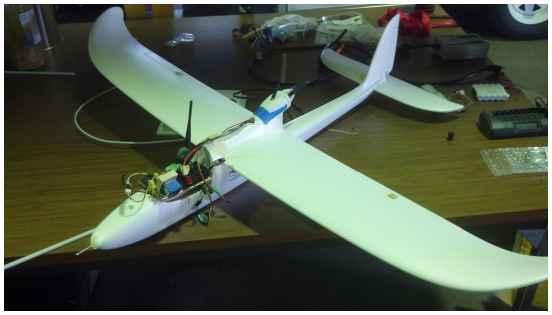


Figure 1.1: Easy Star vehicle modified to carry autopilot and video transmission equipment.

Figure 1.2: Pegasus vehicle in flight.

Given the expense of maintaining and operating the Pegasus vehicle, this platform is not ideal as a controls testbed for hazardous flight tests. However, the wide availability of inexpensive electric aircraft and lightweight microelectronics enables flight testing of relatively sophisticated control laws at a fraction of the cost, risk, and personnel requirements of the Pegasus vehicle. The Vehicles Systems + Controls Laboratory (VSCL) owns several Easy Star and Bixler RC airframes (See Fig. 1.1). These vehicles cost less than $ 100, have a full set of standard control surfaces, and can support a lightweight autopilot. By designing the automatic landing controller and evaluating it on these platforms, this thesis validates the control implementation before automatic landing of the Pegasus vehicle is attempted, reducing risk. The same autopilot hardware can be used on both vehicles, so the only changes required to implement automatic landing on Pegasus are updates to the control gains in software. Furthermore, since the range of flight conditions experienced by these UAVs is relatively small, the modeling and simulation framework established in this thesis for the lightweight vehicle can be used and modified to enable future testing of more advanced control laws on this same low-cost platform.

The modeling uncertainty associated with UAVs must be addressed in the design of the automatic landing controller. Quantitative feedback theory (QFT) is a robust control methodology for synthesizing control laws for linear plants that guarantees satisfaction of frequency domain tracking, stability, and disturbance rejection requirements for a set of uncertain plants specified by the control designer[52]. QFT is used in this thesis for design of linear discrete-time robust control laws, in an effort to ensure overall vehicle safety in flight test.

1.2 Previous research

1.2.1 Automatic landing for unmanned aircraft

Design of automatic landing controllers for manned fixed-wing aircraft is a problem that dates back several decades [6]. Recently, automatic landing for smaller unmanned research aircraft has become a topic of research attention. A variety of different control and sensor approaches have been presented. The literature generally focuses on innovative applications of sensors with relatively traditional control approaches.

Ref. [39] designs a fuzzy logic controller for automatic landing in simulation. This controller shows good performance in controlling a nonlinear aircraft model. Ref. [5] describes flight tests in which optical flow is used for terrain following and attitude estimation at low altitudes. In Ref. [30], an inexpensive attitude filter for a UAV is designed using a Global Positioning System (GPS) receiver, accelerometer, and gyroscope with an Extended Kalman Filter. Ref. [29] explores the use of inexpensive inertial measurement unit (IMU) and GPS hardware for navigation and control. Flight test results are presented for nominal maneuvering, but landing is performed by a remote operator. Ref. [4] experiments with landings using a fusion of barometric sensors and optical flow to estimate altitude, and demonstrated repeated

landings within meters of a target. Proportional-integral-derivative (PID) loops are used for longitudinal-axis control with a lateral/directional control method based on course vector fields. This work is extended in Ref. [3], which presents a vision-based system for longitudinal and lateral control of an unmanned aircraft during landing. A bias correction scheme is implemented to account for errors in relative orientation, parameter estimation, wind estimation, and target velocity estimation. In flight test, the vehicle successfully demonstrated landing onto both fixed and moving platforms, and consistently landed within 5 m of the target position. Ref. [42] performed autonomous takeoff and landing of a lightweight (less than 5 kg) air vehicle using onboard processing only. Sequential feedback loop closures were used for guidance and control. An ultrasonic rangefinder is used for landing to obtain precise altitude values in the absence of optical flow or visual imagery. Ref. [22] uses a vision-based system for visual servoing control of pitch and heading angles to land a small micro air vehicle (MAV) on a small inflatable dome.

1.2.2 Quantitative feedback theory for aircraft control

QFT is a robust control technique originally applied to linear systems. It has received significant attention in the field of automatic control of aircraft. This section highlights prior aircraft control studies that use QFT. Much of the research that has been performed focuses on fault-tolerant control that is to be applied over a wide range of flight regimes.

In Ref. [48], flight test results for a pitch axis controller for the Air Force "Lambda" UAV are presented. Use of QFT is motivated by a desire for a single feedback system to be used across the aircraft's flight envelope. A complete design process for a single-input single-output (SISO) synthesis, including plant uncertainty, is presented. [27] expands upon the previous paper to develop a longitudinal-axis

regulator with robustness to elevator damage. In this work, a pitch-axis regulator is developed, with no additional outside control loops. A complete SISO design is presented in [51], in which an existing longitudinal-axis controller for a remotely piloted vehicle is improved using QFT. Again, both variations in operating conditions and fault-induced loss of control effectiveness are considered. This paper gives the stability and tracking requirements used for synthesis; it also indicates that disturbance rejection performance can frequently be satisfied as a by-product of robust input-output tracking specifications. [21] designed robust inner loop controllers for the "Lambda" UAV using QFT to achieve robustness to varying flight regimes. The design work included both longitudinal and lateral/directional control designs with flight tests and multiple iterations; however, only a fraction of the design process is presented. Santander and Aranda [45] present multiple-input multiple-output (MIMO) longitudinal and lateral/directional QFT controllers validated in nonlinear simulation. More recently, [46] uses QFT to achieve Level 1 handling requirements for a manual control of a manned aircraft.

The utility of QFT for MIMO control design has been questioned in the past[11]. However, subsequent literature finds that the QFT approach provides a sound basis for low dimensional MIMO systems[28]. Furthermore, the history of successful QFT designs in flight test research supports the viability of this approach for control synthesis.

The work of Ref. [49] has closely motivated the present work. In [49], sequential loop closures are used to develop QFT-based control laws for automatic landing of a Commander 700 aircraft in simulation. The autopilot was intended for a commercial aircraft, and therefore used a very different sensor package than would be available on a research UAV. The fundamental control structure, however, can be adapted to any vehicle. A literature search has not revealed any flight test results using QFT

for automatic landing.

1.3 Research challenges

The primary research challenges are: (1) limited sensor accuracy and available payload mass; (2) uncertain plant models; (3) external disturbances. These competing challenges require that the flight controller and sensor apparatus be complex enough to mitigate the effects of process and measurement noise and uncertain plant models, while being simple enough to run in real-time on a commercial microcontroller. This section discusses each of these potential issues in depth.

1.3.1 Hardware limitations

The sensor suite for autonomous landing should be relatively inexpensive and light enough to implement on a micro air vehicle massing less than 5 kg, to ensure that the package can be carried by the Easy Star test vehicle. This assumption greatly limits the available computational power for control computation, as well as sensor quality. Commercial hardware should be used to minimize the cost. A sensor package with existing data processing is desired so that the focus of this research can be on the control synthesis and implementation. Finally, the hardware solution must be readily integrated with the control laws, and should allow access to raw data so that filtering can be added in the future if desired or needed. These requirements limit the available hardware options.

1.3.2 Uncertain plant models

Existing methods for estimating aircraft dynamic properties without the need for aerodynamic analysis, such as DATCOM, are based partly on trends in manned-scale aircraft that do not translate well to lightweight UAVs. Although relatively simple computational methods for aerodynamic analysis are well-established, implementing

a solver and translating the aircraft geometry to a usable format is still expensive compared to the cost of an off-the-shelf ready-to-fly RC airplane and sensors. Ideally, flight tests can be executed to enable system identification, but for the lightweight UAV, it may be difficult to acquire full state measurements at sufficiently fast rates. Model errors of between 10-30% are assumed typical for the linear models used for control synthesis.

1.3.3 Disturbances

The primary disturbances influencing fixed wing aircraft arise from aerodynamic forces. For modeling, aerodynamic disturbances are classified as either transient or steady phenomena. The steady effects are assumed to be a nonzero average wind field, whose magnitude and direction are approximately constant for the duration of autolanding. The steady wind field changes the effective sideslip and angle-of-attack of the vehicle, which in turn influence the aerodynamic forces and moments acting on the vehicle. The primary transient aerodynamic effect is assumed to be air turbulence. Turbulence refers to the effects of chaotic aerodynamic phenomena, and acts as an additive term to the average field velocity. Turbulence tends to affect the entire aircraft as a high-frequency disturbance of variable magnitude, depending on local air conditions.

The nature of academic flight testing is designed to mitigate the influence of aerodynamic disturbances. Operational limits for both steady winds and gusts are established for all flight vehicles. Conditions can change over the duration of flight, but in general the disturbance threshold for any vehicle is clearly defined independently of the control design process. Consequently, the hardware limitations and uncertain models are assumed to be more critical and are given more attention than external disturbances.

This thesis synthesizes automatic landing controllers for two research flight vehicles. The controller is initially implemented on a low-cost demonstrator vehicle for basic validation in a low-risk environment, and is validated in flight test experiments. Subsequently, the control law is re-tuned for the larger Pegasus UAV. Motivated by a desire for computationally simple robust control, quantitative feedback theory (QFT) is used for most of the design synthesis. Control design requires identification of dynamic models for both vehicles, as well as integration with an existing autopilot hardware/software package for flight testing.

2. CONTROL SYNTHESIS AND STRUCTURE

Quantitative feedback theory (QFT) is a control synthesis methodology for satisfying closed-loop performance specifications in the presence of quantified plant uncertainty and/or disturbances. In this section, the QFT formulation and process is detailed, and its selection for control synthesis is justified. A simple example of the QFT design process is presented. Subsequently, the cascaded control loops implemented to achieve automatic landing are detailed in terms of their structure and the controlled variables.

2.1 Quantitative feedback theory

The term "quantitative feedback theory" is first used by Horowitz in 1979 to describe his approach to linear controller synthesis [48]. The foundations of QFT can be dated back further to work by Horowitz in 1959 for single-input single-output (SISO) systems with output feedback[15]. Later work extends the theory to SISO linear time-varying (LTV) systems[17], nonlinear SISO systems[18], and linear and nonlinear multiple-input multiple-output (MIMO) systems [19]. The increase in low-cost computing enabling computer-aided design (CAD) led to a surge of interest in the aerospace field in the 1990s, and the technique is implemented on a variety of fixed-wing aircraft systems in both simulation and flight test, as described in Section 1.2.2.

The fundamental objective of QFT is to achieve a desired performance in the presence of disturbances, sensor noise, and plant uncertainty. The QFT approach is intended to make the trade-off between the major factors that influence control design, such as performance requirements, plant uncertainty, controller complexity, and bandwidth, transparent. QFT is initially developed for linear time-

invariant (LTI) SISO plants, and at its core, the technique facilitates control design for such systems. It is extended to nonlinear and multiple-input multiple-output (MIMO) uncertain problems by rigorously converting these problems into LTI SISO equivalents[16],[17],[18]. The standard SISO loop structure is shown in Fig. 2.1; it consists of a plant P, feedback controller G, prefilter F, and sensor dynamics H. The current research implements sequential loop closures on cascaded SISO loops for assumed linear systems; consequently, the remaining discussion of QFT will be restricted to linear SISO systems.

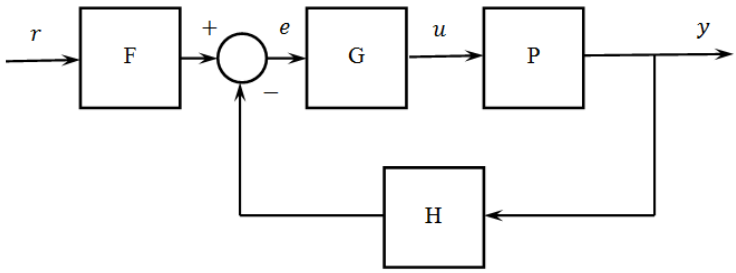


Figure 2.1: Generic QFT feedback loop with plant P, controller G, prefilter F, and sensor dynamics H[52].

The general procedure for control synthesis in the QFT framework is now summarized. In QFT, uncertainties are quantified into a set of uncertain plants, a set of disturbances, and sets of acceptable output responses to each, which must be achieved for all plants. QFT is strictly a frequency domain technique. It is common in engineering practice for output responses to be given in the time domain; if so, the first step is to translate time domain requirements into frequency domain equivalents. If all plants are minimum-phase, this can be achieved by defining upper and lower bounds $a(\omega)$ and $b(\omega)$ such that $a(\omega) \leq |T(j\omega)| \leq b(\omega)$ for any T in the set of accept-

able responses[16]. Subsequently, the next step is to determine the frequency-domain response of the set of plants at a "reasonable" number of frequencies[16], which is largely a matter of the designer's discretion. The responses of the set of plants at a given frequency are referred to as the plant templates. An example Nichols plot of plant templates is shown in Fig. 2.2. For a plant P_i in the set specified, the open-loop plant response with controller is defined to be $L_i = GP_i$. The closed-loop response is then $T_i = \frac{FL_i}{1+L_i}$. For every plant P_i, the value of L_i for which all performance bounds are exactly satisfied can be computed, as in [20].

The Nichols plot shows open-loop magnitude response as a function of open-loop phase, and is used as the principle tool for designing the controller G in QFT. A representative example system response in the Nichols plots is shown in Fig. 2.3. It has been shown that the optimal value of L_i at each frequency lies on the bound defined by exact satisfaction of all associated requirements; here, "optimal" is taken to mean the open-loop response that satisfies the requirements with minimum gain[13],[20]. By plotting a designed L with nominal plant P_i, the control designer can see graphically how far the response is from this optimum, and therefore explicitly judge the tradeoff between increased controller complexity and performance. The prefilter is subsequently designed based on the controller. For example, if the specified closed-loop response requires that $C < |T(j\omega_i)| < D$ at frequency ω_i, and the system response without prefilter yields $X < |\frac{L(j\omega_i)}{1+L(j\omega_i)}| < Y$, then the prefilter response must be $C - X < |F(j\omega_i)| < D - Y$. Repeating this process for all ω_i in the chosen array generates bounds for $|F|$ at all frequencies. If the variations in the closed-loop response without the prefilter are too large, infeasible constraints may be generated, which will necessitate either modification to the controller or to the acceptable output responses.

Contours defined by the performance requirements can be computed for any

desired input-to-output response of a SISO loop[7]. As a practical matter, most examples of QFT in the research literature employ two requirements:

- Bounded stability: $|\frac{L}{1+L}| < SM$, where the static margin SM is constant at all frequencies
- Bounded input-output tracking: $T_{rl}(s) \leq |\frac{FL}{1+L}| \leq T_{ul}(s)$

As noted in [51], disturbance rejection requirements are often satisfied implicitly in the process of addressing input-output tracking requirements. The plant output disturbance rejection requirement, under the assumption of unity sensor dynamics, has the following constraint with respect to the controller:

$$\left|\frac{1}{1+GP}\right| \leq W_d \tag{2.1}$$

In Eq. 2.1, W_d is an upper bound on the output disturbance-to-output response. In the current research, bounded stability and input-output tracking alone are found to provide adequate performance in all QFT loops save for one. In an effort to improve disturbance rejection in one loop, an explicit output disturbance rejection criterion was added, and is discussed further in Sec. 4.2.2.2.

The primary motivation in selecting QFT for designing automatic landing controllers is the robustness to specified plant uncertainties. Small-scale UAS dynamics tend to be effectively nonlinear due in part to their susceptibility to winds and turbulence, according to Chao, Cao, and Chen[8]. The Pegasus aircraft is substantially larger and more massive than many of the small UAS used in current academic research, but is still roughly an order of magnitude less massive than a small commercial or general aviation manned vehicle. As such, exogeneous inputs caused by turbulence or gusts can produce large changes in the vehicle states, making identifi-

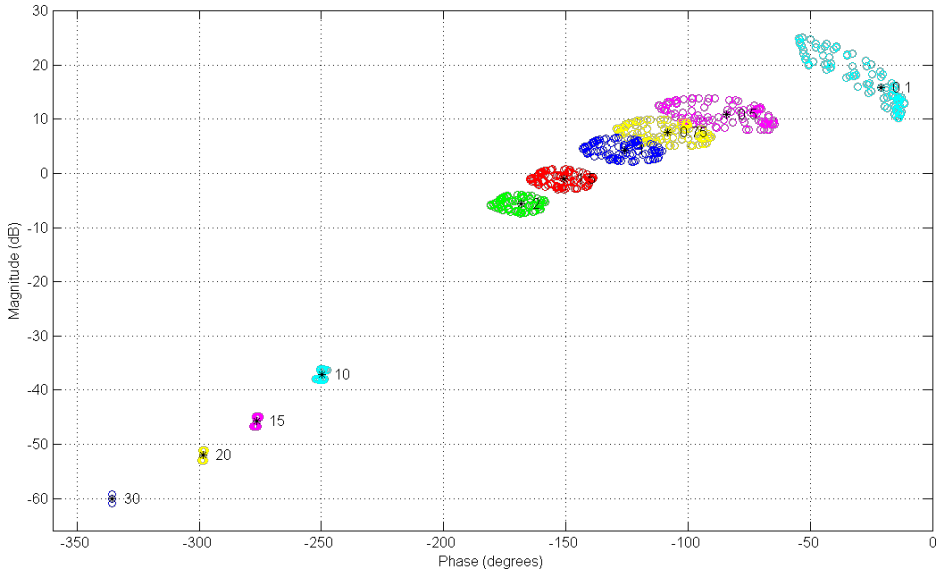


Figure 2.2: Example Nichols plot of plant templates at various frequencies.

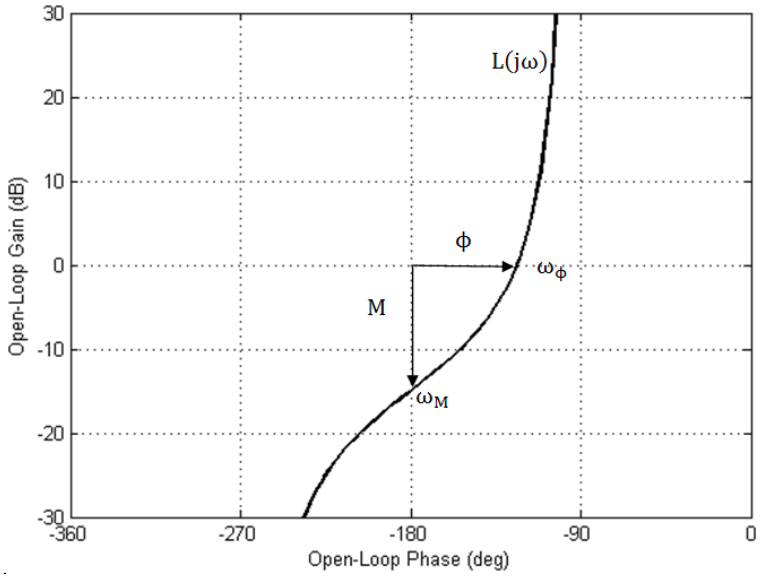


Figure 2.3: Generic Nichols plot for open-loop system $L = GP$. Shown are: (1) Crossover frequency ω_ϕ; (2) phase margin ϕ; (3) Gain margin frequency ω_M; (4) Gain margin M[52].

cation of an accurate linear model difficult even when flight data are available. The ability to account for model uncertainties in the design process is advantageous. The reason QFT is selected instead of a different robust control technique is because QFT allows for direct design of discrete-time linear control laws with a clear representation of the tradeoff between controller complexity and performance. Use of linear control laws minimizes the associated computational overhead, and the transparency in the design stage should enable the design of controllers that satisfy performance requirements reasonably well while remaining relatively low-order. The objective is to make the controllers as simple as feasible to make integration with an existing autopilot system straightforward.

QFT is implemented typically as a computer-aided design (CAD) process. The design procedure is to add, remove, and change the poles and zeros of the controller and prefilter and observe the change of the system response in real-time. In the current research, the Terasoft MATLAB® QFT Frequency Domain Control Design Toolbox enables the CAD process by generating interactive plots and automating the computation of the bounds for performance requirements. Direct digital design is achieved by designing a controller and prefilter with continuous-time poles and zeros, then converting the resulting controller into an equivalent discrete-time controller using a zero-order hold[7]. This conversion is a background process, and the work of the control designer is agnostic with respect to a continuous- or discrete-time controller. To illustrate the design procedure, the next section considers the example of a linearized pendulum with control torque in detail.

2.2 Example - Pendulum Controller Design Synthesis

To illustrate the QFT synthesis process, a simple control design example is presented. A single pendulum in a gravity field with aerodynamic damping and an

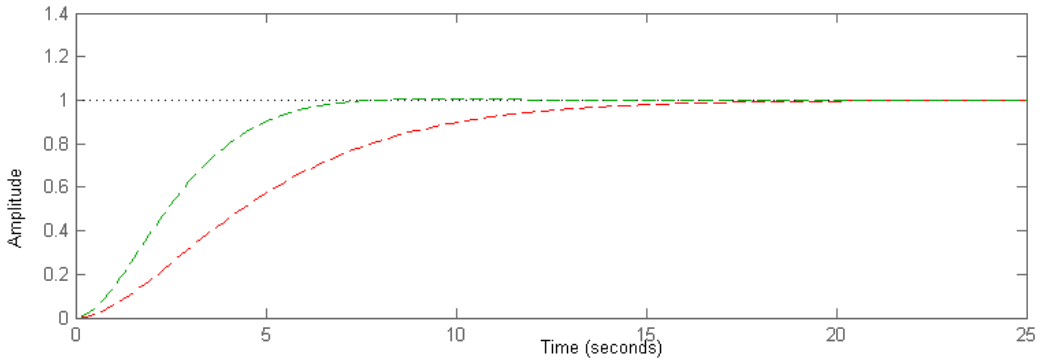


Figure 2.4: Step response of the upper and lower tracking bounds.

available control torque is considered. The pendulum mass and the damping coefficient are inexactly known. The governing equation and linearized form in terms of the mass m, damping coefficient c, length l, control torque u, and gravitational acceleration g are:

$$ml\ddot{\theta} + c\dot{\theta} + mg\sin\theta = ul \tag{2.2}$$

$$ml\ddot{\theta} + c\dot{\theta} + mg\theta = ul \tag{2.3}$$

The control requirements are:

- Stability margin $W \leq 1.1$
- 90% rise time between 10 and 15 seconds
- Zero overshoot in response to a step input

The latter requirement is translated into the frequency domain as bounded tracking up to frequencies of 1 rad/s. The upper and lower tracking bounds are defined by a pair of transfer functions each having a complex pole. The lower bound T_{rl} has a pole at $-.38151 \pm .035359i$. The upper bound T_{ru} has a pole at $-.55147 \pm .33149i$.

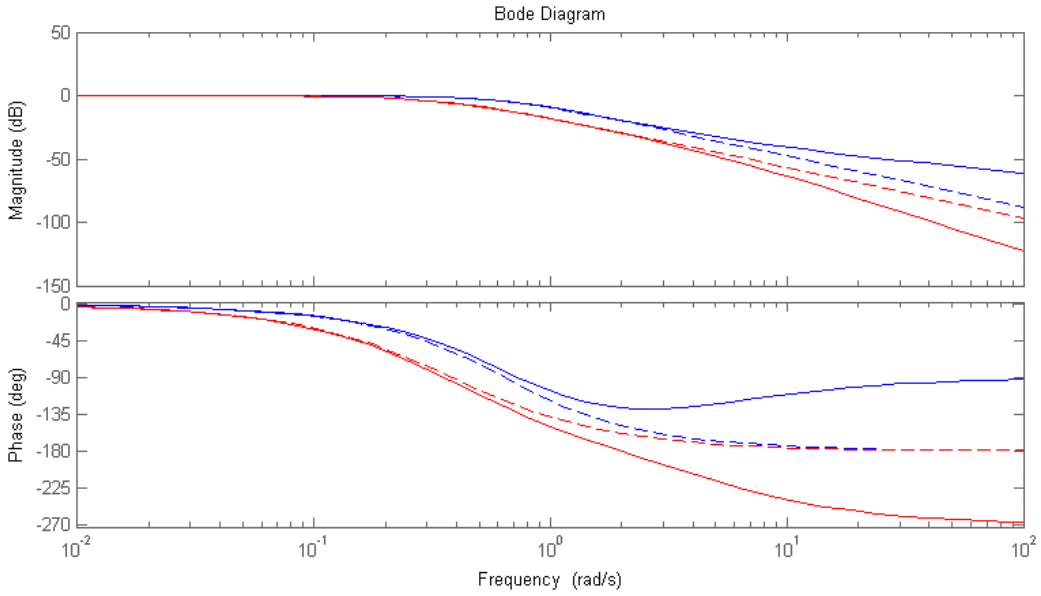


Figure 2.5: Bode magnitude and phase response of the upper and lower tracking bounds. The solid lines are the bounds once augmented with an additional pole and zero.

These functions were selected to give the step responses shown in Fig. 2.4. To increase the separation of the response magnitude of the upper and lower bounds at high frequencies, it is common to add a zero to the upper bound and a pole to the lower bounds[49]. The additional terms do not substantially change the low-frequency response but make satisfying high-frequency tracking feasible. A zero at -5 is added to the upper bound and a pole at -5 is added to the lower bound. The effect of adding these terms can be seen in Fig. 2.5. The full tracking bounds are given in Eqs. 2.4-2.5.

$$T_{rl} = \frac{0.7353}{s^3 + 5.763s^2 + 3.962s + 0.734} \tag{2.4}$$

$$T_{ul} = \frac{0.08264s + 0.4132}{s^2 + 1.103s + 0.414} \tag{2.5}$$

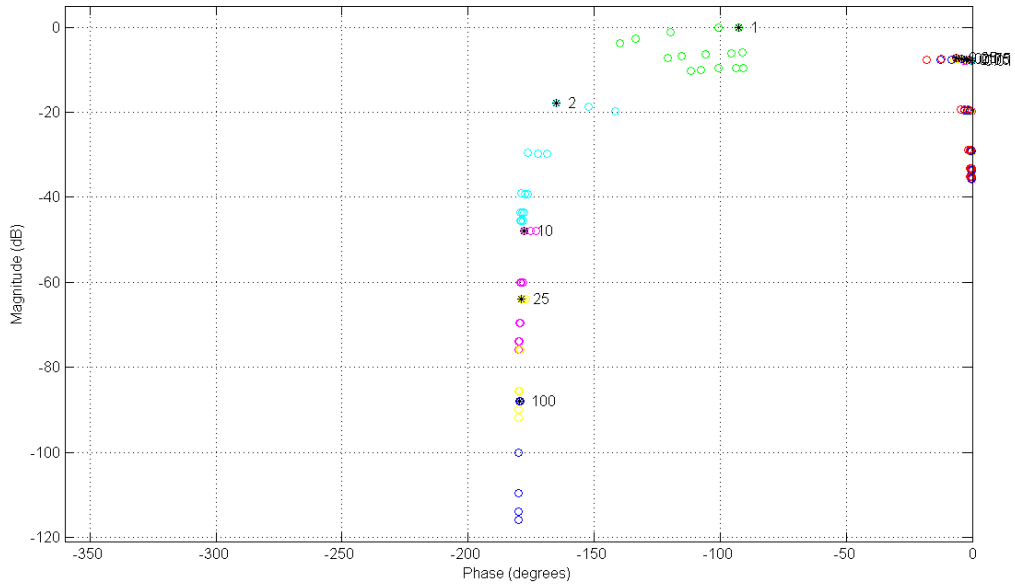


Figure 2.6: Plant templates for the set of uncertain plants and frequency array selected. Frequencies below $\omega = 1$ rad/s are clustered at the upper right.

The plant uncertainty is characterized by known bounds on the mass m and the damping coefficient c. $0.25 < m < 6.25$ and $1 < c < 3$ are assumed. For defining the QFT plant templates, the discrete values $m \in 0.25, 1, 3, 5, 6.25$ and $c \in 1, 2, 3$ are selected and found to give adequate results. The discrete frequency array $\omega \in .01, .05, .1, .175, .25, 1, 2, 10, 25, 100$ is chosen. The resulting plant templates can be seen in Fig. 2.6.

Having defined the plant templates, frequency array, and design requirements, the stability and tracking bounds for the controller G and prefilter F are determined automatically by the QFT software. The Nichols plot bounds, used for designing $G(s)$, are shown in Fig. 2.7. The Nichols plot of the uncertain plants must be above solid lines and must be below dashed lines to satisfy the performance requirements. The intersection of these bounds forms the requirements for the controller G. The prefilter bounds are defined purely in terms of robust tracking and are the upper and

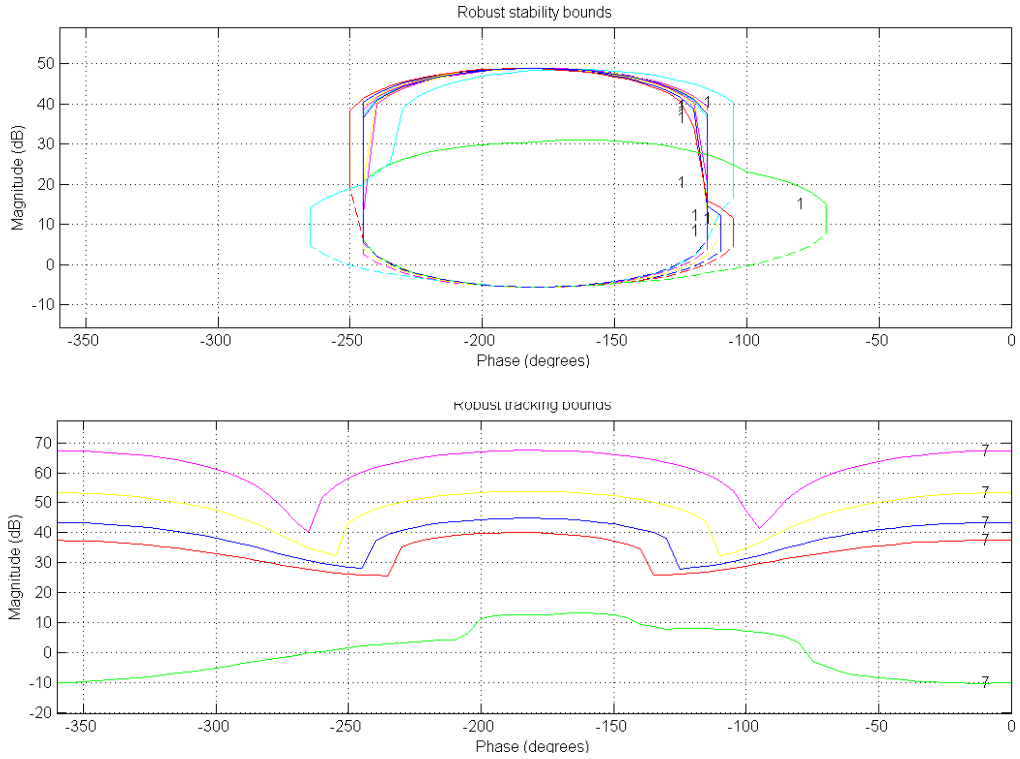


Figure 2.7: Robust stability and tracking bounds for the pendulum tracking problem.

lower tracking Bode magnitude curves.

The synthesis of the controller G for this particular problem is accomplished in four steps. Beginning at the top left of Fig. 2.8, an integrator is added to improve the low-frequency response and better meet the tracking bounds. The gain is then increased to satisfy the low-frequency bounds, yielding the bottom left plot. At this point, the low-frequency requirements are met but the closed-loop response is unstable. A second-order zero is added to stabilize the system and satisfy the tracking requirements for $\omega = 1$ rad/s. Finally, a real pole is added to reduce the gain at high frequencies and ensure an appropriate bandwidth for the closed-loop system. The response with the final controller is shown in Fig. 2.9. The final controller is given

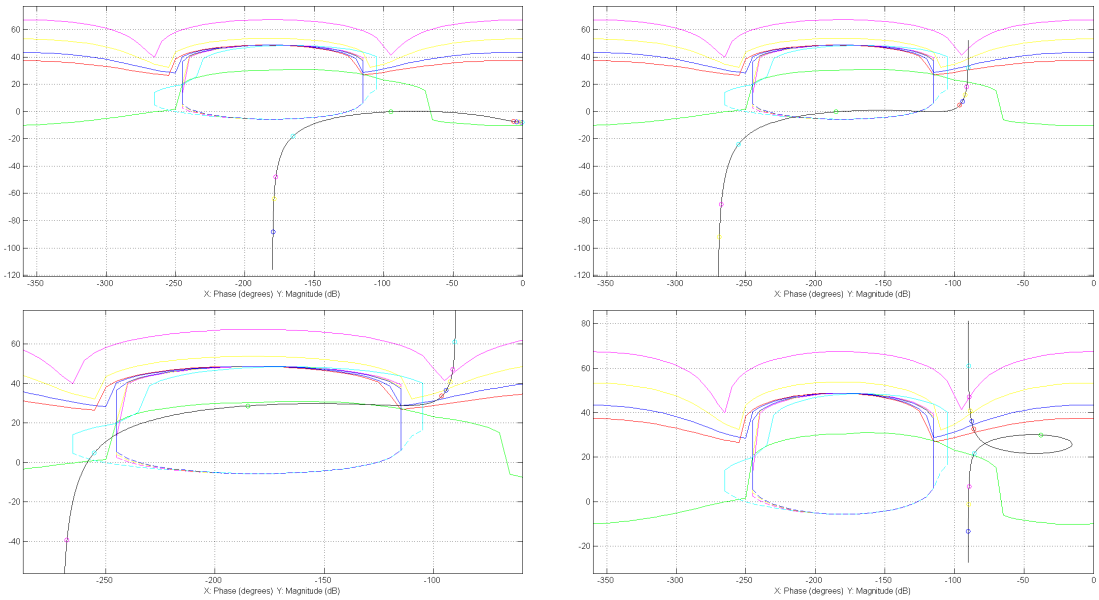


Figure 2.8: Sequence of steps for the synthesis of the controller $G(s)$.

by:

$$G(s) = \frac{1353.2s^2 + 431.39s + 697.94}{s^2 + 24.984s} \tag{2.6}$$

To synthesize the prefilter F, the Bode magnitude plot of the closed-loop response for $\omega \leq 1$ rad/s is considered. The design process is shown in Fig. 2.10. The initial plot is simply the closed-loop response with a unity prefilter. A real pole is added to match the tracking bounds at low frequencies. A notch filter, consisting of a complex pole and zero, is then added to ensure bounded tracking at $\omega = 1$ rad/s. Note that the notch was designed to deliberately push some of the responses slightly below the lower bounds so that the response magnitude at higher frequencies was decreased. Finally, a real zero is added at $\omega = 100$ rad/s to ensure the numerator and denominator orders match; this does not change the Bode plot significantly. The final prefilter is given by:

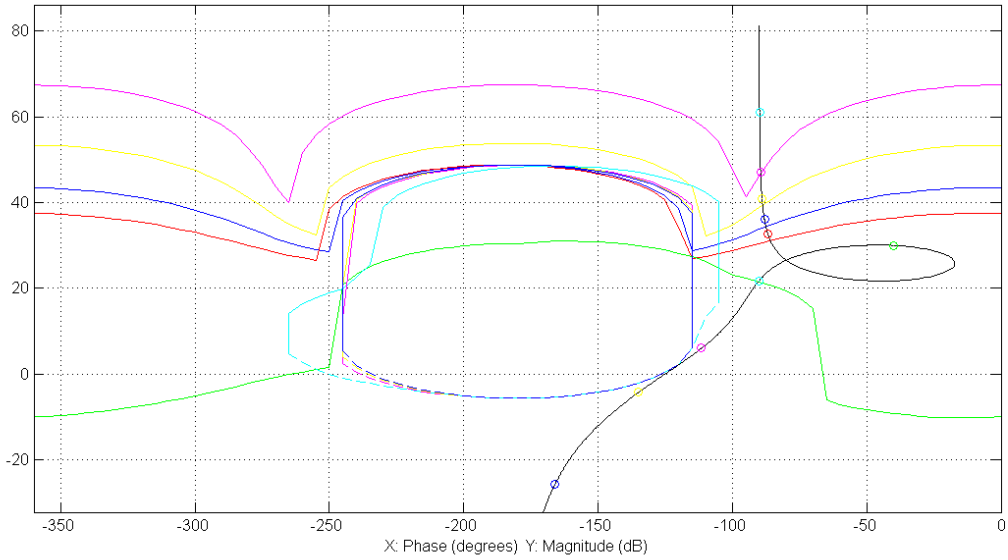


Figure 2.9: Final Nichols plot response of the open loop system $L = GP$ with the designed controller.

$$F(s) = \frac{0.0029967s^3 + 0.30163s^2 + 0.20192s + 0.50983}{s^3 + 1.5088s^2 + 1.477s + 0.50983} \tag{2.7}$$

This simple example is intended to illustrate the fundamental QFT design methodology, and a rigorous performance analysis is outside its scope. The step responses of a set of 10 random plants with mass and damping properties in the ranges defined previously are shown in Fig. 2.11.

2.3 Automatic landing control structure and coordinates

For the automatic landing system, sequential loop closures of SISO feedback loops are used to stabilize the aircraft dynamics and guide the vehicle to a reference flight path. Motivated by the work of Ref. [49], in which a QFT automatic landing controller was designed in simulation, the sensor dynamics H are assumed to be unity for all control loops. Control loops are to consist of prefilter F with feedback

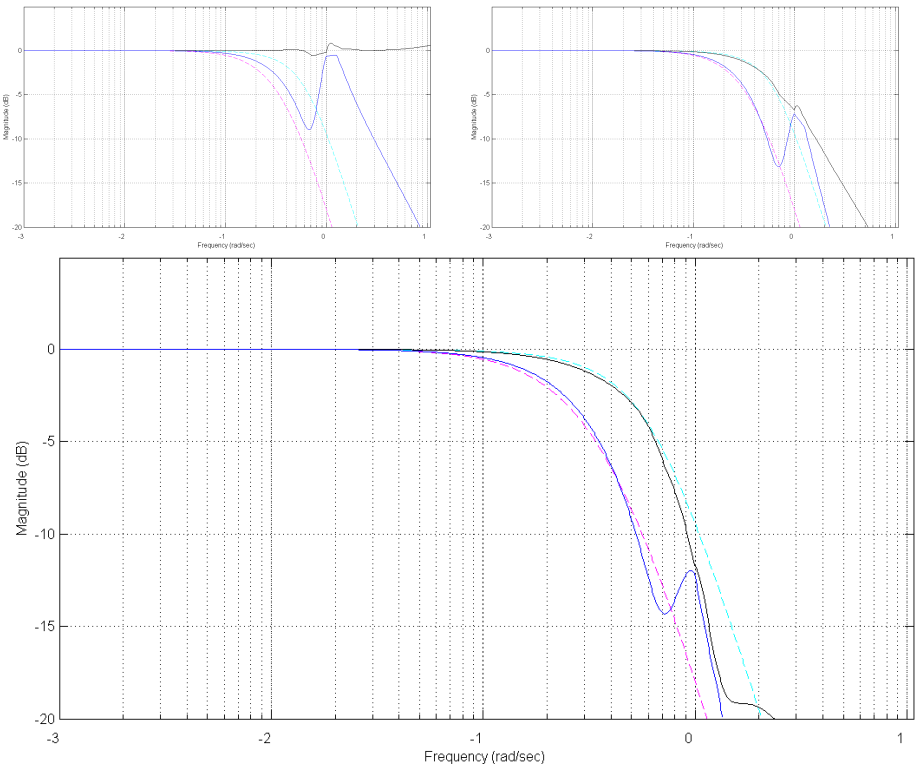


Figure 2.10: Bode magnitude response of the closed-loop system during prefilter synthesis.

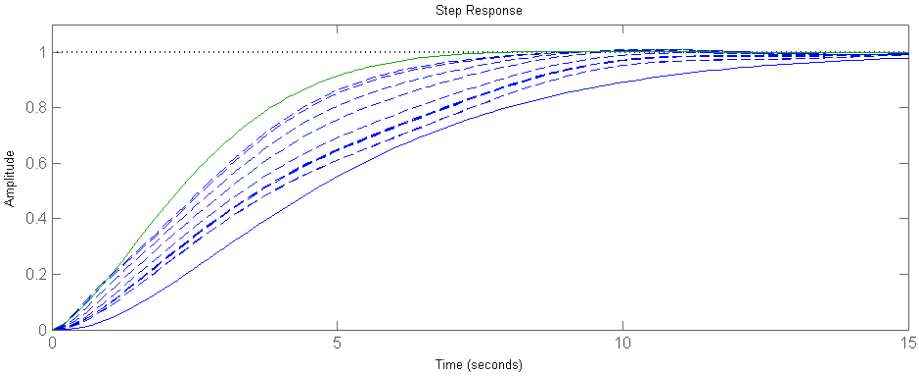


Figure 2.11: Step responses of 10 random uncertain plants (dashed lines). Step responses of the tracking bounds are shown with solid lines.

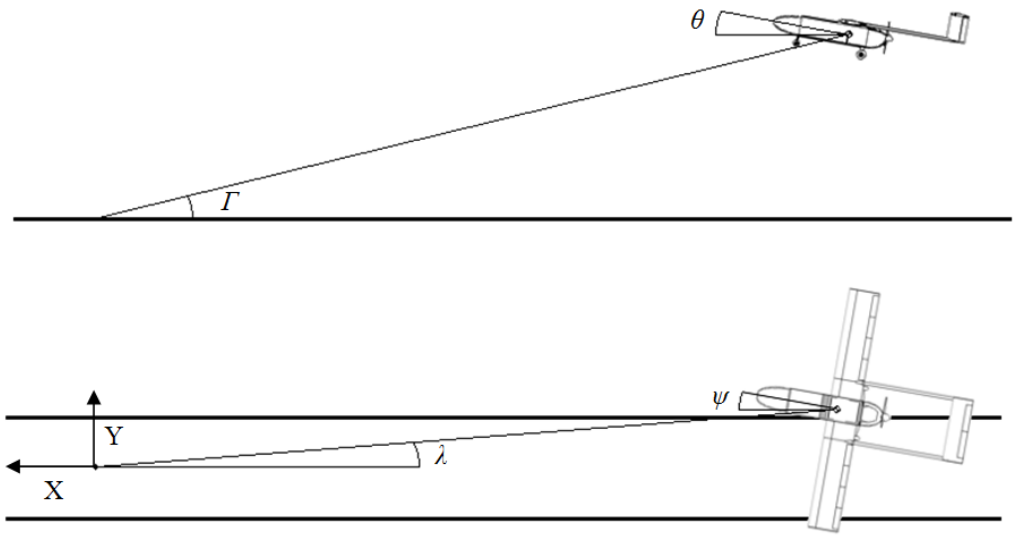


Figure 2.12: Coordinate system used for automatic landing.

G. The control system, which is further detailed later in this section, consists of three primary loop sequences:

- Heading control loop in which aileron is modulated to direct the aircraft's heading to match the runway direction (Three sequential loops)
- Pitch control loop in which elevator is modulated to control the vehicle's descent rate (Two sequential loops)
- Airspeed command and hold loop (Single feedback loop)

The feedback loops used for control are typified by Fig. 2.1. When disturbances are examined in the control design, output disturbances on y, such as aerodynamic turbulence, are considered. Before defining the outer-loop variables, the coordinate systems used are defined.

Two coordinate frames are used in modeling the aircraft dynamics: an inertial frame, designated n^+, and a body-fixed reference frame designated b^+. The inertial

reference frame is defined such that the 1 axis is parallel to the target runway, and the 3 axis points down. For simplicity, the origin is placed at zero altitude at the X-Y coordinates used to define the aircraft glideslope. The body reference frame is fixed to the aircraft at its center of mass. The b^+ frame is defined such that when the aircraft sits at rest on the ground, the body 3 axis is aligned with the inertial 3 axis. The body 1 axis is the projection of the vector from the center of mass to the aircraft's nose onto the plane defined by the 3 axis. This is the body-fixed reference frame traditionally used in aircraft dynamics[44].

The aircraft attitude is parameterized by a sequential 3/2/1 rotation from the inertial frame to the body frame through angles $\psi/\theta/\phi$, also known as heading, pitch, and roll angles respectively. Since the aircraft should not be undergoing aerobatic maneuvers, the kinematic singularity associated with $\theta = \pm\frac{\pi}{2}$ is not considered a problem. The remaining aircraft states are parameterized using traditional aircraft dynamics nomenclature:

- Inertial velocity of the aircraft parameterized in the body frame: $[\mathbf{v}]_b = \begin{bmatrix} U & V & W \end{bmatrix}^T$
- Angular velocity of the body fixed frame relative to inertial frame parameterized in the body frame: $[\boldsymbol{\omega}]_b = \begin{bmatrix} P & Q & R \end{bmatrix}^T$
- Vector from the origin of the inertial frame to the aircraft center of mass, parameterized in the inertial frame: $[\mathbf{r}]_b = \begin{bmatrix} X & Y & Z \end{bmatrix}^T$

The glideslope angle Γ and relative azimuth to the runway λ are defined as in Eqs. 2.8-2.9, and are shown graphically in Fig. 2.12:

$$\tan\Gamma = -\frac{Z}{\|X\|} \tag{2.8}$$

$$\tan\lambda = \frac{Y}{\|X\|} \tag{2.9}$$

The automatic landing control problem can be broken into two stages:

1. Approach to the runway: reference azimuth and glideslope angles are tracked until the aircraft altitude decreases to a specified flare height. Constant airspeed is maintained.

2. Flare maneuver: once the flare height is reached, the aircraft performs a flare by tracking a reference descent rate until touchdown. Target airspeed is reduced in preparation for landing.

Three variables are controlled directly by control modulation: aircraft bank angle ϕ using aileron control; aircraft pitch angle θ using elevator control; and aircraft airspeed using throttle control. Guidance to the runway and landing are achieved by inputting reference angles to the bank angle and pitch angle command and hold loops, while airspeed is maintained at a constant value.

Guidance of the vehicle in the X-Y plane is achieved by designing a SISO loop to track a reference azimuth of 0. This loop is maintained throughout the automatic landing. The plant for this outermost loop is the closed heading command and hold loop multiplied by the transfer function $\frac{\lambda}{\psi}$. The plant for the heading SISO loop is the bank angle command and hold loop multiplied by the roll-to-heading angle transfer function. Guidance in the X-Z plane is effected by a glideslope tracking SISO loop that tracks a constant reference Γ. In this case, P is the pitch angle command and hold loop multiplied by the transfer function $\frac{\Gamma}{\theta}$. For the flare maneuver, a

separate descent rate tracking loop is designed. It also generates reference pitch angle commands. Airspeed is maintained in both phases of the flight by a single SISO loop.

This section presents the basic theory behind QFT, as well as a representative example problem and a description of the control loops used in achieving automatic landing. The following section discusses the modeling environment and assumptions used in developing the dynamic simulations used in evaluating automatic control performance.

3. VEHICLE DESCRIPTIONS AND MODELLING

Control design is performed on the Easy Star and Pegasus UAVs. The VSCL primarily operates the Pegasus UAV, a custom aircraft in the 100-lbf class. The vehicle typically carries the ArduPilot Mega (APM) autopilot, a commercial product capable of enabling waypoint navigation on standard remote-control (RC) hardware. The automatic landing controller, coupled with the standard capabilities of the APM, will enhance the autonomous capabilities of the vehicle. Because the lab has no previous experience integrating new control laws with this autopilot software, it is desired that the control law be tested in a lower-risk environment. The control law and initial autopilot integration will be done on the inexpensive, hobbyist Easy Star. The Easy Star has payload capacity to carry the APM and RC hardware; once the control laws are designed and validated on this aircraft, the same experimental software with updated control gains can be used to achieve automatic landing on Pegasus.

This section begins with a brief description of each vehicle. Next, some further variables are defined to supplement those presented in the previous section. From there, the modelling process used to identify dynamic models for each vehicle is described, followed by a description of the sensors used by the autopilot. The turbulence model used in dynamic simulations is detailed briefly. Finally, the simulation environment and associated assumptions are presented.

3.1 Easy Star system description

The Easy Star is an inexpensive commercial RC product. With its high wing position and large aspect ratio, it has relatively benign flying qualities and is designed for inexperienced RC pilots[38]. The vehicle's payload bay is large enough to carry the

required RC avionics and battery as well as an APM autopilot. This characteristic has made this vehicle a popular choice among hobbyists for autonomous flying with the APM. The vehicle has dual ailerons, rudder, and elevator control, making it an appropriate controls analog for a standard configuration fixed-wing aircraft. The small size and all-electric power system enable multiple test flights of short duration with software changes as required in between.

3.2 Pegasus system description

The Pegasus vehicle was designed and fabricated at Texas A&M. It is designed for a payload capacity of 20 lbf and its fuselage is wide enough to carry a standard ATX motherboard. The APM is employed on this vehicle because it offers basic autopilot capabilities, such as waypoint navigation, in a very small package that does not hinder the vehicle's capacity for remote sensing missions. Since first flight in winter of 2012, the vehicle has approximately five flight hours and 40 flights under remote control.

3.3 Equations of motion and variable conventions

Aircraft reference frames and states are defined in Section 2.3. The aircraft kinetic states, U, V, W, P, Q, R, are heavily influenced by the aerodynamic angles between the aircraft and the relative wind vector, and the governing equations of motion cannot be derived for a general case. The equations of motion for the kinematic states, $X, Y, Z, \psi, \theta, \phi$, are exactly known for an aircraft modeled as a rigid body, and can

be written as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\phi\sin\theta - \cos\phi\sin\psi & \sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta \\ \sin\psi\cos\theta & \cos\psi\cos\phi + \sin\phi\sin\psi\sin\theta & -\sin\phi\cos\psi + \cos\phi\sin\psi\sin\theta \\ -\sin\theta & \cos\theta\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{3.1}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \tag{3.2}$$

Aircraft linear dynamic models are commonly defined about the steady-state $P = 0, Q = 0, R = 0, \phi = 0$ at constant values of U, V, W, θ, ψ. With zero steady-state bank angle, the aircraft dynamics can be decoupled into longitudinal and lateral/directional states:

- Longitudinal states: U, W, Q, θ
- Lateral/directional states: V, P, R, ϕ

Under the assumption of constant steady states, the dynamics can be written in terms of perturbations about the steady state, and the perturbation variables u, v, w, p, q, r are defined for the body axis translational and angular velocities.

The heading angle ψ is sometimes included as one of the lateral/directional states; since its magnitude is not restricted to a linear regime about a constant steady state, it is omitted in the preceding list and subsequent linearized dynamic equations.

The primary aircraft controls are elevator, throttle, aileron, and rudder. The decoupling outlined above has a similar effect on the controls, such that only elevator and throttle deflections δ_e and δ_t are assumed to influence the longitudinal states, and only aileron and rudder deflections δ_e and δ_r are assumed to influence the lateral/directional states. Under this model, the longitudinal and lateral/directional

dynamics can be decomposed into two fourth-order linear systems having two controls each.

Finally, it should be noted that the aircraft body 2 and 3 axis velocities are parameterized in terms of the aerodynamic angle-of-attack and sideslipe angles, α and β. In the absence of external wind, these angles are defined as in Fig. 3.1 or as below:

$$\alpha = \arctan \frac{W}{U} \tag{3.3}$$

$$\beta = \arctan \frac{V}{U} \tag{3.4}$$

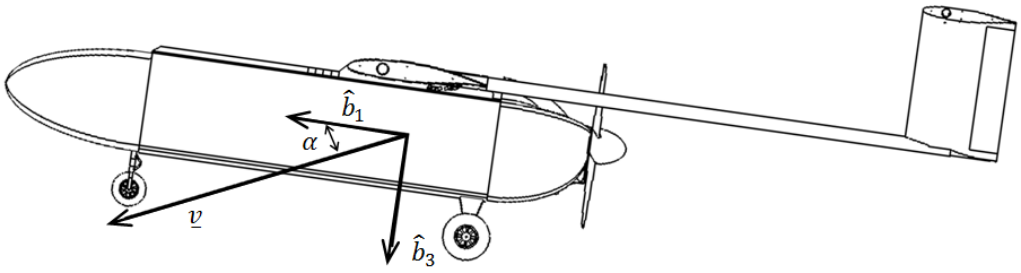


Figure 3.1: Definition of aircraft angle-of-attack and body 1 and 3 axes.

3.4 Easy Star modeling

The QFT controller design requires linear dynamic longitudinal and lateral/directional Easy Star models. Identification of a model from flight data is theoretically possible, but the APM does not directly measure body-axis velocities. Although additional equipment can measure body-axis velocities directly, the required aerodynamic probe with additional tubing and pressure sensors would be difficult to fit in the available

payload space. It is assumed that system identification from flight data is not practical for this vehicle.

The flight simulation program X-plane is used to generate flight histories of longitudinal and lateral/directional axis maneuvers. X-plane uses blade element theory to estimate aerodynamic characteristics based on the geometry of the aircraft[31], and basic aerodynamic properties can be obtained from this technique. The thrust characteristics are determined experimentally from the vehicle hardware, as detailed in Section 3.4.1. A 3D model of the RC Bixler aircraft, which has nearly identical dimensions and configuration to the Easy Star, is flown in simulation to generate the data[41]. The simulation data are then used to fit and evaluate linearized aircraft models. A discrete-time least squares fit is used in model identification. The details of the model identification and the linear models themselves are given in Appendix A.

In addition to the identified model, the control surfaces are assumed to have first-order dynamics with a 0.1 sec time constant, such that the actual deflection δ is related to the commanded deflection δ_c by the transfer function. The Hitec HS-5055MG servos used on the Easy Star will be powered at 4.8 V; the 0.1 second time constant provides a 90% rise time of just under 0.2 seconds, which matches the manufacturer's specification for the time to reach 60°at 4.8 V[14]. The assumed servo dynamics are:

$$\frac{\delta}{\delta_c} = \frac{10}{s+10} \tag{3.5}$$

3.4.1 Thrust modelling

To increase the vehicle's available thrust, the stock electric motor is replaced with a Model Motors AXI 2217/12[37]. The manufacturer does not provide a thrust rating with the 7" x 6 Easy Star propellor, so static testing is conducted to deter-

mine the maximum thrust. The motor with propellor is secured to a scale with the thrust axis normal to the scale. The measured thrust at full throttle is between 1.5 lbf (6.7 N) and 2.0 lbf (8.9 N) in each of several trials. This is consistent with the manufacturer data, which provide thrust values between 9 and 12 N using the same battery and slightly larger propellors. The static tests were conducted with fully charged batteries. Automatic landing will take place after flying, which will deplete the battery and may reduce the available thrust. The maximum thrust at landing is conservatively assumed to be only 1.0 lbf (4.4 N). The maximum thrust is incorporated into the simulation, and replaces the X-plane estimate for the throttle influence on the $\dot{u}$ channel. The throttle's influence on the other states cannot readily be measured without a more extensive experiment but should be less significant. The values derived from X-plane are assumed to be sufficiently accurate.

The dynamic response of the motor to throttle inputs is also characterized for modeling and simulation purposes. A separate experiment is conducted to determine the dynamic response. The Easy Star fuselage with motor installed is secured to a fixed surface. An MPX7002DP air data sensor with Pitot-static probe is then attached to the vertical tail in the propellor wash, at a horizontal distance of 10.5" behind the spinner. Due to the relatively high noise level associated with this sensor, five trials are conducted with the sensor at different locations. After each trial, the air data sensor is moved down 1" from the previous trial, with the first trial beginning at the motor centerline. The Pitot probe extends ahead of the vertical tail surface, so it is assumed that the thrust profile is radially symmetric about the motor centerline with no interference from the vertical tail. The experimental setup is shown in Fig. 3.2. In each trial, a ramp throttle input from 0% to 100% is given and held constant at 100% for three to five seconds. A step input from 100% back to 0% is given.

The thrust dynamics are assumed to be first-order, such that the actual throttle

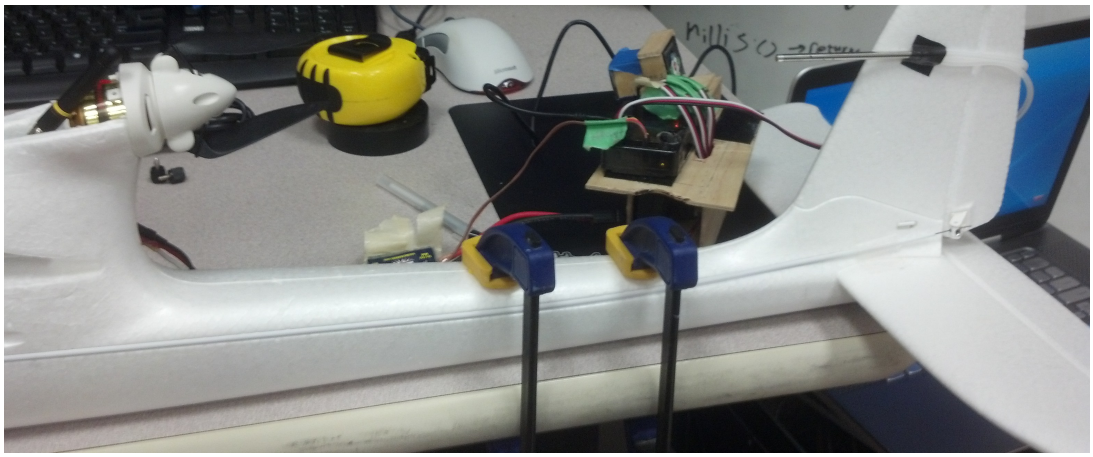


Figure 3.2: Experimental setup determining Easy Star motor dynamic response.

output δ_T is related to the commanded throttle δ_{T_C} by:

$$\dot{\delta}_T = \alpha(\delta_{T_C} - \delta_T) \tag{3.6}$$

Using the above assumption and a first-order forward difference approximation for $\dot{\delta}_T$, a least-squares solution for α is computed using the data from all five tests. It is assumed that thrust is proportional to V^2, so the value of α can be computed by a best fit for V^2 as a function of the recorded throttle inputs. The least-squares solution for α is 1.90. Fig. 3.3 plots the measured time histories against the computed response using Eq. 3.6 with the same inputs and initial conditions. The motor dynamics are also included in the dynamic simulation and QFT transfer functions.

3.4.2 Parametric uncertainty

Aircraft stability properties suffer from large uncertainties in modeling when flight test data are not available. Typical errors are on the order of 20% error (see Table 3.1). Additive uncertainty is assumed in modelling uncertain longitudinal and lateral-directional plants for the QFT control synthesis. Because typical model accuracy

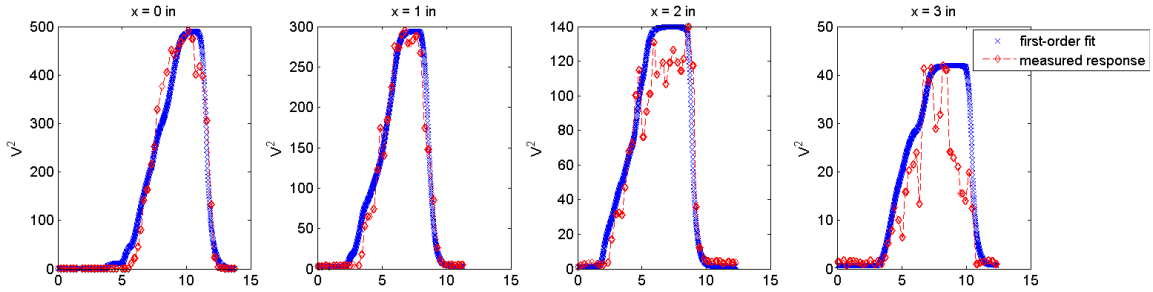


Figure 3.3: Measured and least-squares velocity profile time response from Easy Star motor. x indicates distance radially outward from the thrust centerline.

differs between the longitudial and lateral/directional axes, a different approach is used to populate the set of uncertain plants for each axis:

1. For the longitudinal axis, the uncertainties in the nondimensional coefficients listed in Table 3.1 are assumed to encompass the most significant model uncertainty. Continuous-time models of the form of Eq. 3.7 are assumed. Nondimensional derivatives are computed by performing elementwise division of the continuous $[A]$ and $[B]$ matrices with the matrices of Eq. 3.8. The extrema of the appropriate stability derivatives are computed, and the factorial set of uncertain derivatives is used to populate the dimensional continuous-time linear coefficients.

2. For the lateral-directional axis, stability derivative errors may be as high as 90% [43], which is infeasibly large for use with QFT. Instead, each of the dimensional $[A]$ and $[B]$ matrix entries for non-kinematic equations is assumed to have additive errors of up to $\pm 20\%$ of the nominal value. (Rudder is not used in the autolanding control laws, so rudder derivative errors are not considered.)

Derivative	Accuracy
C_{L_α}	$\pm 5\%$
C_{m_α}	$\pm 10\%$
C_{D_α}	$\pm 10\%$
C_{L_u}	$\pm 20\%$
C_{m_u}	$\pm 20\%$
C_{D_u}	$\pm 20\%$
C_{L_q}	$\pm 20\%$
C_{m_q}	$\pm 20\%$
C_{D_q}	$\pm 20\%$

Table 3.1: Typical longitudinal-axis modeling errors.[43]

$$\begin{bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u & X_\alpha & X_q & -g\cos\theta_1 \\ \frac{Z_u}{U_1} & \frac{Z_\alpha}{U_1} & \frac{Z_q}{U_1}+1 & \frac{-g\sin\theta_1}{U_1} \\ M_u & M_\alpha & M_q & M_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ \alpha \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} X_{\delta_e} & X_{\delta_T} \\ \frac{Z_{\delta_e}}{U_1} & \frac{Z_{\delta_T}}{U_1} \\ M_{\delta_e} & M_{\delta_T} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_T \end{bmatrix} \tag{3.7}$$

$$\begin{bmatrix} -\frac{\bar{q}S}{mU_1} & -\frac{\bar{q}S}{m} & -\frac{\bar{q}S\bar{c}}{2mU_1} & 1 \\ -\frac{\bar{q}S}{mU_1} & -\frac{\bar{q}S}{m} & -\frac{\bar{q}S\bar{c}}{2mU_1} & 1 \\ \frac{\bar{q}S\bar{c}}{I_{yy}U_1} & \frac{\bar{q}S\bar{c}}{I_{yy}} & \frac{\bar{q}S\bar{c}^2}{2I_{yy}U_1} & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & \frac{\bar{q}S}{m} \\ 1 & 1 \\ \frac{\bar{q}S\bar{c}}{I_{yy}} & 1 \\ 1 & 1 \end{bmatrix} \tag{3.8}$$

For both the longitudinal and lateral/directional axes, the set of uncertain plants is populated by computing the extrema of each uncertain matrix element, then computing the set of models using factorial combinations of these extrema. The nominal model is also retained for the control design process. There are a total of 32,768 lateral-directional models and 2048 longitudinal models.

3.5 Pegasus system identification

An accurate model of the Pegasus system is needed for control synthesis and verification in simulation. Traditional longitudinal and lateral/direction axis models are identified. System identification is performed in three primary steps: data collection in flight test, preprocessing for model fitting, and model fitting using Observer-Kalman filter Identification (OKID) and verification.

3.5.1 Flight test

Flight tests of the Pegasus aircraft were conducted under manual control between September and November 2013. A summary of flights is given in Table 3.2. Longitudinal-axis maneuvers consist of an elevator doublet followed by a throttle doublet. Lateral/directional maneuvers consist of a rudder doublet followed by an aileron doublet. A total of fifteen longitudinal axis trials and thirty-two lateral/directional axis trials are conducted over three flight days.

To capture high-frequency content of the system response, aircraft states are recorded at 50 Hz except for position measurements, which are recorded at 5 Hz (see Table 3.3). An APM version 1 is used to collect and log data[10]. The autopilot provides position measurements, altitude, and three-axis acceleration and angular rate measurements and attitude estimates. The IMU is less precise than that of the APM2 used in later flights[10], a fact that will be discussed later. Aerodynamic data are measured by an Aeroprobe five-hole probe that measures airspeed, angle of attack, and sideslip angle. A custom pressure sensor board reads probe values at 50 Hz and communicates them to the APM 1 using the I^2C communications protocol.

Date	Time	Wind (mean/max)	Objectives	Center of gravity	Weight (lbf)	Time of flight (min)
2014/09/14	10:48 CDT	2/7 mph	Longitudinal	51 in	93.5	12.8
2014/11/02	17:57 CST	6/13 mph	Lateral/directional	49 in	99	8.0
2014/11/03	14:32 CST	4/12 mph	Lateral/directional	49 in	99	6.7
2014/11/03	15:04 CST	4/12 mph	Longitudinal	49 in	99	9.2

Table 3.2: Test matrix for Pegasus system identification flight tests. Daily average and high steady wind values, measured at a nearby weather station, are shown[50]. Center of gravity is expressed as the distance along the body 1-axis aft of the vehicle nose.

Measurement	Units	Sample rate (Hz)
Inertial position	degrees latitude/longitude meters altitude	5
Three-axis acceleration	m/s^2	50
Three-axis angular velocity	rad/s	50
Barometric altitude	meters	50
Three-axis attitude estimates	radians	50

Table 3.3: Measured states, units, and sample rates for Pegasus system identification.

3.5.2 Data processing

Angular rates and aerodynamic angles are prefiltered using Butterworth filters in MATLAB^{TM} to reduce measurement noise[33]. In some initial longitudinal axis trials, the attitude estimate from the autopilot appears to approach non-zero steady-state values after excitation, which is inconsistent with the observed behavior during flight. This is interpreted as a possible effect of gyro bias, or simply poor accuracy in attitude estimation. In addition, for two of three flight days, performance of the air data probe is inconsistent. Angle-of-attack and sideslip angle measurements are not usable on two days, and on one of those days airspeed readings are also unusable. The process used to address these issues is described in this section.

3.5.2.1 Attitude estimation

To obtain the Euler angle position history, an extended Kalman filter (EKF) is implemented based on the method outlined in Ref. [30]. This filter is used to process all subsequent data and provide three-axis attitude and gyro bias estimates. The following procedure is used, and is also summarized in the flowchart of Fig. 3.4:

1. GPS-derived inertial position histories are differentiated once to estimate inertial velocity in three axes.

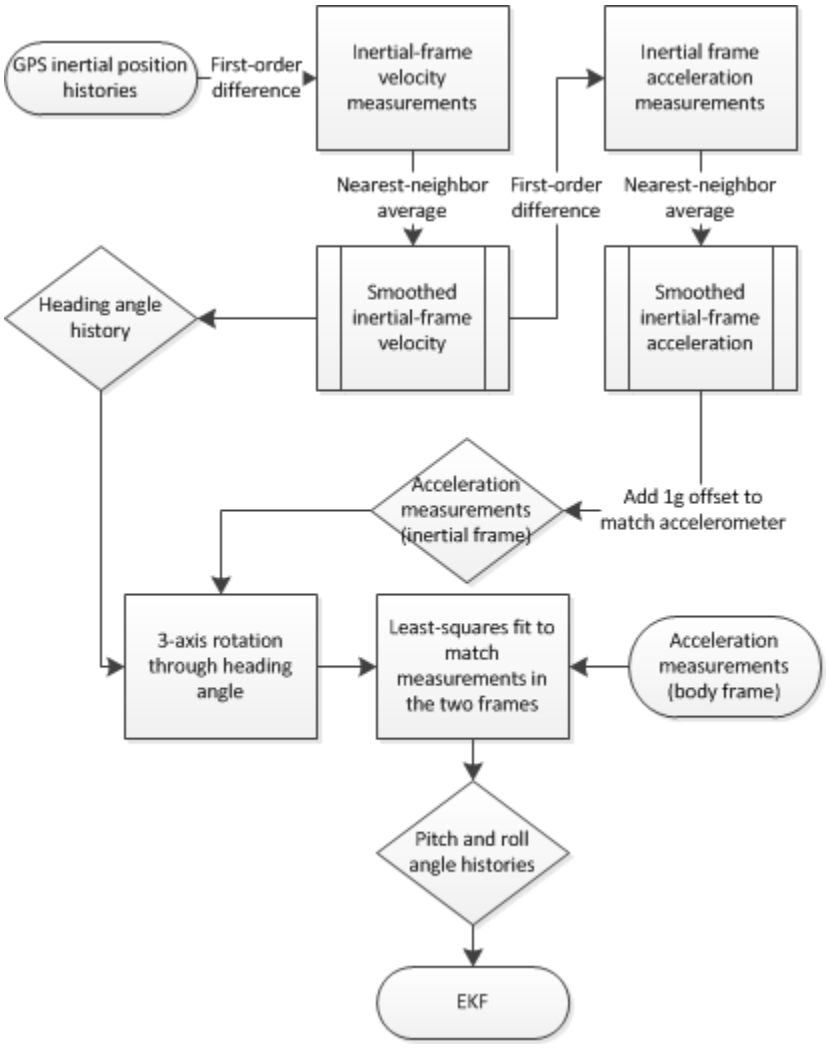


Figure 3.4: Flowchart of attitude estimation from GPS time histories.

2. To reduce noise, a two-term moving average of inertial velocity is taken and used subsequently.

3. In a north-east-down inertial coordinate system, aircraft heading is estimated from $\psi = \arctan \frac{V_{east}}{V_{north}}$.

4. Velocity histories are differentiated to produce acceleration histories, and a two-term moving average is again taken to reduce noise.

5. A value of $-g\hat{\mathbf{n}}_3$ is added to the inertial acceleration histories for consistency with accelerometer measurements, in which n^+ is the inertial coordinate system.

6. The inertial acceleration vector is transformed through a three-axis rotation through ψ into an intermediate reference labeled i^+.

7. The transformed acceleration histories are now related to the aircraft body frame by a 2-axis rotation through θ and a 1-axis rotation through ϕ. In theory this vector should match the accelerometer measurements after rotating through the two still-unknown angles.

8. ϕ and θ are estimated in a least-squares sense by solving the following linearized transformation equation relating the intermediate frame i^+ to the body frame b^+:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_b = \begin{bmatrix} 1 & 0 & -\theta \\ 0 & 1 & \phi \\ \theta & -\phi & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_i \tag{3.9}$$

9. The resulting ϕ and θ histories are treated as measurement updates in a continuous-discrete EKF.

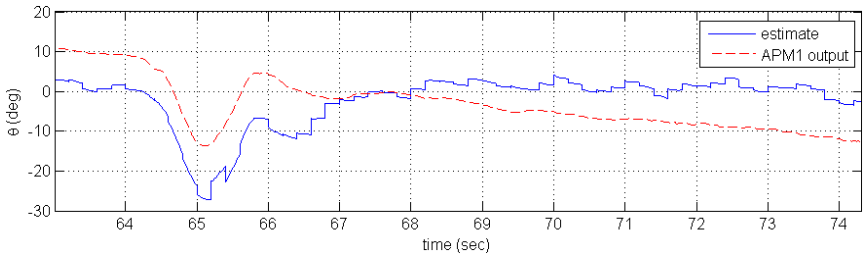


Figure 3.5: Comparison of pitch attitude estimates from flight data and from the EKF for one longitudinal-axis maneuver.

Attitude and gyro bias estimation is a familiar estimation problem with many existing solutions. The EKF attitude estimator in Ref. [9] is modified to use the Euler angles directly instead of the quaternion to parameterize attitude, simplifying the filter by eliminating the need to enforce quaternion normality. A sample of the autopilot and EKF estimates is shown in Fig. 3.5.

3.5.2.2 Aerodynamic angle estimation

The air data measurements are not usable from some flight days. The GPS-derived inertial velocity history and attitude estimates are used to approximate airspeed and aerodynamic angles as necessary. Body-frame velocity components U, V, W are computed by transforming the inertial frame velocity vector. Eqs. 3.3 and 3.4 are used to approximate the aerodynamic angles, with the assumption that the external wind is small relative to the magnitude of the aircraft velocity. Airspeed measurements are judged to be usable from two of three flight days by comparing them to the body 1-axis inertial speed, but all of the aerodynamic data were unusable from the final day of testing. Direct airspeed measurements are found to yield better models and are used whenever available. Since aerodynamic angles are usable on only one flight day, the approximation from inertial velocities described above was used on all flights for consistency in comparing model fits on different flight days.

3.5.3 Model fitting with OKID

Observer-Kalman Filter Identification is used to identify the linear models used for QFT control design. OKID is an established algorithm for system identification[26] and has been used previously in identifying aircraft models from flight data[32]. The algorithm directly fits a discrete-time state-space model, with the following form, to measured outputs and inputs:

$$\mathbf{x}_{k+1} = [A]\mathbf{x}_k + [B]\mathbf{u}_k \tag{3.10}$$

$$\mathbf{y}_k = [C]\mathbf{x}_k + [D]\mathbf{u}_k \tag{3.11}$$

In Eq. 3.10, $\mathbf{x}_k$ is the value of the state at time $t = k\Delta T$, where ΔT is the sample period for the discrete system. $\mathbf{u}_k$ and $\mathbf{y}_k$ are the control inputs and measured outputs at time $t = k\Delta T$. To identify models from the processed flight data, the data are manually segmented into each longitudinal or lateral/directional maneuver. A model is then fitted to each data segment. For validation, control inputs are fed back into the identified model, starting at the measured initial state, and the measured and predicted responses are compared.

Models are verified against both the data used in identification and from other segments. Both mean squared error (MSE) and the Theil inequality coefficient (TIC) are used as metrics of fit. MSE is defined in terms of the measured output $\tilde{y}_k$ and predicted output $\hat{y}_k$ as in Eq. 3.12:

$$MSE = \frac{\sum_{k=1}^{N}(\tilde{y}_k - \hat{y}_k)^2}{N} \tag{3.12}$$

The TIC is defined as follows:

$$TIC = \frac{\sqrt{1/N \sum_{k=1}^{N} (\tilde{y}_k - \hat{y}_k)^2}}{\sqrt{1/N \sum_{k=1}^{N} \tilde{y}_k^2} + \sqrt{1/N \sum_{k=1}^{N} \hat{y}_k^2}} \tag{3.13}$$

Eq. 3.13 yields a vector whose members are between 0 and 1, and can be evaluated across one data set or several. $TIC = 0$ implies the predictions match the data exactly and $TIC = 1$ implies maximum inequality. Acceptable values vary, but a range of $0.25 < TIC < 0.3$ is considered to indicate good agreement[25].

The longitudinal model fits linear perturbation models for airspeed u, angle-of-attack α, pitch rate q, and pitch angle θ. One model is fit to each segment of longitudinal axis flight test maneuvers using OKID. Comparisons of measured outputs with those predicted by the identified model are shown in Fig. 3.6 for two flight maneuvers. The lateral/directional model fits perturbations in sideslip angle β, roll rate p, yaw rate r, and bank angle ϕ as a linear function of these states and aileron and rudder inputs. Fig. 3.7 shows a comparison of measured outputs and model-predicted outputs for three lateral/directional trials. A detailed summary of the model identification and selection process for both axes is given in Appendix B.

3.5.4 Parametric uncertainty

Uncertain linear models are generated for Pegasus by assuming additive errors on the continuous-time state and control coefficients. Uncertainty is assumed to be in non-kinematic states only (that is, no uncertainty is associated with the Euler angle propagation). The objective in defining the uncertain sets is to select the maximum errors for which acceptable control performance can still be achieved. For the longitudinal axis, each continuous-time state coefficient is assumed to have an uncertainty of $\pm 5\%$ with control coefficient errors of $\pm 10\%$. For the lateral/directional axis, the

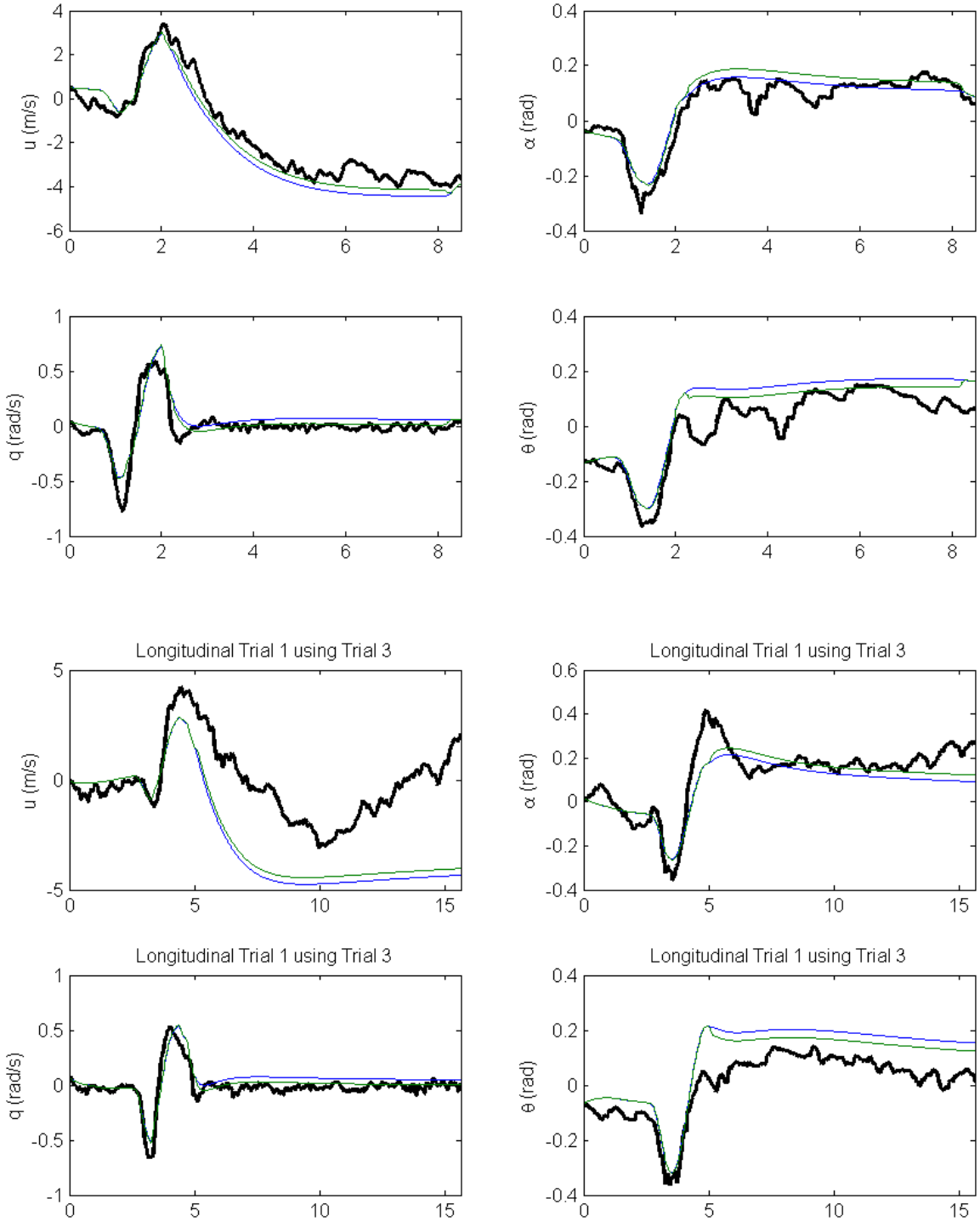


Figure 3.6: Comparison of longitudinal axis time histories and Pegasus model predicted outputs. Flight data are indicated by the thick black lines.

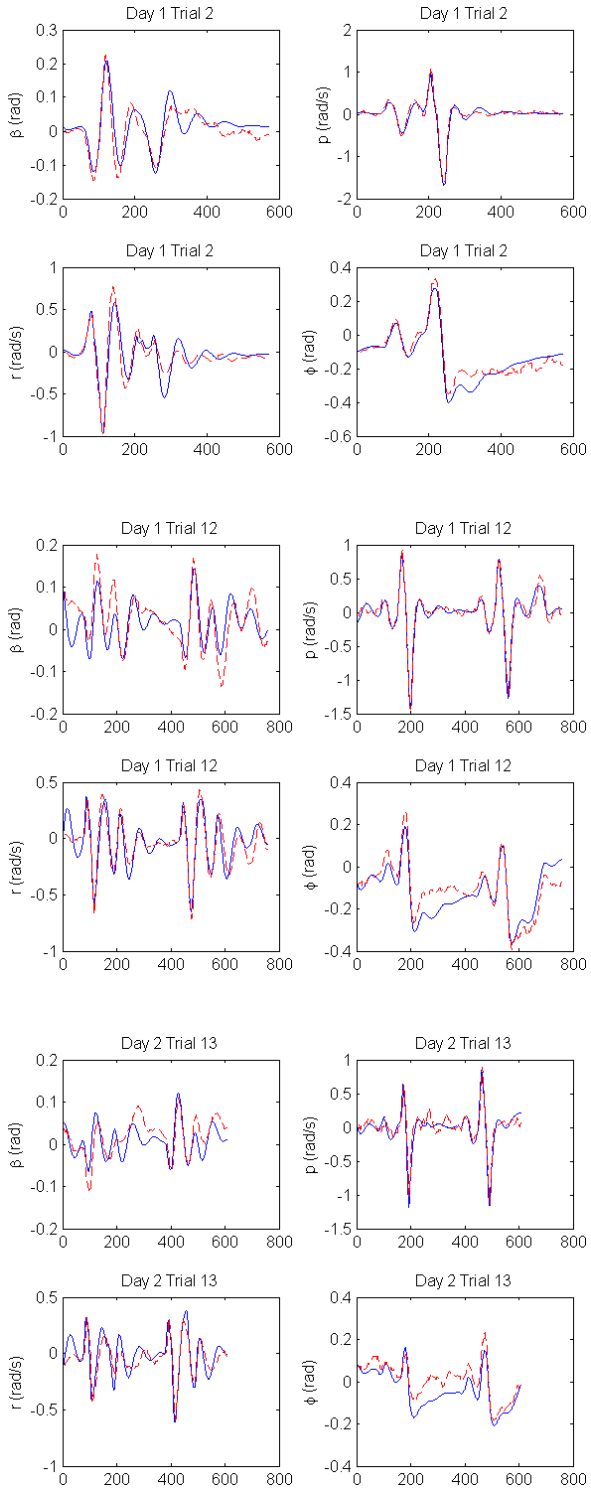


Figure 3.7: Sample model fitting for selected Pegasus lateral/directional model. Solid lines indicate model predicted outputs.

state and control coefficient errors are ±5% and ±20% respectively. The factorial set of combination of the extrema of these variables is used to generate discrete sets of uncertain models for each axis.

3.6 Sensor modeling

To implement the automatic landing feedback control laws, the following aircraft states must be measured or estimated:

- Body 1-axis airspeed U
- Inertial position X, Y, Z
- Body attitude ψ, θ, ϕ

The APM[10] includes a full sensor suite capable of providing the required measurements for normal flight. An ultrasonic rangefinder is added to the standard sensor package for accurate height-above-ground measurements in landing. The sensors used, and errors assumed, are shown in Table 3.4. The remainder of this section details the sensors and assumptions used in modeling them for simulation purposes.

3.6.1 Global positioning system

The APM uses a MediaTek MT3329 GPS running at 5 Hz. Data from the manufacturer indicate that the unit provides latitude/longitude measurements with a 2.5 m RMS error, and inertial north/east velocity is accurate to 0.05 m/s[36]. Due to the limited altitude accuracy of GPS, raw measurements from this sensor are not used to measure altitude or rate of change of altitude during flight.

3.6.2 Airspeed sensor

An airspeed kit using a MPX7002DP differential pressure sensor is employed. This sensor measures the differential pressure Δp with typical errors of 2.5% of the

Sensor	measurement	assumed 3σ error bound	units
MediaTek MT3329 GPS	Global north-east position	2.5	meters
MPX7002DP differential pressure sensor	Body 1-axis airspeed	100 (≈6.5)	Pascals (m/s)
Measurement Specialties MS5611-01BA03 barometric pressure sensor	Altitude (h>6 m)	250 (≈20.8)	Pascals (meters)
AHRS	Heading, pitch, roll angles	11.1,11.1,10.3	degrees
XL-MaxSonar-EZ4 ultrasonic rangefinder	Altitude (h<6 m)	2.5	centimeters

Table 3.4: Sensors and assumed errors for simulation.

full scale reading, and maximum errors of 6.25% full scale[12]. The rated full scale value is 4 kPa, so the sensor noise is assumed to be Gaussian white with a typical 3σ value of 100 Pa. Airspeed is sampled at 10 Hz. Since the air vehicle will operate at low altitudes well within the subsonic regime, airspeed V is calculated from differential pressure by the incompressible Bernoulli equation $\Delta p = \frac{1}{2}\rho V^2$. Here, ρ is assumed to be the sea-level air density of 1.225 $\frac{\text{kg}}{\text{m}^3}$.

To reduce the effects of sensor noise, the APM uses the following lowpass filter equation to estimate the current airspeed, V_k, from the current differential pressure reading, Δp_k, and the previous measured airspeed V_{k-1}:

$$V_k = 0.7V_{k-1} + 0.3\sqrt{1.9936\Delta p_k} \tag{3.14}$$

3.6.3 Barometric sensor

The APM uses a Measurement Specialties MS5611-01BA03 barometric pressure sensor to determine vehicle altitude. This sensor's maximum error band is given

as ± 250 Pa[35], which corresponds to a 3σ error of approximately 20 m at sea level. However, each time the sensor is read in the autopilot software, four pressure measurements are taken and averaged to reduce measurement noise; this repeated sampling is included in simulation. Altitude is updated at 10 Hz in the autopilot loop. To determine the measured altitude $\tilde{h}$, the autopilot uses the following equation, in which $p_{h=0}$ and $T_{h=0}$ indicate the pressure and temperature when the barometer is calibrated upon initialization:

$$\tilde{h} = 29.271267 \ln(\frac{p_{h=0}}{p})(T_{h=0} + 273.15) \tag{3.15}$$

To reduce the effects of sensor noise, the barometer readings are lowpass filtered at the nominal update rate of 10 Hz with a 0.7 Hz cutoff frequency. The current barometer measurement, h_k, is updated by the following digital filter before being used in the autopilot loop:

$$h_k = h_{k-1} + 0.30547(h_k - h_{k-1}) \tag{3.16}$$

3.6.4 Attitude and heading reference system

The APM's attitude and heading reference system (AHRS) estimates the aircraft attitude from gyroscope measurements. Attitude is parameterized with a 3/2/1 Euler angle rotation sequence through $\psi/\theta/\phi$. The Digital Motion Processor used for attitude determination on the MPU-6000 IMU is not documented[24]. Rather than replicating the complete APM calculations in simulation, the AHRS error performance is analyzed experimentally. Images of the setup used in the experiment are shown in Fig. 3.8. In the experiment, a rotating platform is driven by a Hitec HS-311 servo. The APM is then attached to the platform with either the roll, pitch, or yaw

axis normal to the platform. The servo rotates the APM through 90° at an average rate of 66.67° s^{-1}, holds for one second, then returns to 0° at the same rate. This rotation is performed three times for each axis to improve accuracy.

A first-order discrete-time model is fit to each axis in terms of the "true" angle at time t_i, $(\theta_t)_i$ (as determined by the angle commanded to the servo), the previous angle measurement θ_{i-1}, a constant offset b_θ, and a standard deviation σ_θ:

$$\theta_i = a\theta_{i-1} + b(\theta_c)_i + b_\theta + \mathcal{N}(0, \sigma_\theta^2) \tag{3.17}$$

The measured standard deviation between the AHRS outputs and the predicted outputs based on the model is taken as σ_θ. This metric admits some more error than may be actually present in the AHRS, since the servo actuator dynamics are not accounted for. Since the actuation rate is much lower than the rated limit of the servo[47], this error is assumed to be relatively small, and the additional error is retained as a conservative estimate. The 3σ error limits for the AHRS are shown in Table 3.4. The pitch-axis fit is shown in Fig. 3.9.

3.6.5 Ultrasonic rangefinder

Preliminary landing simulations using the standard sensor package make it clear that the worst-case barometer error is too large for reliable autoflare maneuvers. Because of the minimal available onboard processing power, typical filtering schemes are considered impractical for implementation on the Easy Star. Lightweight ultrasonic rangefinders can provide accurate range data, but typically are limited in maximum range to distances right above the planned autoflare altitude. An ultrasonic rangefinder cannot be used to determine altitude throughout the flight envelope, but can be incorporated for use exclusively during the flare maneuver.

The XL-MaxSonar-EZ4 provides range data with nominal 1 cm accuracy at ranges

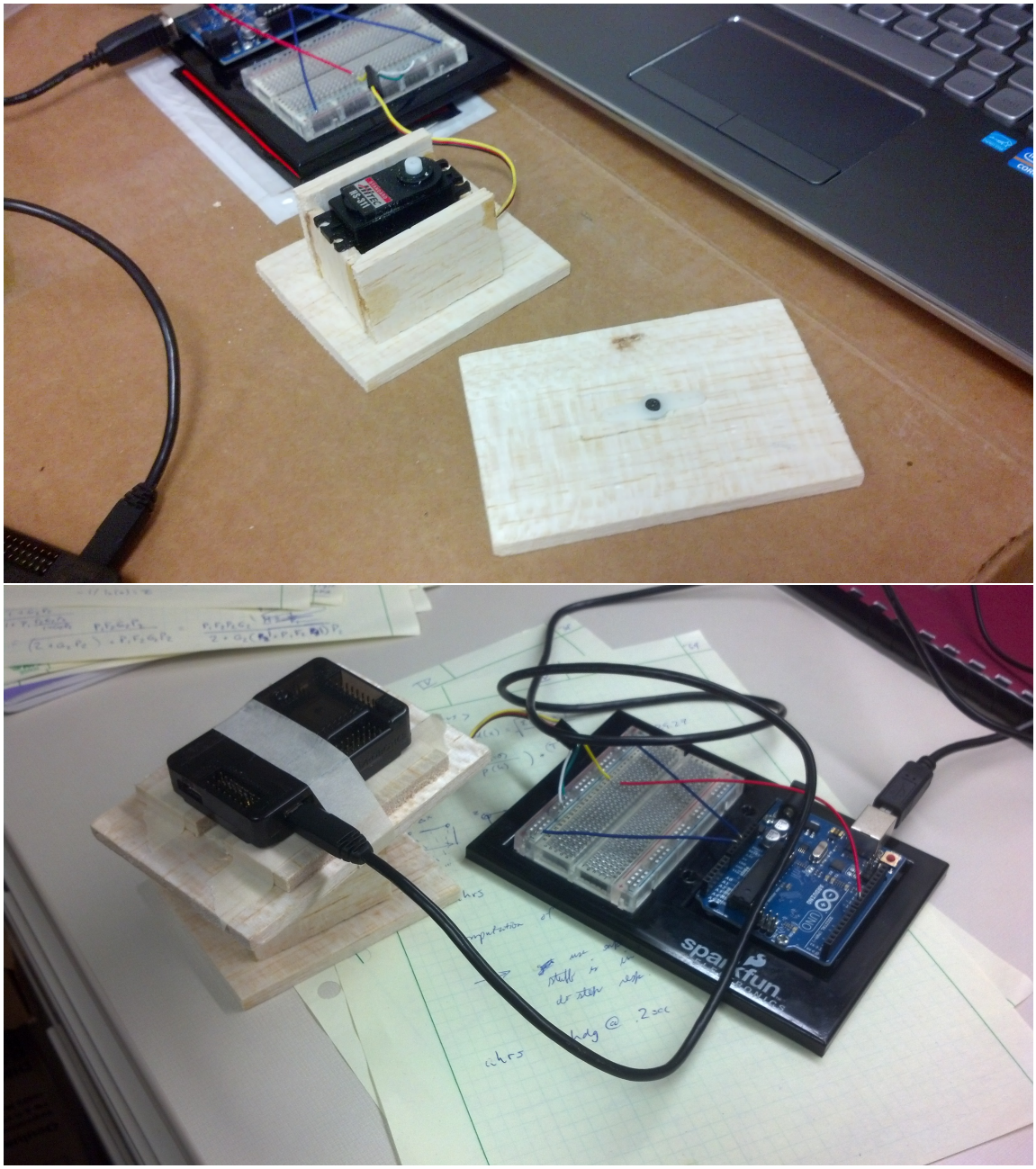


Figure 3.8: Test setup for APM2 AHRS error estimation.

between 20 cm and 7.6 m[1]. Furthermore, it has a footprint of approximately 1" by 1" and weighs only 5.9 grams. At distances less than the minimum or greater than the maximum sensitive range, it is assumed the sensor returns the minimum

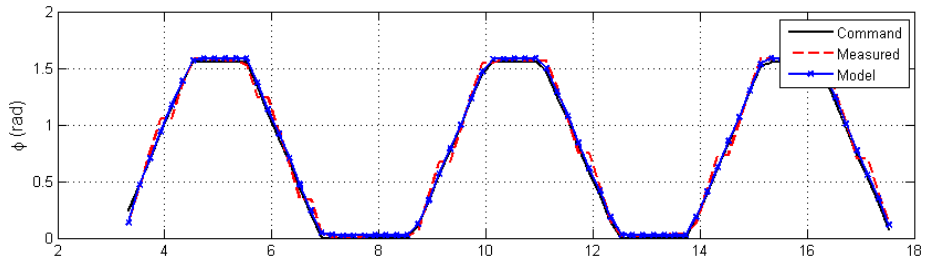


Figure 3.9: Comparison of commanded, measured, and modeled APM2 pitch axis response in experiment.

or maximum range, respectively. As a worse-case estimate, a maximum effective 3σ error bound of 25 cm is assumed to be present in the range readings for simulation. This value is considered to be extremely conservative.

On Pegasus, the sensor is so small that it can be installed almost anywhere; for simulation purposes, it is assumed to be located at approximately the center of mass (i.e., the vector from the center of mass to the sensor is negligibly small). To avoid damage to the sensor during landings on the Easy Star, the sensor must be embedded in the Easy Star's wing, aligned with the vehicle $\hat{\mathbf{b}}_3$ axis. Since the sensor is aligned with the vehicle 3-axis and not the inertial 3-axis, the vehicle attitude must be considered in computing the relationship between vehicle altitude h and sensor measured range R_s. This relationship is used in simulating the sensor response at a given vehicle state.

The position of the sensor relative to the vehicle center of mass, $\mathbf{r}_s$ is assumed to be primarily along the body 2 axis, $\hat{\mathbf{b}}_2$, such that $\mathbf{r}_s = r_s\hat{\mathbf{b}}_2$. The sensor height-above-ground is given by $h_s = h + r_s \sin\phi\cos\theta$. h_s is simply the projection of the sensor's measured range vector, $\mathbf{R}_s$, onto the inertial $\hat{\mathbf{n}}_3$ axis. The measured range

R_s and vehicle altitude h are related by:

$$R_s \cos\phi \cos\theta = h_s = h + r_s \sin\phi \cos\theta \tag{3.18}$$

3.7 Turbulence modeling

The turbulence model implemented is the Dryden wind turbulence model[34] from MIL-F-8785C[2], which gives 3-DOF additive disturbance forcesas functions of altitude and gust intensity. The turbulence is given in an inertial frame aligned with the mean wind direction and transformed into the body axes, where it results in additive noise. Each component of the turbulence is obtained by passing band-limited white noise through the corresponding transfer function. For example, the turbulence aligned with the mean wind direction is found by passing white noise through $H_u(s)$. The governing equations are given in state-space form by Ref. [34], and units of feet and feet/second are used for distance and speed:

$$\begin{aligned} H_u(s) &= \sigma_u \sqrt{\frac{2L_u}{\pi V}} \frac{1}{\frac{L_u}{V}s + 1} \\ H_v(s) &= \sigma_v \sqrt{\frac{2L_v}{\pi V}} \frac{\frac{2\sqrt{3}L_v}{V}s + 1}{(\frac{2L_v}{V}s + 1)^2} \\ H_w(s) &= \sigma_w \sqrt{\frac{2L_w}{\pi V}} \frac{\frac{2\sqrt{3}L_w}{V}s + 1}{(\frac{2L_w}{V}s + 1)^2} \end{aligned} \tag{3.19}$$

The variables L_u, L_v, L_w are functions of altitude h:

$$\begin{aligned} L_u = 2L_v &= \frac{h}{(0.177 + 0.000823h)^{1.2}} \\ L_w &= \frac{h}{2} \end{aligned} \tag{3.20}$$

The turbulence intensities $\sigma_u, \sigma_v, \sigma_w$ are given by:

$$\sigma_u = \sigma_v = \frac{0.1W_{20}}{(0.177 + 0.000823h)^{0.4}}$$
$$\sigma_w = 0.1W_{20} \tag{3.21}$$

In Eq. 3.21, W_{20} is the turbulence intensity, characterized by the wind speed at a 20 ft altitude. The Pegasus is currently restricted to autonomous operations below 400 ft above ground level, or roughly 750 ft above sea level. Based on the probability of turbulence intensities as given in Ref. [34], it is reasonable to expect turbulence intensities no greater than 10 ft/s $\approx$ 3 m/s at operational altitudes of 400 ft or less, with a less than .001 probability of exceedance.

3.8 Dynamic simulation

A key focus of this research is the validation of the control laws through dynamic simulation. The primary objective of dynamic simulation is to evaluate and quantify the controller's robustness to the various uncertainties present in the modelling: process and measurement noise as well as dynamic uncertainty in the system models.

Controller synthesis and initial validation is performed in MATLAB^{TM}. The inner QFT loops (heading, roll, pitch, and airspeed) are evaluated by simulating the step response of the set of uncertain linear plants selected for control design. No sensor uncertainty is implemented. For the outer loops, the potential nonlinear effects of large Euler angles are considered important. A full 12-DOF simulation is conducted using the identified linear models to propagate the vehicle's velocity-level variables, while the nonlinear kinematic equations of motion are used to propagate the translational and rotational states. To address model uncertainty, simulations are conducted using each of the uncertain lateral/directional or longitudinal models. Lateral/directional uncertainty is included when testing the azimuth tracking,

and longitudinal uncertainty is assumed when testing glideslope tracking and autoflare. For preliminary controller evaluation, no process or measurement noise is incorporated.

After the full automatic landing controller has been synthesized, Monte Carlo simulations are conducted to analyze the robustness to various uncertainties. For both vehicles, model uncertainty is considered critical. 12-DOF simulation with nonlinear kinematics is conducted using one randomly selected longitudinal and lateral-directional model from each of the uncertain sets to propagate the velocity-level states. The remaining uncertainty considered varies depending on the vehicle. For the Easy Star, prevailing winds must be low for a flight to be permitted, so the effects of turbulence are considered negligible relative to the sensing uncertainty outlined in the previous section. The effects of steady crosswinds and tailwinds on the landing are considered as the worst-case external disturbances. Steady crosswinds are transformed into the body axes and act as additive noise on the vehicle's u, α, and β perturbed states. For Pegasus, which is two orders of magnitude heavier than the Easy Star, turbulence effects are considered more significant than steady winds since the rated gust limits for Pegasus are much higher than for the Easy Star. Sensor noise is once again considered as a significant influence on vehicle behavior.

The initial control design and nonlinear evaluation for both vehicles was performed in MATLAB$^{\text{TM}}$. The primary robustness Monte Carlo simulations for the Easy Star were also conducted in MATLAB$^{\text{TM}}$. In evaluating the continuous-time plant with discrete control updates, the run time of the code was found to be unacceptably slow, significantly limiting the number of Monte Carlo simulations that could be conducted. For Pegasus batch simulations, the code was ported to C++, which reduced the typical run time required for each simulation by roughly an order of magnitude. The freely available C++ Armadillo library[40] was used to implement

matrix operations; all other code including the Runge-Kutta 45 integrator was written form scratch for this research. The C++ simulation was validated by comparing results against MATLAB simulations with no uncertainty. The C++ simulation replaced the MATLAB nonlinear simulation for the Pegasus control design. It was also adapted for rapid evaluation of the Easy Star outer-loop controllers after initial flight tests demonstrated a need for control re-design.

This section describes the two UAVs and the basic assumptions and modeling used in implementing the dynamic simulations used to evaluate the performance of the automatic landing controllers. The next section presents a detailed description of the design and verification of the automatic landing controllers for each vehicle.

4. CONTROL DESIGN AND VALIDATION

QFT is used to develop robust feedback controllers with guaranteed tracking performance for the set of uncertain plants identified in Section 3. The QFT controller is designed as a set of sequential closed single-input single-output loops. A digital controller with an update rate of 10 Hz is synthesized. This update rate is compatible with real-time operation of the ArduPilot Mega autopilot, whose inner loop runs at 50 Hz with outer loops of 10, 5, and 1 Hz[10]. The basic structure of each loop consists of a prefilter $F(z)$, controller $G(z)$, and plant or inner loop model $P(s)$ or $P(z)$. The primary performance specifications of interest in each loop are:

$$\|\frac{P(z)G(z)}{1+P(z)G(z)}\| \leq SM \tag{4.1}$$

$$T_{rl}(s) \leq \|\frac{F(z)P(z)G(z)}{1+P(z)G(z)}\| \leq T_{ul}(s) \tag{4.2}$$

The first specification ensures robust stability margins (SM) to compensate for the uncertain plant model. The second ensures robust tracking performance in the frequency domain between the lower bound T_{rl} and the upper bound T_{ul}. Both performance criteria place constraints on the controller; the prefilter is restricted only by the tracking bound.

This section presents the detailed control synthesis and preliminary evaluation for both the Easy Star and Pegasus flight vehicles. For the Easy Star, the design of the lateral/directional loops is presented first, followed by the longitudinal-axis loops. Flight test results with this vehicle indicated that some loops should be redesigned, and, where applicable, new or modified controllers are also presented. Similarly, for Pegasus, the lateral/directional loops are presented followed by the longitudinal

control loops. In the Pegasus design, particular attention is given to performance of the outer control loops in the presence of turbulence, which proves to be a major research challenge for the identified dynamic model.

4.1 Easy Star controller design

4.1.1 Lateral/directional control

Lateral/directional control consists of three control loops: (1) an azimuth tracking loop, which generates reference headings ψ_{ref}; (2) a heading command and hold loop, which generates reference bank angles ϕ_{ref}; (3) a bank angle command and hold loop. The control loops are synthesized from the innermost loop outwards. This inner-outer methodology is implemented because of the desire for robust tracking in the inner loops. The bank angle command and hold transfer functions are driven by the large range of plant uncertainty. The performance of the outer loops is strongly influenced by the design of the bank angle loop.

4.1.1.1 Bank angle command and hold loop

The innermost lateral/direction loop is the bank angle command and hold loop. The nominal open-loop transfer function is $L_\phi = G_\phi P_\phi$. The stability margin and tracking bounds for the bank angle command and hold loop are as follows:

- Lower tracking bound: $T_{rl}(s) = \frac{1.25}{s^3+6s^2+5.25s+1.25}$
- Upper tracking bound: $T_{ul}(s) = \frac{0.5s+1.25}{s^2+s+1.25}$
- Stability margin: $SM = 1.4$

The plant uncertainty in P_ϕ is governed by parametric uncertainty in the identified linear model. The lateral/directional uncertainty assumptions have been detailed previously, and a large set of uncertain models has been synthesized. This set of models is impractically large for computing the robust stability and tracking bounds. To

identify a manageable subset of models that encompasses the full range of plant uncertainty, the Nichols plot of the open-loop plants at eight frequencies between $0.1\frac{\text{rad}}{\text{s}}$ and $30\frac{\text{rad}}{\text{s}}$ is created. At each frequency, the convex hull of the plotted templates is computed, and all plants that appear in the convex hull at one or more frequencies are retained. This technique is used for all the inner loops designed in the course of this research. The reduced set of models has 167 plants that represent the extrema of the plant uncertainties assumed.

To satisfy the performance constraints for the full set of plants, the following controller G_ϕ and prefilter F_ϕ are designed:

$$G_\phi(z) = \frac{z^4 - 2.5817z^3 + 2.1787z^2 - 0.5969z}{-1.6136z^4 + 2.6016z^3 + 0.03421z^2 - 1.4361z + 0.4139} \tag{4.3}$$

$$F_\phi(z) = \frac{z^2}{46.29z^2 - 68.812z + 23.517} \tag{4.4}$$

Figs. 4.1-4.2 show the closed-loop system responses with and without the prefilter F_ϕ. Fig. 4.1 is used to verify that the closed-loop system satisfies robust stability, and Fig. 4.2 verifies that the controller satisfies the robust tracking stipulated.

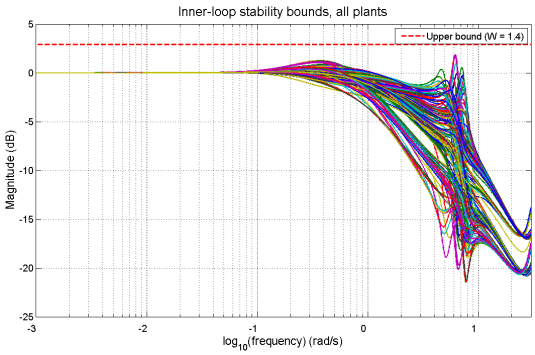


Figure 4.1: Bode magnitude plot showing the closed-loop response of the Easy Star bank angle system without prefilter F_ϕ.

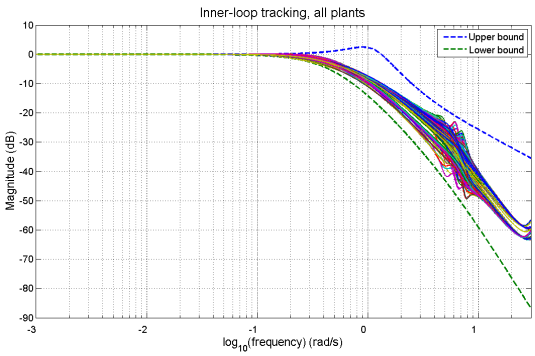


Figure 4.2: Bode magnitude plot showing the closed-loop Easy Star bank angle system response and tracking bounds.

4.1.1.2 Heading command and hold loop

The heading command and hold loop generates reference bank angle commands for the bank angle loop. The actual bank angle, which is the output of the bank angle command and hold loop, is the input to the heading angle plant P_ψ. The assumed transfer function from bank angle to heading angle is $\frac{\psi}{\phi} = \frac{g}{U_1 s}$. Since the gain is dependent on the steady-state aircraft speed, the gain is assumed to be the principle source of error in P_ψ, with extrema of $\pm$ 20% of the nominal value.

Due to the large uncertainties associated with the lateral/directional model, it is not practical to design the bank angle and heading angle loops independently. Rather, the inner loop prefilter and controller are designed iteratively with a heading angle prefilter to satisfy acceptable performance for the inner and outer loops simultaneously. The heading command and hold performance requirements are: (1) 90% rise time of between three and seven seconds; (2) maximum overshoot of less than 30%.

To improve the heading angle response, the following prefilter F_ψ is implemented:

$$F_\psi = \frac{z^3 - 2.9030z^2 + 2.8127z - 0.9084}{0.08783z^3 - 0.1746z^2 + 0.09119z - 0.004118} \tag{4.5}$$

Fig. 4.3 shows the system response to a 45° step command in heading. The heading state histories demonstrate acceptable performance. The aileron deflection histories, which include the effects of actuator dynamics, indicate that the maximum required control deflection is not excessively large (less than 10°). The rise time is between 4 and 6.3 seconds. The maximum overshoot is 31%. This performance is considered acceptably close to the specifications.

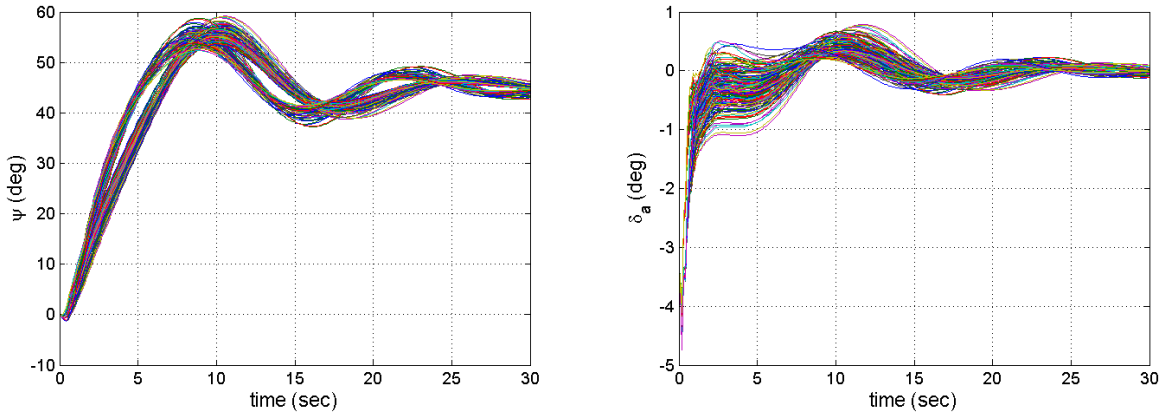


Figure 4.3: Response to a 45° commanded step change in heading with nominal Easy Star linear model.

4.1.1.3 Azimuth tracking loop

The outermost loop in the lateral-directional controller tracks the angle from the aircraft to the inertial origin in the $X - Y$ plane, λ, and generates reference heading commands to drive the angle to zero. Because the reference value of λ is zero, there is no pre-filter associated with this loop, and consequently no tracking requirements. A stability margin of 1.2 is stipulated for the azimuth tracking loop. A feedback controller $G_\lambda(z)$ is designed using QFT to ensure robust stability for the set of plants.

The nominal transfer function from ψ to λ is assumed to be $\frac{U_1}{Rs}$, in which R specifies the range to the origin. For robust stability, the value of $\frac{U_1}{R}$ is assumed to vary by $\pm$ 20% from a nominal speed of 11 m/s and range of 50 m. The principal challenge in designing the azimuth tracking loop is in designing G_λ to regulate λ quickly while achieving acceptable performance from an initial state far from the runway.

To satisfy the performance requirements, the following controller $G_\lambda(z)$ is implemented:

$$G_\lambda(z) = \frac{z^2 - 1.9934z + 0.9969}{0.5226z^2 - .9974z + .4766} \tag{4.6}$$

To analyze the performance of the lateral-directional controller, two sets of deterministic simulations are conducted: (1) the nominal plant is simulated for initial conditions of $X(0) = -500$ m, $Y(0) \in \{-500, -400, -300, -200, -100, 0\}$ m, and $\psi(0) \in \{45, 30, 15, 5, 0\}^\circ$; (2) each of the uncertain plants from the assumed linear model uncertainties is simulated for initial conditions of $(X, Y, \psi)(0) = (-500, -500, 45^\circ)$. For each of these simulations, the longitudinal perturbed state variables are assumed to be zero. The nonlinear translational dynamics including the effects of steady-state forward speed U_1 and steady-state pitch angle θ_1 are included.

Fig. 4.4 shows the results for the set of uncertain plants with the single initial condition. The set shows fairly uniform convergence to the origin. Near the origin, the trajectories are well within the width of the assumed 10 m wide runway. Fig. 4.5 shows the results for the nominal plant with varying initial conditions. The response

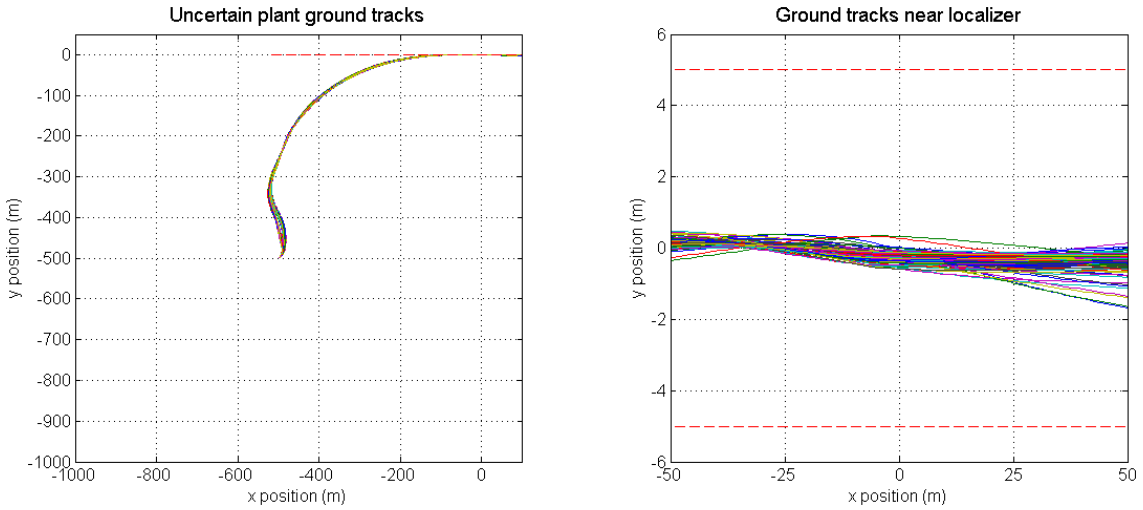
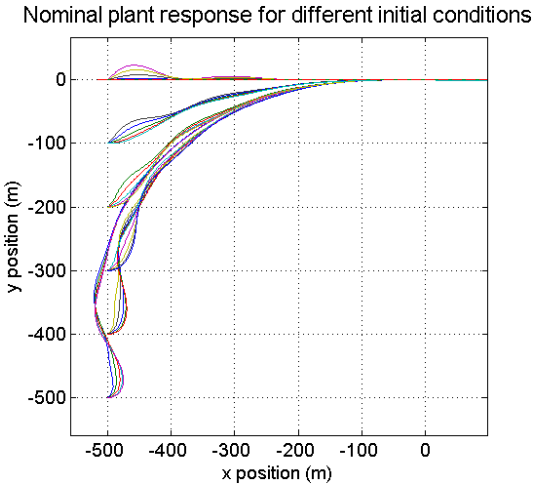


Figure 4.4: Uncertain Easy Star plant responses from initial conditions $(x, y, \psi) = (-500, -500, 45^\circ)$. The boundaries indicated are $y = \pm 5$ m.



2.80in

Figure 4.5: Nominal Easy Star plant groundtrack for various initial conditions

at each set of conditions is acceptable. Each trajectory passes through the origin to within 3 cm, with a maximum deviation from the runway center of less than 1 m within 60 m downrange of the origin.

4.1.1.4 Lateral/directional control redesign

Initial flight tests using the heading command and hold loop to track a constant heading indicate that the heading control loop initially designed is significantly underdamped, as in Fig. 4.6. In this test, the vehicle is flown under manual control, then the autopilot is engaged to regulate heading. The response is obviously underdamped, which is also true of the simulated response shown in Fig. 4.3. When the bank angle controller is evaluated with a reference bank of 0°, the response is relatively well-damped, so the heading loop is modified. A controller G_ψ is added to the heading control loop to improve performance. The azimuth tracking loop is also

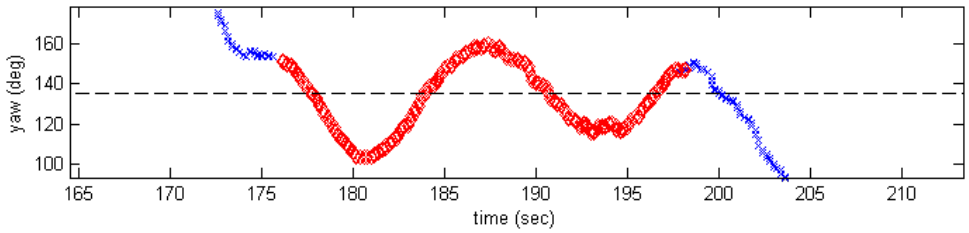


Figure 4.6: Initial Easy Star heading axis response to a hold of 135°.

updated to account for the redesigned inner loop. The new control laws are:

$$G_\psi = \frac{z - 0.9668}{0.03072z - 0.01192} \tag{4.7}$$

$$F_\psi = \frac{z^2 - 1.6567z + 0.6661}{15.268z^2 - 29.926z + 14.67} \tag{4.8}$$

$$G_\lambda = \frac{2.5001z^2 - 4z + 1.5}{z^2 - z} \tag{4.9}$$

Performance of the redesigned controller in flight test is presented in Section 6.

4.1.2 Longitudinal control

The longitudinal control is distinct for two flight phases: (1) glideslope tracking for most of the approach to the runway; (2) flare tracking immediately before touchdown. Glideslope tracking consists of independent control loops for tracking a target glideslope and maintaining constant airspeed. The flare maneuver is performed immediately before landing to reduce the vehicle's vertical speed.

4.1.2.1 Pitch angle command and hold loop

The glideslope tracker consists of a cascaded loop, with the inner loop being a pitch angle command and hold loop for the angle θ. Because the pitch-axis control is so critical to autonomous landing, the stability and tracking requirements are more

restrictive than in the lateral/directional inner loop.

$$SM = 1.2 \tag{4.10}$$

$$T_{rl}(s) = \frac{1.25}{s^3 + 6s^2 + 5.25s + 1/25} \tag{4.11}$$

$$T_{ul}(s) = \frac{.1389s + .6944}{s^2 + .7599s + .6944} \tag{4.12}$$

Uncertain plant models are generated using the approach outlined in Sec. 3.4.2. The full set of uncertain models is downselected using the convex hull of the Nichols plot plant templates. This produces a set of 56 candidate models, including the nominal model. The transfer functions $\frac{\theta(s)}{\delta_a(s)}$ are computed for the QFT control design.

To satisfy the performance requirements, the following controller $G_\theta(z)$ and pre-filter $F_\theta(z)$ are implemented:

$$G_\theta(z) = \frac{z^3 - 2.5588z^2 + 2.2991z - 0.7231}{-2.5636z^3 + 6.0048z^2 - 4.6755z + 1.2343} \tag{4.13}$$

$$F_\theta(z) = \frac{z^3 - 2.2095z^2 + 1.6386z - 0.3984}{23.0885z^3 - 63.6612z^2 + 58.7204z - 18.1170} \tag{4.14}$$

Fig. 4.7 shows the stability and tracking performance of the set of closed-loop plants in terms of the Bode magnitude plots. Both sets of requirements are met. Fig. 4.8 shows the step response of the set of closed-loop uncertain linear plant models. The set of closed-loop plants demonstrates the desired closed-loop rise time and overshoot characteristics.

4.1.2.2 Glideslope tracking loop

The reference glideslope is defined as the line passing through the inertial coordinate origin, aligned with the runway, with a slope of Γ_{ref}. The vehicle must track the

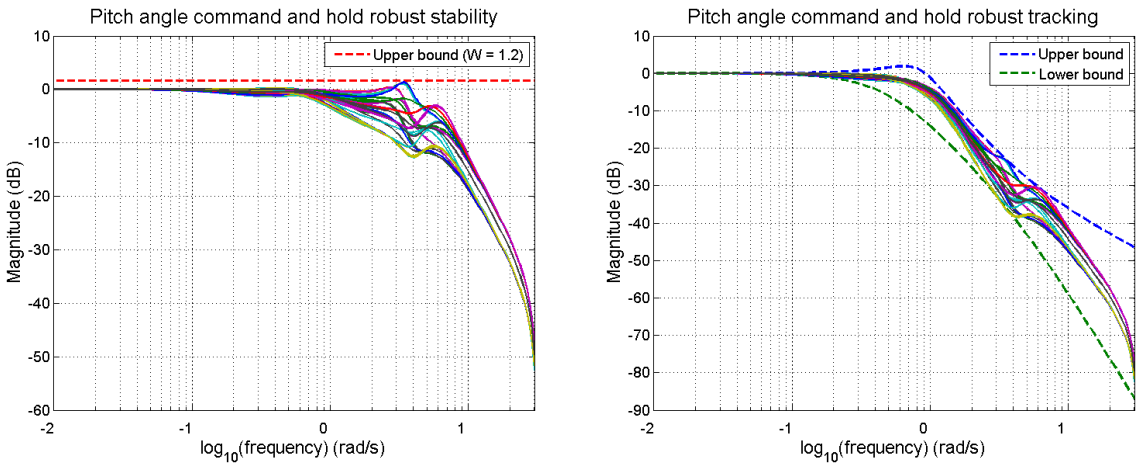


Figure 4.7: (Left) Robust stability margin for Easy Star closed-loop pitch angle control without prefilter. (Right) Robust tracking for the full pitch angle loop.

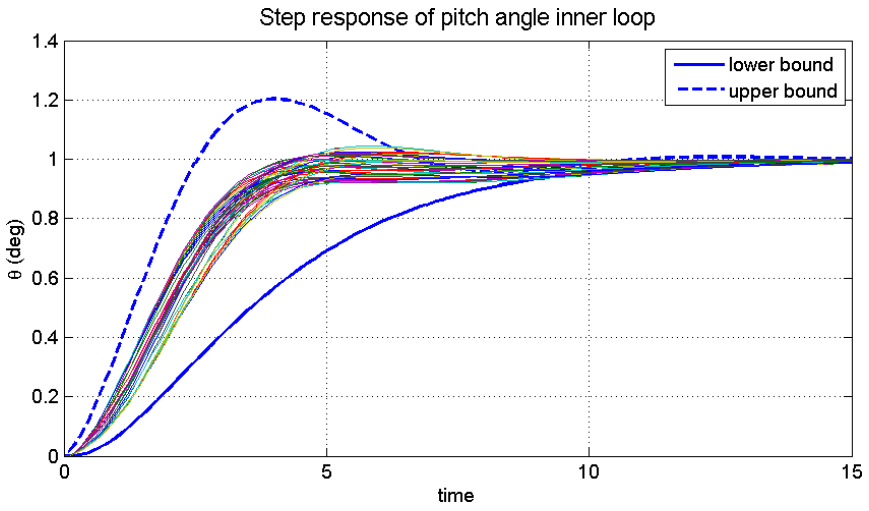


Figure 4.8: Step response for the closed-loop set of candidate longitudinal-axis Easy Star plants.

glideslope until it descends to the flare altitude, at which point the flare maneuver is performed. The reference glideslope is a constant, so no prefilter is implemented in the glideslope tracking loop. The tracking error $\Gamma_{ref} - \Gamma$ is fed into the controller G_Γ, which generates a reference pitch angle for the pitch angle command and hold inner loop.

The primary requirements of the glideslope tracking law are:

- Steady-state error of less than 1°.
- Robust tracking for initial glideslopes between 2°and 15°.
- Maximum elevator control deflections less than 15°.

A potentially large initial error in the glideslope is assumed because the automatic landing mode on the Easy Star may be triggered by a remote pilot during testing. The following controller is implemented to meet the requirements:

$$G_\Gamma = \frac{5.51z^2 - 6.5z + 1}{z(z-1)} \tag{4.15}$$

To evaluate the controller performance, two sets of simulations are conducted with a target glideslope of 5°: (1) The nominal model is evaluated for initial conditions $X \in \{-800, -500\}$ m, $\Gamma \in \{2°, 5°, 10°, 15°\}$; (2) The full set of uncertain models determined for the pitch angle control development is evaluated for initial conditions $X = -500$ m, $\Gamma = 15°$. For simulation, lateral/directional dynamics are neglected, and forward speed is assumed to be constant (due to the presence of a constant speed control loop, which has not yet been developed). The full nonlinear translational dynamics are incorporated. A first-order actuator with a 10 sec time constant is assumed for the elevator deflection. A flare height of 5.3 m is assumed, based on Ref. [49], and simulations terminate at this altitude.

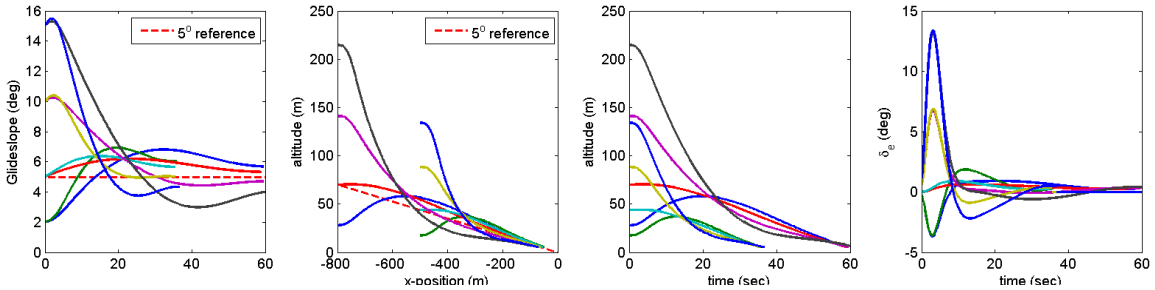


Figure 4.9: Results for the nominal Easy Star longitudinal plant at various initial conditions

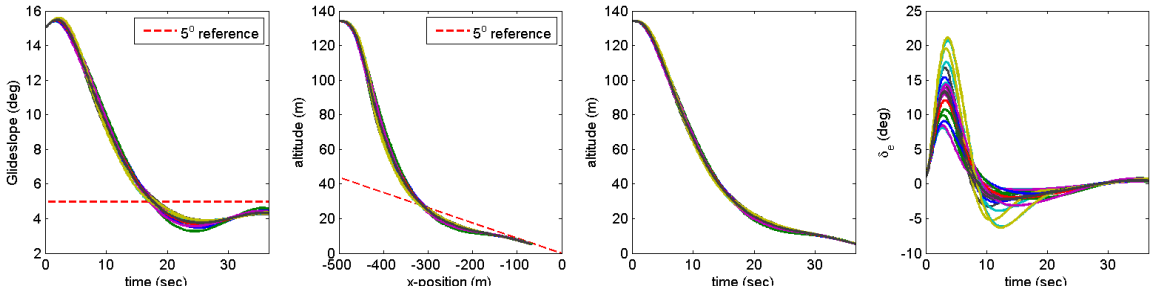


Figure 4.10: Results for the set of uncertain Easy Star plants with initial glideslope of 10°.

Fig. 4.9 shows the simulation results for the nominal model at various initial conditions. The plots show acceptable performance and the requirements are met. The glideslope converges to within 1° of the target Γ_{ref} in all simulations by the time the flare altitude is reached. Fig 4.10 shows the simulation results for the set of uncertain plants. All of the performance requirements are met except for the elevator deflection limit of 15°. Because fewer than half the uncertain plants exceed this bound, and the initial condition of $\Gamma = 15°$ is considered an extreme case, the designed G_Γ is acceptable.

4.1.2.3 Throttle control loop

The throttle control loop is designed to regulate the vehicle body-axis forward speed about its steady-state value during descent and landing. This control loop consists of a single closed loop with the standard prefilter and controller arrangement. The requirements for the throttle control loop are as follows:

- Rise time between seven and ten seconds.
- Overshoot of less than 10%.
- Stability margin of 1.1.

Since the only requirement of the throttle control loop is regulation to a constant value, the rise time is much lower than for the other control loops. The set of uncertain plants is derived using the convex hull of the Nichols plant templates as for the other control loops. The throttle response to commands is modeled as a first-order response as derived experimentally in Sec. 3.4.1. This is intended to represent a conservative estimate of the motor response time. To satisfy the requirements, the following prefilter $F_u(z)$ and controller $G_u(z)$ are implemented:

$$G_u(z) = \frac{z^2 - 1.7143z + 0.7367}{1.3797z^2 - 1.6695z + 0.2901} \tag{4.16}$$

$$F_u(z) = \frac{z^2 - 1.8362z + 0.8468}{9.9495z^2 - 19.3958z + 9.4587} \tag{4.17}$$

Figs. 4.11-4.12 show the performance of the controller and prefilter for the linear uncertain plants. The stability margin requirement is satisfied as shown in Fig. 4.11. The rise time and overshoot can be seen in Fig. 4.12 to be about 7 seconds and 5%, respectively.

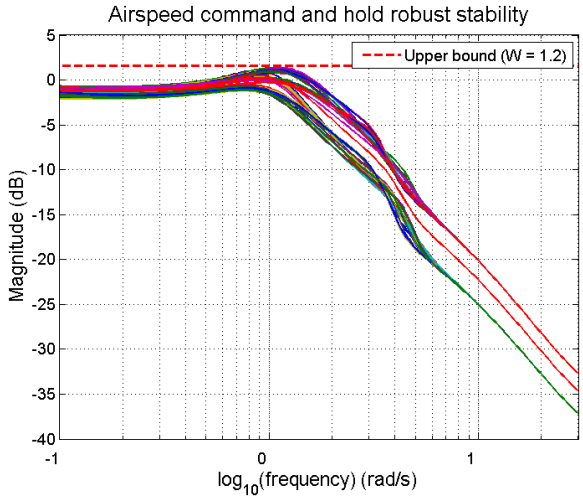


Figure 4.11: Robust stability margin for Easy Star closed-loop airspeed control loop without prefilter.

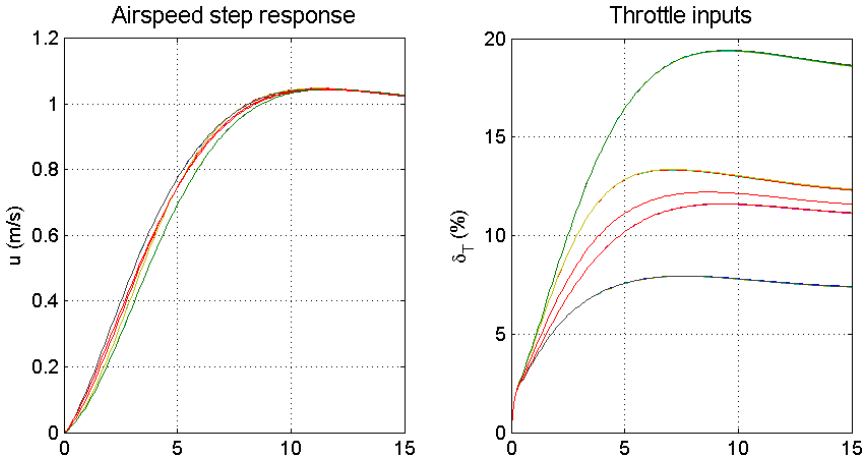


Figure 4.12: Step response for the set of Easy Star plants in the airspeed command and hold loop.

4.1.2.4 Flare tracking loop

The autoflare loop is the most critical loop in the autolanding maneuver. Once the vehicle reaches the flare altitude, h_{flare}, the control system switches from glideslope tracking to the autoflare maneuver, in which the descent rate is directly regulated by commanding reference pitch angles. The autoflare loop consists of a prefilter and controller designed using QFT in the usual fashion. The reference descent rate is defined as a function of current altitude, h, and constant τ as follows:

$$\dot{h}_{ref} = -\frac{1}{\tau}h \tag{4.18}$$

The controller requirements for the flare loop are: (1) stability margin less than 1.1; (2) Rise time between 5 and 7 seconds with a maximum overshoot of 10%. The maximum flare length (i.e. the ground distance traveled during the maneuver) should be less than 150 m. To ensure vehicle safety, the descent rate at landing should be less than 1.8 $\frac{\text{m}}{\text{s}}$. For control design, the descent rate is linearized as $\dot{h} = U_1(\theta - \alpha)$. Descent rate is modeled in simulation by the first-order backwards-difference approximation $\dot{h} \approx \frac{h_k - h_{k-1}}{T}$, which is the assumed microcontroller computation for descent rate. After iteration over different settings, the flare settings $h_{flare} = 4$ m and $\tau = 2.5$ s are selected. The value of τ is selected so that the reference descent rate at the flare height will match the nominal descent rate from tracking the reference glideslope.

The flare controller $G_{flare}(z)$ and prefilter $F_{flare}(z)$ are:

$$G_{flare}(z) = \frac{2.6005z^2 - 5.1000z + 2.5000}{z^2 - z} \tag{4.19}$$

$$F_{flare}(z) = \frac{z^4 - 2.4426z^3 + 2.2758z^2 - 0.9898z + 0.1825}{4.7188z^2 - 8.9315z + 4.2385} \tag{4.20}$$

The performance of the autoflare controller is coupled with the performance of

the airspeed command and hold, as well as the conditions at the start of the flare, and it is not useful to evaluate the step response of the set of uncertain linear plants. The performance of the uncertain longitudinal-axis plants is evaluated by simulating landings with the full set of longitudinal controllers, assuming deterministic state measurements are available, with the nonlinear kinematic equations of motion. The set of plants evaluated is the union of the set of plants considered for both the airspeed and pitch angle command and hold loops. This set has 114 plants. These plants are evaluated for initial conditions $X = 500$ m, $\Gamma = 10°$.

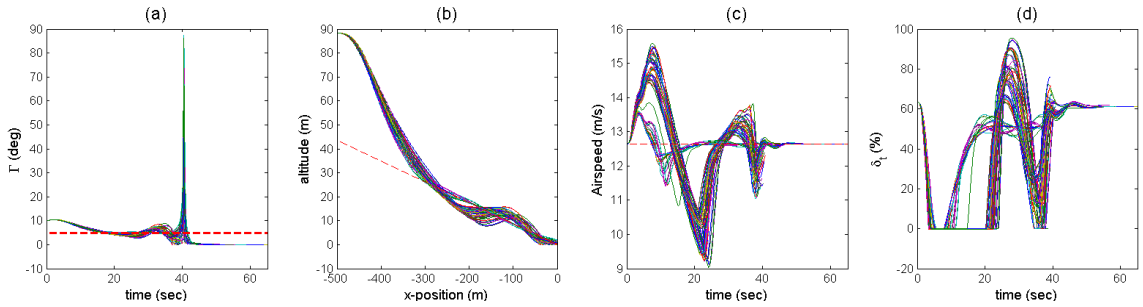


Figure 4.13: (a) Glideslope angle of the set of uncertain Easy Star plants. The singularities in the glideslope history near time $t = 40$ occur when the vehicle passes near $x = 0$, but happen after the autoflare maneuver begins and do not affect the controller's performance. (b) Inertial trajectories of the set of uncertain Easy Star plants. (c) Perturbed body 1-axis speed during the Easy Star simulations. (d) Throttle use during Easy Star glideslope tracking. Limits of 0% and 100% throttle are enforced in simulation.

Fig. 4.13 shows the glideslope tracking performance of the set of uncertain plants. 4.13(a) plots the time history of the glideslope of each plant relative to the origin. The singularities near $t = 40$ occur when the plants pass through $X = 0$. However, these singularities occur after the autoflare maneuver is triggered and do not affect performance. Fig. 4.13(b) shows the inertial trajectories in the X-Z plane. The

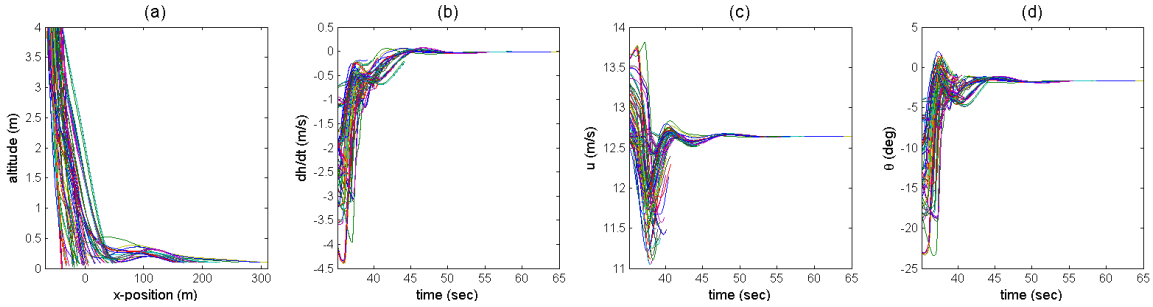


Figure 4.14: (a) Inertial trajectories of uncertain Easy Star plants during the flare maneuvers. (b) Descent rates of the uncertain Easy Star plants during flare maneuvers. (c) Perturbed Easy Star body 1-axis speed during flare maneuvers. (d) Easy Star pitch angles during flare maneuvers.

deviation is larger than ideal for some plants, which is driven by the large initial glideslope tracking error and relatively short ground path to the origin. These initial conditions represent a worst-case scenario, and the performance is considered acceptable. Figs. 4.13(c)-(d) show the perturbed body 1-axis speed and throttle during the flight. Again, although tracking exhibits significant errors for some plants, performance is tolerable given that this is a worst-case scenario. Furthermore, airspeed typically converges to within 2 m/s of the steady-state value by $t = 40$, the typical flare start time. Additionally, these plots indicate that this performance can be achieved with the expected throttle control limits.

Fig. 4.14 shows the flare tracking performance. Fig. 4.14(a) shows the inertial trajectories during the automatic flares. Several plants violate the desired flare length. This is the result of a deliberate design choice to favor a long flare rather than risk an increase in hard landings when uncertain sensing and disturbances are included. As reflected in Fig. 4.14(b), decent rates are relatively low during the flare. Figs. 4.14(c)-(d) are included to demonstrate that airspeed and pitch angle are correctly regulated during the flare maneuver.

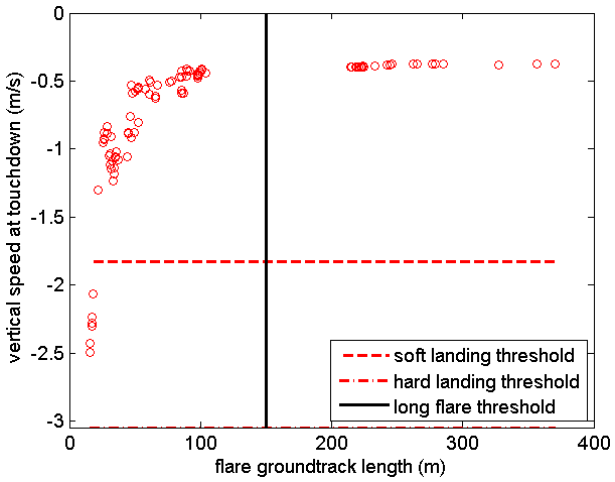


Figure 4.15: Analysis of flare performance of the set of uncertain plants.

Fig. 4.15 plots the descent rate at touchdown versus the flare groundtrack distance with the target limits for each. 108 of the plants are within the assumed "soft landing" descent rate limit. 33 plants violate the flare length requirement. Given the success in meeting the descent rate limit, the violation of the flare length is considered acceptable for preliminary implementation, given the uncertainty present in the plants. The complete controller design for the Easy Star is validated in Monte Carlo simulation with sensor noise; results of this evaluation are presented in the next section.

4.1.2.5 Longitudinal control redesign

Initial flight tests of the Easy Star automatic flare loop indicated that the controller as designed was significantly underdamped. Representative results are shown in Figs. 4.16-4.17. To some extent, this was a result of the design choice to reduce the chance of a hard landing in favor of longer flares. The actual flight performance exhibited much less damping than was present in simulation, and the autoflare controller and prefilter were redesigned to add damping and improve the reference tracking in

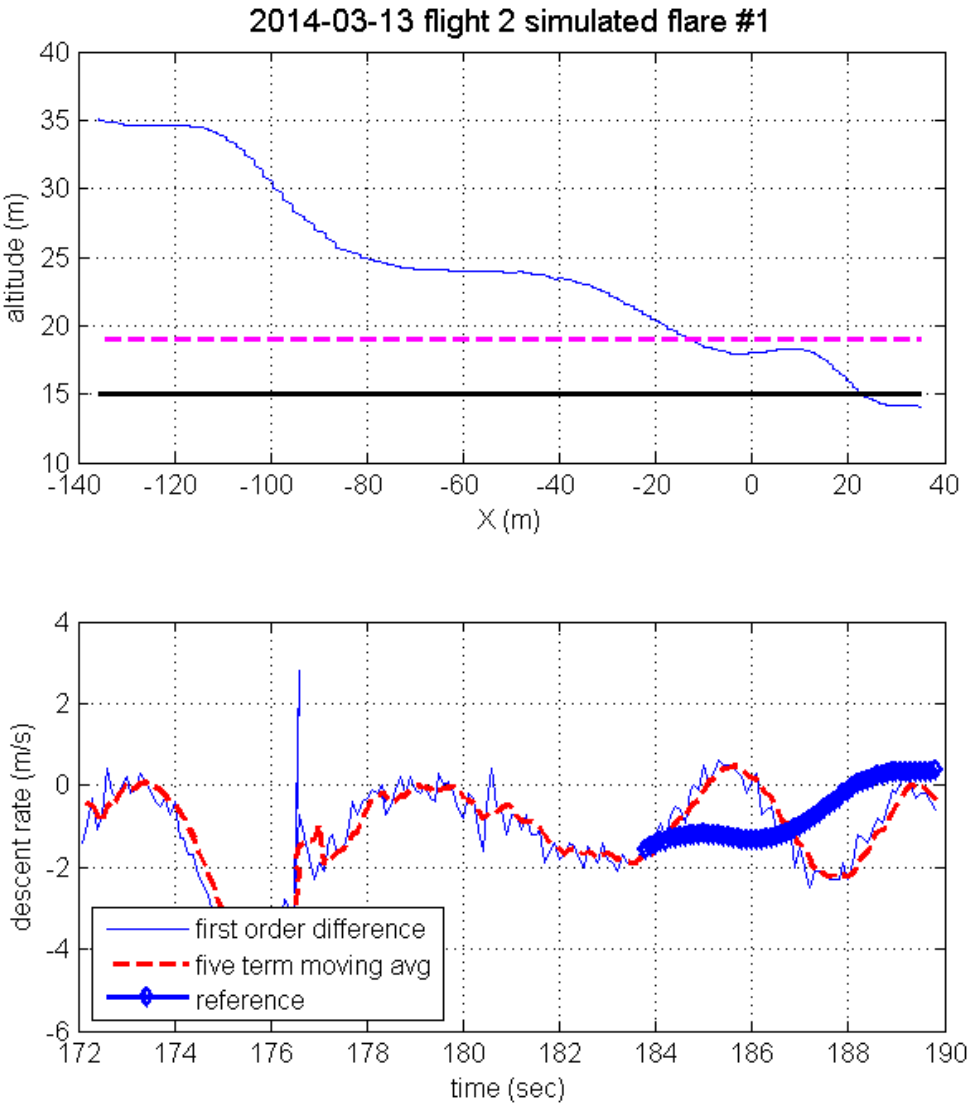


Figure 4.16: Easy Star glideslope tracking and initial flare performance in flight testing. The testing is performed with a 15 m vertical offset from the ground, so the flare altitude is 19 m. The descent rate response tracks the reference but shows very little damping. The dashed line indicates the effective flare altitude and the solid line indicates the effective target landing altitude.

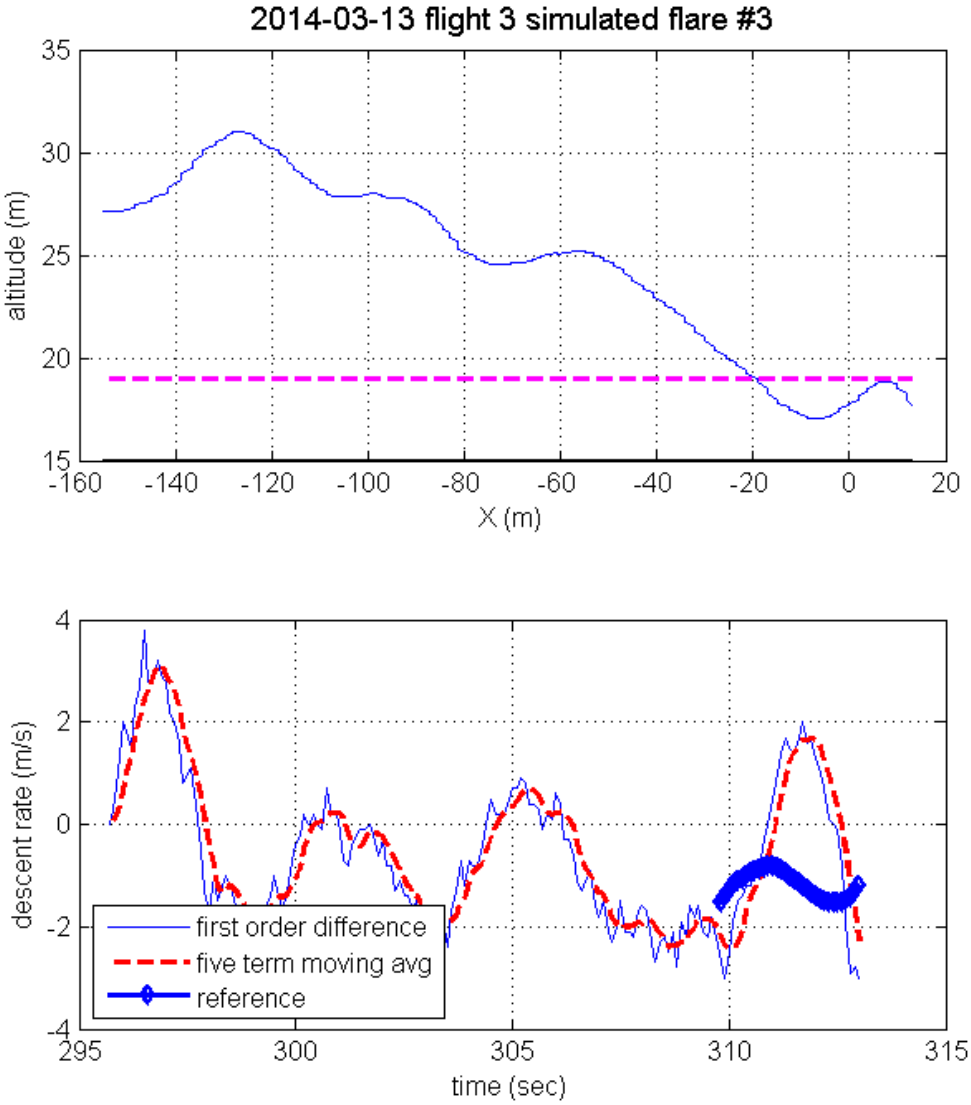


Figure 4.17: Easy Star glideslope tracking and initial flare performance in flight testing. The testing is performed with a 15 m vertical offset from the ground, so the flare altitude is 19 m. The descent rate response shows a large initial overshoot. The dashed line indicates the effective flare altitude and the solid line indicates the effective target landing altitude.

the flare mode.

The redesigned controller and prefilter are:

$$G_{flare}(z) = \frac{2.525z^2 - 5.025z + 2.500}{z^2 - z} \tag{4.21}$$

$$F_{flare}(z) = \frac{.00726}{z^2 - 1.876z + 0.8831} \tag{4.22}$$

Performance of the redesigned controller in flight test is presented in Section 6.

4.2 Pegasus controller design

Pegasus control laws have the same structure and similar requirements to the Easy Star controllers. The Pegasus design is constrained by the requirements of the backup human pilot. Since the Pegasus vehicle has previously been flown manually, the initial range to the runway in past recorded landings has been as short as 500 meters. The vehicle's steady-state airspeed is approximately 30 m/s, so this is a very short approach corridor, and is the primary driver behind the design of the outermost longitudinal and lateral/directional control loops.

4.2.1 Pegasus lateral/directional control design

The lateral/directional control consists of the bank angle, heading angle, and azimuth angle loops. QFT is used for synthesis of each loop. The heading angle and azimuth angle are most challenging to design.

4.2.1.1 Bank angle command and hold

The bank angle command and hold loop consists of the prefilter, F_ϕ with feedback controller G_ϕ. With P_ϕ designating the transfer function $\frac{\phi}{\delta_a}$, the open-loop response is $L_\phi = G_\phi P_\phi$. The primary design requirements of the bank angle command and

hold loop are a 90% rise time of three to five seconds and an overshoot of no more than 20%. Because the bank angle control loop is a part of the heading command and hold loop, the performance of the inner controller strongly influences the design of the robust heading controller. To address the coupling between these loops, an iterative workflow is used. Inner loop controllers are designed, then the outer loop controller is synthesized to satisfy the QFT bounds as best as possible. The performance of both control loops in the time domain is then evaluated, and the inner loop is revisited until performance is acceptable. The bank angle design requirements in terms of stability margin and tracking bounds are given below:

$$\begin{aligned} T_{rl}(s) &= \frac{1.25}{s^3 + 6s^2 + 5.25s + 1.25} \\ T_{ul}(s) &= \frac{0.5s + 1.25}{s^2 + s + 1.25} \\ SM &= 1.4 \end{aligned} \tag{4.23}$$

Plant uncertainty assumptions are outlined in Section 3.5. Using the convex hull of the plant templates, as with the Easy Star, 144 plants, including the nominal model, are used for robust QFT design. To prevent control saturation, the maximum commanded aileron deflection is limited to $\pm 20\%$ in this loop. To satisfy the performance requirements, the controller and prefilter of Eqs. 4.24-4.25 are implemented. The step responses of the closed-loop linear systems with each of the uncertain plants are shown in Fig. 4.18. The responses are not all within the bounds of the step response of the tracking limits; this is not considered to be problematic because the original time domain requirement of a 90% rise time of five seconds is

achieved with an overshoot of 3%.

$$G_\phi(z) = \frac{z^3 - 1.735z^2 + 0.7415z}{25.06z^3 - 60.17z^2 + 47.35z - 12.24} \tag{4.24}$$

$$F_\phi(z) = \frac{z^2 - 1.730z + .7482}{5.667z^2 - 10.78z + 5.135} \tag{4.25}$$

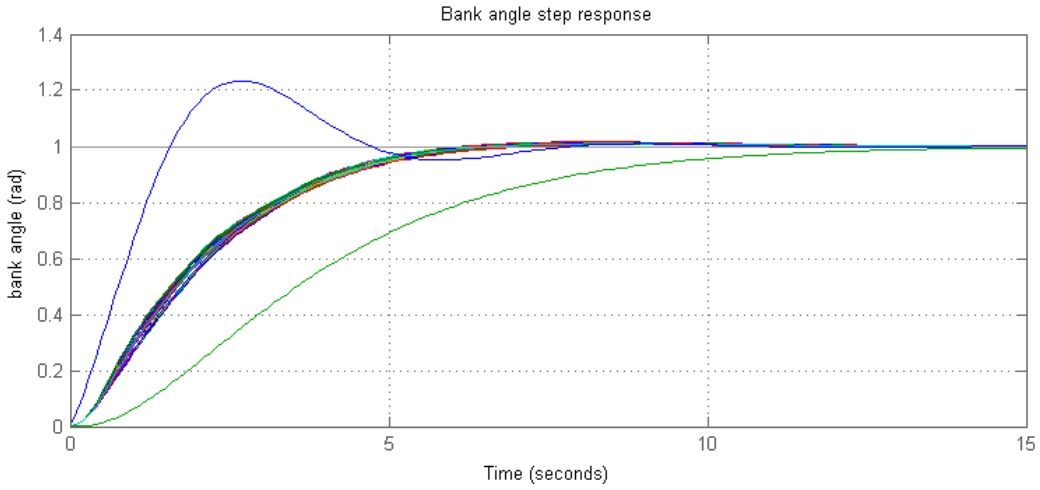


Figure 4.18: Bank angle step response of closed loop uncertain lateral/directional Pegasus system. The upper and lower tracking bounds are plotted for reference.

4.2.1.2 Heading angle command and hold loop

The control structure for the heading command and hold loop is the same as that used for the bank angle command and hold loop. The plant is the product of the transfer function $\frac{\psi(z)}{\phi(z)}$ with the set of uncertain bank angle command and hold closed loops T_ϕ. The assumed linear model for the heading angle is $\dot{\psi} = r$; since this is a kinematic relationship, there is no assumed additional uncertainty and all plant uncertainty for the heading angle command and hold loop originates in the bank angle loop.

For the heading angle command and hold loop, the step response requirements are a 90% rise time of three to seven seconds with an overshoot of no more than 30%. As with the bank angle command and hold loop, an open-loop stability margin of 1.4 is required, and the upper and lower tracking bounds are the same as those of Eq. 4.23. The following prefilter and controller are implemented:

$$G_\psi = \frac{z - 0.9657}{0.1355z - 0.1107} \tag{4.26}$$

$$F_\psi = \frac{0.2600z^2 - 0.4746z + 0.2163}{z^2 - 1.940z + 0.9414} \tag{4.27}$$

The step response of the uncertain plants meets the rise time requirement with no overshoot (see Fig. 4.19). In the full simulation, the maximum commanded bank angle is limited to $\pm 15^\circ$ as a safety measure against controller destabilization.

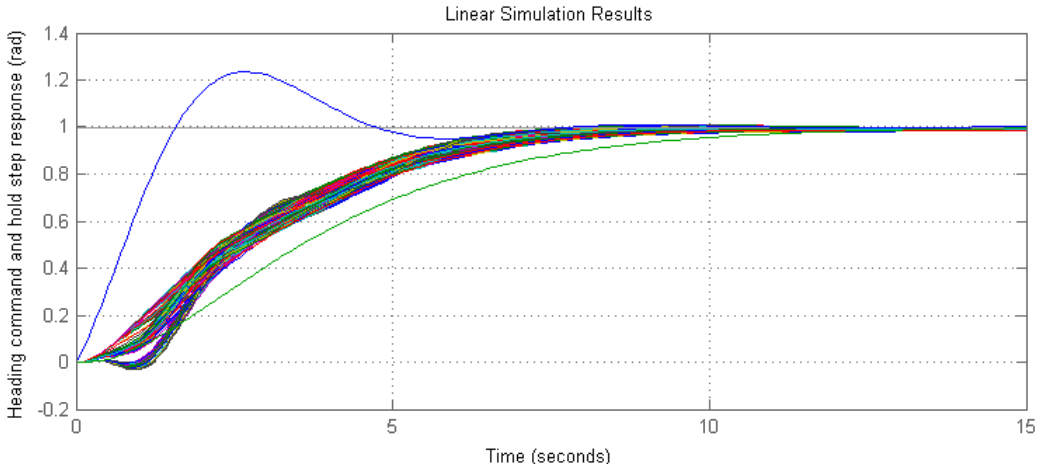


Figure 4.19: Step response of uncertain Pegasus plants to a step change in heading (linear kinematic equation is used for propagation).

4.2.1.3 Azimuth regulation loop

The azimuth tracking loop consists of three cascaded SISO control loops: bank angle command and hold, heading angle command and hold, and azimuth angle regulator. Because the Pegasus system is currently restricted to visual flying rules (VFR), the aircraft excursions are not typically far from the takeoff point. Consequently, the azimuth tracking loop must be capable of centering the aircraft on the runway from a relatively short initial downrange distance. Moreover, the vehicle's cruising speed of about 30 m/s is comparable to a manned light sport aircraft[23], and limits the time between engaging the autolanding controller and reaching the target landing coordinates. Overall, the task of the azimuth regulation loop is very different for Pegasus than for an automatic landing controller on a manned aircraft, for which the approach distance might be several nautical miles[49]. The azimuth regulation loop is primarily designed by considering the response to the most extreme set of initial conditions, which were assumed to be $X = -250$ m and $Y = \pm 75$ m, corresponding to an initial azimuth angle of 16.7°. In addition to the model uncertainty identified previously, the control loops should be robust to at least light turbulence (defined as an intensity of 5 mph), and should be capable of landing in the presence of sensor uncertainty.

The azimuth regulation loop is only required to regulate azimuth angle to zero, so no prefilter is implemented in this loop. The requirements for the azimuth regulation loop are as follows: in the deterministic (zero turbulence) case, the loop must regulate the crossrange position to $\|Y\| \leq 5$ m by the time the vehicle reaches $X = 0$ m, and should maintain this limit subsequently. $X = 0$ m is, conservatively, taken as the earliest point at which touchdown will occur, since the automatic flare should extend touchdown beyond this point. Additional requirements must be satisfied in

the presence of a turbulence intensity of 5 m/s: the vehicle should reach $\|Y\| \leq 11$ m by the time $\|X\| \leq 50$ m, and should subsequently maintain Y position within this 11 m limit for 90% of cases. 11 m is selected because it is approximately three times the wingspan of the vehicle.

The linear approximate transfer function for $\frac{\lambda}{\psi}(s) \approx \frac{U_1}{Rs}$ (in which $R = \sqrt{(X^2 + Y^2)}$ is the range) is used for control synthesis[49]. Relatively low control gains are required to satisfy robust stability bounds. To avoid singularities near the origin, the controller is scheduled with range up to 500 m; i.e., for ranges greater than 500 m, a fixed multiplier is used to avoid control saturation if the initial azimuth error is large.

After iteration multiple times through all three inner-to-outer lateral/directional loops, the controller of Eq. (4.28) is selected as offering acceptable performance. The closed-loop response of the nominal plant with no disturbances is shown in Fig. 4.20(a).

$$G_\lambda = \frac{z - 0.9618}{5.726z - 1.791} \tag{4.28}$$

Monte Carlo simulations are used to evaluate the performance of the azimuth tracking loop in turbulent conditions. A grid of initial conditions is defined by X locations ranging from -250 m to -500 m in intervals of 50 m and λ ranging from 0° to 16.7° in intervals of 5.567°. $\psi(0) \in \{0^\circ, -5^\circ, -10^\circ -, 15^\circ\}$ are considered as initial headings, and all other perturbed lateral/directional states are initially zero. At each grid point of initial conditions, one simulation is performed with each of the 144 uncertain models with turbulence intensities of 2.3 and 4.45 m/s. For both turbulence intensities, it is found that the vehicle is regulated to a lateral position $\|Y\| \leq 11$ m in all cases before reaching $X = 0$ m, and remains with the 11 m bounds until reaching $X = 500$ m, which is the break condition for the simulation.

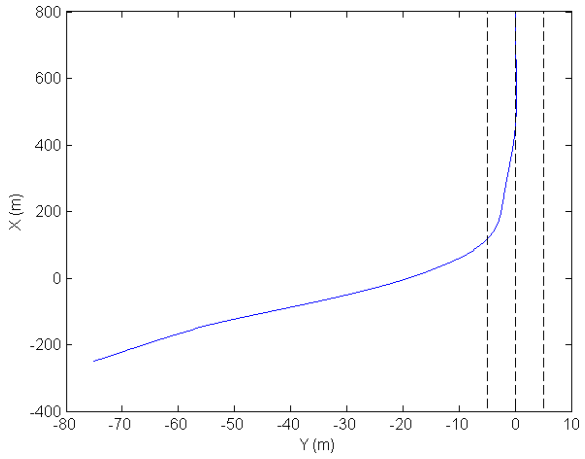


Figure 4.20: Nominal Pegasus plant azimuth response in X-Y space with no turbulence. $Y = \pm 5$ m bounds shown.

This performance validates the lateral-directional control design.

4.2.2 Pegasus longitudinal control design

The longitudinal control system consists of the cascaded inner-outer tracking loop for pitch control and the airspeed command and hold loop. The pitch control loop uses the glideslope tracking loop for most of the descent and the automatic flare loop to track reference descent rate commands near the runway. Both of these loops provide reference pitch angles for the same pitch angle command and hold inner loop. The airspeed command and hold loop is used to regulate forward speed during the flight.

4.2.2.1 Airspeed command and hold loop

The primary time-domain requirements of the airspeed command and hold loop are a rise time of between 10 and 15 seconds with no overshoot. The relatively slow throttle dynamics limit performance, but the requirements of the throttle control system are primarily regulation or tracking of piecewise constant references while

rejecting disturbances, so the requirements are relatively simple to satisfy. The frequency domain stability requirement is a robust stability margin of 1.2. The rise-time requirements are translated into robust tracking bounds by Eq. 4.29:

$$\frac{0.0246}{s^3 + 0.7487s^2 + 0.2186s + 0.02475} \leq \| \frac{F_u L_u}{1 + L_u} \| \leq \frac{0.1659s + .04976}{s^2 + 0.4s + .05} \tag{4.29}$$

Since the two-stroke gasoline engine on the vehicle is known to have a slow response time compared to the servos that actuate the control surfaces, the throttle dynamics are explicitly accounted for during control synthesis. The throttle system consists of a mechanical linkage that is actuated by a servo; therefore, both the servo and engine dynamics are considered. The servo is assumed to have first-order dynamics with a 50% time constant of 0.1 seconds; the engine is assumed to have first-order dynamics with a 1.0 second time constant. These dynamics, which essentially represent a delay in the control application, are included in the plant model P_u.

To satisfy the requirements, the controller and prefilter of Eqs. 4.30 and 4.31 are implemented. The step response is shown in Fig. 4.21, and satisfies the 90% rise time specifications.

$$G_u = \frac{z^3 - 2.551z^2 + 2.1556z - .60277}{.034441z^3 - .045706z^2 + .012181z - .00091285} \tag{4.30}$$

$$F_u = \frac{.00056098z^2}{z^2 - 1.9594z + .95997} \tag{4.31}$$

Disturbance rejection is analyzed by considering Monte Carlo simulations with each of the uncertain longitudinal-axis plants for sixty seconds with a turbulence intensity of 2.3 m/s. The 220 uncertain models from the union of the airspeed and pitch axis plant sets is used. The lateral/directional guidance loop is active in

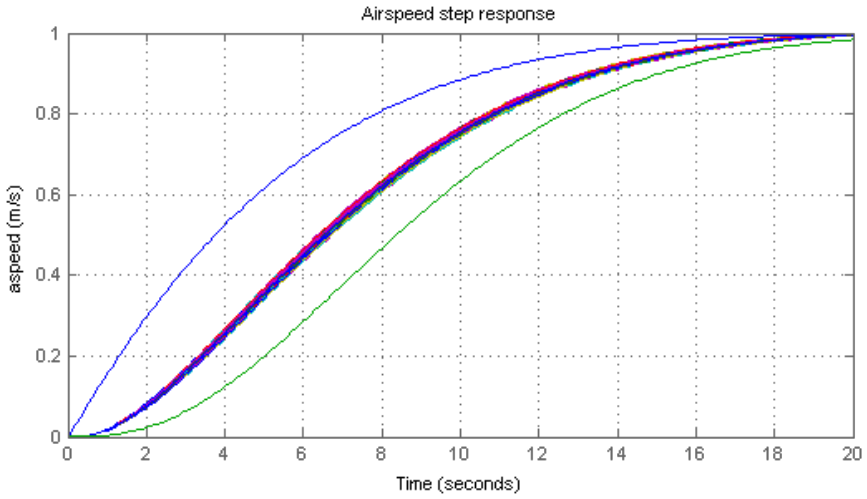


Figure 4.21: Step response of airspeed command and hold closed loop.

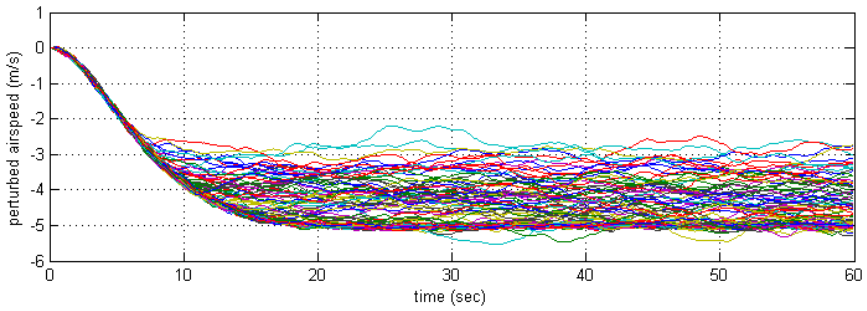


Figure 4.22: Pegasus airspeed time history in Monte Carlo simulations.

these simulations to reject disturbances, but the initial azimuth angle is zero. A constant perturbed airspeed of -5 m/s is the reference. The resulting time histories are shown in Fig. 4.22. The slow response time of the control loop limits its ability to compensate for disturbances; however, steady-state errors of typically $\leq$ 2 m/s are achieved. In 168 simulations, the final airspeed is within 0.25 m/s of the target, a 76.3% success rate by this metric.

4.2.2.2 Pitch angle command and hold loop

The pitch angle command and hold loop is the inner loop for both the glideslope tracking and automatic flare loops. It is one of the most critical loops in the automatic landing sequence, as it must be capable of accurate tracking of reference commands during the short flare maneuver, while rejecting disturbances induced by turbulence and sensor errors. The time domain requirements are a rise time of between one and three seconds. The stability margin is 1.3. The tracking requirement is enforced in the frequency domain by the same upper and lower bounds used in the bank angle controller design: $T_{rl}(s) = \frac{1}{0.288s^3+1.92s^2+2.6s+1}$ and $T_{ul}(s) = \frac{0.4s+1}{0.124s^2+0.6s+1}$. Additionally, after several iterations of the pitch-axis controllers, including the outer loops, a disturbance rejection criterion was added to the pitch angle control loop to minimize the influence of turbulence on overall performance. The requirement is equivalent to Eq. 2.1. For the pitch axis controller, a bounded input disturbance response at all frequencies considered is enforced, expressed by Eq. 4.32:

$$\left| \frac{1}{1 + P_\phi G_\phi} \right| \leq 1.0 \tag{4.32}$$

Since the servos used for control surface actuation are the same as that used to drive the throttle, first-order servo dynamics with a time constant of 0.1 seconds are included in the elevator-to-pitch angle transfer functions in control synthesis.

The controller and prefilter of Eqs. 4.33 and 4.34 are implemented. The step response of 116 uncertain linear longitudinal plants, including servo dynamics is shown in Fig. 4.23. The 90% rise time requirement is met. In addition, the elevator

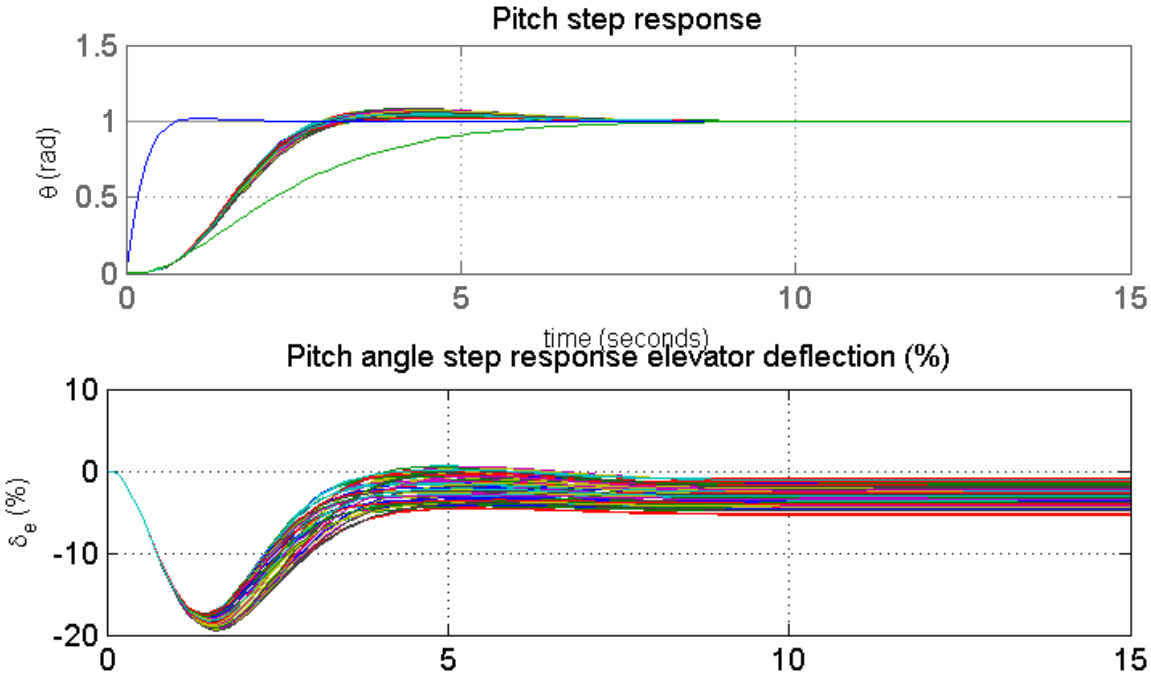


Figure 4.23: Step response of uncertain Pegasus linear plants to a commanded change in pitch angle.

deflection remains bounded within 20% full scale range.

$$G_\theta = \frac{z^5 - 1.968z^4 + 1.258z^3 - 0.2970z^2 + 0.019727z - 0.0004401}{-1.844z^3 + 3.188z^2 - 1.344z} \tag{4.33}$$

$$F_\theta = \frac{z^2 - 0.099574z + 0.0024788}{51.42z^2 - 87.93z + 37.42} \tag{4.34}$$

4.2.2.3 Glideslope tracking loop

The glideslope tracking loop is the outermost loop for pitch axis control. Based on flight data with a remote human pilot, a typical landing approach, from beginning of descent until touchdown, has a ground run of approximately 600 m and an approach glideslope of 5° to 10°. The glideslope tracking loop should provide sufficiently rapid regulation of tracking errors to a reference to land the vehicle near the

specified GPS coordinates. The loop should also be robust to the same level of turbulence considered in the azimuth tracking loop. For safety, the glideslope tracking performance should be evaluated over a range of initial conditions to quantify which initial states are most critical for successful automatic landing. A minimum initial range of $X = 1000$ m to the target waypoint is assumed; when the vehicle is closer initially, performance becomes extremely sensitive to the initial state.

To improve the landing performance of the vehicle, the reference glideslope decays with time, as opposed to the constant reference used with the Easy Star. Assuming a target waypoint at the inertial coordinate frame's origin, the reference glideslope in terms of the initial glideslope Γ_0, initial X-position X_0, and reference final glideslope $\Gamma_f = 1^\circ$ is:

$$\Gamma(X) = \Gamma_0 \left(\frac{\Gamma_f}{\Gamma_0} \right)^{(X_0 - X)/X_0} \tag{4.35}$$

The glideslope tracking loop for the Pegasus consists of a the following controller G_Γ and prefilter F_Γ:

$$G_\Gamma(z) = \frac{0.5383z^2 - 1.038z + 0.500}{z(z-1)} \tag{4.36}$$

$$F_\Gamma(z) = \frac{3.213 \times 10^{-6} z^2}{z^3 - 2.965z^2 + 2.930z - 0.9651} \tag{4.37}$$

To avoid the glideslope singularity at $X = 0$, the controller is scheduled with range, up to a maximum of 500 m (i.e., beyond 500 m a fixed gain of 500 is used to avoid saturating the control if the initial glideslope error is large). The controller G_Γ was designed primarily by iterative evaluation of the performance of the uncertain plant models in nonlinear simulation with turbulence, as was the controller for the azimuth regulator. To evaluate the effectiveness of the controller, Monte Carlo simulations are conducted for the set of uncertain longitudinal models, using a grid

$\Gamma(0)$	4	3	2
no turbulence	0	1	0
turbulence	0.03947	0.5975	0.0848

$\alpha(0)$	-2.5	0	2.5
no turbulence	0.3333	0.3333	0.3333
turbulence	0.2406	0.2406	0.2406

$\theta(0)$	-5	-2.5	0
no turbulence	0.3333	0.3333	0.3333
turbulence	0.2406	0.2406	0.2406

Table 4.1: Proportion of successful Pegasus glideslope approaches with and without turbulence as main effects of initial conditions. All angles are given in degrees.

of the following set of initial conditions:

- $\alpha(0) \in [-2.5^\circ, 0^\circ, 2.5^\circ]$
- $\theta(0) \in [-5^\circ, -2.5^\circ, 0^\circ]$
- $\Gamma(0) \in [2^\circ, 3^\circ, 4^\circ]$

Glideslope performance is evaluated in terms of the X-position error magnitude at the flare altitude; if the error between the actual position and the reference trajectory is less than 60 m, this is considered a success. Simulations are conducted with both zero and light turbulence, to evaluate the performance degradation in the presence of turbulence, from an initial X coordinate of $X = -1000$ m. Performance is presented as the fraction of successful trials for each set of initial conditions.

The overall success rates are 24.06% with turbulence and 33.3% without turbulence. Table 4.1 shows the success rates as main effects of the initial conditions, both with and without turbulence. For safety, the control authority is limited by restricting commanded pitch angles to $\pm 15^\circ$, which is the primary reason that success cannot be achieved in all of the deterministic cases. This restriction also somewhat

limits the disturbance rejection of the controller. The results indicate that the accuracy of the glideslope tracking depends strongly on the initial glideslope angle, with little dependence on initial angle-of-attack and pitch angle. The success rate with turbulence varies, but is typically between 60% and 70% of the success rate without turbulence. Based on the performance with varying $\Gamma(0)$, automatic landings should be attempted if the initial glideslope is near 3°. The initial glideslope is easy to measure, with good accuracy from GPS, and software failsafes can be designed to prevent automatic landings being triggered if the glideslope deviation is more than 0.5°.

4.2.2.4 Automatic flare

The automatic flare loop is the outer loop for pitch control from the point when the vehicle is below the flare altitude until touchdown. Overall vehicle safety is most closely related to this loop. A reference descent rate, which is simply a linear function of altitude, is tracked during the flare maneuver.

For Pegasus, the flare altitude is increased to 10 m, since the vehicle is much larger and faster than the Easy Star. This should provide some additional margin of safety and give the remote operator more time to override if the autopilot experiences an error when switching to the flare controller. The reference descent rate in terms of altitude, h, is:

$$\dot{h} = -0.200h \tag{4.38}$$

The primary requirement is disturbance-free tracking of the reference to within 0.5 m/s for the set of uncertain plants. This threshold is selected to minimize the chance of a hard landing. Tracking should be achieved within three seconds. It is intended that in hardware implementation the commanded pitch angle should be limited to ±5°; however, in simulations with turbulence, this restriction too greatly limited

$\Gamma(0)$ (°)	4	3	2
no turbulence	0.9956	1	1
turbulence	0.0009747	0.01413	0.03119

$\alpha(0)$ (°)	-5	-2.5	0
no turbulence	0.9985	0.9985	0.9985
turbulence	0.01511	0.01706	0.01413

$\theta(0)$ (°)	-5	-2.5	0
no turbulence	0.9985	0.9985	0.9985
turbulence	0.01657	0.01608	0.01365

Table 4.2: Table of the success rate of achieving "soft" landings, defined as a vertical descent rate of magnitude 6 ft/s or less at touchdown, both with and without turbulence, for Pegasus longitudinal-axis control.

control authority, and the controller is unable to properly compensate for 5 mph turbulence. Consequently, no restriction is placed in simulation. It is recommended that the 5° limit on commanded pitch angle be enforced if the measured pitch angle is ±5° or more.

The following controller is implemented for the automatic flare with unity prefilter:

$$G_{flare}(z) = \frac{0.10825z^2 - 0.1675z + 0.0625}{z^2 - z} \tag{4.39}$$

Flare performance is evaluated in terms of the number of hard and soft landings with and without turbulence. As a preliminary analysis of performance, simulations are conducted with the same grid of initial conditions used to evaluate the glideslope tracking, from an initial range of $X = 1000$ m. One simulation is run with each of the 228 uncertain longitudinal models. Both turbulence with an intensity of 5 mph and calm conditions are considered, as before. In evaluating the performance, the primary metrics of interest are the percentage of simulations that terminate in "soft" and "hard" landings. These thresholds are defined by descent rates of 6 ft/s and 10 ft/s, respectively.

Success rates of achieving soft landings are presented in Tables 4.2 and Table 4.3. Although the performance in deterministic simulations is good, the disturbance rejection is evidently very poor. At the initial range of 1000 m, the performance appears to be essentially invariant with respect to initial pitch angle or angle-of-attack. The success rate does improve slightly as the initial glideslope angle decreases. Table 4.3 compares the overall performance with and without turbulence. In both sets of simulations, landing or crashing occurs in every case. The success rate of soft landings in the deterministic case is essentially 100%, and is about 5% in the turbulent simulations. Overall approximately 80% of turbulent simulations are expected to crash in a way that significantly damages the vehicle, although the exact result would depend strongly on the attitude at touchdown.

It is clear from the performance of the flare controller that the performance with turbulence is unacceptable for implementation in flight test. The controller presented in Eq. 4.39 is the result of numerous iterations in both deterministic and stochastic simulations with turbulence. It is felt that the performance cannot be substantially improved without a significant change in the controller parameters, either by using a more sophisticated (i.e. nonlinear) control law or greatly increasing the update rates of the inner and outer pitch loops. It is also felt that this extreme sensitivity to turbulence is not a realistic representation of the Pegasus system performance at low speeds and altitudes. This indicates a limitation of the identified longitudinal-axis model.

This section has summarized the complete control synthesis and verification for both the Easy Star and Pegasus aircraft. In the next section, the performance of the controllers for each vehicle is considering using Monte Carlo simulations with sensor noise, uncertain plants, and disturbances.

	non-landing rate	hard landing rate	soft landing rate
no turbulence	0.0	0.001462	0.9985
turbulence	0.0	.1282	.04971

Table 4.3: Comparison of overall Pegasus landing performance with and without turbulence. Non-landings refer to cases where the vehicle has not reached the ground within the 50 second simulation time. Hard landings occur when the descent rate is greater than 10 ft/s at touchdown. All other simulations correspond to crashes with significant damage to the vehicle.

5. UNCERTAIN MONTE CARLO SIMULATION RESULTS

The previous section describes the initial synthesis and validation of control loops for both the Easy Star and Pegasus flight vehicles. In some cases, such as the outermost guidance loops, results with some uncertainty are used for initial evaluation. In the current section, simulations of both vehicles with the designed controllers are conducted. Simulations include sensor noise models as well as environmental effects. In each simulation the nonlinear kinematic equations of motion are used, while linear models are used to propagate the dynamic states. The sets of uncertain longitudinal and lateral/directional models are too large to exhaustively evaluate all combinations of models. In an effort to consider the full range of the assumed uncertainty, a random pairing of one uncertain longitudinal and one uncertain lateral/directional model is used in each simulation to propagate the velocity-level states. In evaluating the performance, a vertical speed at landing of 6 ft/s is considered a "soft" or nominal landing, a speed of 10 ft/s in considered a "hard" landing (minor damage to vehicle, repairable with no significant effect on performance), and a greater speed at landing is considered a crash (effective loss of vehicle or damage requiring major repair) [49]. Easy Star performance is considered first, then Pegasus performance. For the Easy Star, results are presented in the presence of static winds of varying strength and direction. For Pegasus, simulations are conducted in the presence of turbulent air, since disturbance rejection proved difficult in the control design of Section 4.

5.1 Easy Star results

The controllers designed in Sec. 4.1 are evaluated in simulation using the sensor noise characteristics assumed in Sec. 3.6. To improve performance of the flare maneuver, the target body 1-axis speed during the flare is set at -2 m/s relative

to the steady-state value. In addition, the following simplifying assumptions are made in implementing the control laws, in preparation for implementation on a microcontroller:

- Ultrasonic rangefinder measurements are approximately equivalent to vehicle altitude. This assumption is valid as long as bank and pitch angles are sufficiently small during the flare maneuver. This expedites computation and avoids the use of Eq. 3.18.

- Azimuth and glideslope angles can be approximately determined by $\lambda = \frac{-Y}{\|X\|}$, $\Gamma = \frac{-Z}{\|X\|}$; this avoids computation of transcendental functions, and is valid as long as initial angles are less than approximately 10°.

- Y-axis position errors approach zero before the vehicle reaches the region near the localizer, such that vehicle range to the localizer is well-approximated by X. This simplifies the computation of range. For localizer ranges of less than 10 m, the reference heading angle is set to zero to avoid the singularity at $X = 0$.

The primary performance metrics of interest are as follows:

- Descent rate at touchdown

- Flare length

- Pitch angle at touchdown

- Minimum body 1-axis speed (to prevent aerodynamic stall)

- Y-position at touchdown

- Maximum control surface deflections (Throttle is limited between 0-100% and its extrema are not considered)

To evaluate control law performance, four sets of 1000 Monte Carlo simulations are executed at different levels of static winds with uncertain plants and sensor noise. The initial vehicle states for all simulations are:

- $\psi(0) = -20^\circ$
- $X(0) = -500$ m
- $Y(0) = 100$ m
- $\Gamma(0) = 5^\circ$

All other states are initially their steady-state values. When cross-winds are present, the vehicle is assumed to approach the runway flying into the wind; at the initial X-coordinate specified, the control law does not compensate for the crosswind fast enough when approaching from the opposite direction. It should be noted that cross-wind landings are not generally permitted, it is desirable to know how the Easy Star performance may be affected if conditions change within the course of a flight. The following four sets of simulations are conducted:

1. No external wind
2. 1.35 $\frac{\text{m}}{\text{s}}$ crosswind
3. 2.25 $\frac{\text{m}}{\text{s}}$ wind along the runway, to determine if glideslope singularities are encountered.
4. 2.81 $\frac{\text{m}}{\text{s}}$ wind at 36.5° to the runway. This represents 125% of the maximum steady wind and cross-winds allowable for an Easy Star test flight and is considered the worst-case scenario.

Monte Carlo results are summarized in Table 5.1. The following section will display additional results and analysis for each set of simulations.

Case	Soft	Hard	Non-landings	Landings off runway	Mean flare length (m)
no wind	854	73	0	36	38.3
1.35 m/s crosswind	839	87	0	144	35.9
2.25 m/s tailwind	617	160	0	430	41.6
worst-case	708	150	0	239	31.1

Table 5.1: Summary of Monte Carlo results for the Easy Star flight vehicle.

5.1.1 No external wind

With no external wind, landing performance is limited primarily by state uncertainty at the beginning of the flare maneuver. Results are summarized in Fig. 5.1. Despite significant sensing errors, 85% of landings are within the target tolerance for vertical speed. The remainder of landings are split evenly between hard landings and crashes. Less than 5% of landings are outside the target 10 m wide strip. The mean flare length is well below the 150 m target defined previously; as expected, the very long flares seen in Fig. 4.15 only occur in the absence of sensing errors. Maximum aileron deflections are well within acceptable bounds; a very small number of elevator extrema exceed 20°, which is the target maximum. Most of the pitch angles at touchdown are within acceptable bounds of approximately ± 5°. The small number that exceed these bounds correspond to hard landings or crashes. Performance is acceptable, given the plant uncertainty assumed.

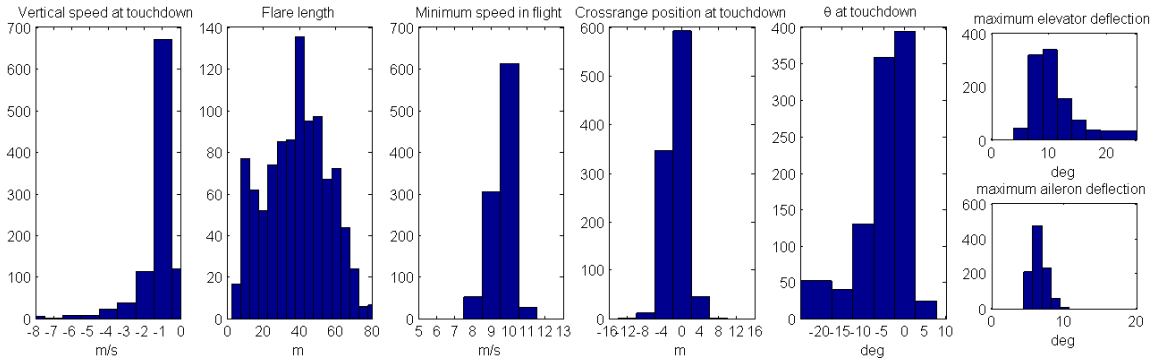


Figure 5.1: Histogram of Easy Star Monte Carlo results with no external wind.

5.1.2 Maximum rated crosswind

This case evaluates the performance with the nominal maximum crosswind. The longitudinal-axis performance in terms of the number of soft and hard landings is essentially unchanged from the zero-wind case. Approximately 15% of landings are outside the target runway. This represents a performance drop from the no-wind simulation. The distribution of the landing position in Fig. 5.2 is overall similar to that in Fig. 5.1, with a constant offset, indicating that the crosswind does not destabilize the system, but simply adds a bias. The control extrema are similar to those in the zero-wind case.

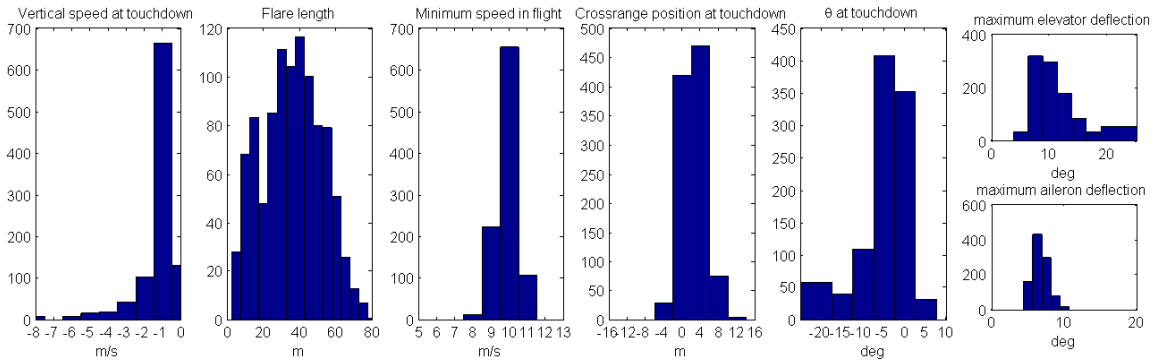


Figure 5.2: Histogram of Easy Star Monte Carlo results with 1.25 m/s crosswind.

5.1.3 Maximum rated steady tailwind

This case evaluates the performance with the nominal maximum steady wind blowing along the direction of landing; i.e., a tailwind. Performance degrades significantly from the zero-wind case with fewer than 70% soft landings. The mean flare length is slightly longer than in the other cases. The minimum flight speed trends toward higher values than before, which is reasonable given the wind conditions. The Y-position at landing is off the runway in a large number of simulations, despite the absence of a crosswind. Further, the number of control extrema above 20° increases substantially. Most likely, the larger control extrema and crossrange position errors can be attributed to the vehicle's close proximity to the landing waypoint, which is driven by the tailwind. The short range to the origin induces singularities in the glideslope and azimuth calculations, which magnify the effects of small tracking errors. Normally the flare altitude is reached well before the localizer, so the singularities are not a problem. Since the direction of a steady wind can usually be determined readily in advance of flight, the runway direction can and should be selected to be into the wind field to maximize the chances of a safe landing. However, if wind patterns reverse during flight, this simulation indicates that there is approximately a 75% chance of landing without severe damage to the vehicle. It should also be noted that the singularity is avoided entirely with Pegasus by scheduling the controllers with range to the target.

5.1.4 Worst-case analysis

The worst-case scenario assumed has a wind magnitude of 125% the rated maximum of 5 mph, with a crosswind component of 125% of the maximum rated crosswind of 3 mph. Results are shown in Fig. 5.4. Results are similar to those with a pure tailwind, indicating that the tailwind more strongly influences success or failure than

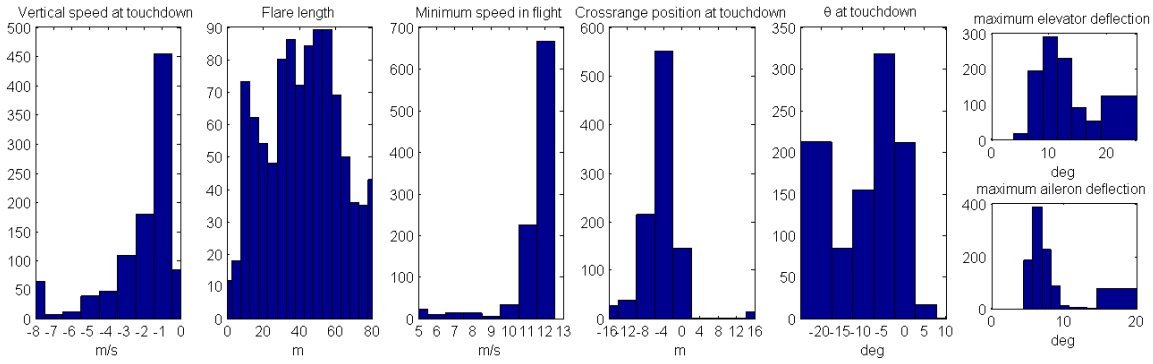


Figure 5.3: Histogram of Easy Star Monte Carlo results with 2.25 m/s tailwind.

the crosswinds (for the relative magnitudes assumed). The number of soft landings is actually much higher in the worst-case analysis than in the pure tailwind case, and the number of landings on the runway is also larger. This most likely occurs because the combination of a larger tailwind and crosswind tend to keep the vehicle farther from the localizer when $X = 0$, reducing the effect of the trigonometric singularity on the landing performance. Overall, the probability of a soft landing in highly adverse conditions with the Easy Star is much lower than would be preferred, but is driven by the hardware limitations and vehicle mass. The control law retains an approximately 80% chance of landing without loss of the vehicle when landing with a tailwind and significant crosswind. Performance is tolerable, given the relatively extreme disturbances present.

5.2 Pegasus results

Performance of the Pegasus control laws is evaluated in Monte Carlo simulations with sensor noise and aerodynamic turbulence using 500 random combinations of one uncertain longitudinal model and one uncertain lateral/directional model. The same basic simplifying assumptions outlined in Sec. 5.1 are used in the calculation of range and glideslope and azimuth angles. In addition, to achieve consistent performance

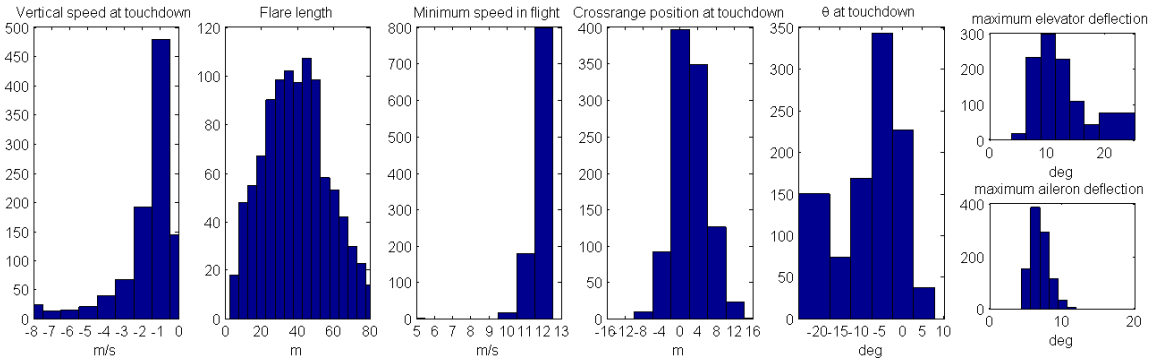


Figure 5.4: Histogram of Easy Star Monte Carlo results with 2.81 m/s winds at 36.5° to the runway.

in the presence of sensor noise, digital lowpass filters are added to measurements of heading angle, glideslope angle, azimuth angle, and descent rate. The filtered values y_k are given in terms of the previous value y_{k-1} and the most recent measurement x_k by:

$$y_k = \alpha x_k + (1-\alpha) y_{k-1}, \quad \alpha \in [0,1] \tag{5.1}$$

The values of α were selected iteratively to reduce the standard deviation in filtered state errors while retaining approximately zero mean error. Values of 0.15, 0.05, 0.1, and 0.1 were used for the heading, glideslope, azimuth, and descent rate states respectively. These additions were found to be sufficient to make turbulence-free performance comparable to the performance in the deterministic simulations already conducted.

Turbulence intensities of 0, 1, 2.6 and 5.1 mph are considered so that performance degradation in turbulence can be quantified. All simulations begin at $X = 1000$ m, $Y = 200$ m, and $\Gamma(0) = 3°$, with initially zero perturbed states. The primary performance metric of interest is the descent rate at touchdown, and the corresponding

Turbulence	Soft	Hard	Non-landings	Landings off runway	Mean flare length (m)
no turbulence	421	35	39	0	272
1.0 mph	346	90	34	0	277
2.5 mph	155	147	74	18	287
5.0 mph	41	58	120	160	332

Table 5.2: Landing performance of Pegasus controllers for various levels of turbulence. All simulations include sensor noise. 500 simulations are conducted at each condition. Target runway width is 11 m (three times the wingspan).

proportions of soft and hard landings, as well as non-landings. Secondary metrics of interest are the pitch and roll angles at touchdown, X-position at touchdown, and the distribution of the final descent rates in simulations.

Table 5.2 shows the landing performance of the Pegasus models for the four sets of simulations conducted. For the zero-turbulence case, the rate of soft landings is about 80%, which is a noticeable degradation from the deterministic simulations. It should be noted that in terms of descent rate, there are no crashes in this initial simulation. Performance is much poorer as the turbulence intensity increases. At a 1 mph turbulence intensity, the effect is relatively minor; there are approximately 50 fewer soft landings, 30 more hard landings, and 20 crashes with significant damage to the vehicle. At 2.5 mph turbulence intensity, the number of soft landings drops by more than 50% compared to the zero-turbulence case and the number of hard landings triples; the descent rate in 137 simulations corresponds to assumed crashes. Performance is even worse when the turbulence intensity doubles to 5.0 mph, to the extent that basic vehicle safety can no longer be ensured in a majority of cases.

In addition, at the highest turbulence level, the proportion of landings outside the 11 m target runway is approximately 30%. It should be noted that this is only a measure of controller accuracy, and not representative of physically landing off the

runway at the Riverside test facility. The physical runways available are approximately 33 m wide, and only 16 landings are outside this value in the worst turbulence. The mean flare length increases only very slightly with turbulence intensity; clearly the turbulence has the greatest effect on the descent rate, of the variables considered.

Figs. 5.5-5.6 show histograms of some of the key states at touchdown in the simulations. These figures demonstrate that the primary effect of turbulence on these states is on the final attitude and descent rate. With no turbulence, final pitch angles are generally between -2.5°and -15°, and almost all bank angles are less than 5°. With 1 mph turbulence, the distributions of descent rate and final pitch angle shift to the left slightly and the standard deviation increases; the spread in the bank angle distribution increases slightly as well. With 2.6 mph turbulence, the bounds on pitch angles increase to approximately 0°-20°, and bank angle limits essentially double compared to the zero turbulence case. In the worst-case turbulence, the pitch angle bounds increase slightly more, and the mean appears to shift to lower pitch angles as well. Bank angles in this case are mainly bounded by ±15°. The final descent rates for each case have already been discussed with the landing performance in Table 5.2; it is sufficient to say that as turbulence increases, the mean descent rate decreases and the 3σ bounds increase. It in interesting to note that the final X-position for most landings is between ±100 m in almost all simulations. The exceptions occur in the presence of turbulence. As turbulence increases, a small number of models do not land after first reaching the flare height, but land much further down the runway. Mostly likely these cases are driven almost entirely by turbulence.

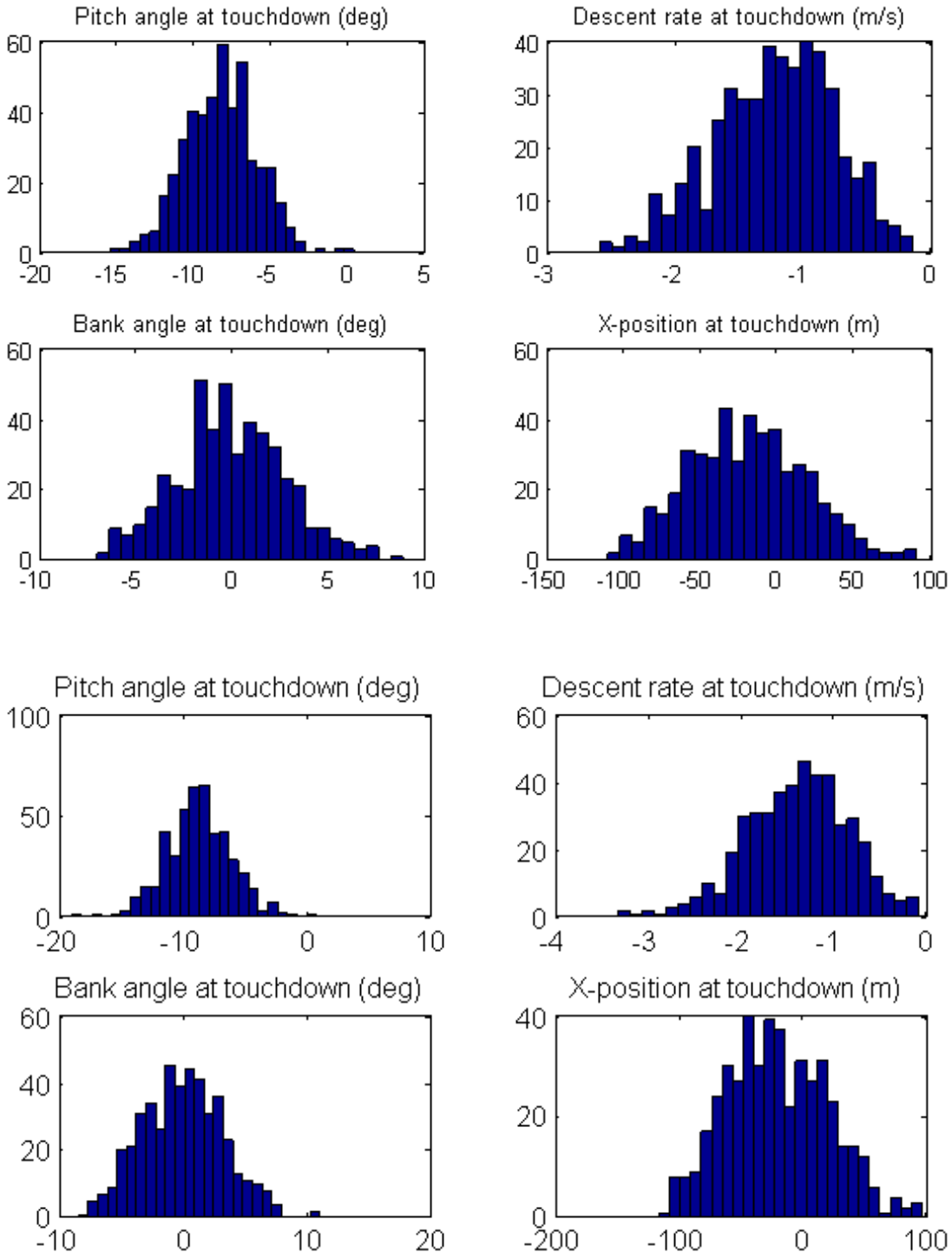


Figure 5.5: Summary of final states for Pegasus Monte Carlo simulations with no turbulence (top) and 1 mph turbulence (bottom).

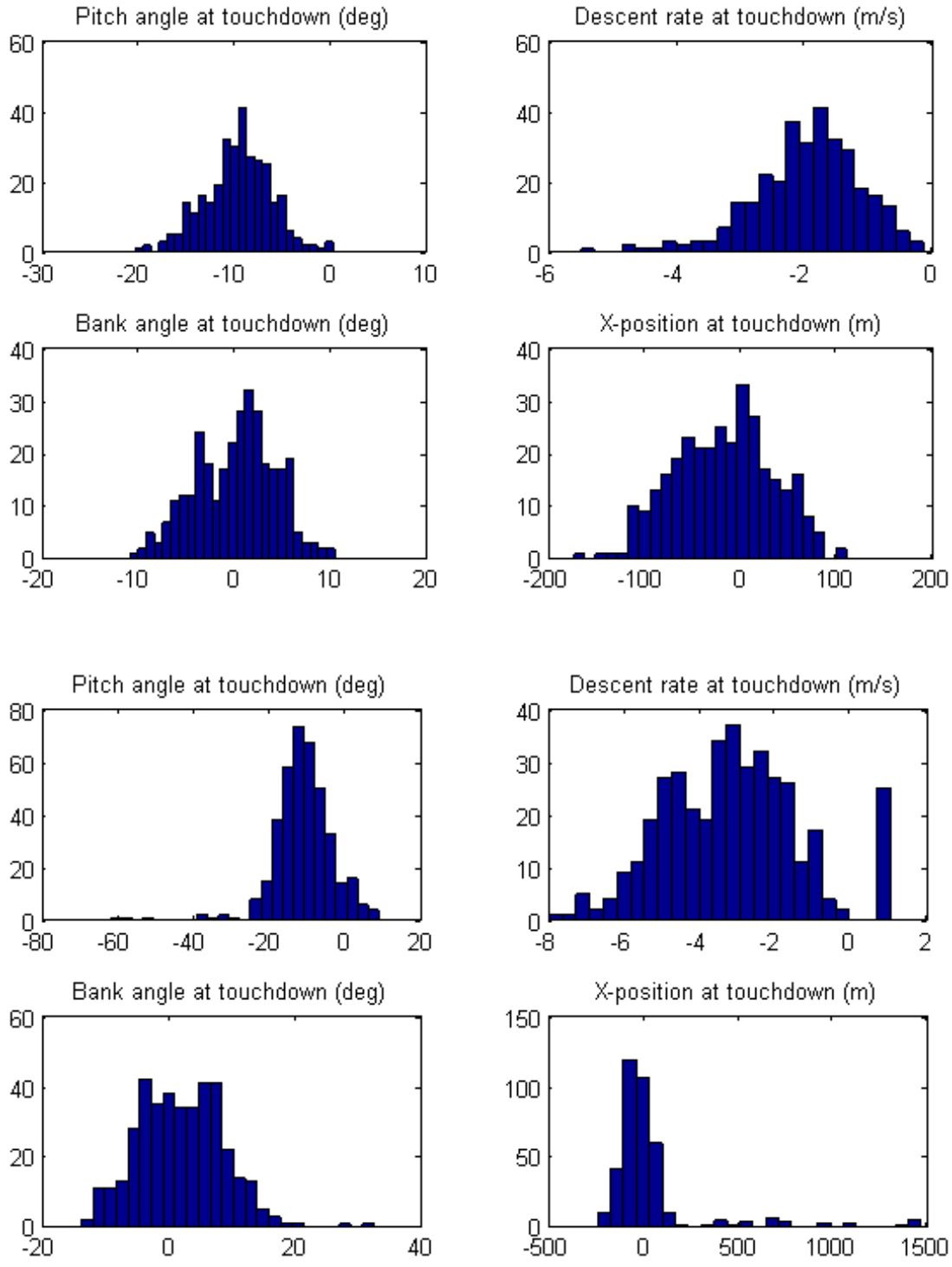


Figure 5.6: Summary of final states for Pegasus Monte Carlo simulations with 2.6 mph turbulence (top) and 5.1 mph turbulence (bottom).

6. FLIGHT TEST RESULTS

Flight tests of the Easy Star began in January 2014. The full autolanding sequence was realized by implementing the innermost loop for each control surface, and adding the outer loops on subsequent flights. The sequence in which each control loop was added and evaluated is:

- Bank angle command and hold to $\phi = 0$
- Heading command and hold to $\psi = 0$
- Pitch angle command and hold to $\theta = 0$
- Airspeed command and hold to $u = 0$
- Azimuth and glideslope tracking loop down to the flare altitude
- Full landing sequence with automatic flare

This section presents the flight tests results using the automatic landing controller on the Easy Star platform. First, a brief description is given of hardware added to the vehicle to increase the ability of the remote pilot to recover in the event of an autopilot failure. Subsequently, the sequence of flight tests performed is described, including representative results of the glideslope, azimuth, and flare tracking loops, before displaying results from the successful automatic landings conducted.

6.1 Modifications to vehicle

Flying the automatic landing control law requires modification of the APM software. The possibility of a software problem in flight cannot be discounted and could very likely lead to the loss of the vehicle. To ensure that a remote human pilot can

Date	Wind	Objectives	Number of flights
2014/01/24	5-10 mph	Manual flying, operator familiarization	1
2014/01/25	5 mph	Bank and heading hold	2
2014/01/29	0-5 mph	Bank, pitch, and heading hold	5
2014/02/01	0-5 mph	APM2 waypoint navigation	1
2014/02/03	5-10 mph	APM2 waypoint navigation	1
2014/02/08	5-10 mph	APM2 waypoint navigation, Full state regulation	2
2014/02/09	5-10 mph	Azimuth and Glideslope	2
2014/03/11	10 mph	Azimuth and Glideslope	1
2014/03/12	15 mph	Azimuth and Glideslope	1
2014/03/13	5-10 mph	Flare, automatic landing	10

Table 6.1: Test matrix for Easy Star automatic control flight tests. This includes all tests for which the vehicle was instrumented with the autopilot, and does not include prior flights for operator familiarization. The center of gravity for all flights is kept at 1/4 of the wing root chord.

always take over in the event of an emergency, the vehicle is equipped with a fail-safe multiplex circuit board that allows the human pilot to override the autopilot by switching from autopilot outputs to RC receiver outputs. This increases the chances of recovery in the event of a major failure in the autopilot hardware or software.

6.2 Flight test results

A total of twenty-six flights were conducted between January and March of 2014 for the purpose of evaluating the automatic landing controller on the Easy Star. A summary of the flight dates and test conditions and objectives is given in Table 6.1. This list includes several flights evaluating the effectiveness of waypoint navigation with the APM2's built-in control loops, which was considered for use in setting up the automatic landings. Ultimately, the RC pilot preferred to manually set up and engage the automatic landing, rather than using waypoint navigation for this task.

The basic regulator loops for roll, pitch, and yaw were evaluated qualitatively during flight. As discussed in Section 4, preliminary heading-axis regulation time histories indicated that the controller was underdamped, leading to a redesign. The response of the other axes was judged to be acceptable.

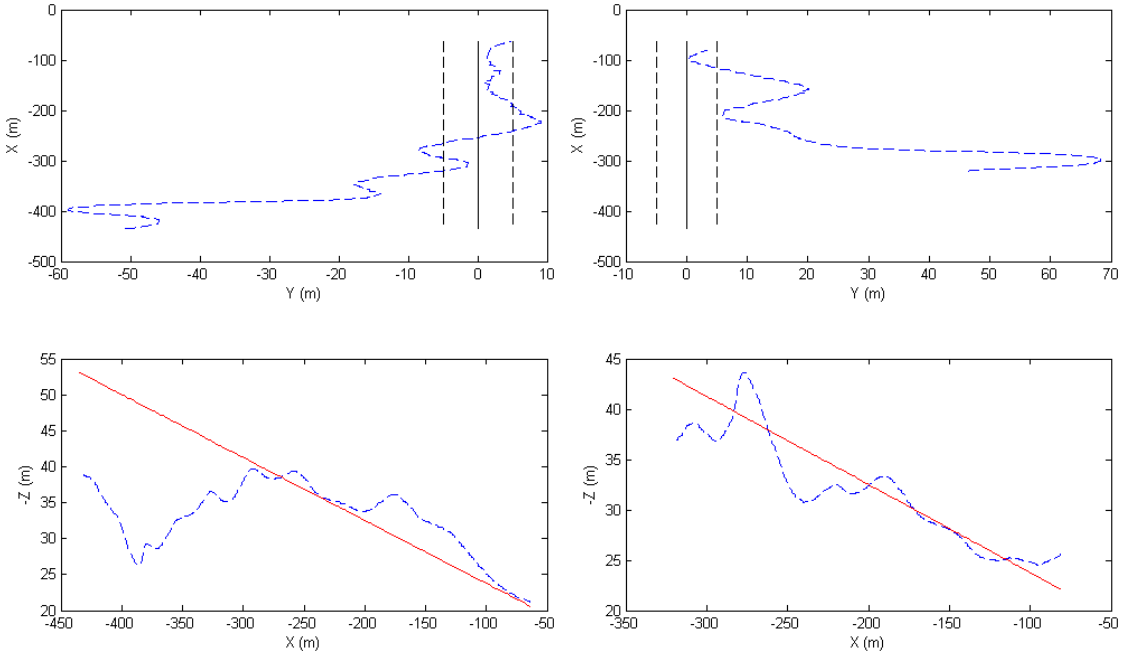


Figure 6.1: Easy Star experimental glideslope and azimuth tracking position flight test histories. Reference 5 m wide landing target and glidepath are shown. The target landing waypoint is at $(X, Y, Z) = (0, 0, -15)$ m.

To test the glideslope and azimuth tracking loops, the autolanding waypoint was artificially placed 15 m above the runway, giving the backup remote operator time to recover in the event of a hardware or software problem. In these trials, the automatic landing flight mode was triggered manually at different vehicle headings to evaluate the controller performance; the landing maneuvers were aborted by the remote pilot after several seconds of automatic flight. Flight paths from two test maneuvers are

shown in Fig. 6.1. Despite initial position errors and somewhat unfavorable initial conditions, the control law converges to the target approach vector and follows it until the automatic flight mode is disengaged.

After the successful tests of the glideslope and azimuth tracking loops, the final remaining step before attemping automatic landing was to test the automatic flare performance with the 15 m altitude offset. Initial trials indicated the flare tracking loop was significantly underdamped, as shown previously in Sec. 4.1.2.5 (see Figs. 4.16-4.17), and the controller was redesigned. The redesigned flare loop shows maximum typical errors of about 1 m/s relative to the reference, compared to about 2 m/s for the initial controller.

On March 13, 2014, two full automatic landings were completed. The vehicle was flown under manual control to a position approximately 300 m uprange of the waypoint that defines the glideslope, then switched into the automatic landing mode. The inertial position histories and flare performance are shown in Figs. 6.2-6.3. The approach performance is evaluated in terms of the inertial frame position histories. The reference glideslope is plotted in the X-Z plane, and the approximate lateral position of the waypoint with 5 m error lines are plotted for the X-Y plane history. In both cases, the vehicle begins with significant initial errors in glideslope position. This initial error in corrected by the time approximately 100 m have been travelled along the groundpath. The azimuth tracking loops also experience initial error of a few degrees, which are corrected within roughly 150 m. The azimuth tracking loop still appears to be slightly underdamped, despite the modifications to the heading control loop, but this effect is likely driven by the lightweight aircraft flying in unsteady winds.

Flare performance is examined in terms of the position histories below the flare altitude and the descent rate during that flight segment. The flare altitude was 4

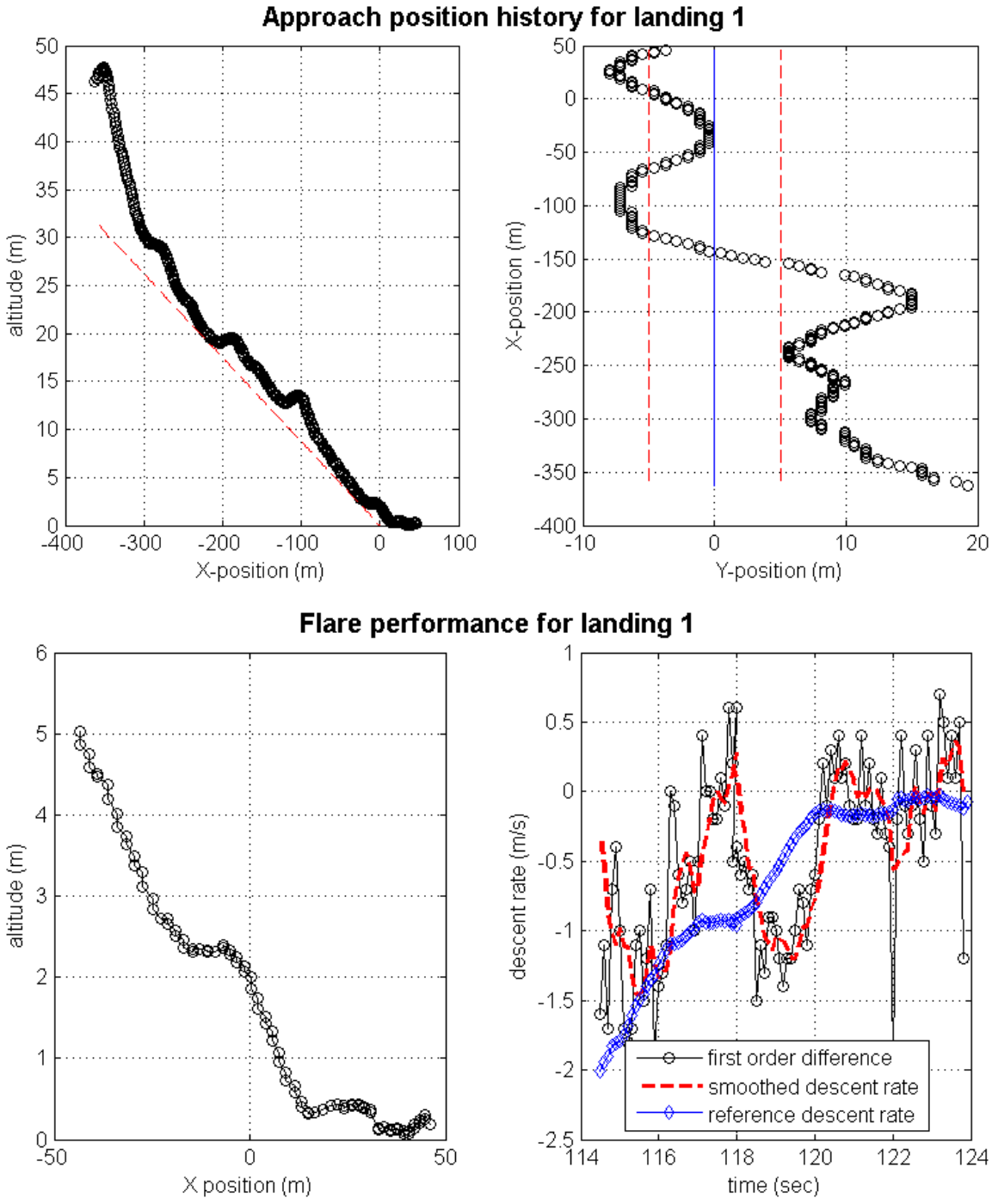


Figure 6.2: Easy Star first experimental automatic landing time history.

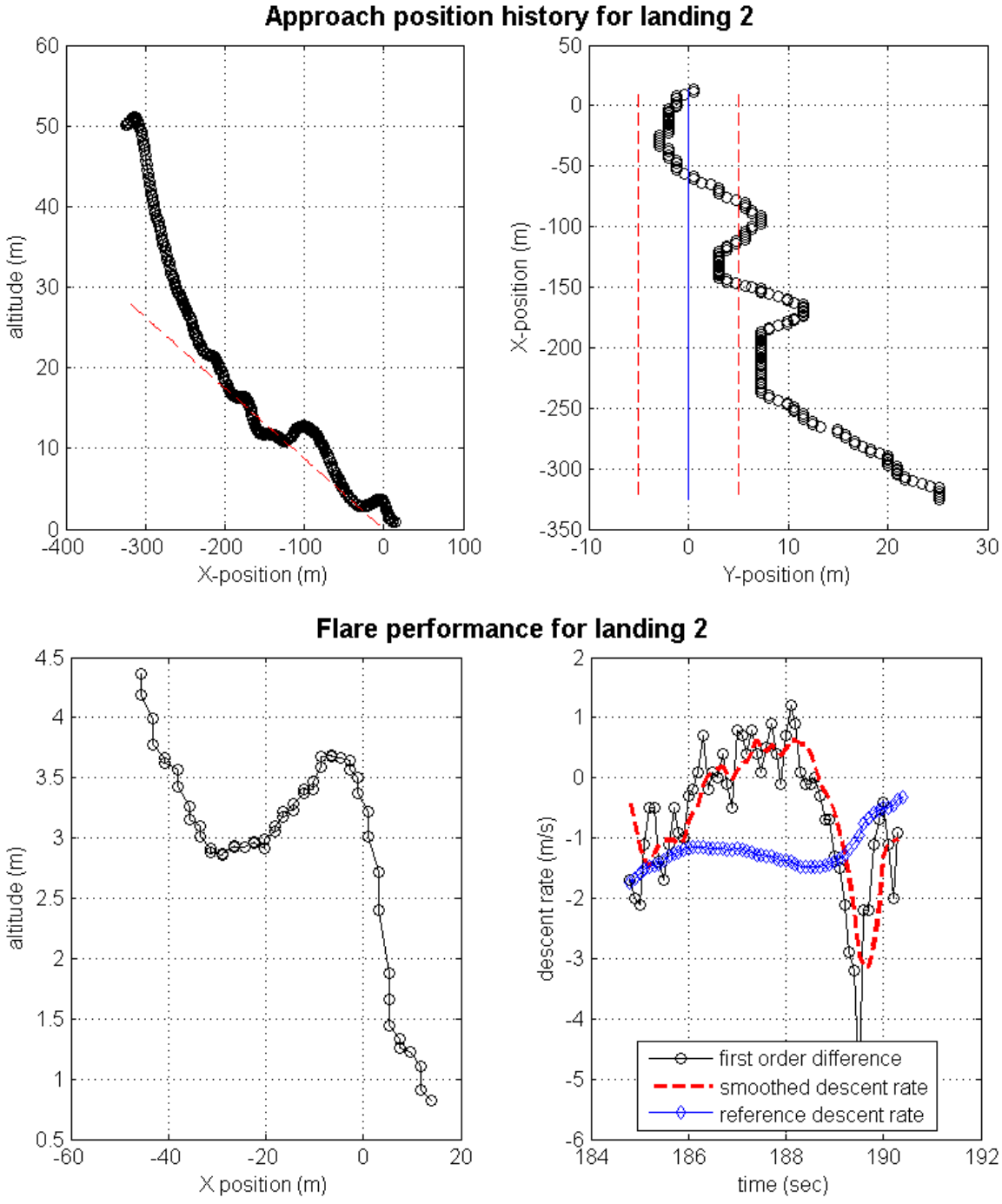


Figure 6.3: Easy Star second experimental automatic landing time history.

m for the Easy Star tests. In Figs. 6.2-6.3, descent rates are computed by a first-order finite difference of the altitude histories; to reduce noise, a five-term moving average is used to approximate a smoothed descent rate history. Flare performance is nearly ideal for the first test. The vehicle travels less than 50 m downrange of the target waypoint and remains within about 1 m/s of the reference descent rate. The final descent rate is nearly zero and is well within the margin for a "soft" landing. During the second automatic landing, the vehicle overcorrected to reduce its descent rate to match the reference, and the touchdown descent rate is on the threshold of a "hard" landing (defined earlier as a descent rate between 1.83 and 3.05 m/s). This manifested as damage to the horizontal tail, and some minor scratching to the pitot-static probe, which was installed in the nose. The aircraft could not be fixed and flown again *in situ*, but was repaired with about two man-hours of work. In examining the flare performance, the response appears to be underdamped. It is not immediately apparent whether this is due more to the design of the inner or outer loop, and unsteady winds most likely are a factor as well. There is room to improve performance with further gain tuning. However, the two flights shown here demonstrate that the basic implementation of the control law on the hardware works as intended and is capable of safely guiding the vehicle to a landing.

This section has outlined experimental automatic landings of the Easy Star platform. Performance in flight test validates the simulation results and, with some adjustments based on experimental performance, the controllers provide adequate performance to achieve autolanding. The next section summarizes the major conclusions and recommendations based on the body of research that has been carried out.

7. CONCLUSIONS AND RECOMMENDATIONS

This section presents a summary of the major findings of this thesis, as well as recommendations for future work. The primary work of this thesis has been to develop an automatic landing controller using Quantitative Feedback Theory. Full controllers have been developed in simulation for the Easy Star and Pegasus UAVs. The controller has been implemented in hardware on the Easy Star platform and two landings have been performed. Based on the research that has been performed, the following conclusions about the control design approach and implementation are presented:

1. The QFT based automatic landing controller has been demonstrated to work in simulation and in flight test. The Easy Star simulations indicate a high probability of a successful landing in relatively calm conditions. The dynamic performance in simulation is qualitatively similar to the flight test results, in that lightly damped dynamic responses in flight test tend to be underdamped in the simulation as well. The Pegasus simulations perform well in the presence of sensor noise and low turbulence. However, the identified Pegasus model is highly sensitive to turbulence on the longitudinal axis, and performance diminishes significantly when the turbulence intensity exceeds 1 mph.

2. Based on the Easy Star flight test results, the X-plane-derived uncertain model captures the true system dynamics reasonably well. The identified linear model should be usable for future linear control synthesis. Additionally, it may be feasible to use X-plane directly for nonlinear dynamic simulations of Easy Star and Bixler aircraft.

3. From a design standpoint, QFT works well for the inner loop controller design but not as well for the outer loops. In general, it is straightforward to use the QFT process on the uncertain inner-loop plants to satisfy the robust performance requirements while providing adequate time-domain performance. For the outer loops, it is, in several cases examined here, prohibitively difficult to satisfy the robust performance criteria and achieve good time-domain performance with the assumed level of uncertainty, while maintaining an acceptably low controller order. In many cases, simple PID controllers were found to give adequate performance on the outer loops, while the inner loops are relied on to ensure robustness and disturbance rejection. Performance in stochastic simulations and flight test supports this control synthesis approach.

Based on the results produced to date, several recommendations are made regarding future work:

1. Additional experimental automatic landings should be performed on the low-cost RC vehicle(s), to ensure the controller performance is repeatable, and to examine the adequacy of the current sensor package. Changes made between the Easy Star and Pegasus simulations should be implemented on the Easy Star to ensure that experimental trials on the two vehicles are consistent, and verify that these additions make landings safer. These changes include a nonconstant reference glideslope, and scheduling with range on the glideslope and azimuth control loops.

2. Pegasus system identification should be repeated and new testing should be performed at different flight conditions for the longitudinal axis. The current longitudinal controllers should then be re-evaluated on the new models.

The identified longitudinal-axis models used in this document show poor consistency between different trials, i.e. the models produced are typically only accurate predictors of the measured history for one or two other trials. This can be partly attributed to the fact that the state excitation in almost all trials exceeded the linear range for angle-of-attack and pitch angle. Additionally, some new trials should be conducted at lower airspeeds, around 20 m/s, which is a typical landing speed based on manual landings. Because of the limitations of the current longitudinal model, simulated landings had to be done near the trim state of 30 m/s, which increased the chance of a hard landing compared to a landing at 20 m/s. Feedback from the human pilot indicates that the vehicle handling characteristics are different, and generally more forgiving, at low speeds than at high speeds, and accurately capturing the low-speed dynamics is essential to ensure vehicle and personnel safety.

3. A rudder control loop should be designed and implemented for lateral/directional control during the flare. The bank angle control loop should be used only to keep the wings level. The primary effect of this change is to allow the Pegasus autopilot to maintain control once touchdown occurs. Currently, directional control cannot be effected by the controller once the vehicle is on the ground, and the remote operator must take over to ensure the vehicle stays on the runway. For the Easy Star, this is not an issue because the vehicle has little momentum and stops after travelling a short distance. For Pegasus, the loss of directional control presents a significant safety hazard, since the vehicle lacks brakes and the pilot's RC transmitter only has an effective range of about 30 m when the vehicle is on the ground. Maintaining zero bank during the flare should also help ensure the aircraft's main wheels touch down nearly

simultaneously, reducing stress on the landing gear.

REFERENCES

[1] XL-MaxSonar-EZ Series. `http://maxbotix.com/documents/XL-MaxSonar-EZ_Datasheet.pdf`. Accessed July 17, 2013.

[2] US Military Specification MIL-F-8785C. Technical report, US Department of Defense, 1980.

[3] Blake Barber, Timothy McLain, and Barrett Edwards. Vision-based landing of fixed-wing miniature air vehicles. *Journal of Aerospace computing, Information, and Communication*, 6(3):207–226, 2009.

[4] D. Blake Barber, Stephen R. Griffiths, Timothy W. McLain, and Randal W. Beard. Autonomous landing of miniature aerial vehicles. *Journal of Aerospace Computing, Information, and Communication*, 4(5):770–784, 2007.

[5] Geoffrey L. Barrows, Javaan S. Chahl, and Mandyam V. Srinivasan. Biomimetic visual sensing and flight control. In *Proc. Bristol UAV Conf*, pages 159–168. Citeseer, 2002.

[6] John H Blakelock. *Automatic control of aircraft and missiles*, volume 6. Wiley New York, 1991.

[7] Craig Borghesani, Yossi Chait, and Oded Yaniv. *The QFT Frequency Domain Control Design Toolbox For Use With MATLAB®: User's Guide*. Terasoft, July 2003.

[8] Haiyang Chao, Yongcan Cao, and YangQuan Chen. Autopilots for small fixed-wing unmanned air vehicles: A survey. In *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*, pages 3144–3149. IEEE, 2007.

[9] John L Crassidis and John L Junkins. *Optimal estimation of dynamic systems*. CRC press, 2nd edition, 2011.

[10] DIY Drones. Official Arduplane Repository. `https://code.google.com/p/ardupilot-mega/wiki/home?tm=6`. Accessed May 24, 2013.

[11] John C. Doyle. Quantitative feedback theory (QFT) and robust control. In *American Control Conference, 1986*, pages 1691–1698. IEEE, 1986.

[12] Freescale Semiconductor. Freescale Semiconductor MPX7002 Series. `http://www.freescale.com/files/sensors/doc/data_sheet/MPXV7002.pdf`. Accessed July 13, 2013.

[13] Amos Gera. *Feedback systems with highly uncertain plants*. PhD thesis, Weizmann Institute of Science, 1980.

[14] Horizon Hobby. HS-5055MG Digital Metal Gear Feather Light Servo. `http://www.horizonhobby.com/products/hs-5055mg-digital-metal-gear-feather-light-servo-HRC35055S#t2`. Accessed March 31, 2014.

[15] I.M. Horowitz. Fundamental theory of automatic linear feedback control systems. *IRE Transactions on Automatic Control*, 4(3):5–19, 1959.

[16] Isaac Horowitz. Quantitative feedback theory. *IEE Proceedings, Pt. D*, 129(6), November 1982.

[17] Isaac M. Horowitz. A synthesis theory for linear time-varying feedback systems with plant uncertainty. *IEEE Transactions on Automatic Control*, AC-20(4):454–464, 1975.

[18] Isaac M. Horowitz. Synthesis of feedback systems with nonlinear time-varying uncertain plants to satisfy quantitative performance specifications. *Proceedings of the IEEE*, 64(1):123–130, 1976.

[19] Isaac M. Horowitz and Moshe Breiner. Quantitative synthesis of feedback systems with uncertain nonlinear multivariable plants. *International Journal of Control*, 12(5):539–563, 1981.

[20] Isaac M. Horowitz and Marcel Sidi. Synthesis of feedback systems with large plant ignorance for prescribed time domain tolerances. Technical report, University of Colorado, Boulder, Colorado, July 1971.

[21] C.H. Houpis and S.J. Rasmussen. Unmanned research vehicle: development, implementation, and flight test of a mimo digital flight control system designed using quantitative feedback theory. In *1999 Intentational Symposium on Quantitative Feedback Theory and Robust Frequency Domain Methods*, pages 1–13, 1999.

[22] Sungsik Huh and David Hyunchul Shim. A vision-based automatic landing method for fixed-wing uavs. In *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*, pages 217–231. Springer, 2010.

[23] ICON Aircraft. ICON A5 Specs. `http://www.iconaircraft.com/a5-model.html`. Accessed January 20, 2014.

[24] InvenSense. MPU-6000/6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking™ Devices. `http://www.invensense.com/mems/gyro/mpu6050.html`. Accessed May 03, 2014.

[25] Ravindra V. Jategaonkar. *Flight vehicle system identification: a time domain methodology*, volume 216. American Institute of Aeronautics and Astronautics Reston, Virginia,, USA, 2006.

[26] Jer-Nan Juang. *Applied system identification*. Prentice Hall, 1994.

[27] M.S. Keating, M. Pachter, and C.H. Houpis. QFT applied to fault tolerant flight control system design. In *American Control Conference, 1995. Proceedings of the*, volume 1, pages 184–188. IEEE, 1995.

[28] M.L. Kerr, C.Y. Lan, and S. Jayasuriya. Non-sequential MIMO QFT control of the X-29 aircraft. In *7th International Symposium on Quantitative Feedback Theory QFT and Robust Frequency Domain Methods*, 2005.

[29] Jong-Hyuk Kim, Salah Sukkarieh, and Stuart Wishart. Real-time navigation, guidance, and control of a uav using low-cost sensors. In *Field and Service Robotics*, pages 299–309. Springer, 2006.

[30] Derek B. Kingston and Randal W. Beard. Real-time attitude and position estimation for small uavs using low-cost sensors. In *AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*, 2004.

[31] Laminar Research. Appendix A: How X-Plane Works. `http://wiki.x-plane.com/Appendix_A:_How_X-Plane_Works`. Accessed June 15, 2013.

[32] Monika Marwaha, John Valasek, and Puneet Singla. GLOMAP approach for nonlinear system identification of aircraft dynamics using flight data. In *Proc. AIAA Atmospheric Flight Mechanics Conference*, pages 1–19, 2008.

[33] MathWorks®. Butterworth filter design. `http://www.mathworks.com/help/signal/ref/butter.html`. Accessed March 31, 2014.

[34] MathWorks®. Dryden Wind Turbulence Model (Continuous). http://www.mathworks.com/help/aeroblks/drydenwindturbulencemodelcontinuous.html. Accessed June 17, 2013.

[35] Measurement Specialties. MS5611-01BA03 Barometric Pressure Sensor. http://www.meas-spec.com/downloads/MS5611-01BA03.pdf. Accessed July 13, 2013.

[36] MediaTek. MEDIATEK-3329 Datasheet Rev.A03. https://docs.google.com/file/d/0B_dHj7E2weiiNmUzNDA3OTktNTNhNy00Y2Y5LTg0YTQtMzIyNzJhZmFiNjcy/edit. Accessed July 13, 2013.

[37] Model Motors. AXI 2217/12 GOLD LINE. http://www.modelmotors.cz/index.php?page=61&product=2217&serie=12&line=GOLD. Accessed July 20, 2013.

[38] Multiplex. Kit Easystar II. http://www.multiplex-rc.de/en/products/categories/products/details/productgroup/rc-accessories/productcategory/elapor-models-1/product/kit-easystar-ii.html.

[39] Kyungmoon Nho and Ramesh K Agarwal. Automatic landing system design using fuzzy logic. *Journal of Guidance, Control, and Dynamics*, 23(2):298–304, 2000.

[40] NICTA. Armadillo C++ linear algebra library. http://arma.sourceforge.net/. Accessed February 11, 2014.

[41] Michael Pursifull. Hilstar17 Public Release. http://www.diydrones.com/profiles/blogs/hilstar17-public-release. Accessed June 11, 2013.

[42] Jan-Cor Roos and IK Peddle. Autonomous take-off and landing of a low cost unmanned aerial vehicle. *University of Stellenbosch*, 2007.

[43] J. Roskam. *Airplane Flight Dynamics and Automatic Flight Controls: Rigid and elastic airplane flight dynamics and automatic flight controls*. Roskam Aviation and Engineering Corporation, 1979.

[44] Jan Roskam. *Airplane Flight Dynamics and Automatic Flight Controls*. DAR-Corporation, 2009.

[45] A. Santander and J. Aranda. QFT for the design of an aircraft flight control. In *IFAC World Congress*, 2005.

[46] Falko Schuck, Matthias Heller, Thaddäus Baier, and Florian Holzapfel. Longitudinal robust controller for excellent handling qualities design of a general aviation aircraft using QFT. 2013.

[47] Servo City. HS-311 Standard. `http://www.servocity.com/html/hs-311_standard.html`. Accessed July 14, 2013.

[48] Stuart N. Sheldon and Steven J. Rasmussen. Development and first successful flight test of a qft flight control system. In *Aerospace and Electronics Conference, 1994. NAECON 1994., Proceedings of the IEEE 1994 National*, pages 629–636. IEEE, 1994.

[49] Thomas William Wagner. Digital autoloand system for unmanned aerial vehicles. Master's thesis, Texas A&M University, 2007.

[50] Weather Underground. Personal Weather Station: KTXBRYAN19. `http://www.wunderground.com/personal-weather-station/dashboard?ID=KTXBRYAN19#history/data/s20130914/e202404/mtoday`. Accessed April 15, 2014.

[51] Shu-Fan Wu, Michael J. Grimble, and Wei Wei. QFT based robust/fault tolerant flight control design for a remote pilotless vehicle. In *Control Applications, 1999.*

Proceedings of the 1999 IEEE International Conference on, volume 1, pages 57–62. IEEE, 1999.

[52] Oded Yaniv. *Quantitative feedback design of linear and nonlinear control systems*. Springer, 1999.

APPENDIX A

EASY STAR MODEL IDENTIFICATION

A.1 Data generation and model fitting

Use of X-plane ensures that full state and control histories with zero noise can be obtained at a rate faster than the expected Nyquist frequency of the UAV. It should be noted that the identified model is only used when considering the vehicle kinetic states. The linearized vehicle kinematic relationships are exactly known, and are used rather than an identified model. In generating state and control time histories for system identification, it is essential that all relevant system dynamic modes be excited without perturbing the aircraft states beyond the regime in which behavior is nearly linear. Two types of control inputs are used to generate data, as in [49]. (1) 3-2-1-1 inputs, in which a control is deflected to one extreme for three seconds, then to the opposite extreme for two seconds, followed by a deflection to each extreme for one second each. (2) Doublets, in which the control is deflected fully to one extreme for one second, then deflected to the other extreme for one second.

To identify a lateral-direction model, a 3-2-1-1 aileron maneuver followed immediately by a 3-2-1-1 rudder maneuver is performed. The lateral/directional model is evaluated on a separate data set in which an aileron doublet is followed by a rudder doublet. The longitudinal model is identified with a 3-2-1-1 elevator maneuver followed by a 3-2-1-1 throttle maneuver, and is evaluated with an elevator doublet followed by a throttle doublet. System identification is performed by generating discrete-time models that best satisfy the given data in a least-squares sense. For a

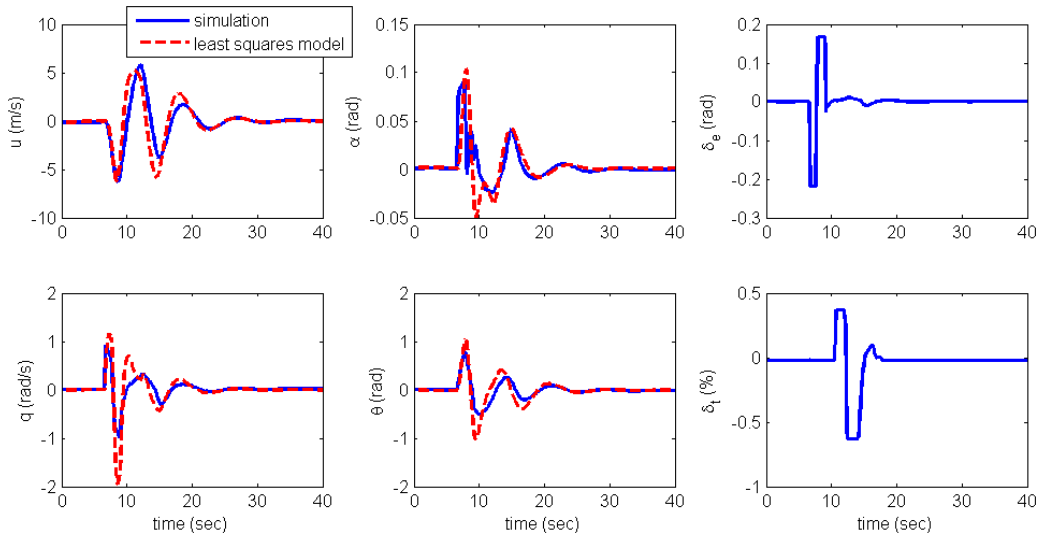


Figure A.1: Identified Easy Star longitudinal axis model response to simulation inputs. Simulation response (red) plotted for comparison.

discrete-time linear system of the form:

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k \ (k \in [0, N-1]) \tag{A.1}$$

The error vector $\mathbf{e}_k$ is defined as:

$$\mathbf{e}_k = \mathbf{x}_{k+1} - (A\mathbf{x}_k + B\mathbf{u}_k) \ (k \in [0, N-1]) \tag{A.2}$$

The least-squares criterion that is minimized is:

$$\sum_{i=1}^{N} \mathbf{e}_k^T \mathbf{e}_k \tag{A.3}$$

This least-squares fit is feasible in this instance because the data have no noise.

In Fig. A.1, the responses of the identified and simulation longitudinal models to the doublet control inputs are compared. The identified model matches the frequency

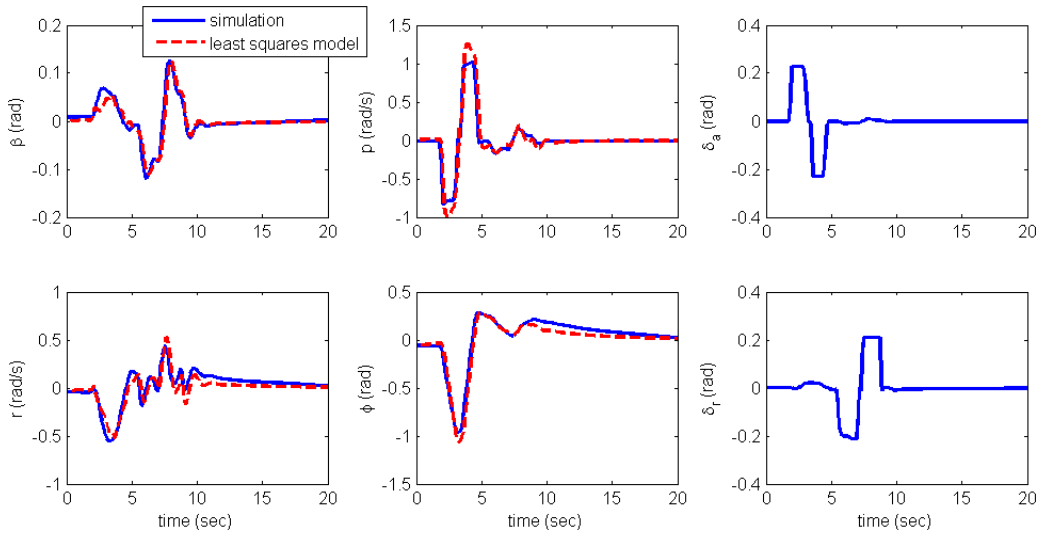


Figure A.2: Identified Easy Star lateral/directional axes Easy Star model response to simulation inputs. Simulation response (red) plotted for comparison.

content of the simulation model well. The most obvious magnitude discrepancies are in the angle-of-attack and pitch rate channels. Since the frequency content of the identified model matches, and the simulated model is expected to have errors compared to the true glider, the accuracy of the identified longitudinal model is considered acceptable.

Fig. A.2 plots the responses of the identified and simulation lateral/directional models to the verification doublets. For the most part, these outputs match well in both amplitude and frequency content. There is a discrepancy in the roll rate response amplitude to the aileron input. Again, the modeled response to considered acceptable for preliminary control design, given the limited accuracy of the simulation model and the similarity of the frequency content.

A.2 Vehicle scaling

Due to the limitations of the X-plane 9 software, vehicles massing less than 2.2 kg cannot reliably be simulated[41]. Consequently, the 3D model used for identification

is scaled up by a factor of 1.7 in length, and a factor of 3 in mass. Defining $\tilde{l}$ and $\tilde{m}$ as the corresponding scale factors, it is assumed that the moments of inertia of the simulation model have been scaled by a factor of $\tilde{I} \equiv \tilde{m}\tilde{l}^2$. Due to the nature of aircraft stability derivatives, the linear models obtained must be rescaled to match the flight vehicle dimensions[44]. Define $\tilde{A}_1$ and $\tilde{B}_1$ as the longitudinal axis matrix variables, and $\tilde{A}_2$ and $\tilde{B}_2$ as the lateral/directional matrix variables. Further, let $\tilde{A}_{ij}$ indicate the entry in the ith row, jth column. The identified continuous-time models are rescaled as follows:

$$[A_1] = \begin{bmatrix} \tilde{A}_{11_1}\frac{\tilde{m}}{\tilde{S}} & \tilde{A}_{12_1}\frac{\tilde{m}}{\tilde{S}} & \tilde{A}_{13_1}\frac{\tilde{m}}{\tilde{S}\tilde{l}} & \tilde{A}_{14_1} \\ \tilde{A}_{21_1}\frac{\tilde{m}}{\tilde{S}} & \tilde{A}_{22_1}\frac{\tilde{m}}{\tilde{S}} & (\tilde{A}_{23_1}-1)\frac{\tilde{m}}{\tilde{S}\tilde{l}}+1 & \tilde{A}_{24_1} \\ \tilde{A}_{31_1}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{A}_{32_1}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{A}_{33_1}\frac{\tilde{I}}{\tilde{S}\tilde{l}^2} & \tilde{A}_{34_1} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$[B_1] = \begin{bmatrix} \tilde{B}_{11_1}\frac{\tilde{m}}{\tilde{S}} & \tilde{B}_{12_1}\frac{\tilde{m}}{\tilde{S}} \\ \tilde{B}_{21_1}\frac{\tilde{m}}{\tilde{S}} & \tilde{B}_{22_1}\frac{\tilde{m}}{\tilde{S}} \\ \tilde{B}_{31_1}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{B}_{32_1}\frac{\tilde{I}}{\tilde{S}\tilde{l}} \\ 0 & 0 \end{bmatrix}$$

$$[A_2] = \begin{bmatrix} \tilde{A}_{11_2}\frac{\tilde{m}}{\tilde{S}} & \tilde{A}_{12_2}\frac{\tilde{m}}{\tilde{S}\tilde{l}} & (\tilde{A}_{13_2}-1)\frac{\tilde{m}}{\tilde{S}\tilde{l}}+1 & \tilde{A}_{14_2} \\ \tilde{A}_{21_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{A}_{22_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}^2} & \tilde{A}_{23_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}^2} & \tilde{A}_{24_2} \\ \tilde{A}_{31_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{A}_{32_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}^2} & \tilde{A}_{33_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}^2} & \tilde{A}_{34_2} \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$[B_2] = \begin{bmatrix} \tilde{B}_{11_2}\frac{\tilde{m}}{\tilde{S}} & \tilde{B}_{12_2}\frac{\tilde{m}}{\tilde{S}} \\ \tilde{B}_{21_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{B}_{22_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} \\ \tilde{B}_{31_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} & \tilde{B}_{32_2}\frac{\tilde{I}}{\tilde{S}\tilde{l}} \\ 0 & 0 \end{bmatrix}$$

After rescaling, the vehicle longitudinal continuous-time model has the following dynamic modes and eigenvalues λ:

$$\lambda_1 = -1.4 \pm 3.9i \quad \lambda_2 = -0.33 \pm .36i$$
$$\zeta_1 = .33 \qquad \zeta_2 = .67$$
$$\omega_1 = 4.2 \ \tfrac{\text{rad}}{\text{sec}} \qquad \omega_2 = .48 \ \tfrac{\text{rad}}{\text{sec}}$$

The lateral/directional dynamic characteristics are:

$$\lambda_1 = -.28 \quad \lambda_2 = -1.6 \pm 6.2 \quad \lambda_3 = -13$$
$$\tau_1 = 3.6 \text{ sec} \qquad \zeta_2 = .25 \qquad \tau_3 = 0.076 \text{ sec}$$
$$\omega_2 = 6.4 \ \tfrac{\text{rad}}{\text{sec}}$$

The continuous-time linear longitudinal model is:

$$\begin{bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -0.548 & 0.0493 & -0.149 & -8.09 \\ -0.0117 & -0.268 & 0.464 & 0.0152 \\ 0.158 & -33.8 & -2.59 & -1.26 \\ 0 & 0 & 1.00 & 1 \end{bmatrix} \begin{bmatrix} u \\ \alpha \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} -1.41 & 4.80 \\ 0.201 & -0.0100 \\ -34.9 & -1.57 \\ 0.000 & 0.000 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix} \tag{A.4}$$

$$U_1 = 12.6 \text{ m/s} \tag{A.5}$$

$$\alpha_1 = -0.0293 \text{ rad} \tag{A.6}$$

$$\theta_1 = -0.0197 \text{ rad} \tag{A.7}$$

The continuous-time lateral-directional model is:

$$\begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -2.23 & 0.248 & 0.770 & -0.326 \\ 35.2 & -12.0 & 4.73 & -4.62 \\ -54.1 & 2.17 & -2.27 & 0.282 \\ 0 & 1.00 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} 1.83 & 0.606 \\ -101 & -6.02 \\ 16.5 & 21.1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{A.8}$$

APPENDIX B

PEGASUS MODEL IDENTIFICATION

B.1 Longitudinal model selection

Models are evaluated by computing the mean squared error in predicting the data in all other segments on the same day. The models on each day with consistently good performance are further downselected by comparison against data from other days. Figs. B.1-B.2 plot the base 10 logarithm of the mean squared error for each model evaluated on each other trial from the same day. Smaller values indicate a more accurate fit.

Based on the preceding analysis, three models were evaluated in terms of MSE against data from other flight days: Models 1 and 3 from day 1 and Model 2 from day 2. Table B.1 collects the MSEs when these models are evaluated in this fashion. Based on these results, either Day 1 Trial 3 or Day 2 Trial 2 have good performance that is relatively consistent across data sets.

To further compare the two models, the Theil inequality coefficient (TIC) is

Model	Day	$log_{10}(MSE_u)$	$log_{10}(MSE_\alpha)$	$log_{10}(MSE_q)$	$log_{10}(MSE_\theta)$
Day 1 Trial 3	1	1.236766	-1.790962	-1.446248	-1.36449
Day 1 Trial 3	2	1.625153	-2.081535	-0.886077	-0.938095
Day 1 Trial 1	1	1.909165	-1.393750	-1.582307	-1.304716
Day 1 Trial 1	2	2.337564	-0.355356	-1.236039	-0.886734
Day 2 Trial 2	1	1.898684	-0.700737	-1.736099	-0.838319
Day 2 Trial 2	2	1.125815	-1.719940	-2.252951	-1.834387

Table B.1: Mean squared errors of the best Pegasus longitudinal models identified using OKID.

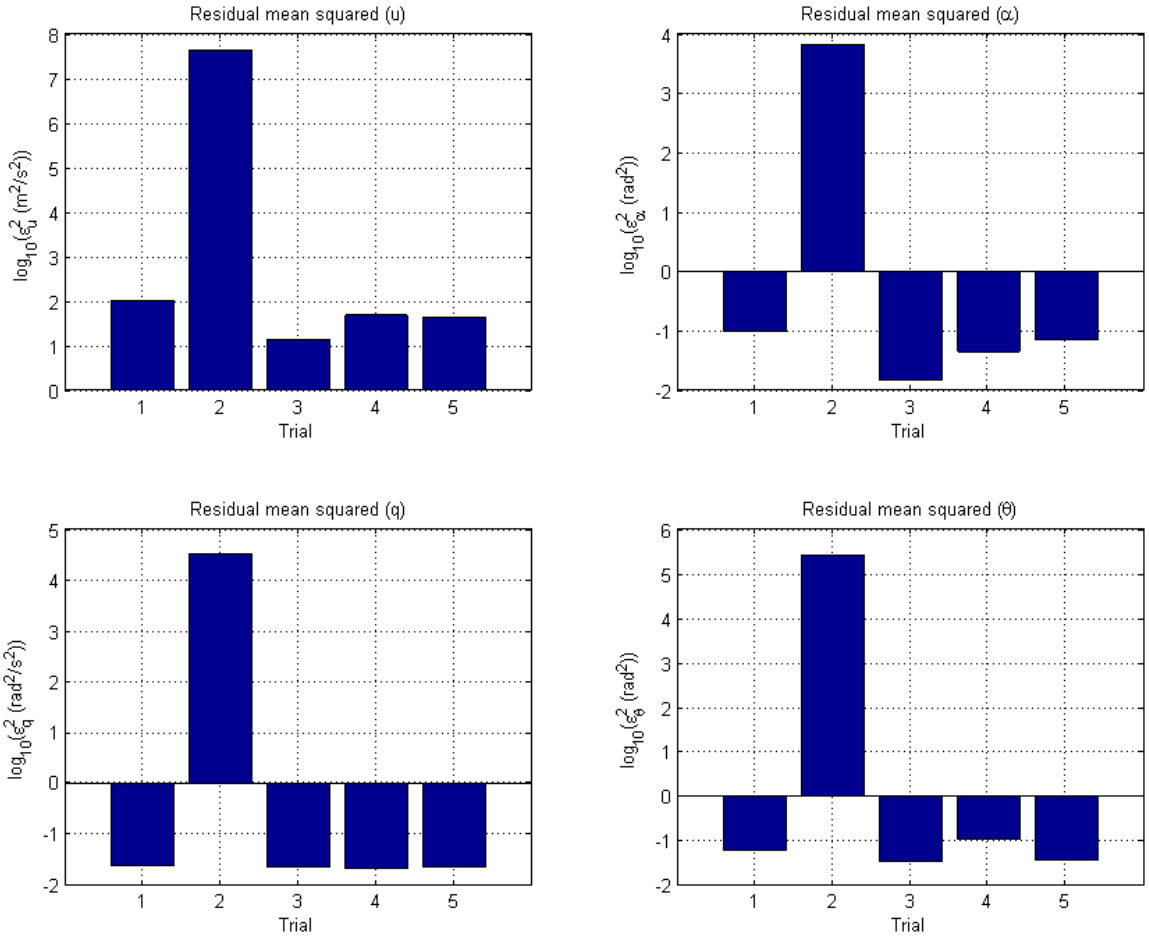


Figure B.1: Pegasus longitudinal-axis MSEs for day 1 of Pegasus flights.

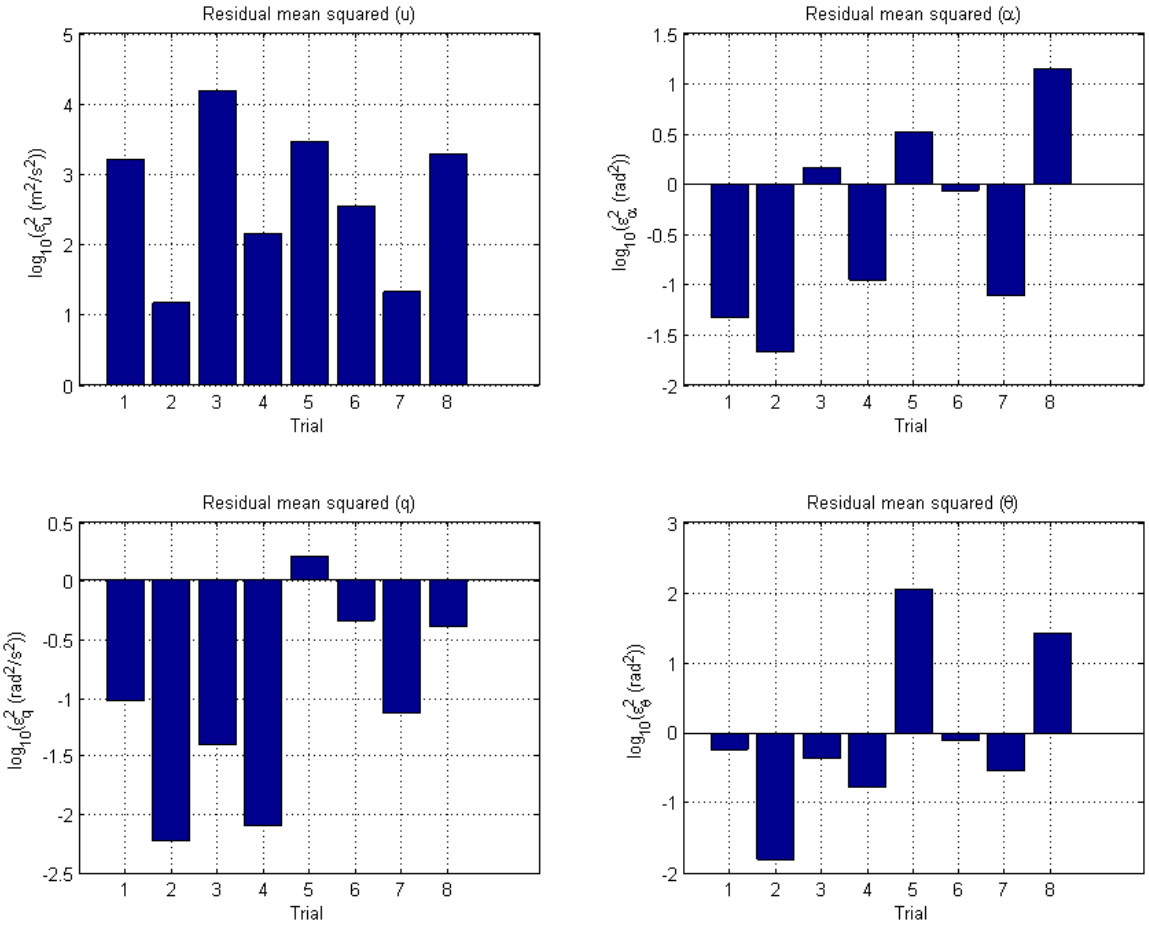


Figure B.2: Pegasus longitudinal-axis MSEs for day 2 of Pegasus flights.

Day 1 Trial 3				
Trial	TIC_u	TIC_α	TIC_q	TIC_θ
Day 1 Trial 1	0.8607	0.2410	0.3922	0.4303
Day 1 Trial 3	0.4673	0.2093	0.2906	0.4096
Day 2 Trial 2	0.8514	0.3459	0.3812	0.5575
Day 2 Trial 3	0.9214	0.2766	0.4391	0.7971
Day 2 Trial 4	0.7214	0.3326	0.3618	0.4647
Total	0.7252	0.4445	0.5527	0.7020
Day 2 Trial 2				
Trial	TIC_u	TIC_α	TIC_q	TIC_θ
Day 1 Trial 1	0.9103	0.5004	0.2863	0.4010
Day 1 Trial 3	0.4866	0.3521	0.2468	0.1611
Day 2 Trial 2	0.4364	0.2213	0.2975	0.4519
Day 2 Trial 3	0.8749	0.2630	0.2606	0.09451
Day 2 Trial 4	0.6627	0.3706	0.2766	0.5007
Total	0.7482	0.5109	0.3841	0.5359

Table B.2: Comparison of Theil inequality coefficient for two selected Pegasus longitudinal models.

computed. The TIC is computed for the final two models over a set of five trials that have similar initial conditions. Two trials are from day 1 and three from day 3. Model accuracy in terms of TIC is quite similar between the two models, which indicates a certain level of consistency in the identification process. Ultimately, the Day 3 Trial 2 model is preferred because of its lower error in predicting q and θ.

The identified continuous-time longitudinal model is given by Eqs. B.1-B.5.

$$\dot{\mathbf{x}} = \begin{bmatrix} -0.796 & -6.86 & -3.199 & -0.620 \\ 0.0266 & -0.744 & 0.169 & 0.664 \\ -0.0983 & 2.466 & -4.21 & -1.64 \\ 0.000732 & 0.181 & 0.603 & -0.392 \end{bmatrix} \mathbf{x} + \begin{bmatrix} -30.2 & 3.2609 \\ -1.70 & -0.114 \\ -15.4 & 0.139 \\ -1.52 & -0.0301 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix} \tag{B.1}$$

$$\begin{bmatrix} u \\ \alpha \\ q \\ \theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} -0.125 & -0.431 \\ 0.0523 & -0.0421 \\ -0.0631 & 0.0651 \\ 0.0409 & 0.0425 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix} \tag{B.2}$$

$$U_1 = 30.3 \text{ m/s} \tag{B.3}$$

$$\alpha_1 = 0.0 \text{ rad} \tag{B.4}$$

$$\theta_1 = 0.0 \text{ rad} \tag{B.5}$$

B.2 Lateral/directional model selection

One model is fit to each of the thirty-two lateral/directional trials from two days of flying. On some maneuvers, the pilot gave multiple doublet commands consecutively; each individual doublet is treated as a trial, and so is the string of consecutive doublets. This means that some data are repeated among the trials. In selecting the best models, one important consideration is the bank angle at the beginning of the maneuver. The decoupling of aircraft longitudinal and lateral/directional axis dynamics is only possible for a steady-state bank angle of zero. Therefore, only trials for which the initial bank angle has magnitude less than 5° are considered. This effectively eliminates twenty-two models. MSE is found not to differ greatly among the remaining models, so each model is evaluated in terms of the TIC against

Trial	TIC_β	TIC_p	TIC_r	TIC_ϕ
Day 1 Trial 2	0.2514	0.0985	0.2415	0.2810
Day 1 Trial 5	0.4720	0.1081	0.3888	0.2144
Day 1 Trial 7	0.4008	0.1332	0.4671	0.2874
Day 1 Trial 10	0.4209	0.1232	0.3814	0.4454
Day 1 Trial 12	0.4869	0.1310	0.3965	0.3308
Day 1 Trial 14	0.4719	0.0837	0.2043	0.4052
Day 2 Trial 2	0.3332	0.1303	0.2752	0.6342
Day 2 Trial 11	0.5627	0.2805	0.4801	0.4012
Day 2 Trial 13	0.5117	0.1942	0.4290	0.3904
Day 2 Trial 15	0.3557	0.1529	0.2733	0.3111
Day 2 Trial 17	0.6151	0.2232	0.5628	0.1623

Table B.3: TIC for Pegasus lateral/directional models with steady-state bank angle less than 5°. Each TIC value is computed by evaluating the model against the data used to generate it.

the data used to generate the model. These TIC values are tabulated in Table B.3.

Based on Table B.3, three models are selected for further evaluation: Day 1 Trial 2, Day 2 Trial 2, and Day 3 Trial 15. The TIC of these three models is then computed for every test in the set of trials in Table B.3. These results are given in Table B.4. For brevity, only the total TIC computed across all eleven data sets is shown for the three candidate models. From these results, it is clear that the model from Day 1 Trial 2 has the lowest TIC for three of the four states. This model also shows reasonably good model fitting qualitatively (see Fig. 3.7), so it is selected as the linear lateral/directional model for the Pegasus system.

The identified continuous-time lateral/directional model is given by Eqs. B.6 and B.7.

Trial	TIC_β	TIC_p	TIC_r	TIC_ϕ
Day 1 Trial 2	0.4799	0.1437	0.2646	0.5310
Day 2 Trial 3	0.5168	0.1698	0.3934	0.5124
Day 2 Trial 15	0.6033	0.3323	0.4616	0.8200

Table B.4: Total TIC for three candidate Pegasus lateral/directional models evaluated across all the sets of data considered in Table B.3.

$$\dot{\mathbf{x}} = \begin{bmatrix} -1.56 & 0.193 & -0.948 & 0.124 \\ -11.2 & -4.79 & 1.12 & -2.77 \\ 12.2 & -2.33 & 0.0539 & 0.845 \\ -0.905 & 0.607 & 0.0131 & -0.2300 \end{bmatrix} \mathbf{x} + \begin{bmatrix} -0.116 & -0.590 \\ 112 & -3.30 \\ 32.3 & 14.1 \\ 6.79 & -0.656 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{B.6}$$

$$\begin{bmatrix} \beta \\ p \\ r \\ \phi \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.0461 & -0.000266 \\ -0.625 & -0.0456 \\ -0.384 & -0.249 \\ -0.0139 & -0.0144 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{B.7}$$