Distributed Coordination and Control Experiments on a Multi-UAV Testbed

by

Ellis T. King

Bachelor of Engineering
The State University of Buffalo, 2002

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2004

Author .....
Department of Aeronautics and Astronautics
August 20, 2004

Certified by .....
Jonathan P. How
Associate Professor
Thesis Supervisor

Accepted by .....
Jaime Peraire
Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Distributed Coordination and Control Experiments on a Multi-UAV Testbed

by

Ellis T. King

Submitted to the Department of Aeronautics and Astronautics
on August 20, 2004, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

This thesis presents the development and testing of a unique testbed consisting of a fleet of eight unmanned aerial vehicles (UAVs) that was designed as a platform for evaluating coordination and control algorithms. A hierarchical configuration of task assignment, trajectory design, and low-level, waypoint following, are used in a receding horizon framework to control the UAV system. Future UAV teams will have to autonomously demonstrate cooperative behaviors in dynamic and uncertain environments, and this testbed can be used to compare various control approaches to accomplish these coordinated missions. Flight demonstrations are made utilizing real-time mixed-integer linear programming techniques, exercising the algorithms in realistic environments with real-world disturbances.Large disturbance sources, computational delay and measurementnoise all represent significant error sources that reduce the ability of UAV teams to interact in a coordinated fashion by increasing uncertainty on higher planning levels. This thesis develops a method that explicitly accounts for this uncertainty by including feedback loops on the task assignment and trajectory design algorithms to prescribe added robustness for the uncertainty at each stage. This approach takes into account low level controller saturation limits that might cause infeasibilities in the plans created at the higher levels of the planning system. Detailed and realistic simulation environments are useful for large-scale multi-vehicle simulations, particularly when logistics prevent flight testing on that scale. This thesis validates one such hardware-in-the-loop simulation environment through the comparison of models obtained from experimentally collected flight data and detailed modeling of environmental disturbances and measurement noise.The product of this thesis is a robust planning system that is tolerant of the types of uncertainty experienced by real aircraft. This robustness has been demonstrated by more than 20 successful flights on a fully automated UAV testbed.

Thesis Supervisor: Jonathan P. How
Title: Associate Professor

Acknowledgments

This thesis would not have been possible without the contributions of many individuals, all of whom invested a great deal of time and effort in making this unique and exciting project a success. I would like to thank my advisor, Prof. Jonathan How, for his guidance, insight and experience throughout every phase of the project. This research is largely an extension from the work of Yoshi Kuwata, and many other colleagues contributed volumes through their conversations and experience.

I would like to thank Col. Peter Young for sharing his knowledge and experience of small-scale aircraft and providing valuable input to the project. Special thanks to the pilots, Todd Wesley, Carl Engel, Maxim Chtangeev and Adam Woodworth and other support members of the team who assisted at various times. This project would not have progressed nearly as rapidly if it were not for the fine engineering and advice from Bill Vaglienti, Ross Hoag, Marius Niculescu and other staff at Cloud Cap Technologies. The Piccolo$^{\text{TM}}$ *system was a pleasure to work with, and worked without fail on every occasion.*

Most importantly, I would like to thank my parents and brother Jesse for their endless support. This work is for them.

This research was funded by AFOSR grant # F49620-01-1-0453 with Lt. Col. Sharon Heise as the technical monitor. The hardware testbed was funded under the DURIP grant (AFOSR Grant # F49620-02-1-0216) with Dr. Belinda King as the technical monitor.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Unmanned air vehicles (UAVs) offer fundamental advantages over conventional manned aircraft in many applications, providing a cost effective and enabling technology which can be used in situations otherwise too dangerous (or mundane) for human pilots to perform [1, 2]. The comparatively cheap cost of UAVs makes them well suited for coordinated activities, since larger numbers of vehicles can be employed to perform tasks, however this requires that the vehicles maintain the ability to make coordinated decisions with less reliance on ground operators. While the roles and capabilities of UAVs are continually growing, current UAV control structures were not designed to account for the interaction of multiple (semi-) autonomous vehicles, particularly for large teams ($N > 4$). To fully take advantage of the types of coordinated actions these teams of vehicles are capable of, it is necessary to improve on this control structure to account for the difficulties that arise with controlling teams of vehicles in real-world operating environments.

Multi-vehicle coordination is comprised of several coupled subproblems, including determining the sub-team composition, allocating resources (task assignment), and optimizing vehicle trajectories [3]. These are all computationally intensive optimization problems that require good situational awareness of the environment to achieve coordinated and cooperative behaviors. However in real-world operating environments the effect of disturbances, computation delay and communication will all limit how much information is available about the environment and other team members in

the fleet, and the high level planning algorithms need to be robust to this uncertainty to be effective [4, 5].

Numerous algorithms have recently been developed to achieve these coordinated behaviors [6, 7], but a key step towards transitioning these high-level algorithms to future missions is to successfully demonstrate that they can handle the implementation challenges on scaled vehicles operating in realistic environments. Previous work has demonstrated many aspects of these algorithms on ground testbeds [8], but some important components of aircraft planning were absent from these demonstrations. This thesis introduces a multi-UAV testbed that provides a more realistic platform for the evaluation of different UAV coordination and control strategies. In moving from ground to air vehicles, previously negligible environmental disturbance effects became apparent and had to be explicitly accounted for in the planning system.

This thesis extends the approach in [8] to compensate for large disturbance sources acting on the vehicle system, which is a key development if cooperative behavior is desired. The approach is demonstrated on an actual vehicle system with real disturbances, computation delay and measurement uncertainty, allowing the robustness of the task assignment and trajectory design algorithms to these sources of error to be quantified. Detailed studies of the disturbances typically encountered by small-scale aircraft are performed and their effect on experimentally determined flight models is validated through accurate hardware-in-the-loop simulations.

1.1 System Configuration

Figure 1-1 shows the control architecture developed for the UAV testbed, with the decomposed graph based planning, trajectory design and task assignment algorithms (described in more detail in subsequent sections) represented in block diagram form [9, 10]. Low-level control and the basic estimation tasks are run onboard the vehicles, while the *planner* outputs dynamically feasible waypoint lists and monitors the uncertain states of the vehicles and the world map. When significant changes to the situational awareness are detected in the environment estimator, the high level task

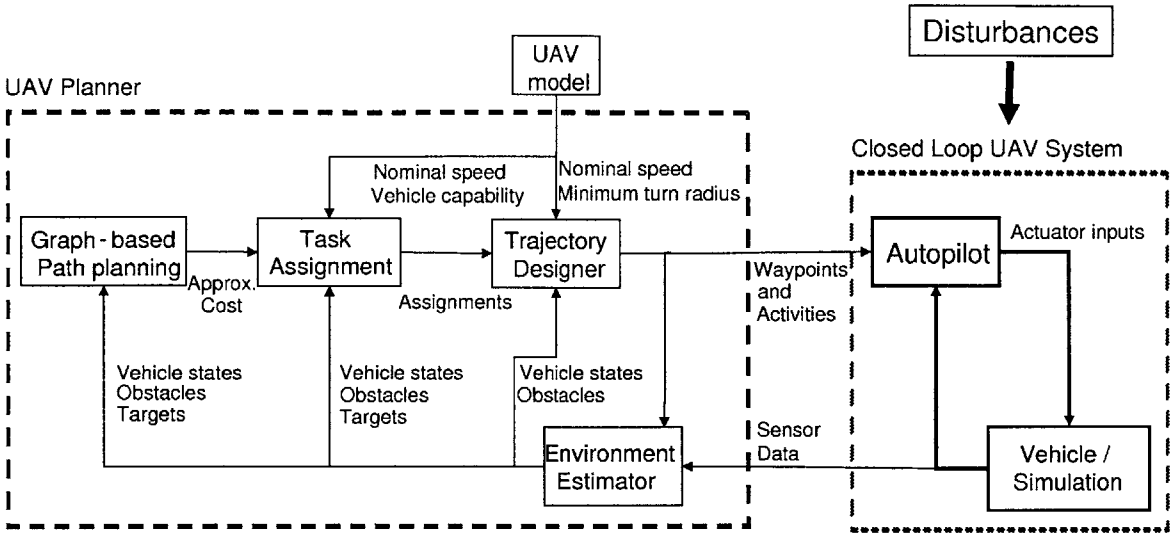


Figure 1-1: System algorithm architecture design.

assignment algorithm reassigns tasks to account for the new change.

In a real-world environment the closed loop aircraft system is impacted by wind disturbance, and these can have a significant effect on the vehicle motion. For sufficiently high disturbance levels, the low-level vehicle controllers will not have the authority to completely reject them, possibly invalidating the assumptions in the approximate UAV model used in the planning level. While the disturbance levels acting on the vehicle are not known explicitly, estimates of the current world state (including disturbance levels) are formed from all of the vehicle sensors in the fleet, and this information can be used by the planners to dynamically compensate for changes to the UAV plant model. The effect of uncertainty in these disturbance estimates can be related to the expected performance of the planner.

1.1.1 Coordination Algorithms

Mixed-integer Linear Programming (MILP) has previously been shown to provide a natural framework for posing coordination problems, because the binary integer variables can be used to encode logical constraints into the problem. Non-convex constraints such as obstacle avoidance, minimum speed constraints and task assignment can be handled using commercially available MILP solvers. CPLEX^{TM} is used

throughout this thesis to obtain solutions to the task assignment and trajectory design subproblems.

By decomposing the task assignment and trajectory design algorithms into separate but coupled sub-problems, accurate yet tractable solutions to the overall problem can be obtained [9, 10, 11]. This approach simplifies the coupling between the assignment and trajectory design problems by calculating and communicating only the key information that connects them. By utilizing approximate graph based planning [12] methods, the planner maintains a *cost map* for the each of the vehicles to the targets, which represents the minimum time path given the presence of obstacles in the environment. The cost map is then used in the trajectory design and task assignment phases to solve for the minimum time path subject to the vehicle model. Recent improvements have also led to the inclusion of incremental and robust versions of the cost map update step [12].

1.1.2 Task Assignment

The task assignment sub-problem deals with the allocation of tasks to vehicles with different capabilities [13], which is a multiple-choice multi-dimension knapsack problem (MMKP) [14], where the number of possible allocations grows rapidly as the problem size increases. *Timing constraints*, which force precedents on the waypoints that are visited during the task assignment process further increase the complexity of the problem [15]. The task assignments are achieved by the enumeration of all the tasks using straight line approximations from the cost map, and the infeasible sets are pruned from the list before passing the enumerations to the MILP solver to be optimized (*i.e.*, the *petal* method) [5, 15]. The use of MILP provides a natural language for encoding the mission objectives and constraints using both binary and continuous variables [11, 16, 9]. Other recent developments in the task assignment algorithm have led to receding horizon and robust formulations [17, 4, 5], which improve the computational tractability of the algorithm for large numbers of vehicles and address uncertainty in the assignment costs, respectively.

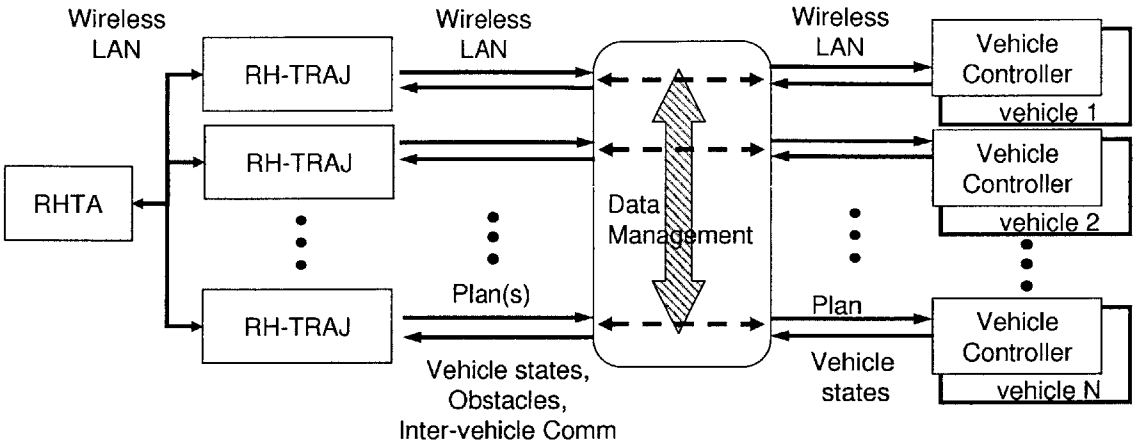
1.1.3 Trajectory Optimization

Using the discrete waypoint lists from the task assignment algorithm, detailed UAV trajectories around the obstacles can be solved using a MILP-based techniques, which allow the inclusion of non-convex constraints, such as collision and obstacle avoidance in the trajectory optimization [18, 19, 20, 21]. However the solution of complete vehicle trajectories by these methods is computationally intensive, and to make them tractable, a receding horizon (RH), or model predictive control (MPC), approach is required. In the receding horizon MILP (RH-MILP) approach of Refs. [22, 8], this is accomplished by using an approximate cost-to-go calculation to obtain good estimates of the costs associated with feasible paths around "obstacles" (e.g. buildings, no-fly-zones) in the environment, and forming short dynamically feasible plan segments for only a portion of the trajectory. After executing one or more steps of the plan the optimization is repeated with updated vehicle locations. This combination gives a good estimate of the cost-to-go and greatly reduces the computational effort required to design the complete trajectory.

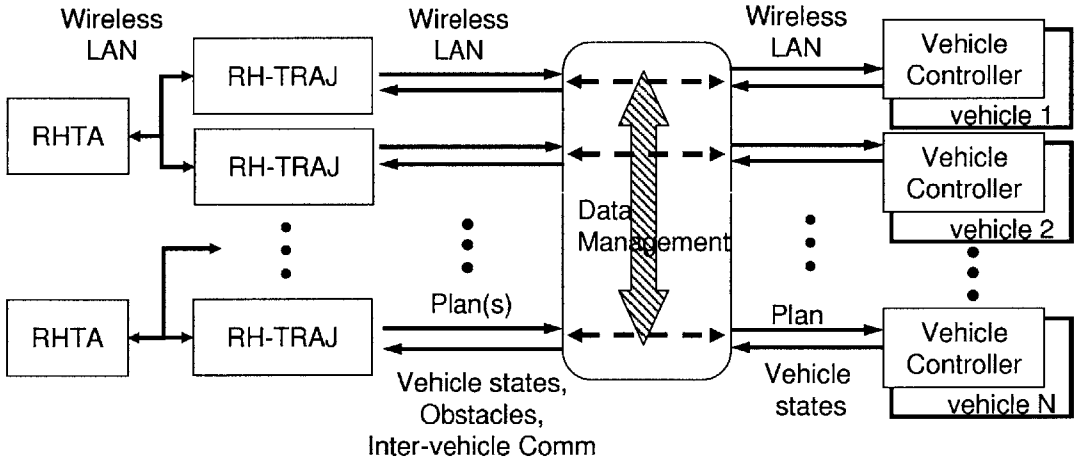
Previous work has shown the RH-MILP approach to guarantee arrival at the target in bounded time, using a simple vehicle dynamics model used in the near term and the straight line approximations for the path on the long term [8, 23, 24]. This approach works well when the vehicle dynamics model is consistent with the vehicle capabilities and no measurement noise is present, but when the state estimate is not perfect and if disturbances act to drive the dynamics away from their nominal values dynamical inconsistency in the form of "drifting" plans can be observed.

1.2 Testbed Infrastructure

The system infrastructure was setup to emulate a fully integrated fleet of UAVs, but maintain as much simplicity as possible in the vehicles themselves. All data passes through a central hub that performs data management between the planning computers and vehicles, simulating the communication delays, vehicle sensors and uncertainty in the environment. Using this approach greatly simplifies the setup of



(a) Centralized task assignment



(b) Decentralized task assignment

Figure 1-2: Testbed infrastructure showing ability to evaluate different architectures for task assignment.

the testbed, while maintaining nearly all of the functionality of a fully integrated system. For example, as shown in Figures 1-2(a) and 1-2(b), the testbed can be used to investigate the impact of communication networking issues on the coordination problem by imposing various limitations and constraints on how the planning laptops communicate (using their own wireless or Ethernet links). The testbed will be used to demonstrate the effectiveness of various control architectures on the task assignment process, as would be seen in utilizing dynamic sub-teams of various compositions.

A key feature of the setup of this testbed is that much of the complexity of the system has been kept off-board the vehicles. This allows the aircraft to be scaled

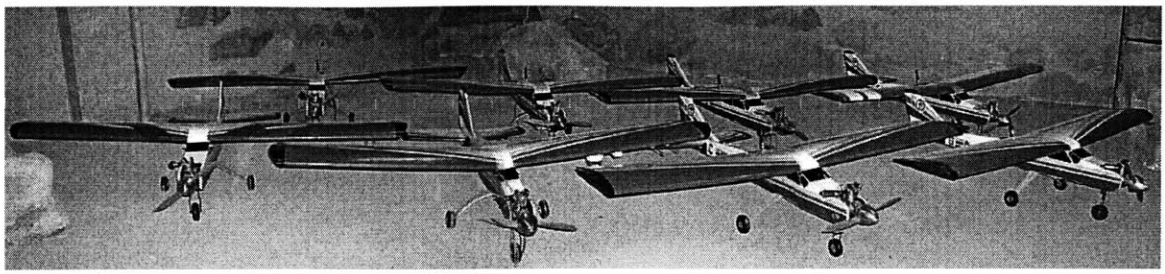


Figure 1-3: The fleet of 8 identical trainer ARF 60 aircraft used in the multi-UAV testbed at MIT.

smaller than would be necessary if additional computers, batteries and other sensors were on board, yet the performance is similar because waypoint plans and high level control commands can be uplinked to the vehicles at a rate of up to 1 HZ. Under the presumption that the low-level vehicle controllers are working well, the planning system would not need to respond faster, and in practice transmissions are made less frequently.

1.2.1 Tower Trainer ARF 60 Aircraft

In order to make successful demonstrations of multi-vehicle flights, the logistics require that the vehicles all have adequate minimum flight durations to ensure that there is sufficient time to perform the required ground operations. For a fleet of four vehicles, flight times greater than 40 minutes are needed in order to have sufficient time in the air to perform experiments. In addition, the vehicles must have sufficient wing loading capacity to carry additional weight from sensors and batteries. The vehicles selected for the testbed are commercially available Tower Trainer ARF 60 aircraft, which have easy handling characteristics and relatively large payload capacities. Only minor modifications are required to augment this class of aircraft to suit the mission requirements, which means that they can be quickly constructed and standardized across the entire fleet. In addition, maintenance and repairs are made much simpler by utilizing cheap, standardized aircraft for the fleet, and the logistics of flight tests are made much simpler by having vehicles with similar handling characteristics. The

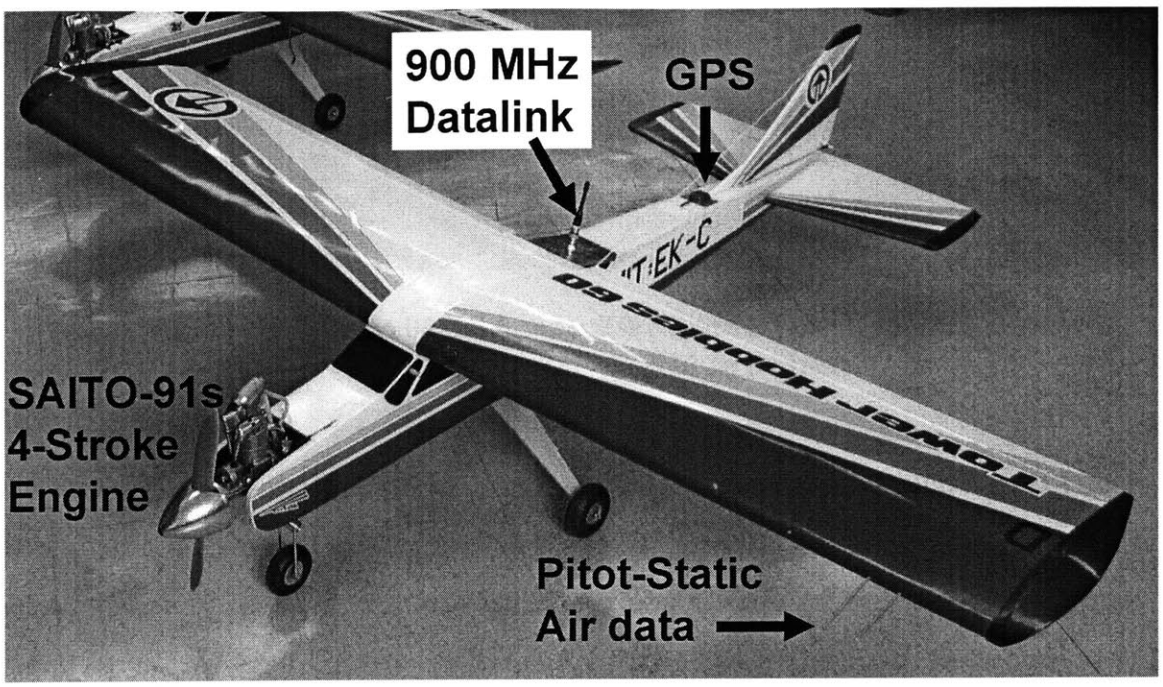
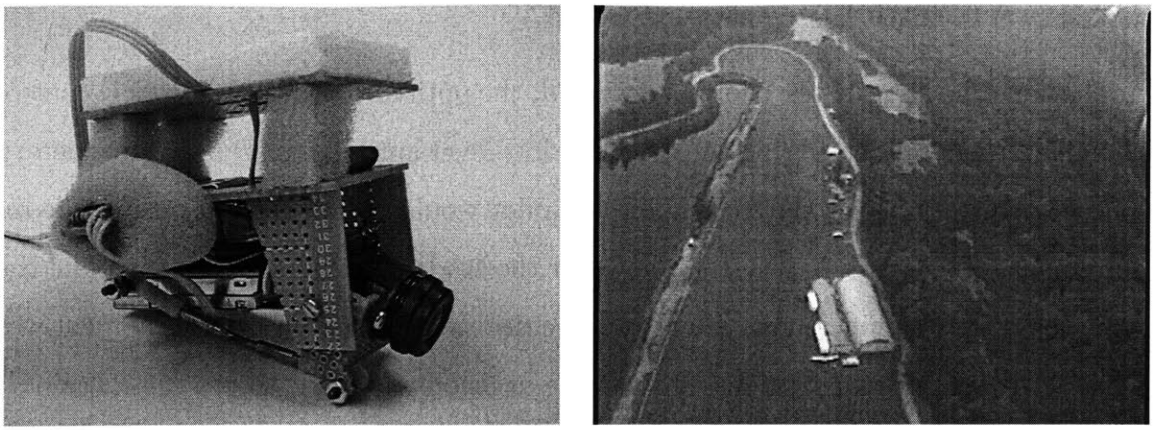


Figure 1-4: The trainer ARF 60 aircraft used in the multi-UAV testbed at MIT.

fleet of 8 trainer ARF aircraft are shown together in Figure 1-3.

The ARF 60 aircraft is shown in Figure 1-4, with a table of important aircraft parameters in Table 1.1. The large wing area of the aircraft, combined with the four-stroke, Saito-91s (91 ccs) engine provides more than 3 lbs. of payload capacity, which is sufficient for the avionics, batteries, and additional sensors. An external fuel tank more than doubles the fuel capacity of the aircraft, which allows for extended flights of greater than 50 minutes with moderate throttle settings. The integration of GPS and air data sensors are minor modifications, providing the necessary measurements for autopilot control.

The tower trainer aircraft are well suited for autopilot control because of their stable design for pilot training purposes. The stable configuration causes them to be less susceptible to upsets caused by turbulence, and the aircraft trim states are easily determined. However because the aircraft is so stable, maneuverability is sacrificed. The reduction in performance, combined with minimum flight speeds of $\approx$20 m/s, dictate that slightly larger test areas be utilized to perform effective demonstrations,



(a) (b)

Figure 1-5: Overhead video of the local flying field at Crow Island in Maynard, MA taken with the onboard video system.

Table 1.1: Trainer 60 ARF aircraft parameters

Measurement	Value	Units (SI)
Wing Span	1.707	m
Wing Area	0.5200	m^2
Chord Length	0.305	m
Wing Incidence	1	deg
Wing Dihedral	5	deg
Gross Mass	5.267	kg
Empty Mass	4.798	kg

however this tradeoff makes sense for the proof-of-concept missions attempted in this phase of the project.

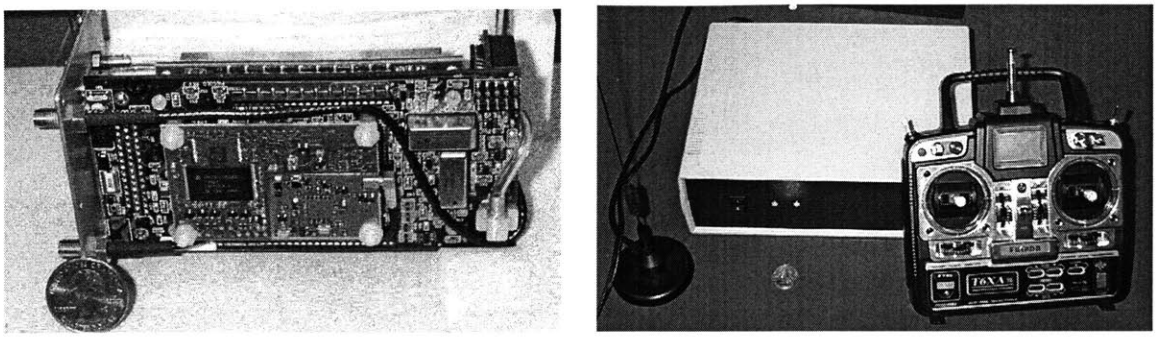
Additional video and magnetometer sensors have also been integrated onboard the aircraft to provide added real-time measurements about the environment. The pan/tilt video camera (shown in Figure 1-5(a)) transmits video over the 2.4GHz band to the ground-station where it can be processed to track ground objects. Figure 1-5(b) shows a captured image from the video system in flight. The onboard magnetometer provides true heading estimates of the aircraft in flight, which can also be used to provide estimates of the ambient winds acting on the vehicle.

1.2.2 Autopilot

At the time the aircraft testbed was designed, the options for low-level vehicle control included either purchasing or constructing low-level autopilots. Due to proprietary source code restrictions, purchasing an autopilot would be less flexible to suit the requirements of the planning system, however the tradeoff in the time to complete the same tasks were significant[1]. As a result, the decision to purchase the equipment was made. After considering various options, the decision was made to use the Piccolo^{TM} autopilot from Cloud Cap Technology (Figure 1-6(a)). This autopilot is used onboard the aircraft to perform the autonomous vehicle stabilization and waypoint navigation. One watt transmission of data over the 912 MHz datalink permits the vehicles to navigate up to three miles from the ground station (Figure 1-6(b)) and this link can also be used to continuously upload new flight plans and other control commands from ground based planning algorithms. Real-time aircraft telemetry is utilized in ground based processing, including GPS position and velocity ($\pm$2m, $\pm$0.1m/s respectively) air data, attitude estimates, static wind estimates and other control data. The attitude solutions are real time estimates obtained using measurements from Cloud Cap's Crista IMU, which provide high bandwidth, angle-rates and accelerations ($\pm$300°/s at 16 bits and $\pm$10g at 16 bits respectively). This information is obtained at a rate of 1 HZ through the robust 912 MHz link providing sufficient bandwidth for high level commands.

Benefits of purchasing a commercially available system are the significant time and effort saved in developing the required infrastructure to tune and test the system. The well-designed and user-configurable Cloud Cap architecture comes complete with a high fidelity hardware-in-the-loop (HWIL) simulation mode that enables real-time testing of the system on the ground before flight tests are performed. While this is primarily meant to simulate the system for controller tuning purposes, it is also interfaced with the planning system so that multi-vehicle simulations can be executed with high levels of accuracy on the ground.

[1]Similar projects at MIT had taken 4 years to develop their own autopilot. Recent projects at Stanford University have taken similar periods of time.



(a) Commercially available Piccolo^{TM}Autopilot from Cloud Cap Technologies.

(b) Cloud Cap groundstation with 912 MHz radio antenna and Pilot Console.

Figure 1-6: The commercially available Cloud Cap autopilot system.

Figure 1-7 displays the setup of the system with the avionics performing HWIL simulations and the planning system in the loop. The groundstation communicates with each of the avionics through the 912 MHz data link, and the telemetry data from the vehicles is passed on to the planning system through integrated TCP/IP protocols, exactly as would be performed during an actual flight test. An integrated GUI is connected to the system to allow user feedback to the planning system and visualize the state of the mission in real-time as shown in Figure 1-8, using FlightGearV0.9.2 operated in network connection mode.

While performing HWIL simulations, each of the avionics is connected through a USB-CAN adapter to a simulator CPU which stimulates the avionics sensors with simulated measurements. The HWIL simulator application allows the specification of a detailed aircraft model that is built up from aircraft geometry and inertia measurements or alternatively specified through calibrated wind tunnel data. By specifying the appropriate aircraft parameters and selecting suitable models for sensors onboard, actuator delays and turbulence parameters, an accurate simulation of flying characteristics can be built up. This is an essential part of the process in validating control settings and testing the performance of the system before attempting an actual flight. In Chapter 2, the HWIL simulation environment is described in more detail and the flight parameters for the ARF 60 aircraft are explicitly determined.

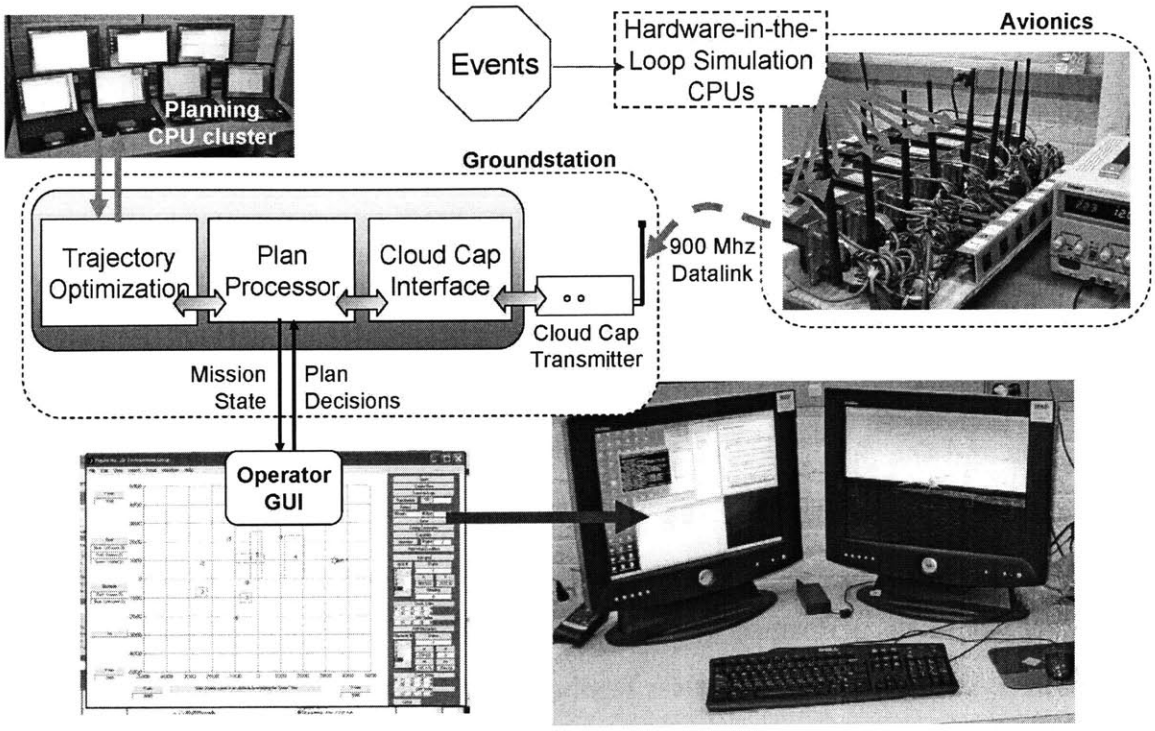


Figure 1-7: Hardware-in-the-loop configuration for the UAV testbed allowing simultaneous simulation of 8 aircraft with the integrated planning system connected through the 912 MHz data link, exactly as would be performed in flight.

1.3 Thesis Outline

Chapter 2 uses identification and analytical methods to find approximate models for the ARF 60 aircraft and verifies that the hardware-in-the-loop simulator reproduces motion consistent with these models. The Cloud Cap autopilot is tuned for the testbed aircraft and closed loop responses are determined to provide more accurate models to be used on the planning level. Chapter 3 introduces the notion of disturbance estimation into the planning level and accounts for uncertainty in the bounded errors in these estimates. Part of the strategy for rejecting these disturbance estimation errors includes implementing *timing control* to vary the reference speed of the aircraft to reject relative timing errors on the vehicle level. Chapter 4 implements

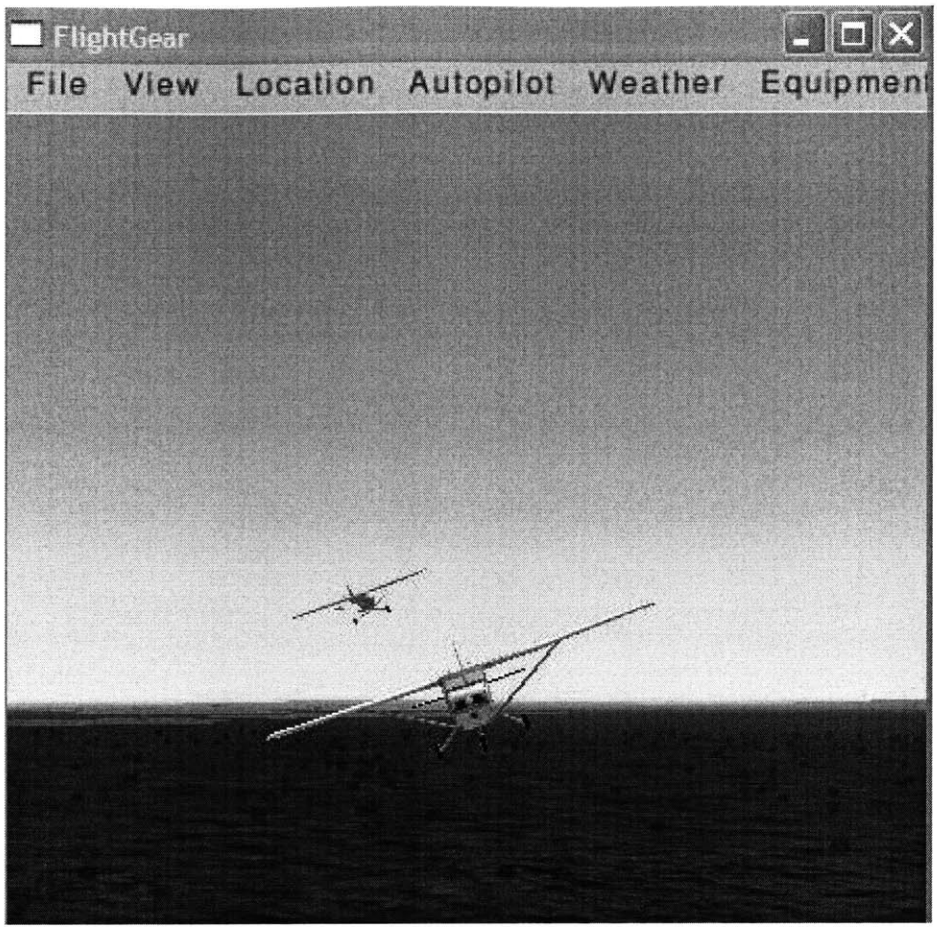


Figure 1-8: FlightgearV0.9.2 visualization support for the HWIL simulation environment.

position and heading feedback on the planning level to account for difference between the planned MILP trajectory and the one that is actually flown by the aircraft. This includes dealing with the effect of initial condition uncertainty due to measurement errors and computation delay, as well as the required reductions in authority on the planning level.

Chapter 2

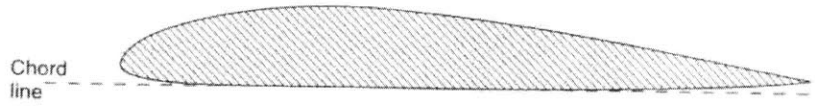
Hardware In the Loop Modeling and Simulation

The hardware-in-the-loop (HWIL) simulation is only useful if it accurately portrays the vehicle dynamics and if the behaviors observed during flight tests can be replicated on the ground. This chapter focuses on identifying some of the dynamical modes of flight for the 60 ARF Trainer aircraft, and verifying that the HWIL simulations reflect the dynamics expected from the aircraft being employed. Reduced order models for 4 of the 5 dynamical modes are determined for the trainer ARF 60 aircraft using identification techniques on experimental flight data and analytical predictions based on aircraft geometry and aerodynamic data. Section 2.1 describes the simulation settings used to create the hardware-in-the-loop (HWIL) simulations, and Section 2.2 details the procedures used to create models of the aircraft dynamics from data collected during flight tests and hardware-in-the-loop simulations. In Section 2.3, the Cloud Cap autopilot is tuned for the trainer ARF 60 aircraft and the closed loop response for several of the modes is measured using the HWIL simulator.

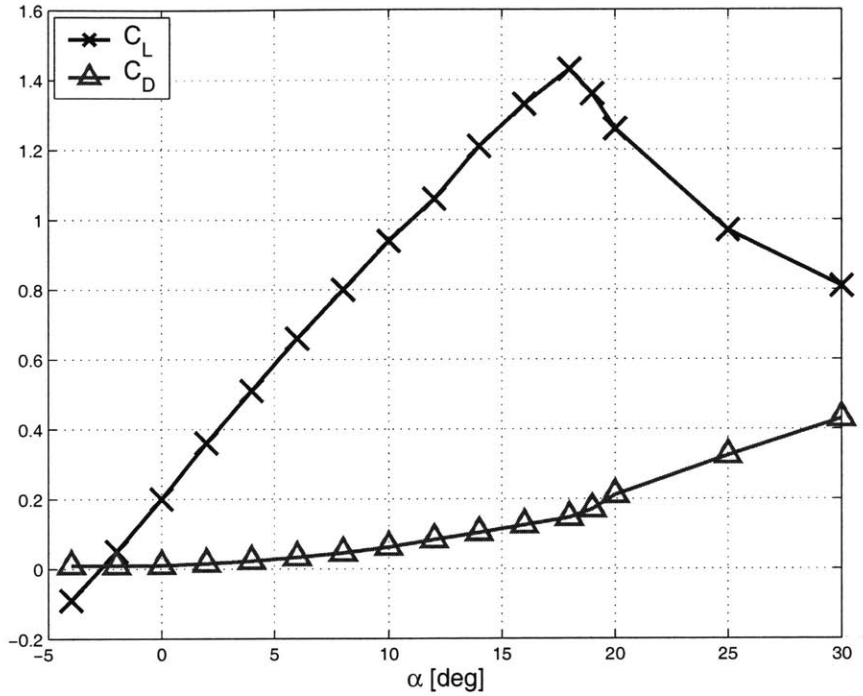
2.1 Hardware in the loop simulations

2.1.1 Aircraft simulation Model Parameters

Aerodynamic, inertial and engine calibration information is provided to the Cloud Cap HWIL simulation application to model the aircraft being flown. For simply



(a) The Clark YH airfoil geometry.



(b) The Clark YH airfoil lift and drag curves.

Figure 2-1: The Clark YH airfoil closely resembles the airfoil used on the trainer ARF 60 aircraft and is used to model wing aerodynamics.

configured aircraft such as the tower trainer 60 ARF used in the testbed, many of the performance characteristics can be obtained using the geometry of the aircraft, such as the data found in Table 2.1. Detailed descriptions of the surface geometry, wing lift curves, and engine performance curves enable simulations of the aircraft under realistic flight conditions, providing the input parameters are configured accurately. For example, the Clark YH airfoil closely resembles the trainer ARF 60 airfoil and is used to describe the aerodynamic properties of the main wing on the aircraft [25]. Some of the data is shown in Figure 2-1. A more detailed description of the simulator input files is given in Ref. [26].

Table 2.1: Trainer 60 ARF measurements, experimentally determined inertias as shown in Subsection 2.1.1. Symbolic notation is borrowed from Ref. [27]

Measurement	Value	Units (SI)	Symbol
Wing Span	1.707	m	b
Wing Area	0.5200	m^2	$\mathcal{S}$
Chord Length	0.305	m	$\bar{c}$
Wing Incidence	1	deg	
Wing Dihedral	5	deg	Γ
Wing Sweep	0.0	deg	Λ
Tail Area	0.0879	m^2	$\mathcal{S}_t$
Tail Span	0.606	m	b_t
Tail Offset X	1.14	m	l_t
Tail Sweep	9	deg	Λ_t
Fin Area	0.0324	m^2	$\mathcal{S}_f$
Fin Span	0.216	m	b_f
Fin Offset X	1.143	m	l_f
Fin Offset Z	0.120	m	h_f
Fin Sweep	53	deg	Λ_f
Fin Volume Ratio	0.0189	-	$\bar{V}_f$
Fuselage CX Area	0.130	m^2	$\mathcal{S}_b$
Fuselage Length	1.270	m	l_b
Gross Mass	5.267	kg	m
Empty Mass	4.798	kg	m_e
Roll Inertia*	0.31	$kg \cdot m^2$	I_{xx}
Pitch Inertia*	0.46	$kg \cdot m^2$	I_{yy}
Yaw Inertia*	0.63	$kg \cdot m^2$	I_{zz}

Aircraft Inertia Experiment

The aircraft pitch, roll, and yaw inertias are important parameters for the accurate HWIL simulation of the aircraft dynamics. Fortunately, due to the small scale of the aircraft, experimental measurements can be easily made for each axis of the aircraft. The experimental setup for the roll axis is shown in Figure 2-2. From the aircraft free body diagram, the tension in each cable, T, is

$$2T = mg \tag{2.1}$$

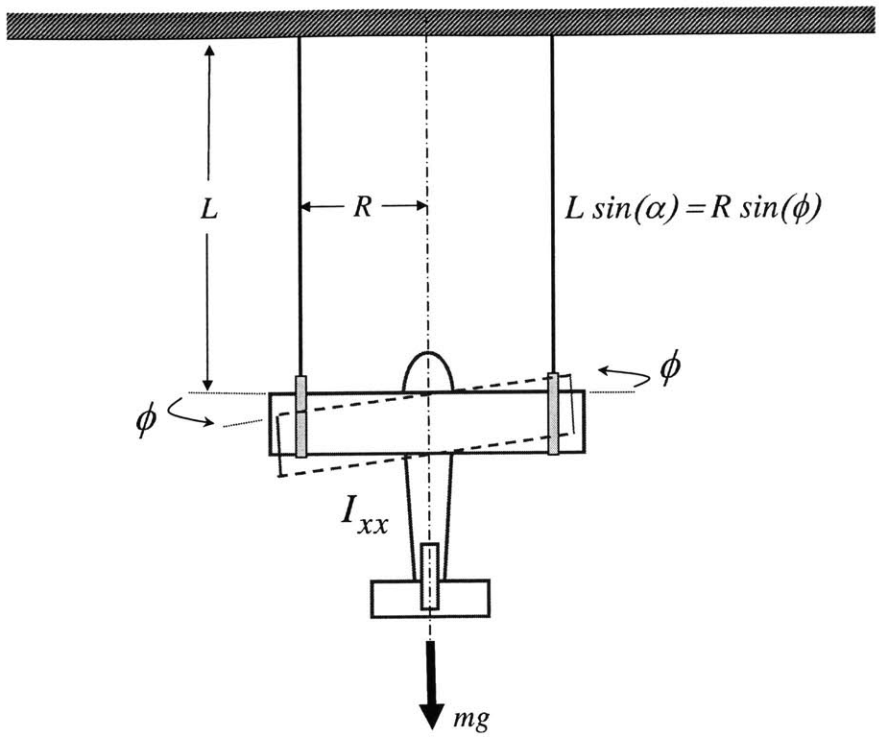


Figure 2-2: Torsional pendulum experimental setup to determine roll axis inertia, I_{xx}, for the trainer aircraft. The period of oscillation of a roll angle perturbation, ϕ, is measured to parameterize the aircraft inertia. The angle α is the small angle deviation of the supporting cables from the vertical position. This experiment was also repeated for the pitch and yaw axes to determine I_{yy} and I_{zz} respectively.

For rotational perturbations applied to the airframe, the product of interior angles and distances must be constant

$$R\phi = L\alpha \tag{2.2}$$

where ϕ is the aircraft roll angle perturbation and α is the small angle deviation of the supporting cables from the vertical position. The differential equation describing the motion of the torsional pendulum is governed by a torsional inertia term and the restoring moment due to tension forces

$$I_{xx}\ddot{\phi} + 2TR\sin\alpha = 0 \tag{2.3}$$

Table 2.2: Experimentally determined aircraft inertias [kg-m^2]

Aircraft No.	Roll Axis I_{xx}	Pitch Axis I_{yy}	Yaw Axis I_{zz}
1	0.28	0.46	0.65
2	0.30	0.44	0.61
3	0.33	0.47	0.63
Mean	**0.30**	**0.42**	**0.63**
Std Dev.	0.029	0.011	0.021

Using the small angle approximation for α since $L \gg R$, and substituting known values from Eqs. 2.1 and 2.2, Eq. 2.3 reduces to

$$I_{xx}\ddot{\phi} + \frac{mgR^2}{L}\phi = 0 \tag{2.4}$$

which is characterized by the undamped natural frequency, ω, and period of oscillation, T_p

$$\omega = \sqrt{\frac{mgR^2}{I_{xx}L}} \tag{2.5}$$

$$T_p = \frac{2\pi}{\omega} \tag{2.6}$$

By finding averaged values for the period of oscillation, T_p, in each of the pitch, roll, and yaw axes, the inertia about each axis can be approximated. This experiment neglects aerodynamic and other forms of damping as well as the cross-axis inertias (e.g., I_{xz}, I_{yz}). The experimental results are summarized for each of the axes and three different aircraft in the same configuration in Table 2.2, showing agreement between different vehicles used in the tests. The largest variation was found in the roll axis due to the difficulty of mounting the aircraft through the center of gravity, which is essential in this experimental setup.

2.1.2 Actuator Models

The servos used on the aircraft have saturation limits, limited bandwidth, and limited slew rates which are captured in the actuation models of the Cloud Cap hardware-

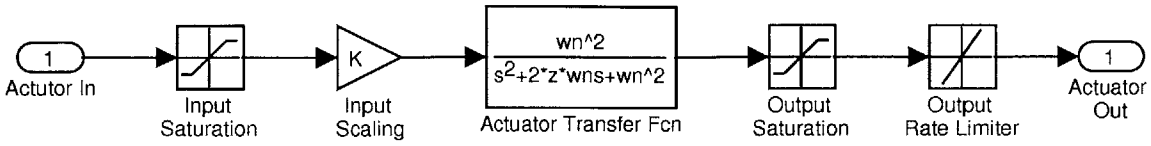


Figure 2-3: Actuator models used the Cloud Cap Hardware-in-the-loop simulations

in-the-loop simulator. As shown in Ref. [26], the actuator transfer function, $G_{act}(s)$, is given by specifying the bandwidth limit, B_W

$$G_{act}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{2.7}$$

$$\omega_n = 2\pi B_W \tag{2.8}$$

$$\zeta = \zeta_c = 0.707 \tag{2.9}$$

where the damping ratio, ζ, is selected at the critical value to set the actuator bandwidth equal to the natural frequency ($\omega_b = \omega_n$) . The aileron, elevator, and rudder channels all respond with approximately the same characteristics ($B_W = 10$ Hz), but the throttle is modeled with less dynamic range ($B_W = 2$ Hz) as the engine RPM requires added time to ramp up to produce thrust. The input/output saturation and slew rate limits are determined as per manufacturer specifications ($\pm 60°$, 2 Hz respectively), and applied as shown in Figure 2-3.

2.1.3 Sensor Noises

The Cloud Cap hardware-in-the-loop simulator includes detailed sensor models based on information from the manufacturer to corrupt the simulation measurements. For the purposes of simulation, noises on the pressure, rate gyros and accelerometers onboard the aircraft are modeled using band-limited white noise and specified drift rates [26]. Although the same noise and drift models could be applied to GPS position and velocity measurements, this information is typically assumed to be perfect in the HWIL tests. The values used to parameterize the Piccolo^{TM} pressure sensors, the Crista^{TM} IMU angle-rate sensors, and the accelerometers are shown in Table 2.3.

Table 2.3: Crista IMU HWIL Sensor Noise Models

Sensor [unit]	PDynamic [Pa]	PStatic [Pa]	Gyro [deg/s]	Accel. [m/s/s]
Resolution [unit]	3.906	2.00	1.6E-4	6.0E-3
Min [unit]	-300	0.0	-5.20	-100.0
Max [unit]	4000	110,000	5.20	100.0
Noise Gain	20.0	20.0	0.10	0.0
Butterworth Order	2	2	2	2
BW Cutoff Freq. [Hz]	11.0	11.0	20.0	20.0
Drift Rate [unit/s]	0.05	1.0	1.5E-4	2.0E-3
Max Drift value [unit]	15.0	100.0	0.01	0.20
Drift Update Rate [s]	5.0	5.0	1.0	1.0

2.1.4 Dryden Turbulence

Stochastic turbulence disturbances are required for accurate HWIL simulations, as real world experiments are characterized by unpredictable winds acting on the vehicle. The Dryden turbulence model is one of the accepted methods for including turbulence in aircraft simulations [28]. By applying shaped noise with known spectral properties as velocity and angle rate perturbations to the body axes of the vehicle, the effect of turbulence is captured during discrete time simulations. The noise spectrum for each of the perturbations is predominantly described by a turbulence *scale length* parameter, L, the airspeed reference velocity, V_o, and the *turbulence intensity*, σ. The selection of these parameters allows for the turbulence to be modeled according to the prevailing wind conditions.

The spectral frequency content for generalized aircraft turbulence have been well studied [29, 28] and are given for each of the aircraft body axes:

$$S_{ug}(\omega) = \frac{2\sigma_u^2 L_u}{\pi V_o} \cdot \frac{1}{1+(\frac{L_u}{V_o}\omega)^2} \tag{2.10}$$

$$S_{vg}(\omega) = \frac{\sigma_v^2 L_v}{\pi V_o} \cdot \frac{1+3(\frac{L_v}{V_o}\omega)^2}{\left(1+(\frac{L_v}{V_o}\omega)^2\right)^2} \tag{2.11}$$

$$S_{wg}(\omega) = \frac{\sigma_w^2 L_w}{\pi V_o} \cdot \frac{1+3(\frac{L_w}{V_o}\omega)^2}{\left(1+(\frac{L_w}{V_o}\omega)^2\right)^2} \tag{2.12}$$

$$S_{pg}(\omega) = \frac{\sigma_w^2}{L_w V_o} \cdot \frac{0.8\left(\frac{\pi L_w}{4b}\right)^{1/3}}{1+\left(\frac{4b}{\pi V_o}\omega\right)^2} \tag{2.13}$$

$$S_{qg}(\omega) = \frac{\left(\frac{\omega}{V_o}\right)^2}{1+\left(\frac{4b}{\pi V_o}\omega\right)^2} \cdot S_{wg}(\omega) \tag{2.14}$$

$$S_{rg}(\omega) = \frac{\left(\frac{\omega}{V_o}\right)^2}{1+\left(\frac{3b}{\pi V_o}\omega\right)^2} \cdot S_{vg}(\omega) \tag{2.15}$$

where ω is the spectral frequency of the turbulence and the aircraft wingspan, b, is used as a parameter in the angle rate filters to scale the effect of rotation on the main lifting surface. The subscripts u, v, w and p, q, r refer to the familiar body frame aircraft wind velocities and angle rates, respectively, thereby allowing independent classification of the turbulence in each axis. These spectral shaping functions are used to form shaping filters to give the body axis noise transfer functions [30]

$$H_{ug}(s) = \sigma_u\sqrt{2\frac{L_u}{\pi V_o}} \cdot \frac{1}{1+\frac{L_u}{V_o}s} \tag{2.16}$$

$$H_{vg}(s) = \sigma_v\sqrt{\frac{L_v}{\pi V_o}} \cdot \frac{1+\sqrt{3}\frac{L_v}{V_o}s}{\left(1+\frac{L_v}{V_o}s\right)^2} \tag{2.17}$$

$$H_{wg}(s) = \sigma_w\sqrt{\frac{L_w}{\pi V_o}} \cdot \frac{1+\sqrt{3}\frac{L_w}{V_o}s}{\left(1+\frac{L_w}{V_o}s\right)^2} \tag{2.18}$$

$$H_{pg}(s) = \sigma_w\sqrt{\frac{0.8}{V_o}} \frac{\left(\frac{\pi}{4b}\right)^{1/6}}{L_w^{1/3}\left(1+\left(\frac{4b}{\pi V_o}s\right)\right)} \tag{2.19}$$

$$H_{qg}(s) = \frac{\frac{s}{V_o}}{1+\left(\frac{4b}{\pi V_o}\right)s} \cdot H_{wg}(s) \tag{2.20}$$

$$H_{rg}(s) = \frac{\frac{s}{V_o}}{1+\left(\frac{3b}{\pi V_o}\right)s} \cdot H_{vg}(s) \tag{2.21}$$

The block diagram for the full 6 DOF Dryden turbulence model is shown in Figure 2-4. Note the cross axis couplings of the angle rate filters q_g and r_g.

Example turbulence perturbations values are plotted in Figure 2-5 as a function of the scale lengths and intensities for each of the body axes. The same 4×1 white noise input was used for each trial set. Larger scale lengths, L, increase the time constant of

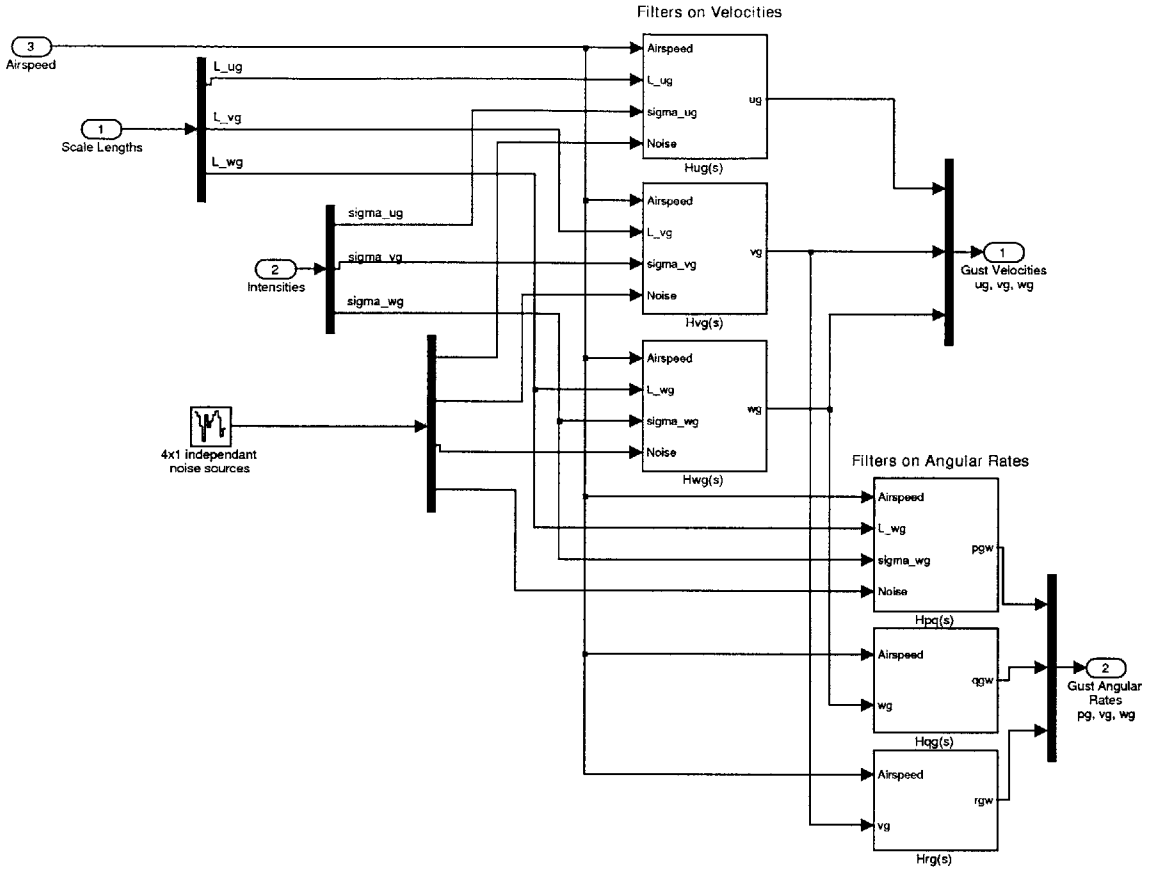
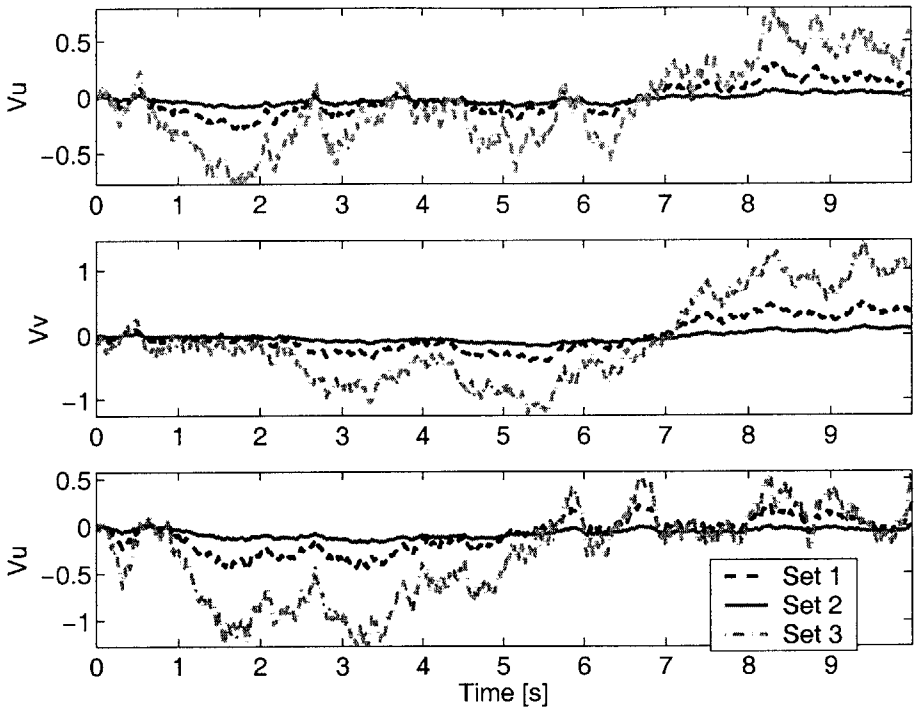


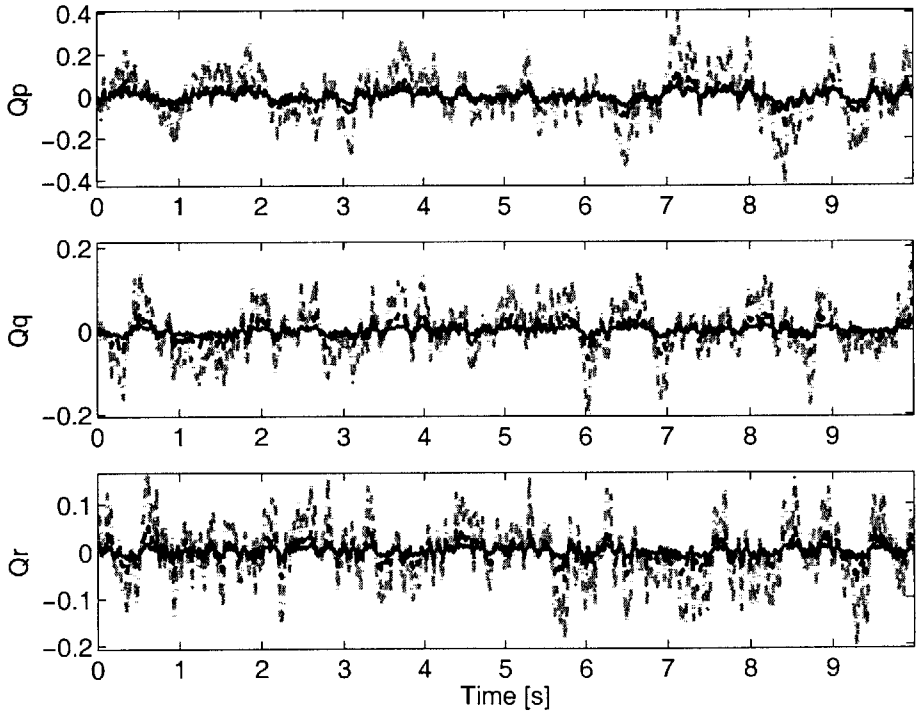
Figure 2-4: Block diagram for the 6 DOF Dryden Turbulence model. The velocity perturbations u_g, v_g, w_g are independent outputs of the filtered values of the turbulence scale lengths, L, intensity values, σ and the white noise input sources. The principle axis coupling of the aircraft is taken into account through the inputs to the angle rate perturbation filters.

the turbulence seen for a given airspeed, while larger σ values increase the deviation about zero. L and σ typically vary with altitude in the lower atmosphere as ground effects become more prominent, but for HWIL simulations they are usually fixed.

The frequency response of the Dryden filters are shown for the same three cases in Figure 2-6. The filter cutoff frequency is determined by the ratio of the scale length to airspeed, and this effectively produces lower bandwidth filters for larger scale lengths. The scale length parameter is chosen according to one of several specifications, all of which take into account the variation of L with altitude. The military reference **MIL-F-8785C** provides one such model of the scale length at low altitudes, h, which



(a) Velocity perturbations



(b) Angle rate perturbations

Figure 2-5: The output of the Dryden velocity and angle rate filters for different selections of the intensity and scale lengths. Set 1: $L = 150, \sigma = 0.5$, Set 2: $L = 1500$, $\sigma = 0.5$, Set 3: $L = 150$, $\sigma = 1.5$

is valid up to 1000 feet [29].

$$L_w = h \tag{2.22}$$

$$L_u = L_v = \frac{h}{(0.177 + 0.000823h)^{1.2}} \tag{2.23}$$

The turbulence intensity is a gain factor that scales the magnitude plots in Figure 2-6 to values appropriate for different wind levels (*i.e.,* light, moderate, severe). The intensity level has been defined for low altitude flight according to **MIL-F-8785C** as

$$\sigma_w = 0.1W_{20} \tag{2.24}$$

$$\sigma_u = \sigma_v = \frac{\sigma_w}{(0.177 + 0.000823h)^{0.4}} \tag{2.25}$$

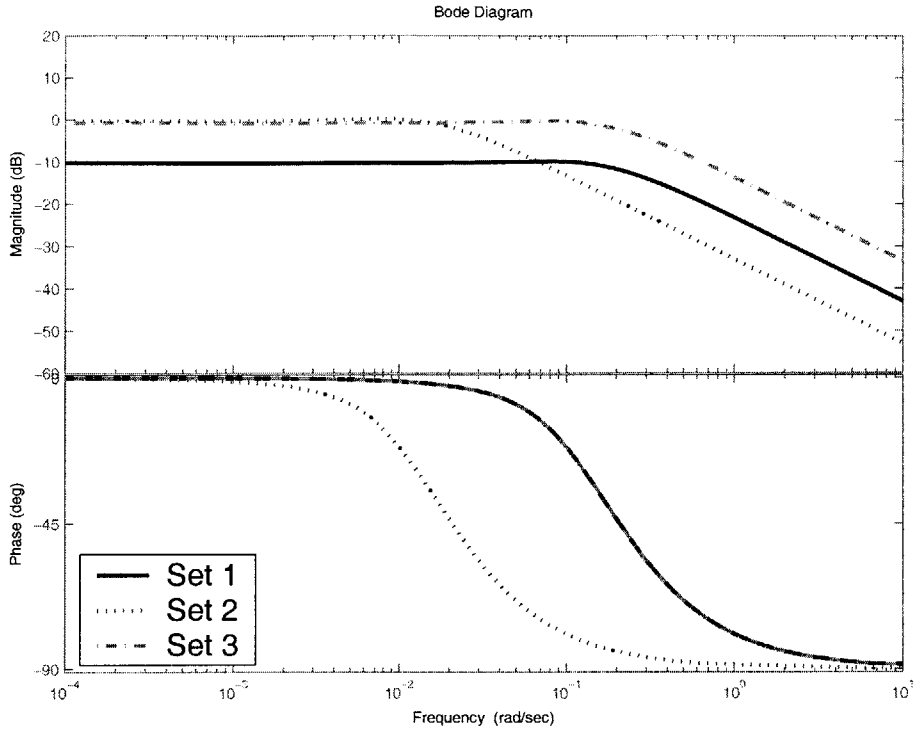
where W_{20} is the wind speed as measured at 20 ft in altitude. According to **MIL-F-8785C**, $W_{20} < 15$ knots is classified as "light" turbulence, $W_{20} \approx 30$ knots is "moderate", and $W_{20} > 45$ knots is "heavy". Other military specifications such as **MIL-HDBK-1797** exist for the low altitude cases [29], and different types of models are used for other regions of the atmosphere. For the purposes of the UAV application, the low altitude models are sufficient.

The utility of the Dryden turbulence model is that it allows the expected turbulence levels to be described for an aircraft flying at a given reference speed for more realistic HWIL simulations. Turbulence is applied to the vehicle body axes consistent with the known parameterized values for scale length and intensity, which effectively defines the appropriate filters with cutoff frequencies and magnitudes needed for simulation. Note that in addition to turbulence, wind is also usually modeled with a static component, $\overline{W}$, that represents a prevailing magnitude and direction in an inertial axis. Together these define an arbitrary three-axis wind vector

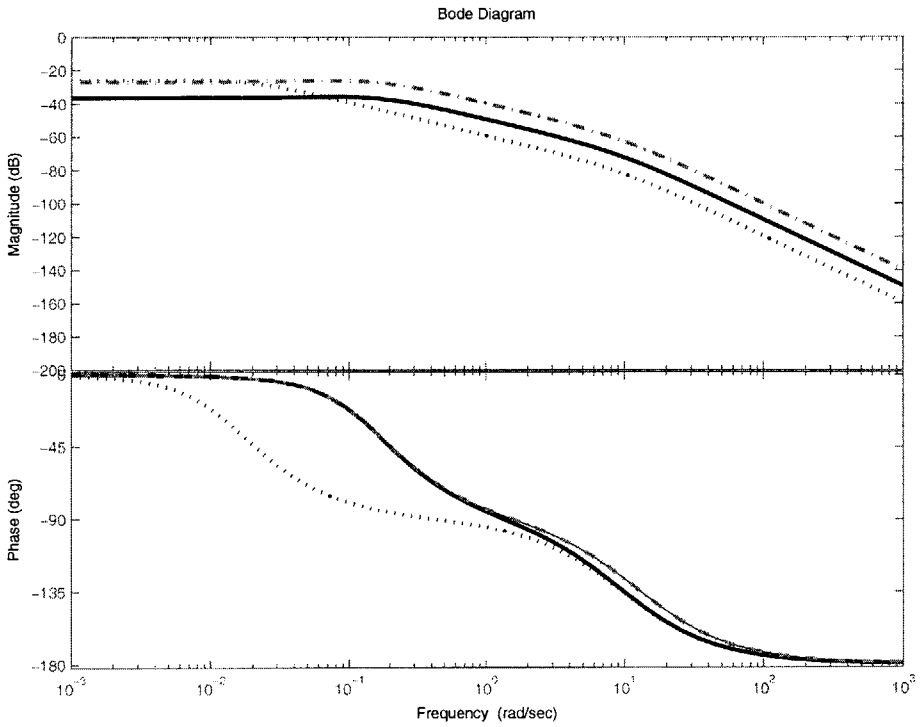
$$W = \overline{W} + \delta W^I \tag{2.26}$$

where δW^I is the effective Dryden wind turbulence in each axis after being rotated through the appropriate body to inertial transformation *direction cosine matrix.*

For small scale aircraft, the static wind component is usually a gross disturbance relative to the aircraft airspeed, and it can have a large effect on high level planning



(a) Velocity filter for w_g



(b) Angle rate filter for q_g

Figure 2-6: The Bode plots of the Dryden velocity and angle rate filters, given white noise inputs to $H_{vg}(s)$ in (a) and $H_{qg}(s) \cdot H_{wg}(s)$ in (b). Various selections of the intensity and scale lengths are shown in different sets. Set 1: $L = 150$,$\sigma = 0.5$, Set 2: $L = 1500$, $\sigma = 0.5$, Set 3: $L = 150$, $\sigma = 1.5$

algorithms. The effect of this type of disturbance on the planning system and aircraft dynamics is discussed in more detail in Chapters 4 and 5.

2.2 Open Loop Aircraft Modeling

2.2.1 Longitudinal Dynamics

A common model for the longitudinal motion of the aircraft is [27, 31]

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} x_u & x_w & x_q & x_\theta \\ z_u & z_w & z_q & z_\theta \\ m_u & m_w & m_q & m_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} x_{\delta_e} \\ z_{\delta_e} \\ m_{\delta_e} \\ 0 \end{bmatrix} \delta_e \tag{2.27}$$

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u} \tag{2.28}$$

where the state variables $\boldsymbol{x} = [u\ w\ q\ \theta]^T$ refer to the longitudinal velocities, u and w, the pitch rate, q and the angle of inclination, θ. The elements of the A matrix in Eq. 2.27 represent the *concise form* aerodynamic stability derivatives referring to the airplane body axis. Tables of values relating the concise form derivatives to the dimensionless or dimensional derivatives are available in numerous sources [27, 32]. The control input $\boldsymbol{u} = \delta_e$ is the elevator defection angle with the engine thrust fixed and is input to the dynamics through the aerodynamic control derivative matrix B.

The Longitudinal Dynamics in Eq. 2.27 are typically resolved into two distinct *phugoid* and *short period* modes, which represent dynamics of the aircraft on different timescales. The short period is characterized by high frequency pitch rate oscillations and can have high or low damping, depending on the dynamic stability of the aircraft. In contrast, the phugoid mode is characterized by lightly damped, low frequency oscillations in altitude and airspeed with pitch angle rates, q, remaining small.

Short Period Mode

A simple approximation for the short period mode of the aircraft can be obtained by assuming the speed of the aircraft is constant over the timescale of the short period dynamics ($\dot{u} = 0$), that the aircraft is initially in steady level flight and that the

derivatives refer to a wind-axis system ($\theta = \alpha = 0$). The equations of motion then reduce to

$$\begin{bmatrix} \dot{w} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} z_w & z_q \\ m_w & m_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} z_{\delta_e} \\ m_{\delta_e} \end{bmatrix} \delta_e \tag{2.29}$$

Following further approximations shown in [27] which make assumptions about the relative size of the m_q, z_q and z_w derivatives, the transfer functions for the two short term equations describing the response to elevator are:

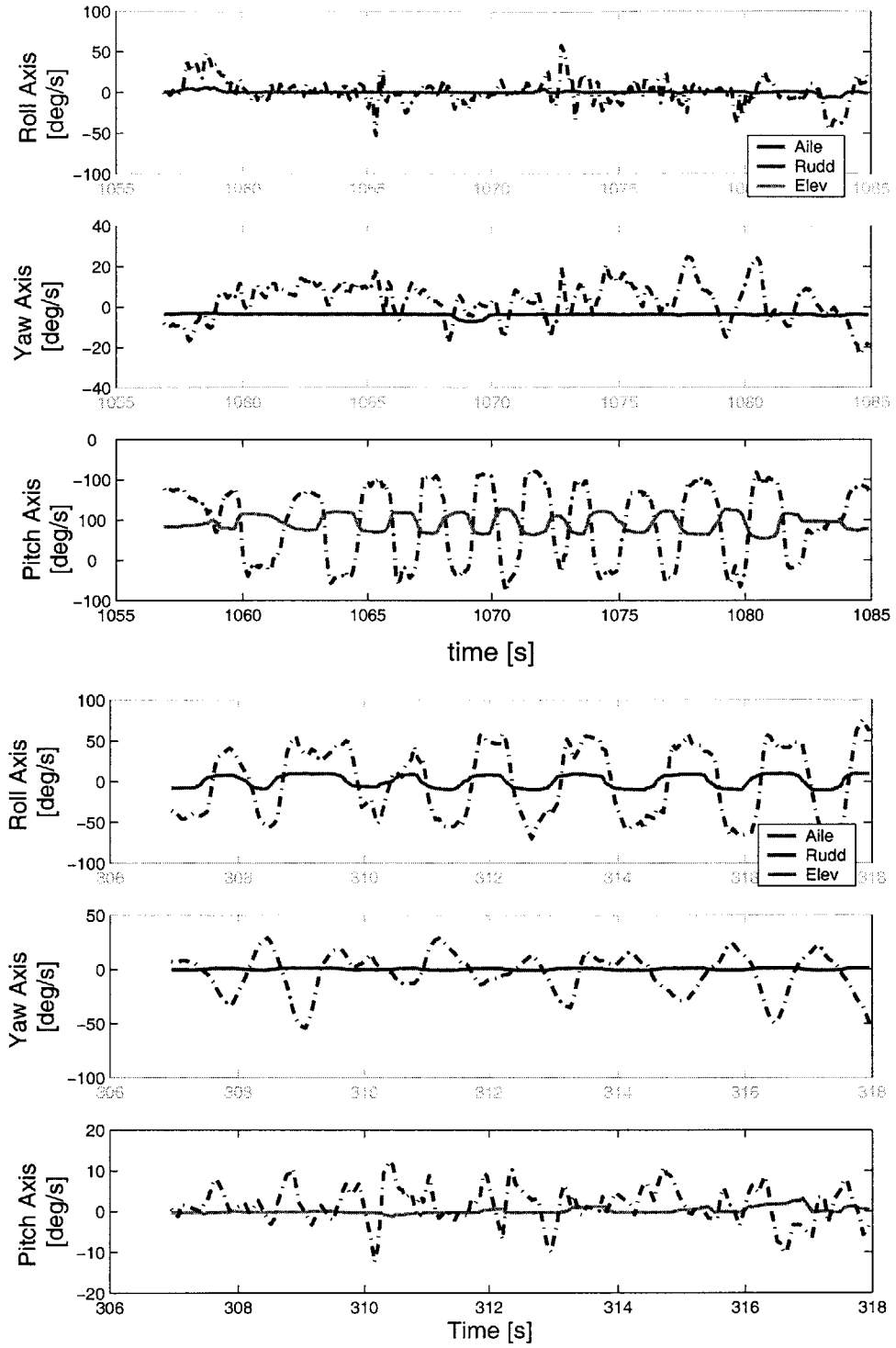
$$\frac{w(s)}{\delta_e(s)} = \frac{z_n \left(s + V_o \frac{m_{\delta_e}}{z_{\delta_e}}\right)}{(s^2 - (m_q + z_w)s + (m_q z_w - m_w V_o))} \triangleq \frac{k_w(s + 1/T_\alpha)}{s^2 + 2\zeta_s \omega_s s + \omega_s^2} \tag{2.30}$$

$$\frac{q(s)}{\delta_e(s)} = \frac{m_n(s - z_w)}{(s^2 - (m_q + z_w)s + (m_q z_w - m_w V_o))} \triangleq \frac{k_q(s + 1/T_\theta)}{s^2 + 2\zeta_s \omega_s s + \omega_s^2} \tag{2.31}$$

where $k_q, k_w, T_\theta, T_\alpha, \zeta_s$, and ω_s represent approximate values for the short period mode and V_o is the vehicle reference speed.

One of the most accurate ways to obtain models for the aircraft data is to use actual flight data. Identification algorithms such as those in the Matlab System Identification Toolbox [33] can be used to used to obtain open loop models of the system dynamics from flight data collected during experiments. These models can then be used to validate the HWIL simulation environment as well as to help determine the gain settings for the autopilot control loops as shown in Section 2.3. Input-output data was collected by disengaging all of the autopilot loops and performing a series of maneuvers to measure the aircraft response to deflections from the elevator and aileron control surfaces. Example data from two experiments are shown in Figures 2-7(a)-(b) depicting the longitudinal and lateral modes, respectively.

To capture the longitudinal dynamics, the bank angle was held fixed at zero degrees, while a series of pitch oscillations were commanded using the elevator. Figure 2-7(a) shows a sample of data that was collected on one run of the pitch test. From the plot it is clear that the longitudinal modes are being excited due to input from the elevator, while the lateral motions in the roll and yaw axes are essentially fixed. Sample data from a roll excitation run is plotted in Figure 2-7(b). This plot also clearly shows coupling in the yaw axis due to the dihedral angle of the wing.



(b) Roll test data induced with aileron deflections. Note the yaw axis coupling due to the large dihedral angle of the wing

Figure 2-7: Sample flight test data used in the estimation algorithms. Dash-dot lines represent rate gyro data output for each axis. Control surface inputs are plotted for each corresponding axis in degrees of deflection

Parametric models for the input-output transfer functions can be formed using experimentally collected data and used to determine the unknown coefficients in Eq. 2.31. Then from the characteristic equation, the longitudinal short period dynamics can be inferred from models of the transfer function from elevator angle to pitch rate. Also note that qualitative predictions about the values of the parameters in Eq. 2.31 are available, since they depend on the concise stability derivatives which all have physical significance. Once models are obtained the approximate parameters can then be verified against these qualitative predictions.

Figure 2-8 shows the output of a parametric subspace model based on experimental data such as that shown in Figure 2-7(a). The model output (solid line) tracks the actual measurements of pitch rate (dashed line) quite well and was validated on data sets from different test days and aircraft. Model residuals within the 99% confidence bounds for the auto- and cross correlation functions are plotted in Figure 2-9 and indicate that the 3^{rd} order model is sufficient to describe the input-output dynamics. This model is represented by the continuous transfer function in Eq. 2.32 and has zero-pole pairs as indicated in Figure 2-10(a). The short period is well represented by the high frequency, oscillatory mode of the system, while there is one low frequency pole located near the imaginary axis. The transfer function is:

$$\begin{aligned} T_{q\delta_e}(s) = \frac{q(s)}{\delta_e(s)} &= \frac{9.539s^2 - 1440s + 60.52}{s^3 + 21.77s^2 + 325.8s + 29.94} \\ &= \frac{9.539(s - 150.9)(s - 0.0420)}{(s^2 + 21.7s + 323.7)(s + 0.0925)} \end{aligned} \tag{2.32}$$

The third order model $T_{q\delta_e}(s)$ was selected because it provided the best fits to a large number of data sets and residuals that remained below the 99% confidence intervals in Figure 2-9. Approximate values for the short period dynamics can be obtained by resolving the oscillatory mode in Eq. 2.32 to determine the corresponding values for ω_s and ζ_s. A step response for this mode is plotted in Figure 2-10(b), indicating a reasonable short period response time with settling time 0.4 seconds, and $\zeta_s = 0.6$. Second order models produced from the same data set were found to have difficulty reproducing the outputs of the experiment, and exceeded the confidence bounds as shown in Figure 2-9. Higher order models (4^{th} and higher) tended to

place additional pole-zero pairs near the origin without achieving better tracking or residual bound performance, therefore indicating the 3^{rd} order model as the best representation for the system.

The zero locations in Eq. 2.32 are not consistent with the expected values for the stability derivatives in Eq. 2.31, which is an indication of the delay acting on this input-output channel and consistent with the servo response time in an actual physical system. In addition, it should be noted that the zero location for these types of models is generally more uncertain than the pole location. As a result, the zeros in Eq. 2.31, are difficult to identify without more sophisticated validation techniques.

In order to validate the HWIL model the same set of tests was performed in simulation and the results are compared in Figure 2-11. As shown in Figure 2-11(a), both models reproduce the experimental outputs with excellent tracking. Since the data set used to validate these two models is different from either of the sets used to generate them, this model can be relied upon with much higher confidence. The transient responses are shown in Figure 2-11(b) and the fast acting short period mode is shown to agree well, however there is variation on the longer timescales. The HWIL model for pitch response is given by

$$\begin{aligned} T_{q\delta_e}(s) = \frac{q(s)}{\delta_e(s)} &= \frac{2.232s^2 - 1265s + 2.449}{s^3 + 19.2s^2 + 283.7s - 2.3} \\ &= \frac{0.0022(s - 566.92)(s - 0.0019)}{(s^2 + 19.209s + 283.8924)(s + 0.0081)} \end{aligned} \tag{2.33}$$

which has a short period characteristic equation similar to Eq. 2.32, identified using experimental flight test data. The discrete P-Z plot in Figure 2-12 shows the pole and zero locations for the longitudinal models. The short period mode is shown to agree well, however the slower frequency dynamics are not as well modeled.

Phugoid Mode

The data from the preceding experiments captures the short period mode of the system well, however the phugoid mode is not well represented. This is physically consistent because the phugoid mode excites the airspeed and pitch response over longer timescales, specifically when $\dot{q} = \dot{w} \approx 0$. Setting the corresponding terms to

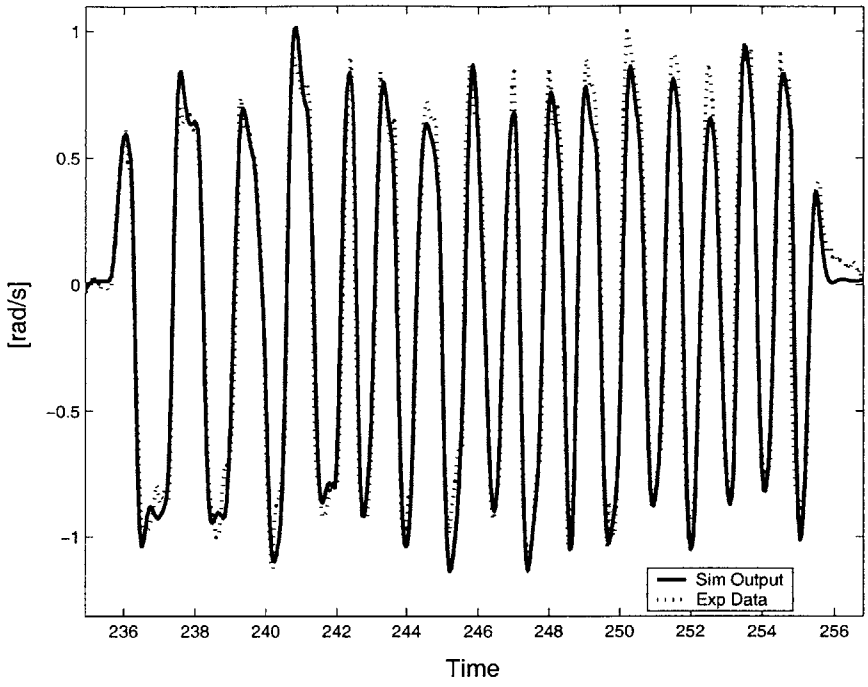


Figure 2-8: Simulated and measured outputs from elevator to pitch rate. Validation on a different data set than that used to create the model.

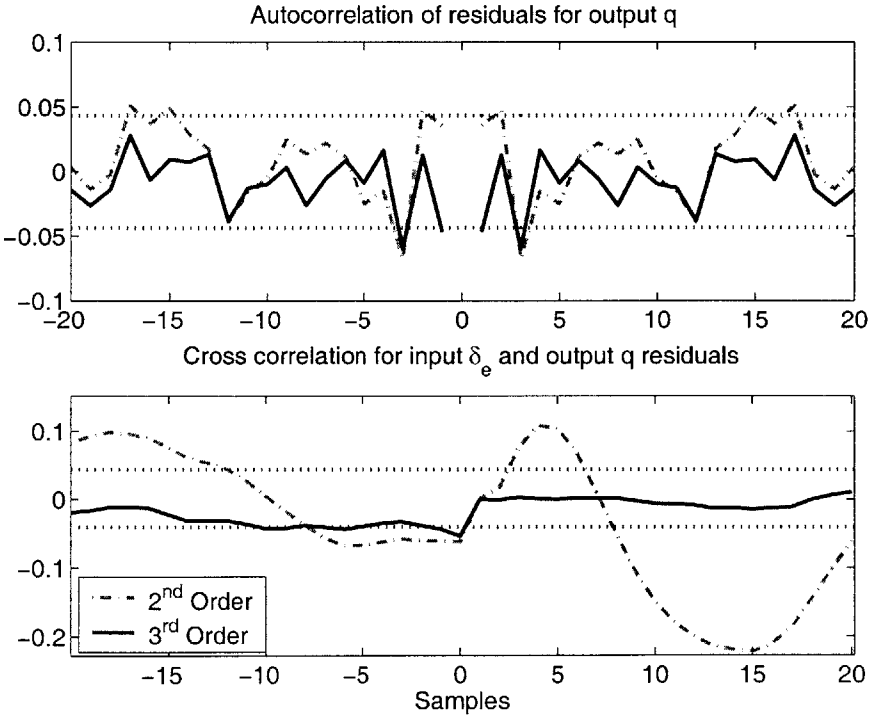
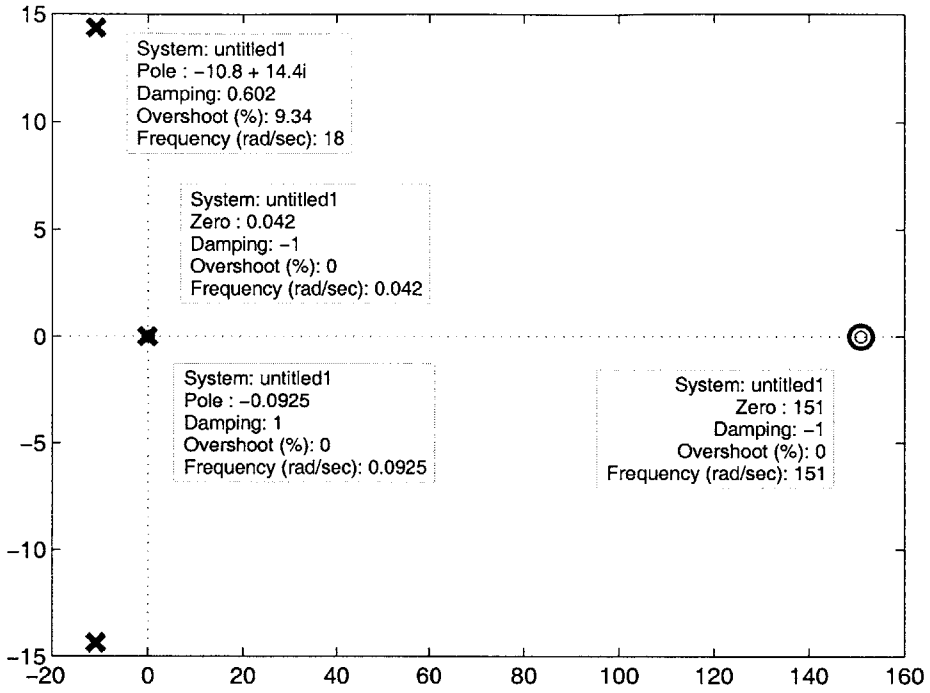
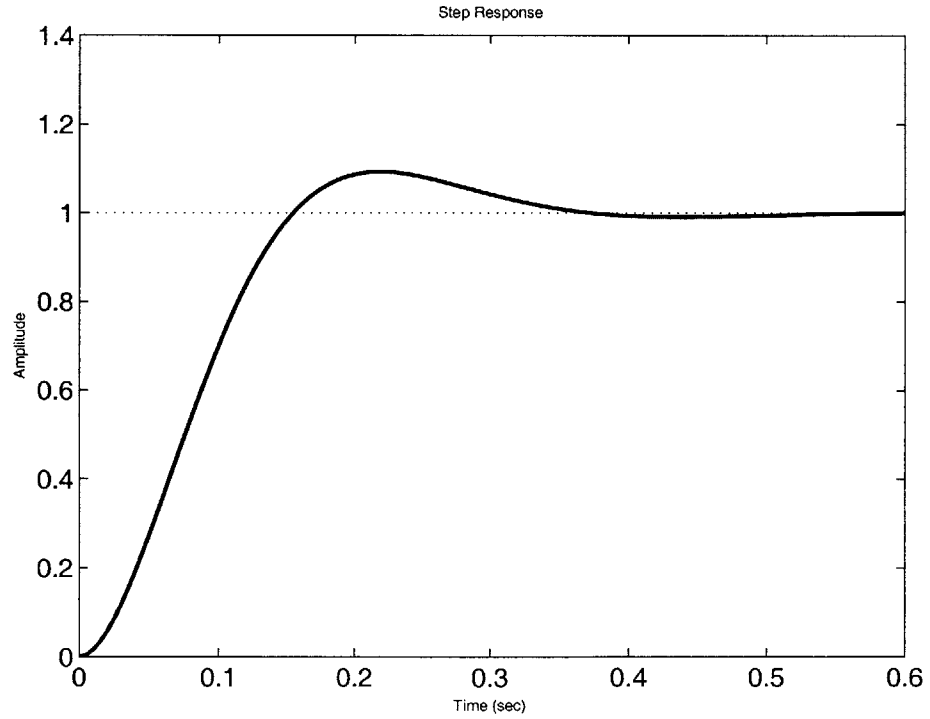


Figure 2-9: Auto- and cross-correlations with 99% confidence intervals (dashed lines) for the elevator to pitch rate models obtained from experimental data. 2^{nd} order model has high cross correlations for positive sample delays, indicating unmodeled higher order dynamics, but 3^{rd} order models stay within the 99% bounds.

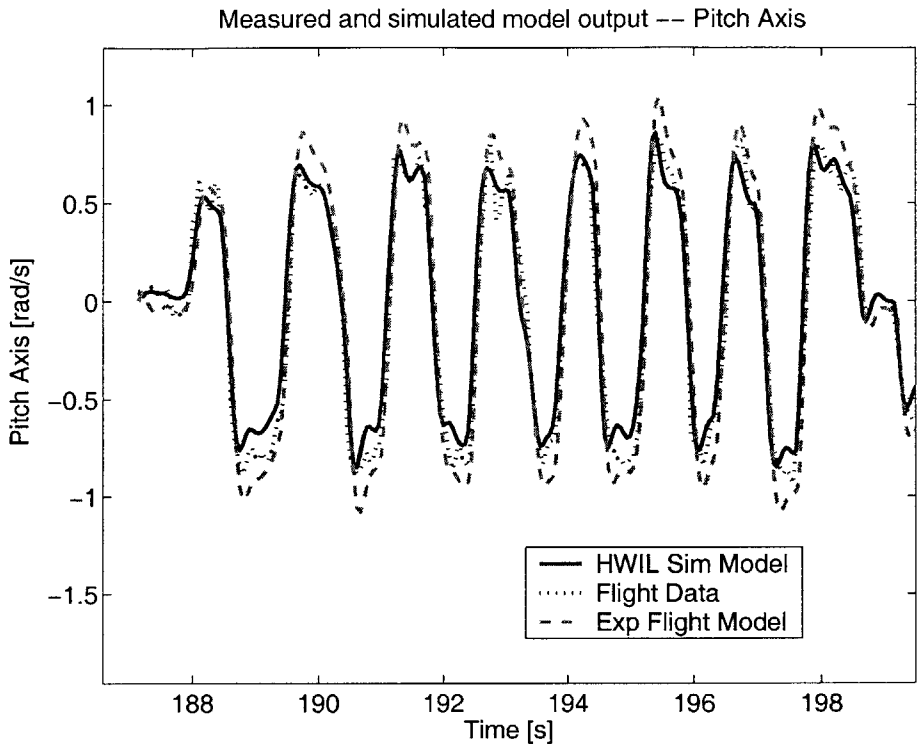


(a) The open loop pole and zero locations indicate the short period dynamics identified by the estimation algorithms.

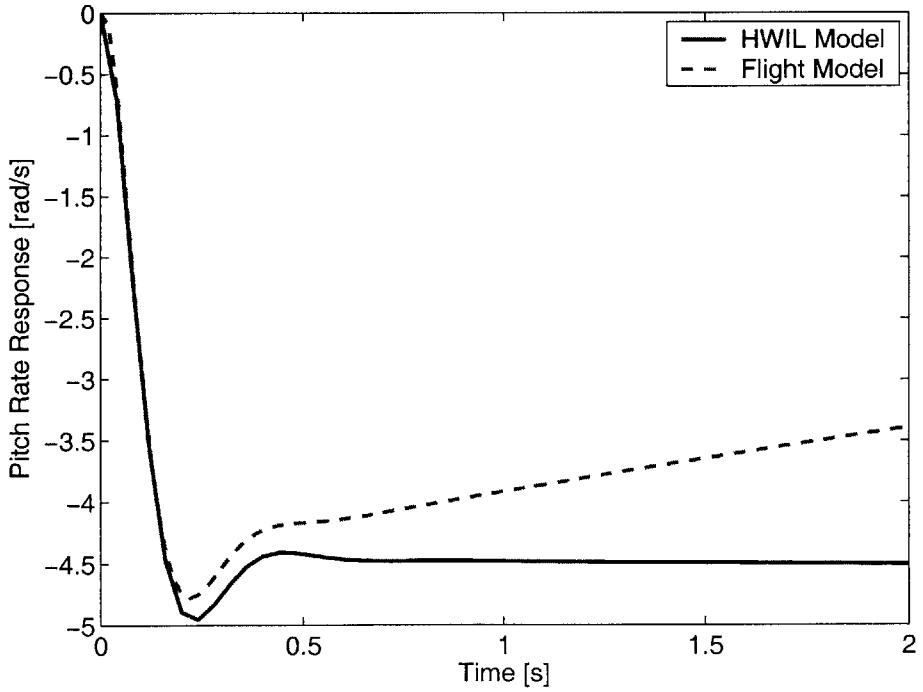


(b) Short period approximation step input

Figure 2-10: Estimated longitudinal model, $T_{q\delta_e}(s)$, from elevator input to pitch rate.



(a) Both models created from the actual flight data and the HWIL simulator are able to track the experimental data well.



(b) Transient response for the longitudinal models.

Figure 2-11: Transient response for the 3^{rd} order models are agree in time constant and damping, showing agreement in the short period dynamics of the HWIL and flight test models.

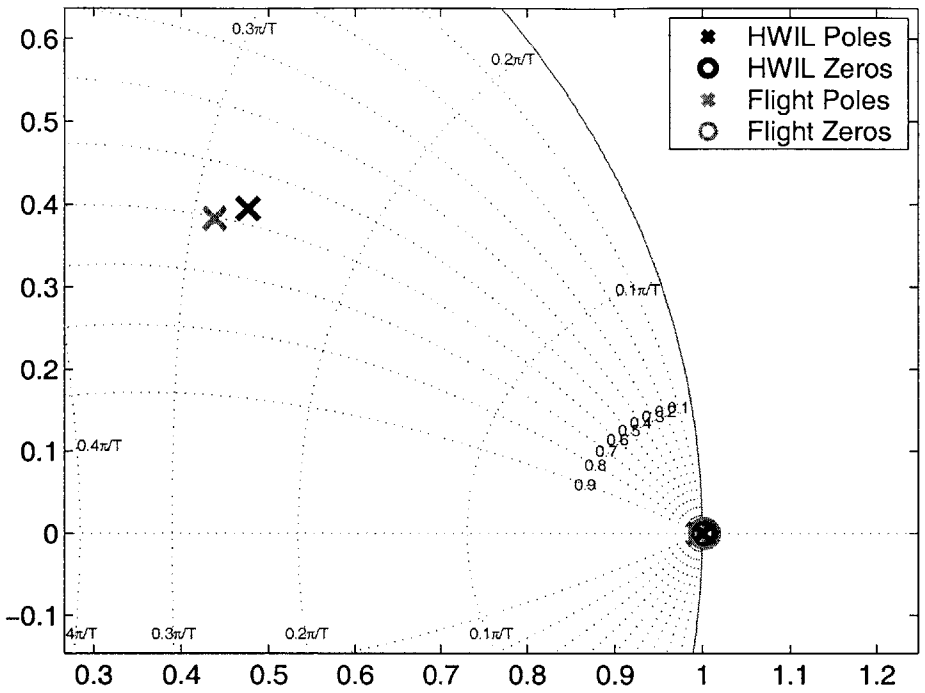


Figure 2-12: Discrete P-Z plot for the experimental and HWIL sim models. Damping and time constant of short period mode agree to a certain margin, but the slow dynamics are not captured well in this experiment.

zero in Eq. 2.27, the dynamics can be simplified to a second order system involving only the body x-acceleration, $\dot{u}$, and the pitch rate, $\dot{\theta}$ [27]

$$\begin{bmatrix} \dot{u} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} x_u - x_w \left(\frac{m_u V_o - m_q z_u}{m_w V_o - m_q z_w}\right) & -g \\ \left(\frac{m_u z_w - m_w z_u}{m_w V_o - m_q z_w}\right) & 0 \end{bmatrix} \begin{bmatrix} u \\ \theta \end{bmatrix} + \begin{bmatrix} x_{\delta_e} - \left(\frac{m_{\delta_e} V_o - m_q z_{\delta_e}}{m_w V_o - m_q z_w}\right) \\ \left(\frac{m_{\delta_e} z_w - m_w z_{\delta_e}}{m_w V_o - m_q z_w}\right) \end{bmatrix} \delta_e$$

$$\dot{\boldsymbol{x}}_{\boldsymbol{p}} = A_p \boldsymbol{x}_{\boldsymbol{p}} + B\boldsymbol{u} \tag{2.34}$$

The phugoid mode can then be approximated by finding the poles of Eq. 2.34,

$$\begin{aligned} \Delta(s) &= |sI - A_p| = s^2 + 2\zeta_p \omega_p + \omega_p^2 \\ &= s^2 - \left(x_u - x_w \left(\frac{m_u V_o - m_q z_u}{m_u z_w - m_u z_u} \right) \right) s + g \left(\frac{m_u z_w - m_u z_u}{m_w V_o - m_q z_w} \right) \end{aligned} \tag{2.35}$$

For conventional aircraft in subsonic flight several approximations can be made [27]

$$m_u \approx 0; \qquad |m_u z_w| \ll |m_w z_u|; \qquad |m_w V_o| \gg |m_q z_w| \tag{2.36}$$

These simplifications yield closed-form expressions for the damping and natural frequency of the mode

$$2\zeta_p\omega_p \approx -x_u \tag{2.37}$$

$$\omega_p \approx \sqrt{\frac{-gz_u}{V_o}} \tag{2.38}$$

As outlined in [27], the concise stability derivatives can be transformed into their dimensionless equivalents and related to the aerodynamic parameters of C_L and C_D. The phugoid frequency and damping can then be expressed in terms of the aerodynamics and airspeed, assuming lift is equal to the weight in the trimmed condition

$$\omega_p \approx \frac{g\sqrt{2}}{V_o} \tag{2.39}$$

$$\zeta_p \approx \frac{C_{D_{total}}}{\sqrt{2}C_{L_w}} \approx \frac{C_{D_w} + C_{D_b} + C_{D_p}}{\sqrt{2}C_{L_w}} \tag{2.40}$$

The subscripts w, b and p denote the contributions due to the wing, body and parasitic forms of the drag for the aircraft, respectively. Eq. 2.39 is known to provide reasonable approximations for the phugoid natural frequency, but due to the simplifying assumptions and uncertainty in the lift and drag coefficients, Eq. 2.40 is less accurate in determining the damping ratio, ζ_p.

In order to observe the phugoid dynamics, an experiment under trimmed conditions with airspeed and/or pitch angle dynamics is required. Unfortunately due to space constraints, an experiment requiring long term "hands off" dynamics on the actual aircraft is not feasible, however this test can easily be performed on the HWIL simulator. The results of a 3 m/s airspeed impulse response with initial airspeed, $V_o = 28$ m/s and zero angle of attack are shown in Figure 2-13. The best fit curve for the oscillation was found to have damping and natural frequency

$$\omega_{p(\text{fit})} = 0.0732 \text{ rad/s and } \zeta_{p(\text{fit})} = 0.20 \tag{2.41}$$

which are close to the predictions from the phugoid approximations in Eqs. 2.39 and 2.40 with predicable uncertainty in the damping term

$$\omega_{p(\text{theory})} = 0.0786 \text{ rad/s and } \zeta_{p(\text{theory})} = 0.14 \tag{2.42}$$

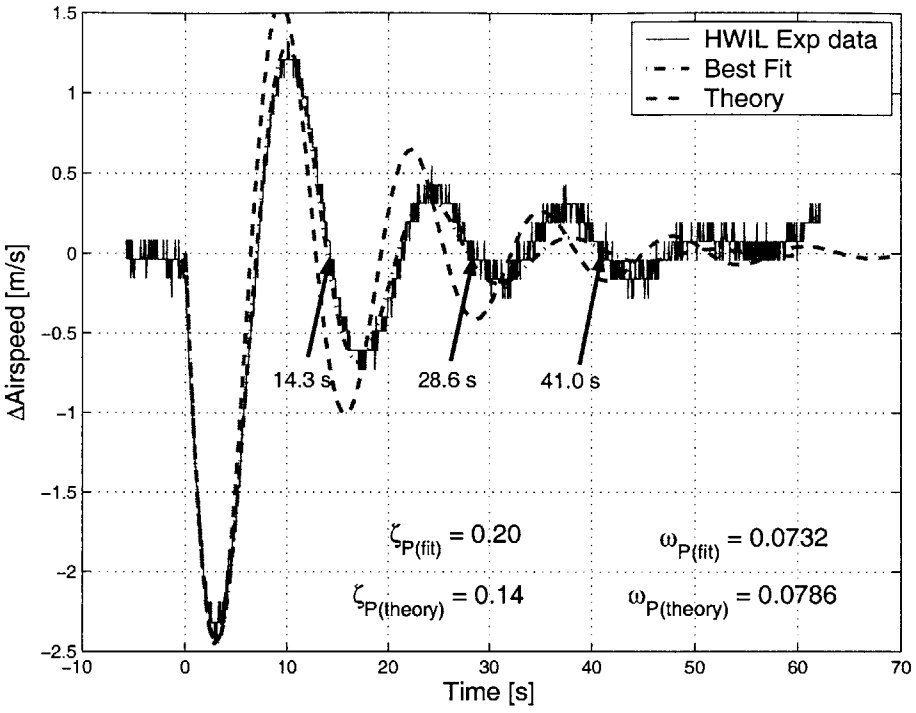


Figure 2-13: HWIL simulation of the phugoid mode using an airspeed perturbation confirms analytical predictions from the initial speed, $V_o = 28$ m/s and Lift/Drag curves. C_{L_w} and C_{D_w} are determined from the wing lift curve at $\alpha = 0$. The values calculated for the body and parasitic components of drag are $C_{D_b} = 0.021$ and $C_{D_p} = 0.01$, respectively.

For these values, the wing lift and drag, C_{L_w} and C_{D_w} terms were determined from the wing lift curves with $\alpha = 0$ in Figure 2-1(b), while the body drag coefficient can be estimated from the power curve at the operating speed and the fuselage geometry.

2.2.2 Lateral Dynamics

Similar to the longitudinal model, the dynamics describing lateral perturbations about an equilibrium trim condition can be written in concise state space form [27]

$$\begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} y_v & y_p & y_r & y_\phi \\ l_v & l_p & l_r & l_\phi \\ n_v & n_p & n_r & n_\phi \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} y_{\delta_a} & y_{\delta_r} \\ l_{\delta_a} & l_{\delta_r} \\ n_{\delta_a} & n_{\delta_r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{2.43}$$

where v is the sideslip velocity, p and r represent the roll and yaw rates, and ϕ is the roll angle. The aileron control deflection angle is denoted by δ_a, and since the effect of the rudder cannot be neglected, it provides an additional control input δ_r.

Roll Subsidence Mode

While the two longitudinal modes decouple rather easily, this is not the case in lateral aircraft dynamics. The lateral motion is characterized by one oscillatory mode known as the *Dutch roll* mode, and two first-order lags known as the *roll subsidence* and *spiral* modes. Because all of the lateral motion is coupled to some extent, it makes identifying individual modes more difficult. Simplifying approximations can still be made for small roll angles, since there is little yaw or sideslip induced, particularly for fast dynamics. In this case the lateral-directional model can be reduced to [27]

$$\begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} l_p & l_\phi \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p \\ \phi \end{bmatrix} + \begin{bmatrix} l_{\delta_a} & l_{\delta_r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_a \end{bmatrix} \tag{2.44}$$

If aircraft wind axes are assumed ($l_\phi = 0$), then Eq. 2.44 further simplifies to a first-order equation representing the roll subsidence mode

$$T_{p\delta_a}(s) = \frac{p(s)}{\delta_a(s)} = \frac{k_p}{s + \frac{1}{T_r}} \tag{2.45}$$

Similar to the longitudinal modes, data can be sampled in the roll and yaw axes and transfer function models can be formed to represent the input-output relations. The first-order dynamics in Eq. 2.45 represent the initial response of the system after the ailerons are actuated. Figure 2-14 presents the experimental and simulated output for the roll axis – the first-order model (solid line) is shown to track the output (dashed line) well. The output of the model in Figure 2-14 is represented by the first order system

$$T_{p\delta_a}(s) = \frac{106.48}{s + 14.36} \tag{2.46}$$

where the time constant $T_r = \frac{1}{14.36} = 0.070$ sec. identifies the dominant physical properties for the roll mode.

Figure 2-15(a) demonstrates the tracking of two models for the roll subsidence mode created from experimental flight test data and the same test performed in HWIL

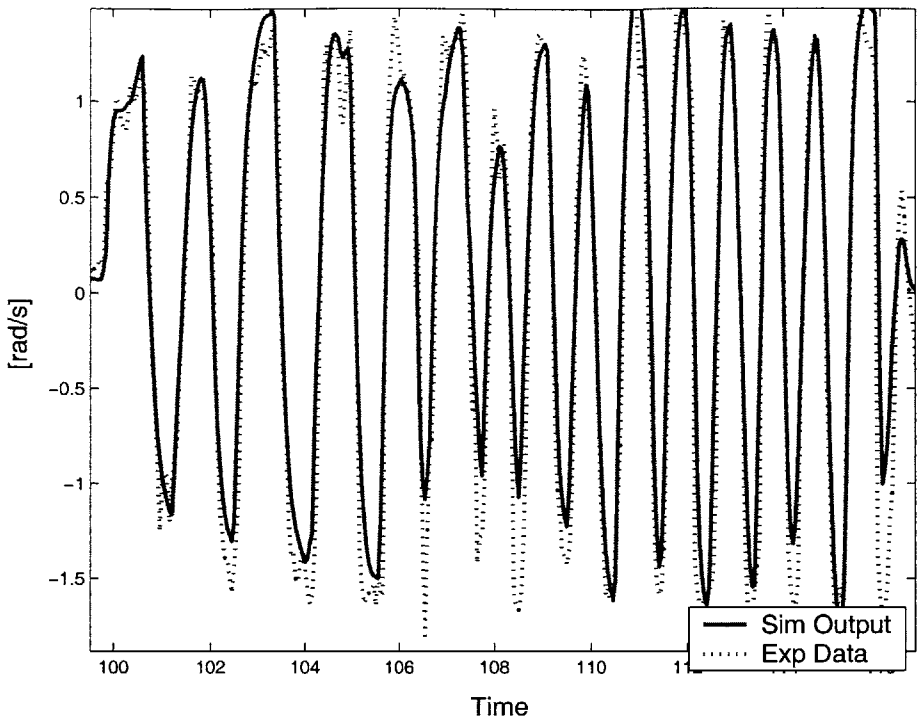


Figure 2-14: Simulated and measured outputs from aileron to roll rate.

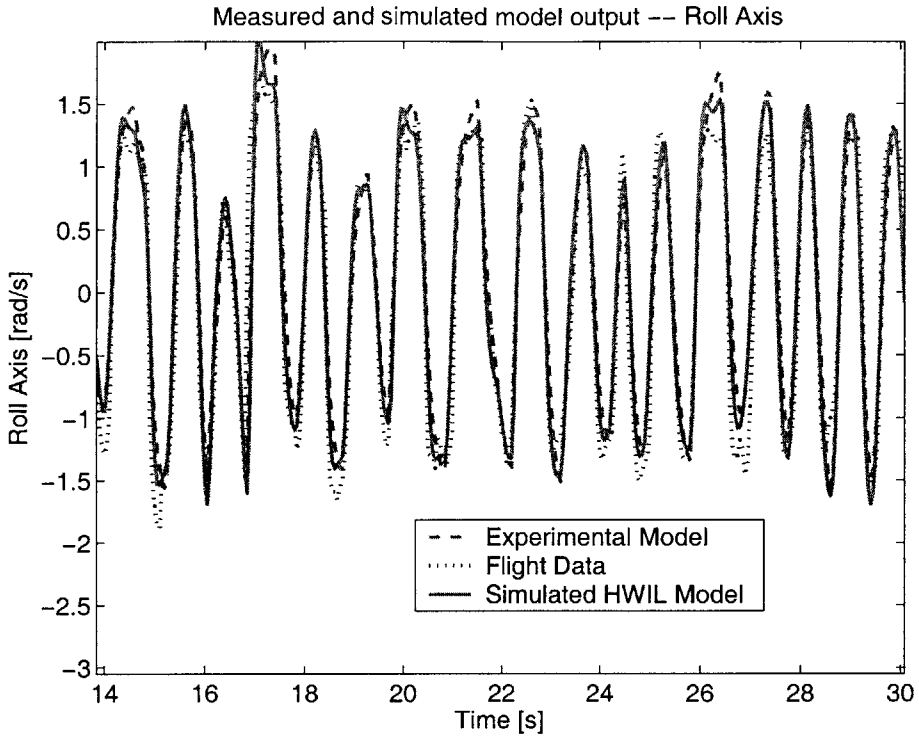
tests. The data sets show strong agreement indicating that the HWIL simulation provides an accurate representation of the testbed aircraft in roll response. As shown in Figure 2-15(b), the time constant value for both models was found to be roughly 0.1 seconds, which is reasonable for aircraft of this size. Subtle model differences created a discrepancy in the overall gain factor of the roll-rate transfer function, but this was corrected in the HWIL simulator by adjusting the aileron chord length from its true physical value until the transfer functions responses were in better agreement. The roll subsidence transfer function for the HWIL simulator is

$$T_{p\delta_a}(s) = \frac{108.10}{s + 14.27} \tag{2.47}$$

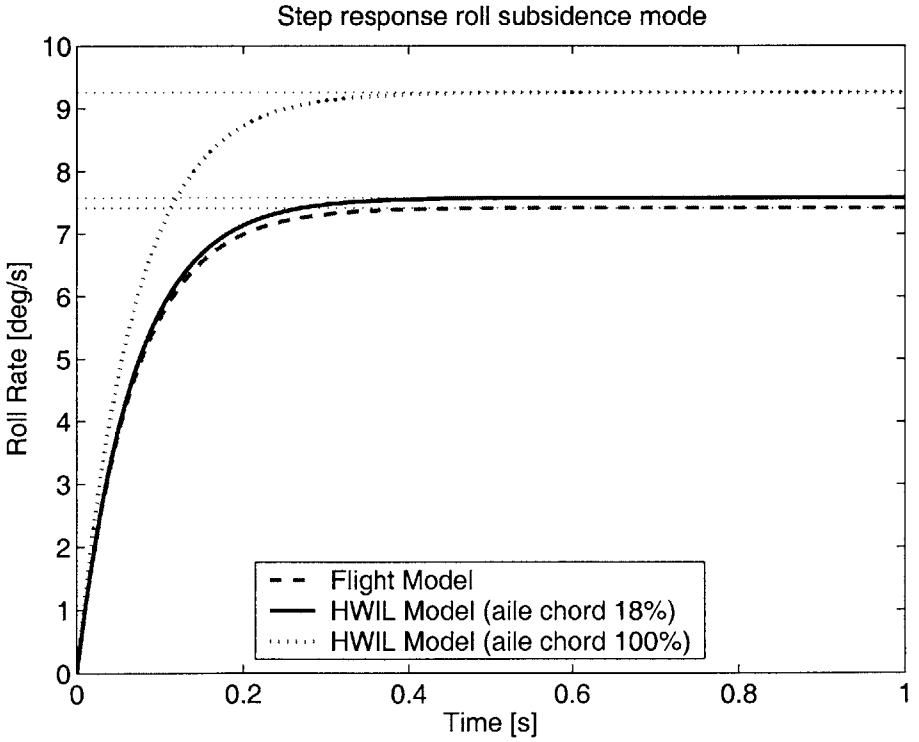
which has time constant of $T_{r_{HW}} = \frac{1}{14.27} = 0.07$ sec. This model agrees well with Eq. 2.45, indicating strong correlation with the true roll response of the aircraft.

Spiral Mode

The roll subsidence mode can be simply decoupled from the lateral dynamics, however the spiral and Dutch roll modes are more difficult to identify. The spiral mode



(a) Models created from both the actual flight data and HWIL simulator data can track the experimental data.



(b) Transient response for roll subsidence models. Time constant is ≈0.1 sec. Aileron chord length was varied from true value to obtain better agreement with model obtained from flight test data.

Figure 2-15: Agreement between the models from aileron input to roll rate output for the HWIL and flight data models.

is a non-oscillatory, slowly acting mode that captures complex motions in roll, yaw, and sideslip. It is characterized by the interaction between directional stability (also known as weathercock or fin stability) and the lateral stability (dihedral) of the aircraft. These two effects counterbalance one another resulting in the slowly acting spiral mode, which can be stable, neutrally stable or unstable depending on the relative strengths of the lateral directional effects.

Over long periods, the motion variables v, q, and p can be assumed steady, and Eq. 2.43 can be reduced using the approximations $\dot{v} = \dot{q} = \dot{p} \approx 0$. This allows the reduced order model for the spiral mode to be formed as shown in [27]

$$\begin{bmatrix} 0 \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} y_r \frac{(l_v n_p - l_p n_v)}{l_r n_v - l_v n_r)} & y_\phi \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p \\ \phi \end{bmatrix} \tag{2.48}$$

Since $\dot{\phi} = p$, Eq. 2.48 can be reduced to a first order differential equation describing the unforced spiral mode dynamics. With reference to Appendix 1 of [27], the time constant of the mode, T_s can be expressed in terms of the dimensionless derivatives

$$T_s = -\frac{V_o(L_v N_p - L_p N_v)}{g(L_r N_v - L_v N_r)} \tag{2.49}$$

Due to the timescales over which the spiral mode acts, it typically is not possible to identify the mode using input-output estimation techniques. Instead, a series of analytical approximations based on aircraft geometry and aerodynamic data can be used to identify the parameters of the spiral mode, and this prediction is used to validate the HWIL settings. Approximations for the derivatives in Eq. 2.49 were found in terms of the aerodynamic and geometric properties of the ARF 60 aircraft [27]

$$N_v = a_{1F} \bar{V}_f \tag{2.50}$$

$$N_p = -\frac{1}{12} C_L \Big|_{\alpha=0} \tag{2.51}$$

$$N_r = (N_r)_{wing} + (N_r)_{fin} = -\frac{1}{6} C_D \Big|_{\alpha=0} - N_v \frac{l_f}{b} \tag{2.52}$$

$$L_p = -\frac{1}{12}(C_{L_\alpha} + C_D|_{\alpha=0}) \tag{2.53}$$

$$L_v = (L_v)_{wing} + (L_v)_{fin} = -\frac{1}{4} C_{L_\alpha} \Gamma - a_{1F} \bar{V}_f \frac{h_f}{l_f} \tag{2.54}$$

Table 2.4: Dimensionless derivatives for the ARF 60 aircraft contributing to the lateral spiral mode

	Derivative	Approx. ARF 60
N_v	*Yaw Stiffness*	0.0792
N_p	*Yaw Moment due to Roll Rate*	-0.0167
N_r	*Yaw Damping*	-0.0547
L_p	*Roll Damping*	-0.3550
L_v	*Dihedral Effect*	-0.1103
L_r	*Roll Moment due to Yaw Rate*	0.0389

$$L_r = \frac{1}{6} C_L|_{\alpha=0} - (L_v)_{fin}\frac{l_f}{b} \tag{2.55}$$

a_{1F} denotes the wing lift curve slope for the tail, which is analytically approximated as a thin wing in subsonic flow [32]

$$a_{1F} = (C_{L_\alpha})_{fin} = \frac{2\pi AR_f}{2 + AR_f\sqrt{1 + \tan^2\Lambda_f + 4}} \tag{2.56}$$

where AR_f is the aspect ratio of the fin. Using the measurements of the aircraft geometry from Table 2.1, the derivatives for the ARF 60 aircraft are computed and summarized in Table 2.4. The time constant of the spiral mode can then be computed from Eq. 2.49 as $T_s = 28.95$. From this rough analysis it is confirmed that the spiral mode will be a very slow, but stable mode with the positive root given by

$$\Delta_s(s) = (s + 0.0345) \tag{2.57}$$

This mode was difficult to observe in flight tests or HWIL simulations, but it was confirmed that the trimmed open loop ARF 60 model does not exhibit divergent behavior over extended periods of time (> 10 min), showing that the low frequency modes are stable as predicted.

2.3 Autopilot Tuning

The Cloud Cap autopilot has a predefined set of controllers that need to be tuned to suit the 60 ARF aircraft modeled in the previous sections. The HWIL simulator is a

useful tool that can be used to tune the autopilot gains and control settings before flight testing. Previous sections have verified the accuracy of the HWIL simulations, it is expected these predictions will map well to the actual aircraft being flown. This section deals with the tuning of the autopilot to suit the requirements of the MILP trajectory planner, which pushes the limits of the waypoint tracking strategy employed. By uploading closely spaced waypoints and making frequent changes to the plan the autopilot is tracking, the closed loop system is required to be responsive, however as with any control system, care needs to be taken to protect against instability.

In addition to path following, the closed loop tracker is required to reject any disturbances acting on the system in order to fly the trajectory plans designed. Later sections incorporate methods for estimating the disturbance levels and accounting for them on higher planning levels, however these estimates will be imperfect, requiring that the autopilot be successful in rejecting bounded levels of wind and measurement noise error. Subsection 2.3.1 addresses the lateral autopilot controllers, the waypoint tracker is described in Subsection 2.3.2, and Subsection 2.3.3 describes the process in tuning the airspeed/altitude controllers. All of the tunings completed using the HWIL simulator are also validated during actual flight testing of the ARF 60 aircraft.

2.3.1 Lateral Autopilot

Effective strategies for autonomous control are often accomplished by nesting numerous control loops to build up to a complicated desired level of functionality. However the overall performance of the system is then limited by the bandwidth and performance of the innermost loops. Because the autopilot will eventually be used for tight trajectory following, the tuning of the inner loops is important to gain adequate system performance. Quick, tight response is needed on the turn rate loops on the order of the roll dynamics response (from Subsection 2.2.2, $T_r = 14.27$), however settings that are too aggressive quickly lead to roll oscillations and eventual instability.

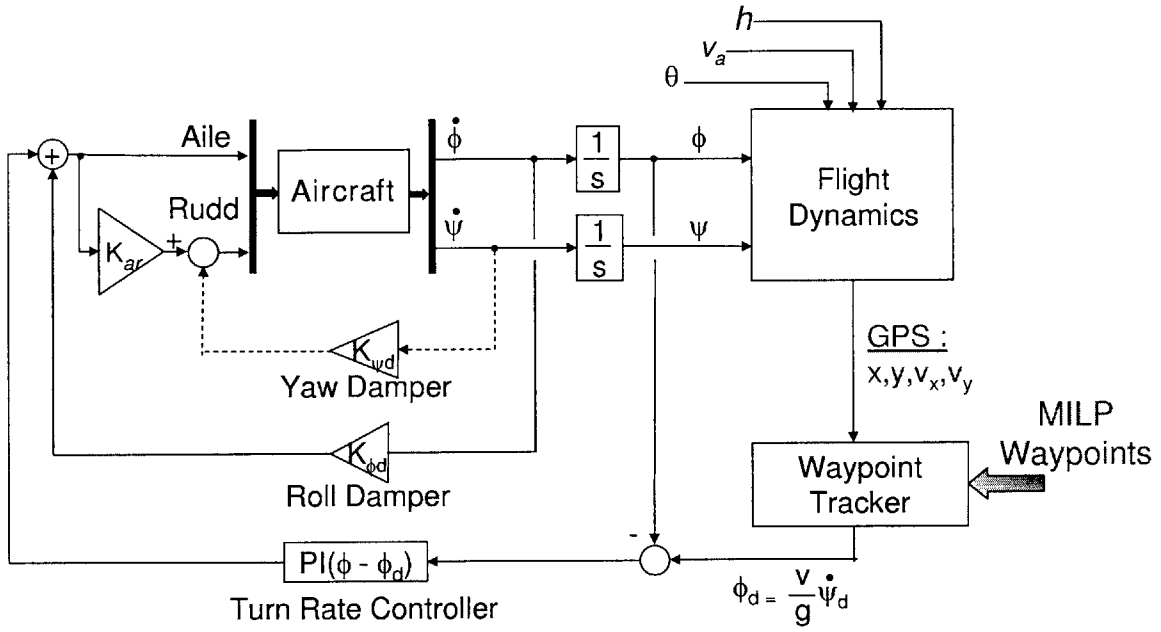


Figure 2-16: Lateral Autopilot block diagram

Turn Rate

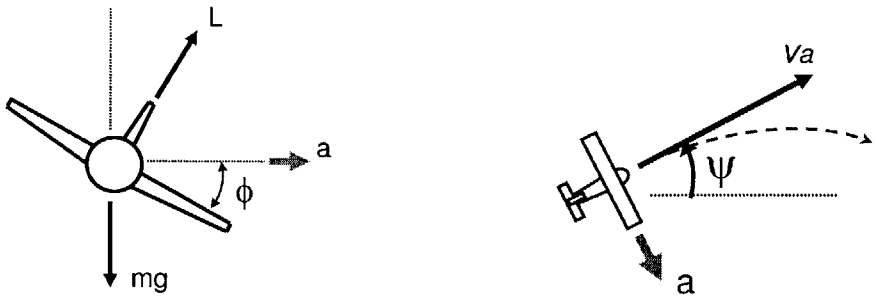
The turn rate of the aircraft is regulated by maintaining a desired bank angle, giving an acceleration in the radial direction as shown in Figure 4-3. Assuming a coordinated turn, the desired bank angle, ϕ_d, is

$$\phi_d = -\frac{v_a}{g}\dot{\psi}_d \tag{2.58}$$

where the desired turn rate, $\dot{\psi}_d$, is the output of the waypoint tracker controller, v_a is the aircraft airspeed, and ϕ is the bank angle (positive for right wing down). The bank angle is regulated by generating a control signal to the ailerons, u_{aile}, using PI control on the bank angle error, $e_\phi(t) = \hat{\phi}(t) - \phi_d(t)$, where $\hat{\phi}$ is the bank angle estimate obtained from filtered roll rate measurements with bias correction,

$$u_{aile}(t) = K_{\phi p}\left(e_\phi(t) + \frac{1}{T_\phi}\int_0^t e_\phi(\tau)\mathrm{d}\tau\right) \tag{2.59}$$

and K_p and T_ϕ are the controller gain and reset rate, respectively. This PI control strategy is appropriate for this application, as integral action is needed to ensure the tracking of the desired turn rate command from the waypoint tracker, while the separate roll damper loop handles upsets caused by turbulence. Note that another



(a) Tail view, with the roll angle, ϕ, and lift forces, L, shown.

(b) Overhead view, with the airspeed vector, v_a, and yaw angle, ψ, shown.

Figure 2-17: Aircraft in a coordinated turn undergoing lateral acceleration, a. The figure assumes that the airspeed, v_a, is constant.

strategy using strictly yaw rate measurements is available, if the roll angle estimates cannot be used due to high noise levels.

Using the HWIL simulator the turn rate controller can be tuned to provide good performance before flight testing. Figure 2-18 shows the response to a bank angle command as given by Eq. 2.58 for varying K_p gains and constant reset rate, $1/T_\phi = 0.1$. The plot shows the increased rise time and settling time for increasing K_P gains, however at the cost of increased turn rate oscillations. The most appropriate gains found for the HWIL simulation model are shown with the thicker lines, indicating reasonable settling time of about 5 seconds, and only slight oscillations in the turn rate.

Roll damper

The roll damper is used to counteract roll turbulence by direct feedback of the roll rate measurement, $\dot{\phi}$ to the ailerons before it integrate to the effect the turn rate controller. A low pass filter (10 rad/s bandwidth) is applied to the roll rate measurement, and GPS is used to correct for the drift bias before feeding it through the roll damper, K_ϕ. Figure 2-19 shows the effect of applying the added derivative action to the turn rate loop, with increasing damper gain, $K_{\phi d}$. The derivative action softens the roll response of the autopilot and increases the bank angle oscillations. Since the 60 ARF trainer aircraft already has a high degree of roll stability, very little, if any, damper

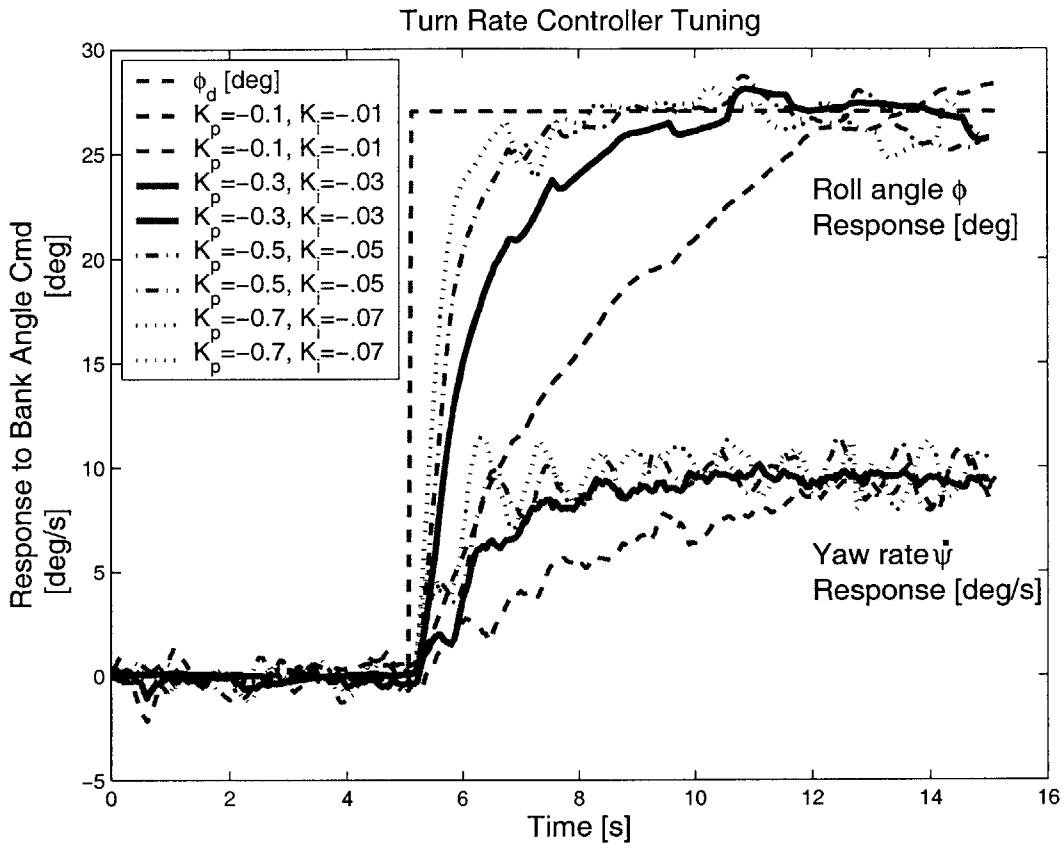


Figure 2-18: The performance variation of the turn rate loop with variation of the proportional and integral gains, $K_{\phi p}$, and $K_{\phi i} = \frac{K_p}{T_\phi}$ respectively. The bank angle estimate, $\hat{\phi}$, and a filtered yaw rate measurement, $\hat{\dot{\psi}}$, is shown in response to a step change in desired bank angle. The thick line provides the best tradeoff between roll response and undamped oscillations in turn rate.

gain is needed to prevent turbulence upsets. The thicker line in Figure 2-19 shows the response for the slight damper settings that might be used in flight. Note that from the time constant of the graph the bank angle dynamics of the aircraft may be determined. Modeling the bank angle as a first order lag of the desired command, the dynamics can be written

$$\frac{\phi(s)}{\phi_d(s)} = \frac{1}{T_c s + 1} \tag{2.60}$$

with $T_c \approx 1.25$ seconds from Figure 2-19. This measurement will be later used in Section 4.2 to modify the MILP dynamics formulation to account for the lag in acceleration that would be seen as a result of the step change.

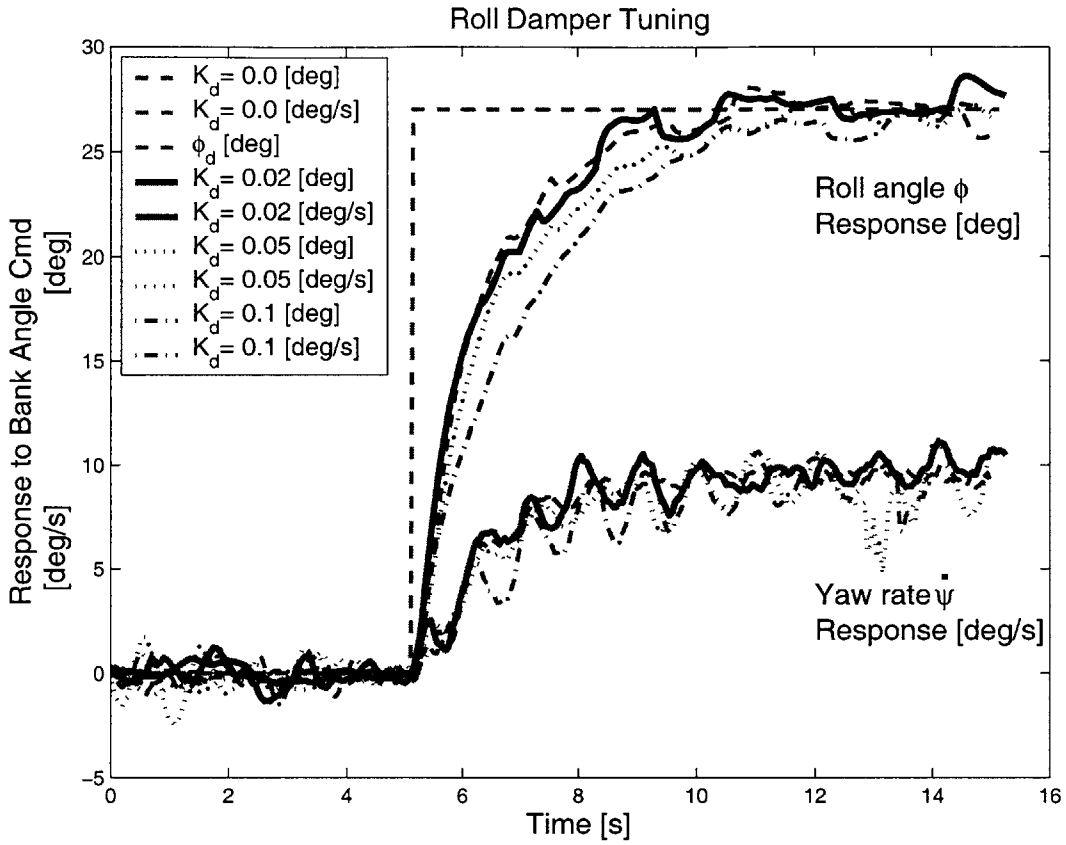
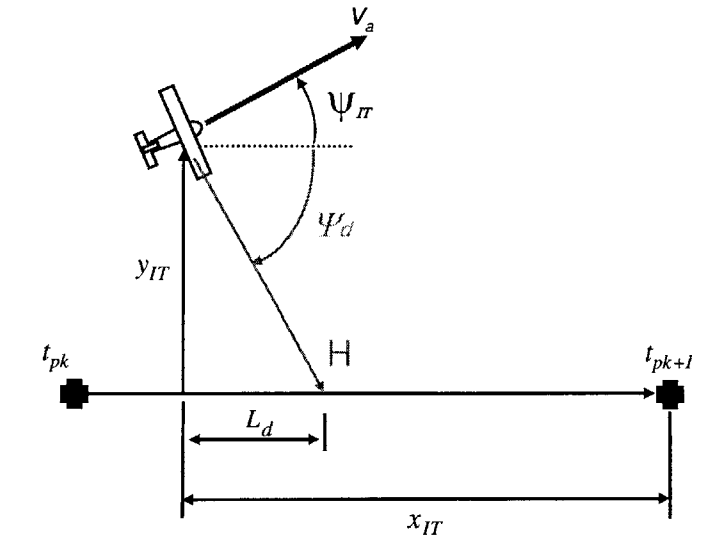


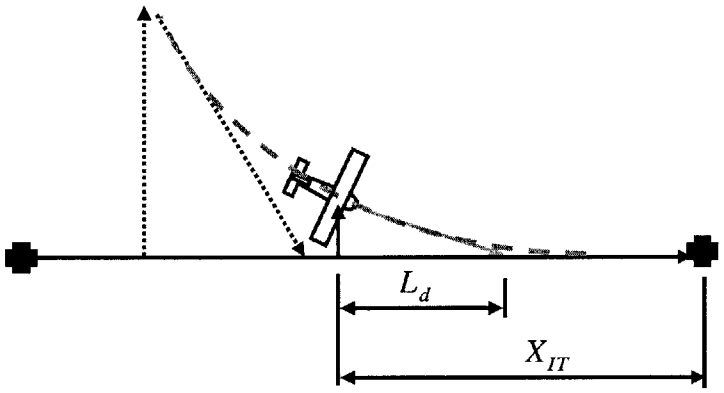
Figure 2-19: Variation of Turn Rate performance with selection of the appropriate roll damper gains, $K_{\phi d}$ with constant proportional and integral gains, $K_{\phi p} = 0.3$, $K_{\phi i} = 0.01$ respectively

Rudder Mixing

Aileron-Rudder mixing rules are employed to assist in making coordinated turns for the aircraft, reducing the sideslip motion while banking the aircraft. For this trainer aircraft this is achieved by setting the aileron to rudder mixing ratio, $K_{ar} = 0.15$, which helps to reduce the errors in GPS heading when making turns. The effectiveness of the mixing ratio could not be determined without additional measurements on the aircraft true heading, however by including an onboard magnetometer the sideslip motion can be measured through comparison of the GPS and true heading estimates. Added true heading measurements would facilitate the selection of the optimum rudder mixing ratio for a variety of flight conditions and airspeeds, thereby reducing the GPS errors due to sideslip motion in bank turns.



(a) Waypoint tracker geometry.



(b) Tracker Response Depicted

Figure 2-20: Lateral track control law for the Cloud Cap autopilot.

The yaw damper loop as shown in Figure 2-16 is typically needed on aircraft with less lateral directional stability, however the trainer ARF 60 aircraft have a large enough dihedral angle, $\Gamma = 5°$, and vertical fin to reject perturbations in yaw (note the relative stability of the derivatives in Table 2.4). Yaw damper augmentation of the plant dynamics is therefore not needed and disabled for the trainer ARF 60 aircraft.

2.3.2 Waypoint Tracker

The Piccolo^{TM}autopilot uses a robust nonlinear waypoint tracker, originally designed for implementation on the Aerosonde UAV [34], which is capable of tracking the vehicle through series of inertially fixed waypoints using GPS position and velocity

measurements. The strategy employs PD control on the error signal $e_\psi = \hat{\psi} - \psi_d$, where $\hat{\psi}$ is the aircraft heading estimate and ψ_d is the desired heading vector as shown in Figure 2-20(a). The heading estimate, $\hat{\psi}$, is approximated using the GPS velocity vector, which is a valid assumption for low wind conditions and small aircraft bank angles (*i.e.,* coordinated turns),

$$\hat{\psi} = \arctan\left(\frac{v_{ay}}{v_{ax}}\right) \tag{2.61}$$

The output of the waypoint tracker is a turn rate command, which is subsequently related to the bank angle of the aircraft as shown in Eq. 2.58 [35].

Convergence Parameter Scaling

As shown in Figure 4-5, the desired heading vector, ψ_d, is determined in the intrack reference frame by selecting a point L_d meters ahead of the current along track position. From Figures 2-20(a) and 2-20(b), the distance L_d, (also referred to as the *tracker convergence parameter*), determines how sensitive the waypoint tracker will be to cross-track errors and is selected to be approximately equal to the vehicle turn radius. For an aircraft in a coordinated turn at a specified maximum bank angle, $\phi_{\max} = 30°$, with airspeed $v_a = 25$ m/s the vehicle turn radius is given by

$$L_d \approx \rho_{\min} = \frac{v_a^2}{g \tan \phi_{\max}} \approx 120\text{m} \tag{2.62}$$

Simulations confirm that selecting $L_d < 120$ leads to instability in the waypoint tracker, as noted by highly oscillatory flight patterns of the vehicle in Figure 2-21. With the selection of the convergence parameter, the desired heading vector is given by

$$\psi_d = -\arctan\left(\frac{y_{IT}}{L_d}\right) \tag{2.63}$$

where y_{IT} is the cross track distance of the vehicle and the negative sign is required for correct operation.

The convergence parameter, L_d, is in fact not constant for all operating conditions, as there are certain situations which require adjustment to the closed loop dynamics. The control law described in Figure 2-20(a) works well when the GPS vector provides

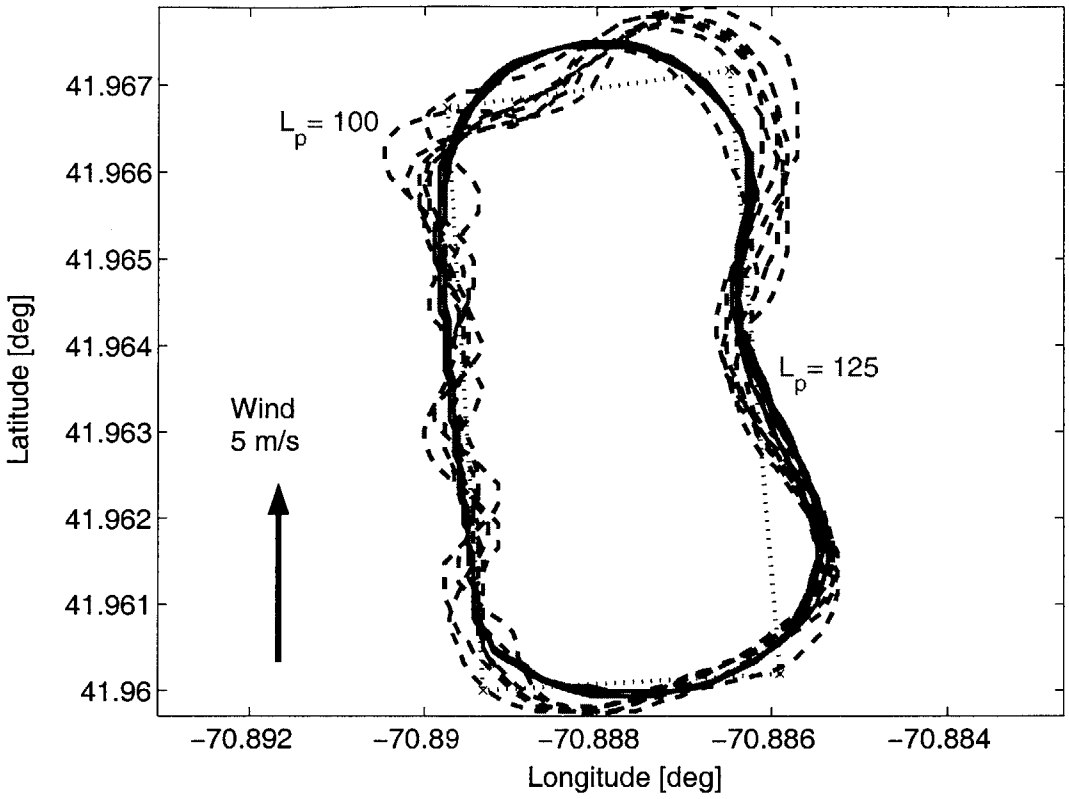


Figure 2-21: Groundtrack of a fixed counter-clockwise waypoint plan for different selections of the intrack convergence parameter, L_d. Different selections determine the autopilot sensitivity to cross-track error, creating unstable oscillations for small values of L_d. Headwinds tend to excite oscillations for the marginally stable system, as the turn rate command provides too much authority to the aileron actuators.

an accurate estimate of the vehicle wind-relative heading. However, due to the effects of wind acting on the vehicle, the aircraft will experience changes in the effectiveness of the turn rate command with changes in the wind relative to the vehicle body axis. By scheduling L_d with varying airspeed to increase the tracker performance in the case of winds, the sensitivity the tracking algorithm is mitigated over the variation in the no-wind case.

The convergence parameter is also scaled according to the vehicle cross-track error, in order to provide higher performance for small cross-track errors, $y_{IT} < L_d$. In order to achieve this one of several proprietary methods is employed in the Cloud Cap

autopilot including a nonlinear scaling of L_d with cross-track error [36]. This scale factor smoothly transitions the convergence parameter to small values for $y_{IT} < L_d$, providing a means to keep the waypoint tracker sensitive for all regions of operation.

While detailed simulations of the system in Figure 2-16 can be performed using the aircraft models developed in earlier sections, the nonlinear scalings and ambiguities in the autopilot internal control loops make accurate predictive simulations difficult to achieve. The simulations lead to mismatches between the predicted and actual gains needed for HWIL or flight testing, although general trends and limits can be confirmed. In practice these tests do not capture all the dynamics required to accurately simulate the closed loop for controller tuning purposes. As a result, the most effective way to select the control gains is to test on the HWIL simulator and subsequently verify on actual flight tests.

Tracker Tuning

The performance of the tracker loop depends primarily on the inner turn rate loop performance as well as the airspeed of the aircraft being flown. Gains that are set at low airspeeds will typically not have as much authority as the same gains at higher airspeeds, while gains set at the high end of the airspeed range can lead to oscillations if left unchecked. For this reason, the HWIL simulator is exercised at several points of the flight envelope to ensure proper operation.

One suitable method for finding autopilot gains is to increase the proportional term, K_p, until the critical point for sustained oscillations is reached. Once this critical point is found, the proportional term can be reduced by 20-30% in magnitude, depending on the airspeeds the aircraft will be flying. The derivative gain, K_d, is then used to soften the autopilot response and further reduce oscillations as needed. The results of this process are shown in Figure 2-22, for an aircraft being flown at 26 m/s, close to the upper limit of the speed range. With the proportional gain set at $K_p = -0.25$, the derivative gain of $K_d = -0.5$ is shown to significantly reduce the undamped response, with a settling time of 17 seconds and 10% overshoot.

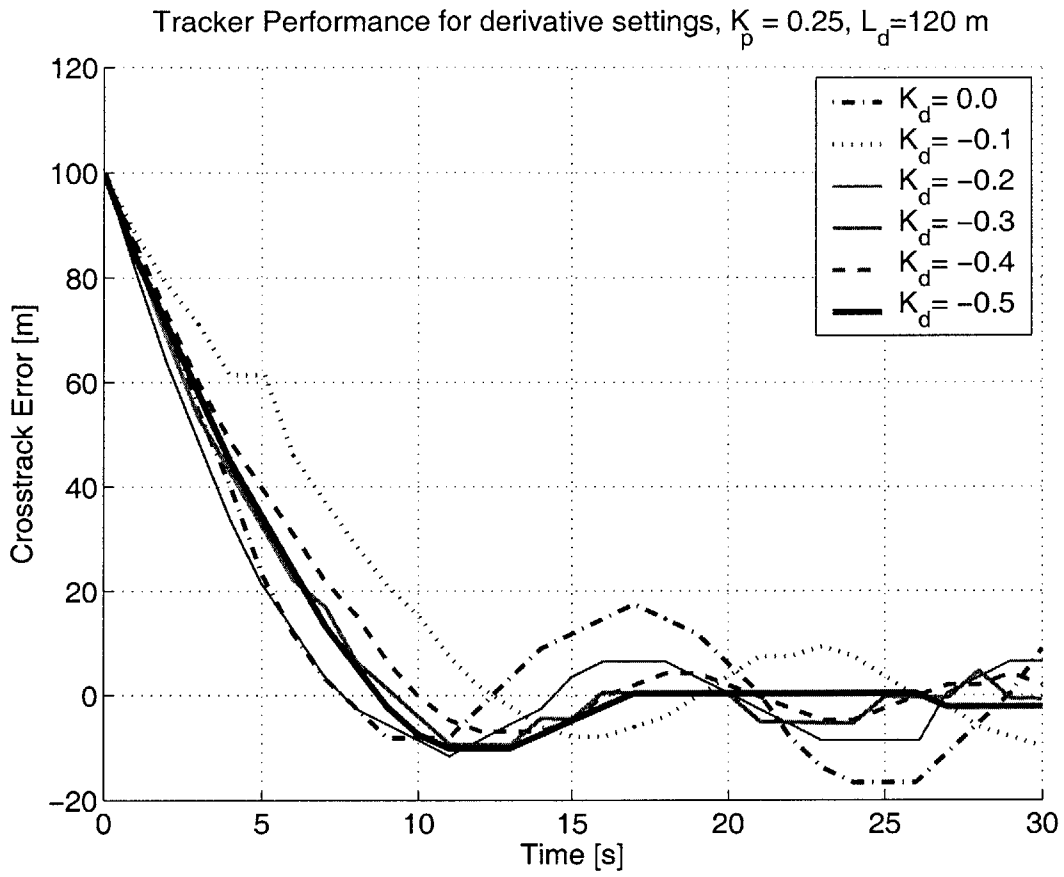


Figure 2-22: The response of the closed loop system is shown for varying derivative tracker gains, reducing the undamped oscillations of the system. Closed loop settling time from a step disturbance of 100 m in cross-track error is shown to be reduced to 17 seconds with 10% overshoot.

Dynamic MILP Trajectory Conversion

The output of the MILP planner is a series of ordered waypoint lists (*flight plans*) that are uploaded to the vehicle in real-time as each optimization completes. These waypoint lists represent the physical locations in (x, y) space that the MILP planner assumes the vehicle will be passing through at a given time, and they need to be converted into a form that is consistent with the closed loop response of the waypoint tracker. The waypoint lists are updated at a rate of roughly 1 plan every 4-5 seconds, depending on the wind conditions. As shown in Ref. [8], the process is repeated when the vehicle reaches the *horizon point* of the current plan.

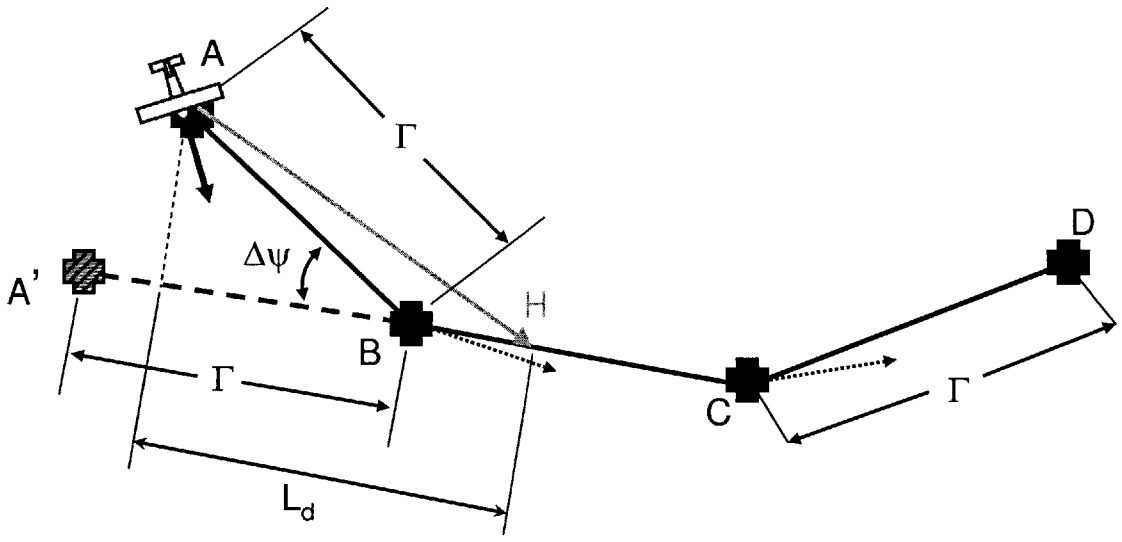


Figure 2-23: Conversion of MILP waypoints to vehicle flight plans, utilizing the waypoint tracker control law to keep bank angle constant throughout a turn. With a small enough waypoint spacing, $\Gamma \approx 100$m, the desired heading vector, H, will point as shown for the trajectories with large changes in heading angle, $\Delta\psi$.

Figure 2-23 shows the trajectory points $\{A, B, C, D\}$ with spacing Γ that were designed by the MILP solver. To prevent the vehicle from experiencing undesirable transients when new plans are uploaded mid-turn, the modified plan $\{A', C, D\}$ is uploaded to the vehicle. A' is selected as the straight line projection of the line $\overline{CB}$ of distance Γ, as shown in Figure 2-23. The point B is not needed and is therefore removed from the plan upload to the vehicle, however B is used as the reference to the start of the next optimization (*i.e.,* the *Horizon Point*). The modified plan, $\{A', C, D\}$, closely resembles the situation in Figure 4-5, and allows the vehicle to maintain a constant bank angle as each point in the plan is reached. Note that in the current implementation the vehicle receives new plans approximately every 4 seconds ($\Gamma = 100$m), meaning that large cross-track errors will not have time to accumulate before new plans are uploaded to the vehicle. As a result, the convergence parameter is scaled using the nonlinear scaling law for small cross-track, and for most of the MILP trajectory design the waypoint tracker will be operating in the scaled nonlinear region. This provides benefits to the performance of the system, but makes the closed loop response more difficult to predict.

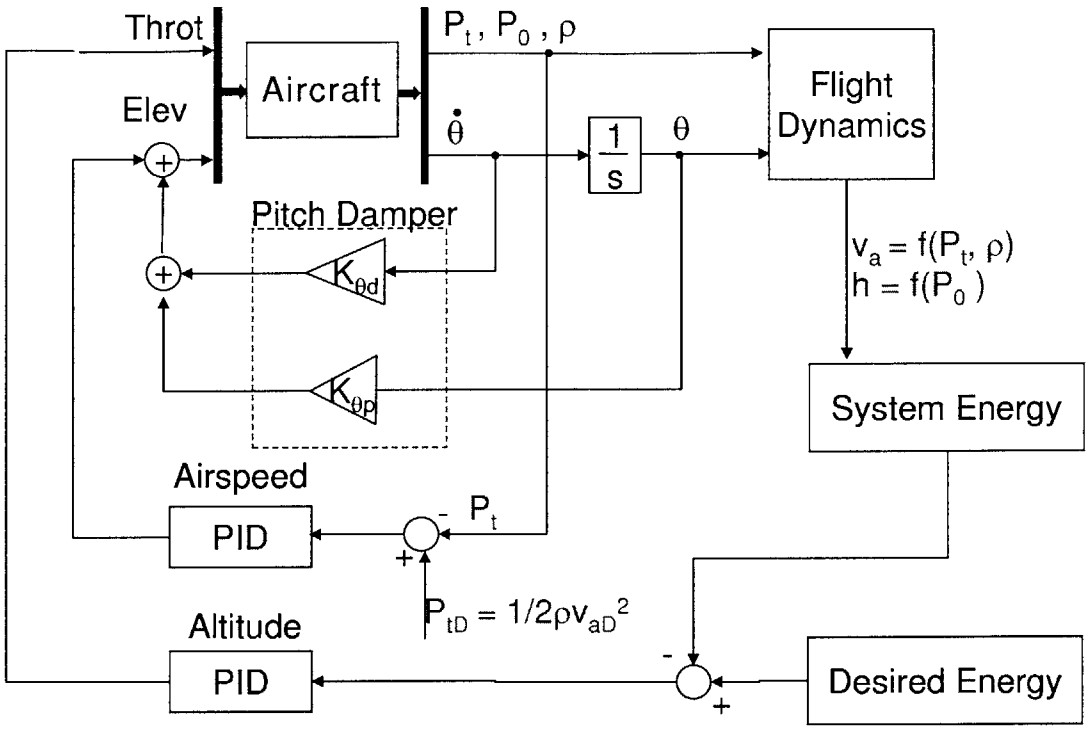


Figure 2-24: The longitudinal altitude, h, and airspeed, v_a, control loops shown for the Cloud Cap autopilot. The total and static pressure, pitch rate and air density $P_t, P_0, \dot{\theta}$, and ρ, respectively, are measured and used to determine the throttle and elevator control inputs δ_{throt} and δ_e. There is also an optional PD loop from altitude to elevator not shown here.

2.3.3 Airspeed/Altitude Control

Due to the inherent coupling between airspeed and altitude, the Cloud Cap autopilot utilizes several coupled air, pitch and altitude control loops to maintain the desired reference values. Altitude is not controlled directly, rather using an energy expression to help damp phugoid oscillations. Due to the coupling between altitude and airspeed, the tuning of the various controller settings in Figure 2-24 is an involved process requiring several iterations to ensure satisfactory performance in both variables. Later sections will utilize the reference speed commands to perform timing control of the vehicle, however the altitude will remain essentially fixed for the most flight experiments.

The true airspeed of the aircraft is estimated by measuring the total and static

pressures, P_t and P_0, of the incident airflow

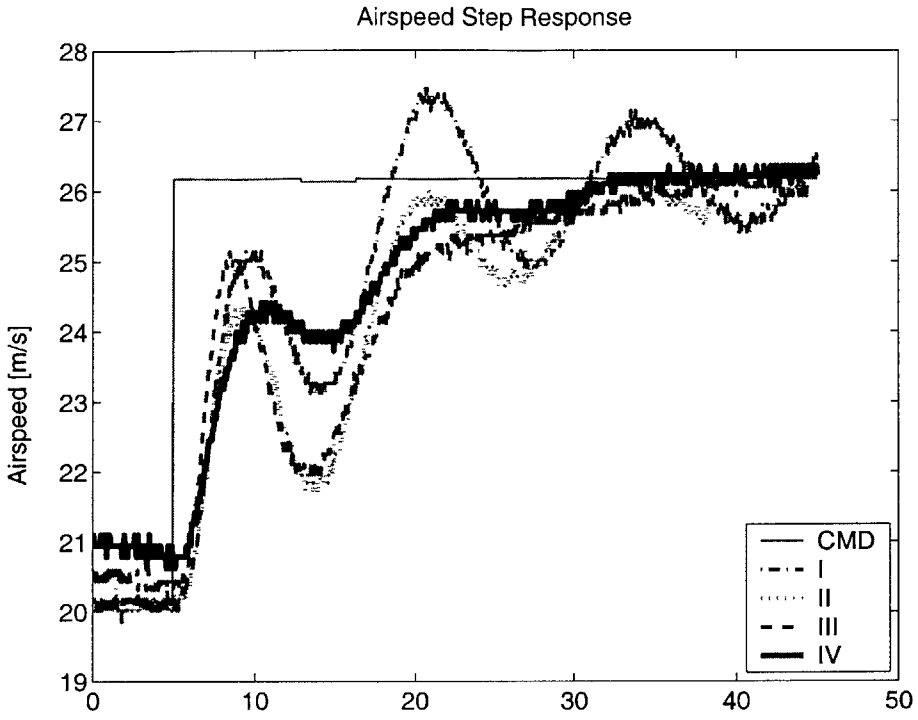
$$v_a = \sqrt{\frac{2P_t}{\rho}} \text{ and } \rho \approx \frac{P_0}{287T}$$

where ρ is the air density estimated using onboard measurements of the air temperature, T. The airspeed is regulated using PID control on the dynamic pressure error signal, $e(t) = P_t(t) - P_{tD}$, where $P_{tD} = 1/2\rho v_a^2$ is the desired total pressure corresponding to the target airspeed of the aircraft.

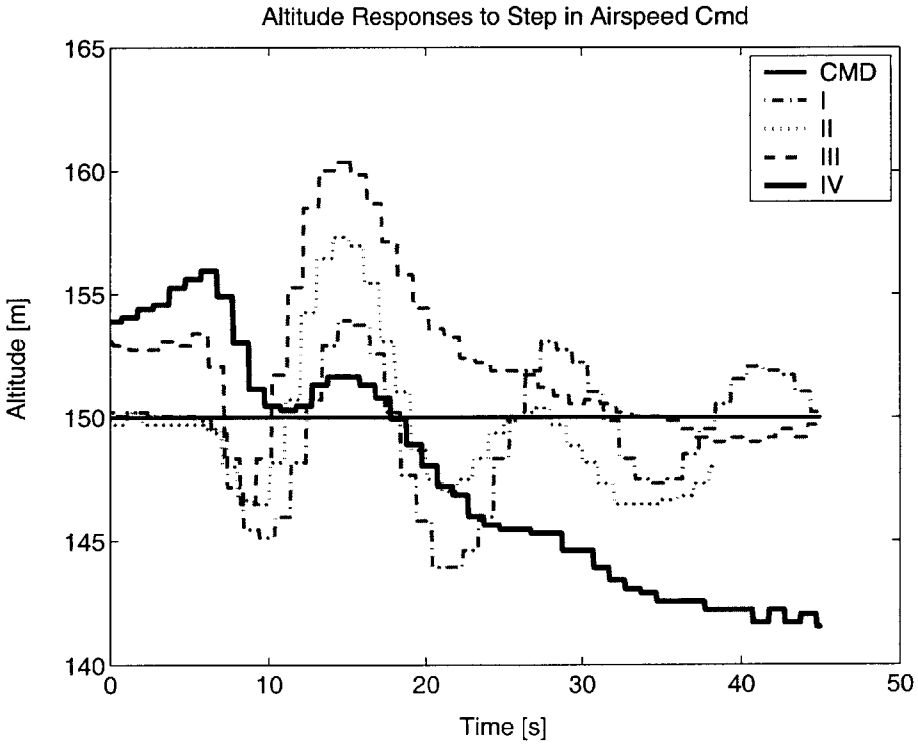
The tuning of the airspeed loop is shown in Figure 2-25, which shows the airspeed response to a step input of 6 m/s for varying cases of control selections, I, II, III, and IV. Case I shows the effect of a high proportional gain in the airspeed loop, as the elevator is excited too heavily producing undamped oscillations. The airspeed proportional gain is shown reduced by 25% in II, and setting the altitude derivative gain 25% higher in Step III reduces some of the oscillation due to throttle. Finally in step IV, the pitch damper term, $K_{\theta p}$, is used to further decrease the oscillations, and the closed loop response is shown to have a settling time of approximately 25 seconds (note that the initial step at $t = 5$ seconds). In Figure 2-25(b), the altitude variation for the same series of tests is shown to lie within ± 10 m of the nominal for these control settings. These control settings provide adequate performance for the tests required, however the entire space of options for control selection in this axis was not fully explored. It is therefore possible that higher performance solutions exist for the loops shown, and could be found with more detailed modeling of the closed loop dynamics. The closed loop response of the airspeed loop will be required in later chapters to develop an additional control loop on the relative timing of the mission plan being flown.

2.4 Conclusions

This chapter presented dynamics models for the trainer 60 aircraft which were used to validate the Cloud Cap HWIL simulation environment. By analyzing the response to specific modes of lateral and longitudinal aircraft dynamics, the reduced order



(a) Airspeed step response.



(b) The altitude response to a step change in desired airspeed, showing the coupling between the airspeed and altitude loops.

Figure 2-25: Airspeed step response input and output of the airspeed controller. Note the timescale indicates the initial step at $t = 5$ seconds.

models developed here for the ARF 60 aircraft were shown to have similar responses and characteristics to the models created from HWIL simulations. While a complete model of the flight dynamics could not be formed, it was shown that the HWIL simulator does represent certain key aspects of the dynamics in both the fast (short period, roll subsidence modes) and slow (phugoid and spiral modes) regions, indicating its suitability to represent the ARF 60 aircraft for autopilot controller tuning and high level HWIL simulations.

The dutch roll mode was not modeled due to the difficulty in collecting experimental data isolating this response. The inherent roll-yaw axis couplings are notorious for being difficult to experimentally verify, and reduced order analytical models require making gross approximations in the dynamics and aerodynamic derivatives. Had difficulties arisen in making the transitions from HWIL simulation to flight testing more detailed models could have been considered.

Section 2.3 described the Piccolo^{TM} autopilot tunings set up to be utilized in the trainer ARF 60 aircraft. The closed loop system performance has been identified for the lateral track waypoint controller, as well as the airspeed controller shown in Subsection 2.3.3, which will allow planning control loops to be closed in later sections with greater accuracy. The performance of the inner loops has been verified on repeated occasions during actual flight tests.

Chapter 3

Timing Control for Distributed Vehicle Systems

3.1 Overview of the Timing Problem

Coordination and control for UAV teams implies that the vehicles will have the capability to reach the desired positions in 3-space, and also that the timing of the mission plan will be correctly executed. These issues become a very important function for teams of UAVs, particularly for coordinated search and strike, or suppression of enemy air defense (SEAD)-type, scenarios. In these mission classes, vehicles are required to reach targets at times within a certain tolerance in order to achieve the necessary sequencing of mission goals (*e.g.*, target classification, strike and assessment). This must be accomplished within the operating limits of the vehicle and in the presence of unknown disturbances acting on the system.

Trajectory design and task assignment for small-scale aircraft becomes increasingly challenging even with relatively low speed wind disturbances. Small-scale aircraft typically have airspeeds in the range of 20-30 m/s, and commonly occurring windspeeds at altitude are roughly 5 m/s, which represent substantial fractions of the aircraft airspeed. At these disturbance levels (up to 20-30% in some cases), substantial variation will be seen in target arrival time, depending on the relative angle between air and groundspeed vectors. Figure 3-1 depicts the relationship between air

and groundspeed vectors for an arbitrary wind. As a result of the wind disturbance in Figure 3-1, the groundspeed can vary as low as $0.75V_a$ in the case of a headwind, or as much as $1.25V_a$ in the case of a tailwind. The effective time of arrival at ground locations is then determined by the groundspeed vector, and the relative timing of vehicle paths will be strongly coupled to the *disturbance magnitude ratio*,

$$\mathcal{T}_{WV_a} \equiv \frac{\| W \|}{\| V_a \|} \tag{3.1}$$

In addition to magnitude variation the relative wind also impacts the relative heading error, $|\psi_g - \psi|$, which is a strong function of the magnitude ratio between air and groundspeed. The relative heading angle is plotted in Figure 3-2 as a function of β (as defined in Figure 3-1) and the disturbance magnitude ratio, $\mathcal{T}_{WV_a}$. From these curves the worst case heading error can be seen for varying disturbance levels acting on the aircraft. Heading angle errors become important in the next chapter when dealing with position and velocity feedback, however Figures 3-1 and 3-2 show that the disturbance magnitude ratio is the one of the most significant factors for predicting performance for aircraft of this class. With other aircraft types operating at potentially higher airspeeds, wind disturbance will have less effect on the timing of the mission. However in the case of the trainer ARF aircraft with $\mathcal{T}_{WV_a} \approx 0.25$, to ensure that proper trajectory design and task assignment was accomplished, more sophisticated compensation techniques were required.

Physical limits on the maximum throttle setting and the minimum airspeed required to keep the aircraft aloft produce saturation limits on the allowable aircraft airspeeds, which limit how much the groundspeed can be varied using simple airspeed control. While low level speed control can be included to mitigate some of the effects of wind disturbance, a typical small-scale aircraft only has enough airspeed range to compensate for very slight wind conditions. As a result, feedback on the task assignment and trajectory design levels are necessary to guarantee that the timing and plans made are consistent with the vehicle capabilities. However, including disturbances in the higher levels requires vehicle level estimation of the current wind conditions, and this estimation process will undoubtedly have some error associated with it. Each of

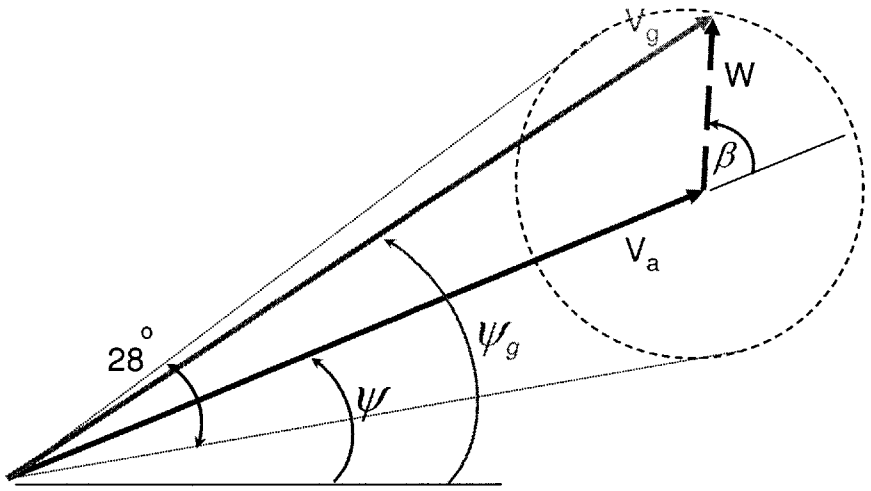


Figure 3-1: Graphical relation between the groundspeed and airspeed vectors with an arbitrary wind vector shown by the dashed arrow and $\| W \| = 0.25 \| V_a \|$. Assuming coordinated turns, the maximum heading difference between the groundspeed heading angle, ψ_g, and the true azimuth angle, ψ, is shown to be ± 14 degrees.

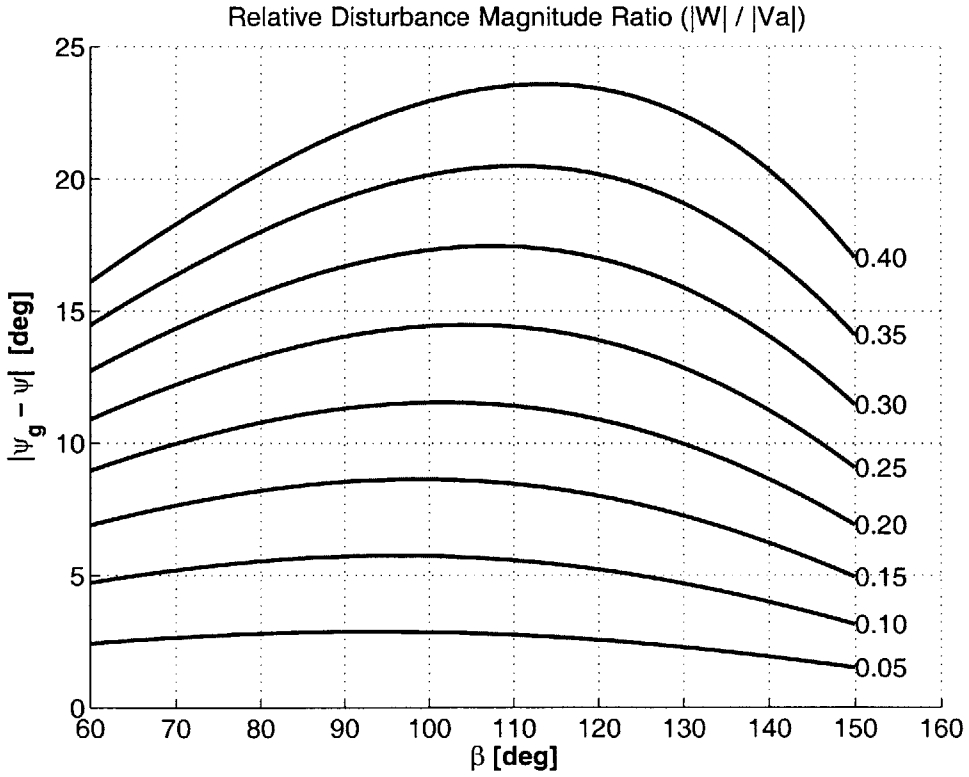


Figure 3-2: The relative disturbance magnitude ratio $\mathcal{T}_{WV_a}$ determines the heading angle error, $|\psi_g - \psi|$, as a function of β, as defined in Figure 3-1. For wind disturbances small compared to the vehicle airspeed, the maximum heading angle errors remain small.

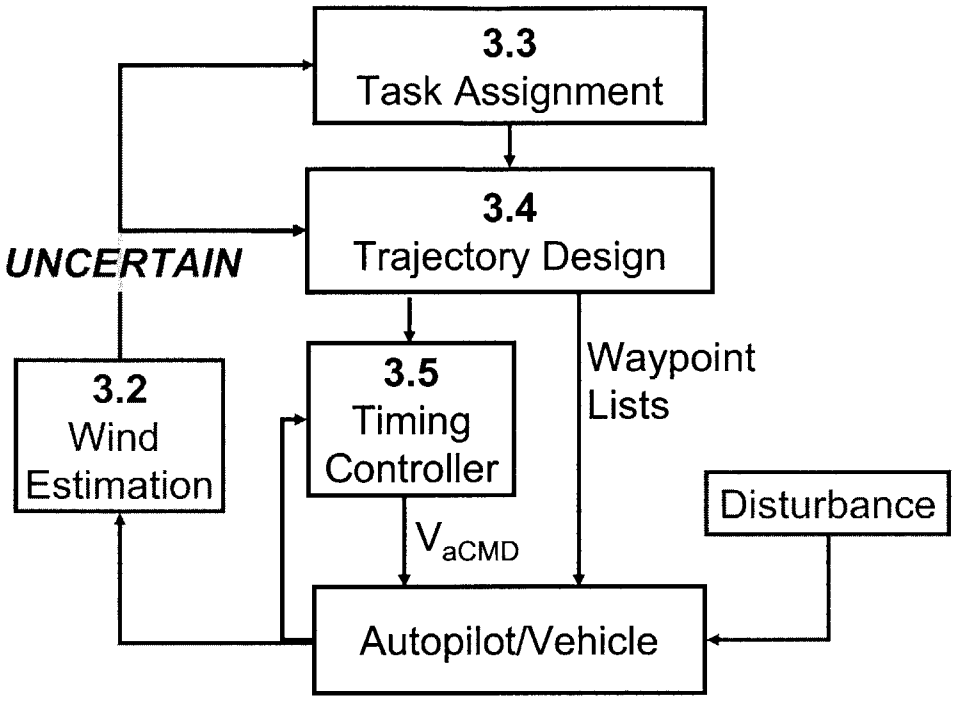


Figure 3-3: Overview of timing control scheme with planning system in the loop. Each level handles the input disturbances differently to make guaranteeing UAV arrival time at targets a multilevel estimation and control process with embedded uncertainty.

the task assignment, trajectory design and autopilot (vehicle) levels will be affected by this uncertainty in different ways, as such the process of guaranteeing the arrival time of UAVs at targets becomes a multi-level task of estimation and control with embedded uncertainty.

Figure 3-3 depicts the chapter layout for the remaining sections. Section 3.2 describes some of the methods in place on the vehicle testbed to perform static wind estimation. Section 3.3 discusses how uncertainty in the wind estimate can impact the task assignment algorithm as one of many sources of error, including the methods used to mitigate assignments being made that are inconsistent with vehicle flying capabilities. Section 3.4 describes the trajectory design algorithm with added input disturbance terms to compensate for the wind acting on the vehicle system. Finally, Section 3.5 describes the development of a *timing controller* to autonomously vary the vehicle reference airspeed between the minimum/maximum saturation limits to

compensate for timing errors in the execution of the optimum plan. The operation of the trajectory design and timing control loop is demonstrated on the HWIL simulator as well as in actual flight tests.

3.1.1 Chapter Definitions

As seen earlier in Subsection 2.1.4, wind can be modeled using the Dryden wind model as an input disturbance with both static and turbulent components,

$$W = \overline{W} + \delta W \tag{3.2}$$

where $\overline{W}$ is a slowly changing average value capturing the gross motion of the atmosphere, and δW captures the turbulence experienced by the aircraft as a function of reference speed, V_{ref}, turbulence scale length, L_w, and intensity, σ_w. The Dryden model for turbulence uses the second order model $W(s)$ to capture the correct spectral content for δW[28]

$$W(s) \triangleq \frac{\delta W(s)}{w_1(s)} = \sigma_w \sqrt{\frac{L_w}{2\pi V_{ref}}} \frac{(1+\sqrt{3}\frac{L_w}{V_{ref}}s)}{(1+\frac{L_w}{V_{ref}}s)^2}; \qquad w_1 = \mathcal{N}(0,1) \tag{3.3}$$

Define the wind estimate, $\widehat{W}$, which is estimated using the algorithms presented in Section 3.2, with some bounded error, $\widetilde{W}$,

$$\widehat{W} \equiv W - \widetilde{W} \tag{3.4}$$

where the estimation error is assumed to be bounded by a fixed set, $\mathcal{W}_b$,

$$\|\widetilde{W}\| \in \mathcal{W}_b \tag{3.5}$$

Note that for the remainder of this thesis, the wind vector is defined as pointing with the prevailing airflow, however the wind is called from the direction it comes. Hence, the wind vector $(0,1,0)$ in the $\mathcal{E}$-$\mathcal{N}$-$\mathcal{U}$ coordinate system is called a *southern wind.*

3.2 Static Wind Estimation

The assumptions about when and how a vehicle arrives at a target location are largely dependent on the best estimate of the static wind component, $\overline{W}$, which will be shown

to be one of the limiting factors in the design of trajectories for small-scale aircraft due to the relatively slow airspeeds being flown. Because trajectory planning is done in the inertially fixed $\mathcal{E}$-$\mathcal{N}$-$\mathcal{U}$ coordinate system, while the dynamics are subject to the disturbance from W, assumptions about time of flight using the vehicle groundspeed vector, V_g, depend on W and the wind-relative velocity vector, V_a

$$E\{V_g\} = E\{V_a + W\} = E\{V_a\} + \overline{W} \tag{3.6}$$

$$E\{V_a\} = \widehat{A}_s[\cos(\hat{\psi})\ \sin(\hat{\psi})]^T \tag{3.7}$$

Figure 3-1 depicts Eq. 3.6 graphically for the case where the wind vector magnitude is 25% of the airspeed, a regular flight condition for small-scale aircraft. V_g can be measured with a high degree of accuracy using GPS [37] and for negligible sideslip motion (*i.e.,* coordinated turns), V_a can also be explicitly written using the airspeed magnitude, A_s, and ψ, the true aircraft azimuth angle measured from an inertially fixed axis as shown in Eq. 3.7.

There are several means available to estimate the static wind component, $\overline{W}$. The first is through repeated ground measurements using a digital weather vane or other similar device. This provides good estimates of the wind near the ground, but not necessarily at flight altitudes (100 - 500 m). Mounting the sensor on a 6 m pole would provide some indication of the general trends, and these estimates, in conjunction with local weather forecasts can be used to validate other methods.

As noted previously, the GPS groundspeed vector, V_g, is very accurately known, and an estimate of the aircraft airspeed is available from pressure and temperature measurements, $\widehat{A}_s \approx \| V_a \|$. Figure 3-4 shows the ambiguity that exists in finding $\overline{W}$ at a given instant, however if $\overline{W}$ can be assumed to be static over the course of a maneuver, and if there is sufficient variation in the groundspeed heading angle, ψ_g, the static wind estimate can be determined through least squares estimation. Cloud Cap Technologies has implemented a maneuver based (S-Turn) wind finding algorithm based on this concept [38], and these estimates are available to be used at the planning level. Errors using this method depend on the airspeed measurement errors and how frequently S-Turns are made. Hardware-in-the-loop testing as well

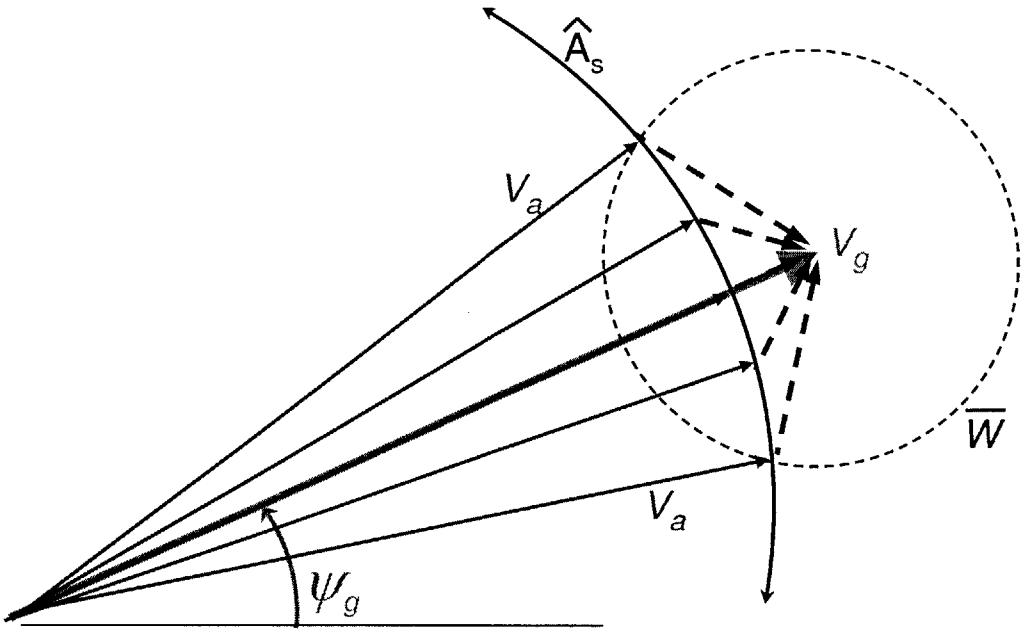


Figure 3-4: Ambiguity of static wind estimate given the GPS groundspeed velocity vector, V_g, and an airspeed estimate, $\widehat{A}_s = ||V_a||$. The dashed lines show possible solutions for the case $\mathcal{T}_{WV_a} \leq 0.25$.

as flight experimentation has shown that this method can provide estimates with up to 90% accuracy (as validated using the Cloud Cap HWIL simulator) in magnitude and direction for static values changing slowly on the order of 1 minute, but if there are significant variations in magnitude and/or direction, as seen on particularly gusty days, this estimation scheme is less robust.

From Figure 3-1, note that if both vectors V_g and V_a are independently measured, $\overline{W}$ can be determined explicitly. The vector V_g is well known through GPS measurements, provided the aircraft sideslip motion is negligible (*i.e.,* assuming coordinated turns), and V_a can be estimated using Eq. 3.7, the airspeed estimate, $\widehat{A}_s$, and an estimate of the true aircraft heading, $\hat{\psi}$. The true heading must be estimated using another measurement device, such as an onboard magnetometer, but for more rapid updates and proper filtering of the $\overline{W}$ estimate. This extra sensor information can

be used to resolve the ambiguity in Figure 3-4,

$$\widehat{W} = V_g - \widehat{V}_a = V_g - \widehat{A}_s \cos(\hat{\theta}) \begin{bmatrix} \cos(\hat{\psi}) \\ \sin(\hat{\psi}) \end{bmatrix} \tag{3.8}$$

where the $\cos(\hat{\theta})$ term is applied using the pitch angle estimate, $\hat{\theta}$ to correctly scale the airspeed to represent the horizontal component of velocity.

As a result of Eq. 3.8, the wind estimate, $\widehat{W}$, will be a function of the measurement uncertainty from GPS (negligible), the airspeed measurement errors, $\tilde{A}_s \equiv |A_s - \widehat{A}_s|$, the errors in the pitch angle estimate, $\tilde{\theta} \equiv |\theta - \hat{\theta}|$, and the magnetometer heading angle error, $\tilde{\psi} \equiv |\psi - \hat{\psi}|$. The groundspeed measurement is assumed to be perfect from GPS measurements ($\widetilde{V}_g \approx 0$), so the wind estimation error, $\widetilde{W} \equiv |W - \widehat{W}|$, can therefore be approximated using the geometry in Figure 3-5, which depicts the worst-case airspeed estimation error, $\widetilde{V}_a$, for uncertainties in the true heading, $\tilde{\psi}$, and the airspeed estimate, $\widetilde{A}_s$. Assuming the pitch angle, θ, and pitch angle estimation error, $\tilde{\theta}$, in Eq. 3.8 are both zero, the uncertainty in the magnitude of the airspeed error is bounded by

$$\| \widetilde{V}_a \| < \sqrt{\widetilde{A}_s^2 + \tilde{\psi}^2 (\| V_a \| + \widetilde{A}_s)^2} \tag{3.9}$$

for small heading angle errors, $\tilde{\psi}$. As a result of Eq. 3.9, bounds on the estimation error for $\widetilde{V}_a$ (and thus $\widetilde{W}$) can be computed as a function of manufacturer-specified measurement errors for the measurement devices. Note that the airspeed is estimated using pressure and temperature measurements as shown in Subsection 2.3.3, and the true heading can be measured using an onboard magnetometer. For the sensors onboard the testbed aircraft, $\widetilde{A}_s \approx 0.75$ m/s and $\tilde{\psi} \approx 3°$ [39], allowing the estimation error bound to be computed from Eq. 3.9 as

$$\mathcal{W}_b \approx 1.5 \text{ m/s}; \qquad \| V_a \| = 25 \text{ m/s} \tag{3.10}$$

This method may provide more accurate estimates, however it begins to blur the distinction between $\overline{W}$ and δW in Eq. 3.2. Different filters designed for the different update rates on each of the task assignment, path planning and vehicle levels could also be used to find suitable estimates for each level. With bounds established on the

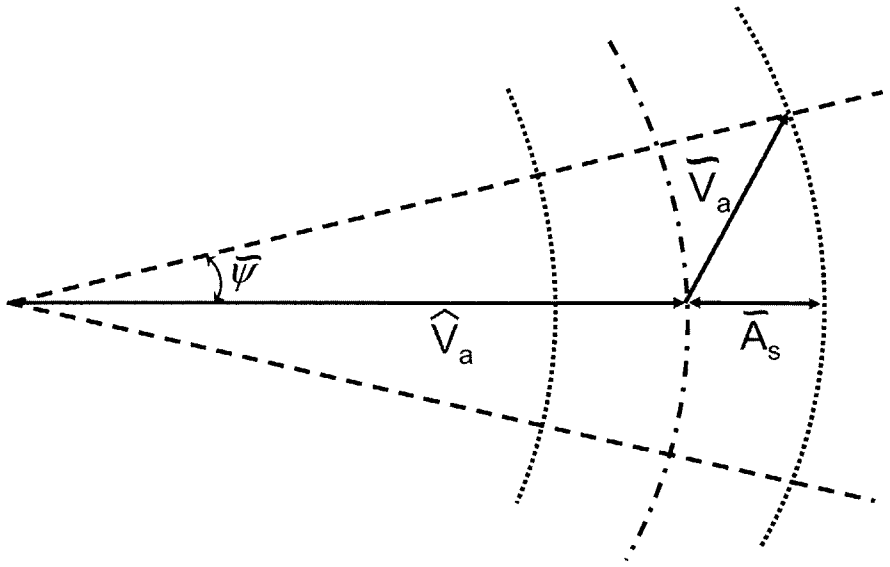


Figure 3-5: The worst-case true airspeed error, $\widetilde{V}_a$, is estimated given (small) uncertainties in the true heading measurement, $\tilde{\psi}$, and the true airspeed, $\widetilde{A}_s$. The heading angle and airspeed errors shown are not to scale for the typical measurement uncertainties encountered here.

type of estimation errors that can be expected, the task assignment algorithm can be extended to account for the uncertainty in the wind estimation error, as shown in Section 3.3.

3.3 Robust Task Assignment with a Steady-State Wind

The allocation of tasks to fleets of UAVs has been addressed in several sources [5, 15], including the formulation of *robust task assignment* [4], which looks at the assignment problem with uncertainty entering through the UAV target costs. Of course there are multiple sources of uncertainty in the assignment problem and they all require consideration in a truly robust problem formulation. As shown in Figure 3-3, this section investigates the effect of an uncertain wind vector applied to finding the robust task assignments, which also relates to work done to find the *robust shortest*

path [12].

Section 3.2 describes several methods for estimating the wind vector, $\widehat{W}$, however there will also be errors associated with this vector, $\widetilde{W}$. A uniform and symmetric upper bound on the magnitude of the wind estimation error is assumed,

$$\| \widetilde{W} \| \leq \mathcal{W}_b$$

where the uncertainty bounds, $\mathcal{W}_b$, can be determined by the expected performance of the wind estimation schemes in Section 3.2. The following subsections highlight the robust task assignment algorithm for the case of steady state winds by the following means:

- The static wind estimate, $\widehat{W}$, determines the vehicle times of arrival at the targets in each permutation using the distance cost map.
- The worst case static error, $\mathcal{W}_b$, is used to compute bounds on the scores obtained on each of the solution permutations, which is analogous to the situation described in Ref. [4] for task assignment with robustness to uncertain target values.
- The *loiter times* presented in Ref. [15] are used to calculate a feasible reference velocity, V_{ref}, which is consistent with the vehicle capabilities and used on the trajectory design level to design the path the vehicle will follow.

In order to estimate the flight times between nodes in the graph, the timing assignment algorithm assumes the vehicle will be flying at a "maximum" reference speed, V_{maxTA}, which can be adjusted to a lower reference speed, V_{ref}, based on the loitering time calculation as shown in Ref. [4] and Subsection 3.3.3. Note V_{MaxTA} should not be confused with the upper saturation limit on the vehicle airspeed, V_{MaxVeh}, for reasons to be discussed in Subsection 3.3.3.

3.3.1 Flight Time Computation

For small-scale aircraft flying at airspeeds with high disturbance magnitude ratios, $\mathcal{T}_{WV_a} > 0.10$, the flight times between the nodes of the graph will depend on the

relative wind acting on the system, as well as the task assignment reference velocity, V_{maxTA}. The first step in the task assignment algorithm is finding the visibility graph between the UAV positions, waypoints and obstacle vertices. This determines a node map consisting of distances between all the points in the graph, given known obstacle and target locations. The times of flight between nodes will depend on the relative wind, and these times can be computed for each distance segment in the graph. For a set of N_n nodes in the graph, define the distance vector, d_{ij}, as the vector from node i to j. The wind-scaled flight times from the visibility graph are then

$$t_{ij} = \frac{1}{V_{maxTA}} \left(\| d_{ij} \| - \frac{d_{ij} \cdot \widehat{W}}{V_{maxTA}} \right) \tag{3.11}$$

where the dot product $d_{ij} \cdot \widehat{W}$ determines the relative scaling direction and magnitude. The flight time from one waypoint to the next is then found by summing the flight times for each segment in the waypoint list (around obstacles, etc.),

$$t_k = \sum t_{ij} \qquad \forall_{i,j} \in k \tag{3.12}$$

Once the visibility graph has been formed and the flight times determined, the algorithm proceeds through the remaining three phases [8]: I) The cost calculation II) planning and pruning and III) task assignment. The cost calculation consists of calculating the shortest feasible paths between all the updated nodes using Dijkstra's algorithm [8], and then calculating the cost map based on the time-discounted scores for each path. The planning and pruning algorithm remains essentially unchanged, removing those nodes which are unlikely to be optimal solutions [8]. Finally, the robust task assignment proceeds as shown in Subsection 3.3.2.

3.3.2 Robust Task Assignment with Uncertain Winds

Due to the errors in the static wind estimate, $\widetilde{W}$, there will be some uncertainty in the times of arrival of each UAV at their targets. This can become problematic if tight timing constraints need to be enforced and there is no allowance for the errors that could be expected. The robust task assignment factors in extra margin into the

assignments based on the expected wind uncertainty $\widetilde{W}$. Generally speaking, this means that target lists with lower average scores and low uncertainty in the scores will be selected over lists with high average scores and higher uncertainty, although the actual assignments depend on many factors. As shown in Ref. [4], the upper and lower bounds on the expected score can be used to determine robust assignments for vehicles in an uncertain environment, where in this case the uncertainty is caused by the static wind estimate, $\widehat{W}$. The time-weighted mission score for each permutation, p, is determined by [5, 4],

$$S_p = \sum C_k \lambda^{-t_k} \qquad k = 1 \ldots N_W \tag{3.13}$$

where C_k is the expected target value, λ is a weighting parameter on the time discount factor, t_k is the time to reach the target and N_W is the number of waypoints in permutation p.

Given set values for the targets visited in each permutation, upper and lower bounds on the score S_p can be calculated using minimum and maximum completion times for each segment, assuming the best and worst case disturbance acting during each interval

$$\bar{t}_{ij} = \frac{1}{V_{maxTA} - \| \widetilde{W} \|} \left(\| d_{ij} \| - \frac{d_{ij} \cdot \widehat{W}}{V_{maxTA}} \right) \tag{3.14}$$

$$\underline{t}_{ij} = \frac{1}{V_{maxTA} + \| \widetilde{W} \|} \left(\| d_{ij} \| - \frac{d_{ij} \cdot \widehat{W}}{V_{maxTA}} \right) \tag{3.15}$$

where the "best" case causes the vehicle to arrive at the target early, $\underline{t}$, and the "worst" case causes the vehicle to arrive late, $\bar{t}$. These definitions were chosen because in a practical sense the vehicle can still physically reach the target on time if a tailwind is encountered by flying a longer path, but if an unexpected headwind is encountered, the assignment already assumes the vehicle is flying at its maximum airspeed, so the plan will be infeasible. Using the minimum/maximum times of flight for each segment in the path allows computation of the corresponding values for the maximum/minimum score values in each permutation, $\overline{S_p}$ and $\underline{S_p}$, respectively

$$\underline{S_p} = \sum C_k \lambda^{-\bar{t}_k} \tag{3.16}$$

$$\overline{S_p} = \sum C_k \lambda^{-\underline{t}_k} \qquad k = 1 \ldots N_W \tag{3.17}$$

Note that the maximum score corresponds to the minimum time completion and vice versa.

The bounds on time weighted target score obtained here prescribe an interval, $(\underline{S_p}, \overline{S_p})$, for which the robust task assignment algorithm can use to take into account the uncertainty in the wind vector, $\widetilde{W}$, using the approach in Ref. [4]. In contrast to assigning tasks based on target score uncertainty, $\widetilde{C}$, the approximate path distance from the vehicle to the target enters the cost calculation through the $\bar{t}_{ij}, \underline{t}_{ij}$ terms in Eqns. 3.14 and 3.15. Since the time to fly this distance is uncertain, the interval $(\underline{S_p}, \overline{S_p})$ will become larger as the target distance from the vehicle increases. The result is that for a given bound on the wind estimation error, there will be growing score uncertainty with target distance from the vehicle. The approach here allows the task assignment algorithm to be cognizant of this error and incorporate $\widetilde{W}$ uncertainty in the assignments. The effect of this type of robustness is yet to be quantified, however as shown in similar examples in Ref. [4], the results can be very scenario-dependant.

3.3.3 Reference Velocity Calculation

For tightly coupled timing constraints Ref. [15] presents an algorithm for task assignment using continuous decision variables for assigning *loitering times*, $\mathcal{L}_k$, to the vehicle at each waypoint, W_{p_k}. While the loitering times could be interpreted as the amount of time a vehicle stops moving at a given waypoint, for an actual aircraft system with minimum/maximum speed constraints the loiter times represent the time loss due to a reduction in speed from the maximum task assignment reference velocity,

$$\mathcal{L}_k = t_k \left(\frac{V_{maxTA} - V_{ref}}{V_{ref}} \right) \tag{3.18}$$

where V_{ref} is the reference velocity used on the path planning level to design the vehicle trajectory and t_k is the total flight time (around obstacles, etc.) from waypoint $k-1$ to waypoint k using Eqns. 3.11 and 3.12.

The task assignment reference velocity, V_{maxTA}, should be selected such that it is strictly less than the maximum vehicle airspeed, V_{maxVeh}. This allows some authority to remain at the low level controllers such that they will not be pushed to their

saturation limits and can still correct for unexpected errors. In order to guarantee that the vehicle will have enough authority to completely reject the static wind error, $\widetilde{W}$, the task assignment minimum/maximum velocities should be set such that the margin above/below the vehicle speed bounds are

$$V_{minTA} = V_{minVeh} + \mathcal{W}_b \tag{3.19}$$

$$V_{maxTA} = V_{maxVeh} - \mathcal{W}_b \tag{3.20}$$

which is depicted graphically in Figure 3-6. The task assignment algorithm nominally assumes the maximum speed bound, V_{maxTA}, for computing the flight times between waypoints and then subsequently assigns loiter times to help in satisfying the timing constraints [15]. As a result, the "worst case" loiter time corresponds to the situation when the slowest possible velocity is flown between waypoints

$$V^*_{ref} = V_{minTA} \tag{3.21}$$

Replacing V^*_{ref} in Eq. 3.18 and substituting for the minimum/maximum task assignment setpoints V_{minTA}, V_{maxTA} from Eqns. 3.19 and 3.20, the corresponding constraints on the loiter time are

$$0 < \mathcal{L}_k < t_k \left(\frac{V_{maxVeh} - V_{minVeh} - 2\mathcal{W}_b}{V_{minVeh} + \mathcal{W}_b} \right) \qquad \forall k \tag{3.22}$$

which can be easily implemented in a linear program. Solving for the planner reference velocity from Eq. 3.18 yields,

$$V_{ref} = \frac{t_k V_{maxTA}}{t_k + \mathcal{L}_k} \tag{3.23}$$

In the event that the wind estimation errors are larger than the interval $V_{maxVeh} - V_0$, where $V_0 = (V_{minVeh} + V_{maxVeh})/2$, the constraint in Eq. 3.22 will become active removing the loiter times, and setting $V_{maxVeh} = V_0$. The effect of the speed bounds $\{V_{minVeh}, V_{maxVeh}\}$ in performing autonomous speed control, are discussed further detail in Section 3.5.

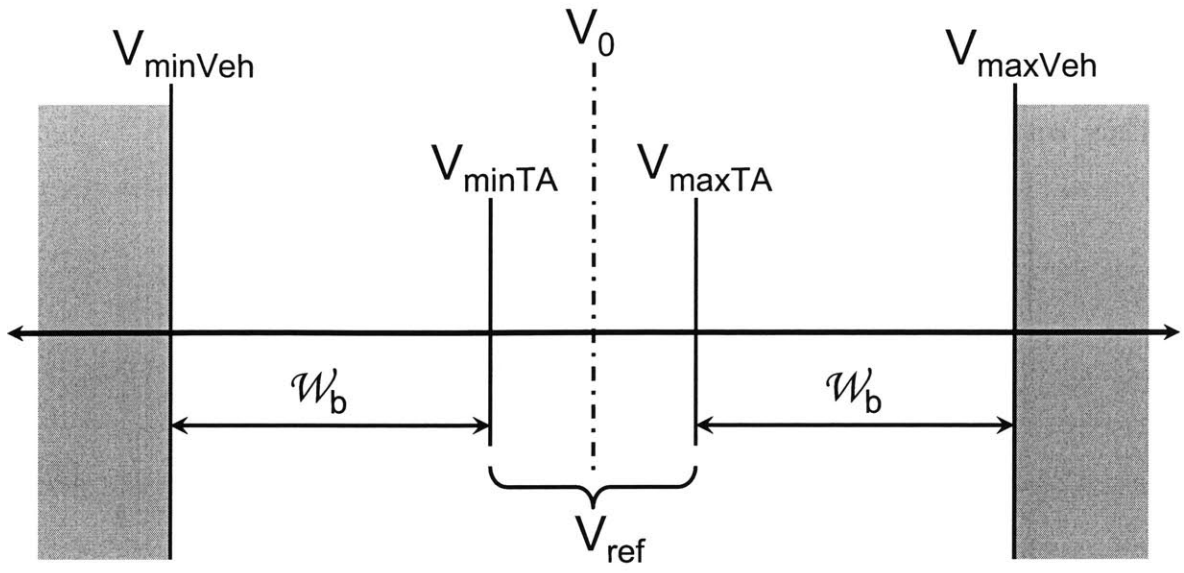


Figure 3-6: The bounds on the trajectory design reference velocity, V_{ref}, are set according to the vehicle minimum/maximum vehicle speeds V_{minVeh}, V_{maxVeh} and the expected estimation errors, $\mathcal{W}_b$.

3.4 Trajectory Planning with Static Wind Disturbance

The trajectory planner designs the path the vehicle will follow to each of the targets designated by the task assignment algorithm. Given a set of waypoints to visit and a feasible reference airspeed V_{ref}, the path planner will design a minimum time path to each of the target locations. This section describes how the input disturbance $\widehat{W}$ is used on the planning level to keep the paths that are designed consistent with the expected response of the vehicle. The wind estimation error, $\widetilde{W}$, also has a large effect on the trajectory design algorithm, however it is studied in more detail in Chapter 4.

As shown in [8], aircraft dynamics can expressed as a simple point mass with distance and velocity state variables, $X = [x\ y\ V_{ax}\ V_{ay}]^T$, and acceleration control inputs, $u = [a_x\ a_y]^T$. The static wind estimate is resolved into its components to provide the disturbance input to the system $\widehat{W} = [w_x\ w_y]^T$, and it enters the dynamics

through the disturbance input matrix, G,

$$X_{k+1} = AX_k + Bu_k + G\widehat{W} \tag{3.24}$$

Thus, with a zero-order hold the discrete time system becomes

$$\begin{bmatrix} x \\ y \\ V_{ax} \\ V_{ay} \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ V_{ax} \\ V_{ay} \end{bmatrix}_k + \begin{bmatrix} (\Delta t)^2/2 & 0 \\ 0 & (\Delta t)^2/2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_x \\ a_y \end{bmatrix}_k + \begin{bmatrix} \Delta t & 0 \\ 0 & \Delta t \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \end{bmatrix}_k \tag{3.25}$$

where k is the time step and Δt is the planning time interval between discrete MILP trajectory points. Note that the control input $[a_x\ a_y]_k{}^T$ is assumed to be constant over each time interval Δt.

As seen from Eq. 3.25, the MILP-based trajectories consist of position and velocity states at discrete times $\{t_{p_k}, t_{p_{k+1}}, \ldots, t_{p_{k+n}}\}$, but minimum/maximum speed constraints are also included to keep the plan velocity at an approximately constant reference value [8], corresponding to the desired airspeed for the vehicle, $A_s \approx V_{ref}$.

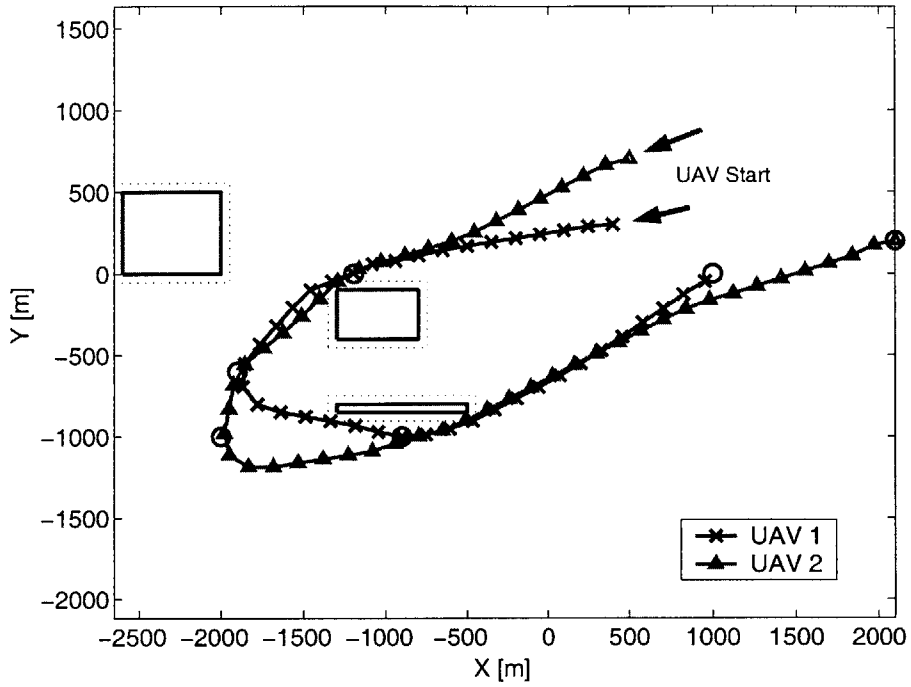
Maximum acceleration constraints on the control inputs, $[a_x\ a_y]_k{}^T$, are also imposed to effectively limit the minimum turn radius of the aircraft to a value consistent with the actual vehicle dynamics, however Section 4.2 discusses the need for added dynamics to account for the bank angle of the aircraft in the acceleration command. While the dynamics, speed and turn radius constraints are all defined with respect to the wind-relative reference frame, the added input disturbance effectively allows the planner to design trajectories for a vehicle operating in the inertially fixed ground-frame. Since target locations and waypoints specified to the autopilot are both referenced in the ground-frame this is an important conversion for dynamically feasible trajectory design.

Figure 3-7(a) and 3-7(b) show the effect of a westerly wind input disturbance in two vehicle scenario with $\mathcal{T}_{WV_a} = 0.25$. The vehicles are assigned targets to visit

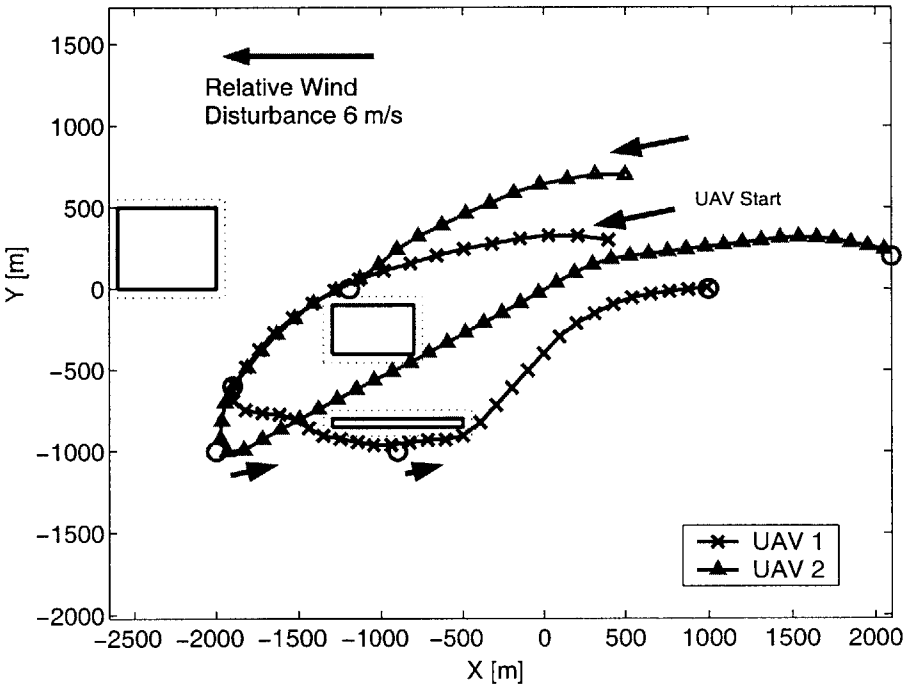
(indicated by circles) in an obstacle field, with both vehicles starting in the upper right part of the figure. The differences in the trajectories in (a) and (b) are due to the influence of the wind on the vehicle dynamics and in both cases the vehicles are shown to navigate the obstacle field in the presence of the disturbance $\overline{W}$ to each of the designated targets. In Figure 3-7(b) the disturbance alters the plans to have variable groundspeed, which can be seen in the trajectory spacing, since a constant time interval is assumed between each of the points in the plan. The plans also have variable turn radius as a function of the aircraft heading relative to $\overline{W}$, as expected. With the overall motion of the atmosphere taken into account in $\overline{W}$, trajectories designed using the updated disturbance dynamics, Eq. 3.24, will be approximately consistent with the ground motion of the vehicle, and low level control is used to remove the integrated effect of the turbulence, δW, as well as any errors, $\widetilde{W}$, on the plan timing.

3.5 LQG Timing Control for Aircraft in Uncertain Winds

Referring to Figure 3-3, the output of the wind estimation algorithm, $\widehat{W}$, is used on the planning level to design dynamically feasible trajectories that are consistent with the capabilities of the UAV. If $\widehat{W}$ is not well known, or there are significant deviations about $\widehat{W}$ due to the turbulent component δW, the vehicle will nevertheless accumulate arrival time errors. The approach proposed here is to include mid-level speed control feedback that issues reference airspeed commands, V_{aCMD}, to the autopilot to compensate for both of these effects. The airspeed commands will be bounded from above and below by saturation limits due to the maximum engine throttle setting and minimum airspeed needed to keep the aircraft aloft. The purpose for using airspeed control in this fashion is to maintain the relative timing of the mission so that vehicles not only follow the reference trajectories, but also arrive at target locations at the anticipated times.



(a) Nominal plan solution without wind disturbance, $V_{ref} = 24$ m/s



(b) 6 m/s Easterly wind disturbance, $V_{ref} = 24$ m/s, showing sensitivity of plan to $\overline{W}$

Figure 3-7: Disturbances acting on the planning level influence the optimal planned paths and create dynamically feasible plans. With $\mathcal{T}_{WV_a} = 0.25$, the sensitivity of the plan to $\overline{W}$ becomes significant. Note the spacing of the plan points reflect constant time step, Δt, although the ground velocity is not constant when wind acts on the system. Goal locations are indicated by circles.

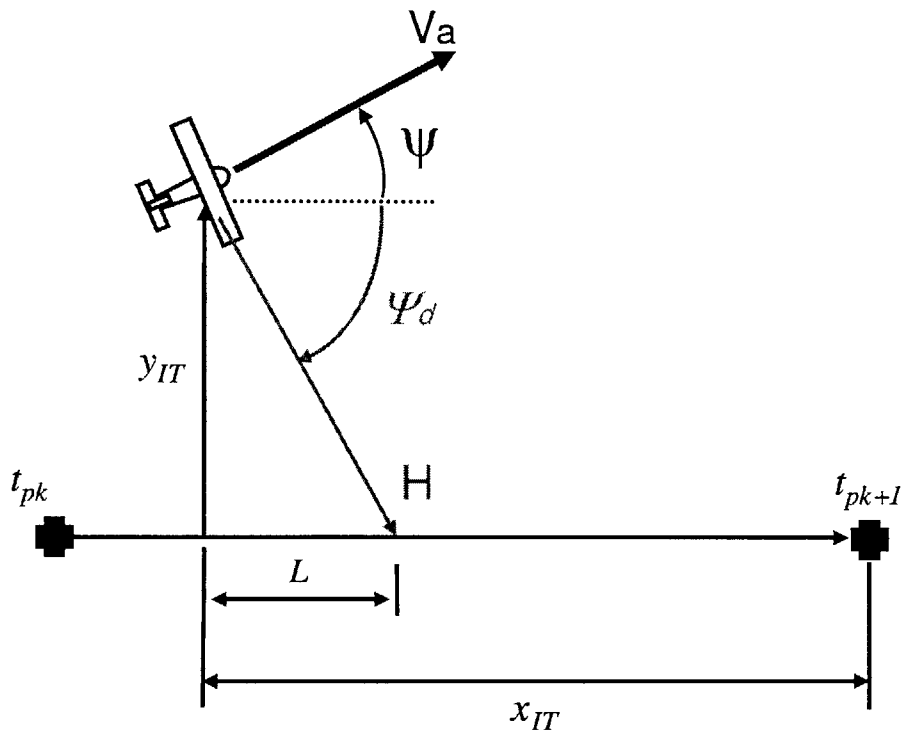


Figure 3-8: Autopilot Tracker Geometry

3.5.1 Timing Dynamics

The discrete model assumes a constant time interval, Δt, between each of the MILP trajectory states $\{X_k, X_{k+1}, \ldots, X_{k+n}\}$, as such the *planning times* $\{t_{p_k}, t_{p_{k+1}}, \ldots, t_{p_{k+n}}\}$, are a natural selection for timing reference. The GPS clock time can be used to find the *actual times of arrival* at targets $\{\tau_k, \tau_{k+1}, \ldots, \tau_{k+n}\}$, which will differ from the planning times due to modeling errors and the input turbulence, δW, present.

Define the *timing error* as the deviation from the reference time used to design MILP trajectories,

$$e_{t_k} \triangleq t_{p_k} - \tau_k \qquad \forall k \in \{0, \ldots, n\} \tag{3.26}$$

As shown in Figure 3-8, the timing error, e_t, can be measured at the GPS update rate (1 Hz) as a function of the along track distance, x_{IT}, to interpolate between discrete MILP points.

$$e_t = t_{p_{k+1}} - \left(\tau + \frac{x_{IT}}{\| V_g \|}\right) \tag{3.27}$$

Eq. 3.27 assumes a small variation between the actual and planned aircraft heading ($\phi_g \approx \phi_{ref}$), and that the groundspeed magnitude is obtainable from GPS measurements. It is a measurement of the current timing error based on the estimated time of arrival at state X_{k+1}. To convert the timing error to a distance error in meters, it is simply scaled by the vehicle groundspeed magnitude,

$$\Delta d = \| V_g \| e_t = \| V_g \| \, (t_{p_{k+1}} - \tau) - x_{IT} \tag{3.28}$$

The vehicle velocity error, Δv, is the difference between the planning reference velocity and the true airspeed of the aircraft, which can be directly measured at 1 Hz using measurements of the aircraft airspeed, $\widehat{A}_s$,

$$\Delta v = V_{ref} - \widehat{A}_s \tag{3.29}$$

Eq. 3.29 assumes that the aircraft altitude is essentially constant such that airspeed measurements contain small contributions in the vertical direction. The distance error, Δd, is then the integral of the velocity error, Δv, plus a disturbance input δW which captures the expected turbulence levels for the aircraft. Or in differential form,

$$\Delta \dot{d} = \Delta v + \delta W \tag{3.30}$$

δW is modeled using the Dryden transfer function $W(s)$ in Eq. 3.3, which is appropriately scaled for the vehicle reference airspeed, V_{ref}, scale length and expected intensity level. Δv is simply modeled as a first order lag of the velocity command, V_{aCMD}, through the plant transfer function

$$G_p(s) = \frac{A_s}{\Delta V_{a_{CMD}}} = \frac{1}{T_p s + 1} \tag{3.31}$$

where T_p is an engine lag time constant, and Δu is a deviation from the planner reference velocity,

$$V_{aCMD} = V_{ref} + \Delta u \tag{3.32}$$

so then the differential equation for Δv is

$$\Delta \dot{v} = -\frac{1}{T_p}\Delta v + \frac{1}{T_p}\Delta u + \frac{\gamma}{T_p} w_2 \tag{3.33}$$

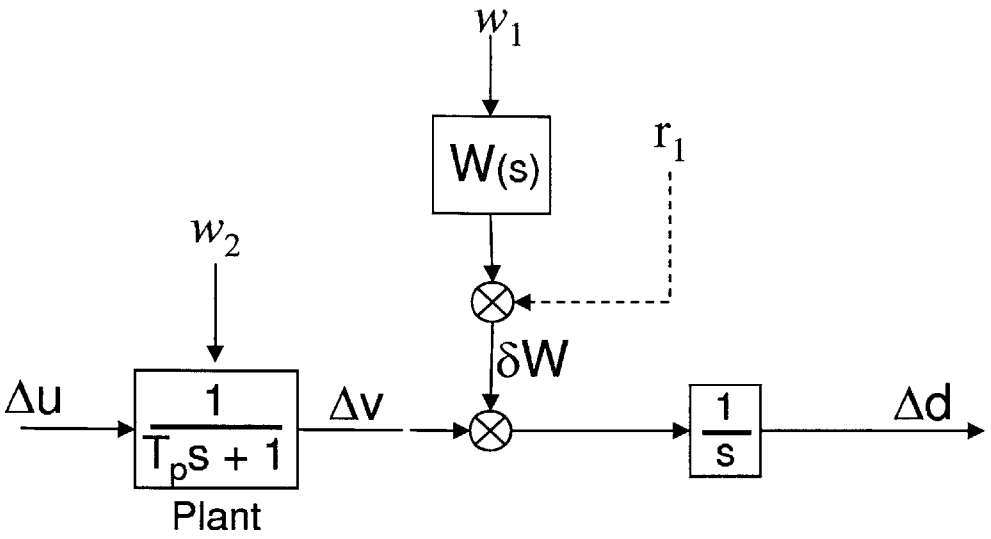


Figure 3-9: The timing dynamics are shown in the diagram for distance error, Δd, velocity error, Δv and the input disturbance δW. In addition to turbulence, $W(s)w_1$, input disturbance r_1 can enter the timing dynamics to represent disturbances seen by the vehicle as a result of static wind planning errors, $\widetilde{W}$, and its effect is similar to adding a slowly drifting bias to the δW state.

Figure 3-9 captures the timing dynamics described in Equations 3.30, 3.33 and 3.3. The two process noise inputs, $\boldsymbol{w} = [w_1\ w_2]^T$ represent the stochastic driving terms for the system. w_1 drives the Dryden turbulence model through $W(s)$, which provides appropriately scaled turbulence levels for the parameters in Eq. 3.3. Process noise w_2 enters the dynamics through $G_p(s)$ to model the deviation of the plan time constant, T_p, from the expected value. The engine lag can vary with motor and flying conditions and it should be noted that ramping up and slowing down to speed can produce different responses as a result of the aircraft drag. γ in Eq. 3.33 is used to vary the noise levels affecting the plant dynamics. Based on experimental flight data from Figure 2-25(a), the value for T_p used here is 7.5 seconds, and $\gamma = 1$.

In addition to the turbulence input, $W(s)w_1$, there is an additional disturbance, $r_1(t)$, that affects the timing of the mission as a function of the wind estimation error, $\widetilde{W}$. The disturbance referred to here represents a systematic offset that will be observed on the vehicle level as an inconsistency in the planned path, and will

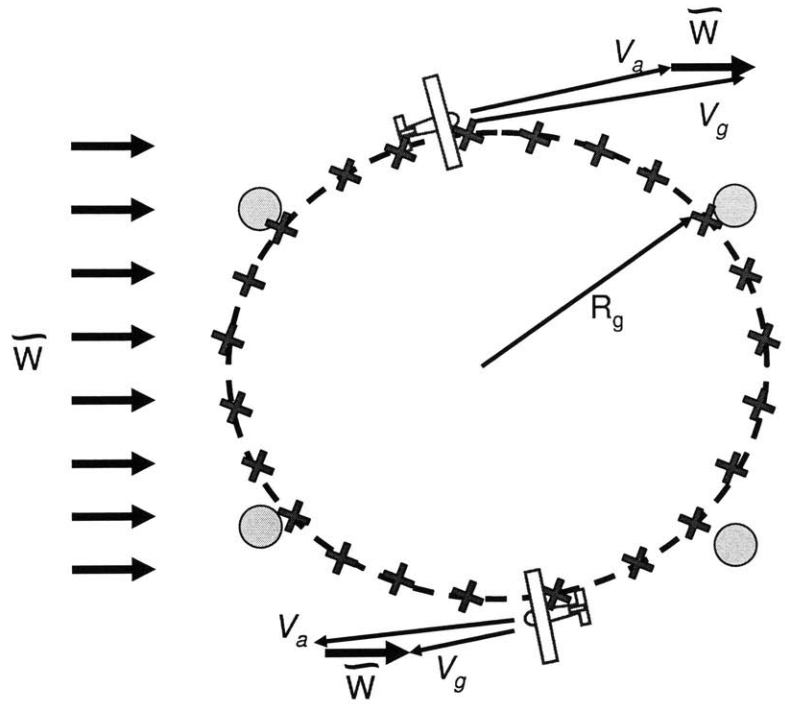


Figure 3-10: Schematic diagram for timing control simulation, where $\widetilde{W}$ represents an error in the static estimate, $\widehat{W}$, used on the planning level to design a circular trajectory to the targets (indicated by circles).

therefore appear as a drifting bias in the δW state.

$$\delta W = W(s)w_1 + r_1(t) \tag{3.34}$$

where $r_1(t)$ is a function relating the static wind error to the trajectory being flown. In the next subsection $r_1(t)$ will be shown to be observable given measurements of the distance and velocity errors, Δd and Δv, and added performance can be gained by using this estimate in the controller design. For further clarification consider the motivating example in Figure 3-10, described here for the case with *no turbulence*:

The circular pattern indicated by the $\times$ symbols represents a plan that was designed using the discretized MILP dynamics in Eq. 3.24. The mission is to fly to each of the targets, while maintaining small timing errors even in the presence of a wind estimation error, $\widetilde{W}$. The planning error causes the discrete MILP trajectory points to become spaced inconsistently with the actual environmental conditions, causing

the vehicle to experience an approximately sinusoidal disturbance as it travels around the loop,

$$r_1(t) \approx \| \widetilde{W} \| \sin\left(\frac{V_{ref}}{R_g} t\right)$$

In order to compensate for the resulting planning error, the vehicle must increase the airspeed when traveling in a headwind, and decrease the airspeed when traveling downwind. Provided that the magnitude of $\widetilde{W}$ is small, the vehicle should have the authority to handle this disturbance, although the frequency of $r_1(t)$ will also play a significant role in the effectiveness of speed control. This example is revisited in the next section and actually demonstrated on the UAV hardware as a practical application of timing control.

The timing dynamics shown here include process noise to model uncertain and stochastic states and there is also measurement noise on the distance and particularly velocity errors. The estimate of δW will be useful in determining a closed loop control law to regulate the timing error, however the proper balance between measurement and process noise weightings will need to be found. This makes the timing dynamics a good application for LQG control, and the next section closes the loop on the dynamics shown here.

3.5.2 Timing Control

LQG control will be useful for this application since the system measurements Δd and Δv are noisy and can also be easily modeled using the equations developed in the previous subsection. LQG control will allow the selection of appropriate measurement and process noises weightings based on observed values and also the selection of the optimal control input to the plant, $G_p(s)$, based on the estimated full state vector. Part of the premise of this scheme is that by estimating the disturbance levels impacting the system through δW, the LQR controller will be better able to "anticipate" the effect on the timing error e_t. This becomes useful when systematic biases creep into the δW state through errors at the planning level, $\widetilde{W}$, and the LQR controller is able to compensate before the disturbances integrate into the distance error state,

Δd. As shown in Figure 3-9, $r_1(t)$ represents the effect of the planning error at the vehicle level, and it can effectively appear as a drifting bias in the δW state. In general $r_1(t)$ would not be known for an arbitrary path beforehand, but with proper tuning of the LQE estimator it can be estimated on-the-fly as it is experienced by the vehicle. More complicated estimation schemes could couple the expected planning level error, $\widetilde{W}$, with knowledge about the aircraft heading, however these will not be addressed here.

The tuning of the LQG controller is accomplished through the appropriate selection of the state and control weighting matrices, $Q_{(6\times6)}$ and $R_{(1\times1)}$ respectively, which penalize state and control deviations from zero in the LQR control design. In addition, the LQE process and measurement noise weightings $Q_{N(2\times2)}$ and $R_{N(2\times2)}$, appropriately select Kalman filter (LQE) gains based on the observed levels of process and measurement noise.

The measurement equations Eq. 3.28 and 3.29 determine the position and velocity errors respectively, however there will also be errors associated with these measurements,

$$\boldsymbol{Y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \Delta d \\ \Delta v \end{bmatrix} + \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix} \tag{3.35}$$

In principle the Kalman estimator could be used to smooth the measurements, $\boldsymbol{Y}$, through high measurement noise weightings, however the accuracy of disturbance signal estimates from r_1 inputs in Figure 3-9 are a function of how strongly the measurements y_1 and y_2 are trusted over the model dynamics. This is demonstrated through a simulation example in Figure 3-11, which represents a similar scenario to the one seen Figure 3-10. The estimates of distance, velocity and disturbance states $X = [\Delta d\ \Delta v\ \delta W]^T$ are shown with the truth values for two different measurement noise weightings, and a sinusoidal input $r_1(t)$. Since the estimator has no knowledge of the disturbance signal, $r_1(t)$, low measurement noise weightings on y_1 and y_2 are required to provide accurate results. The tradeoff on increasing the δW estimation accuracy is that even relatively low noise levels pass through the estimator and are seen amplified in the δW estimates. As a result, some caution is needed in implement-

ing control based on these estimates, since high noise levels in the airspeed command could exceed throttle actuation limits.

To mitigate the noise levels on the control signal that actuates the engine throttle, one approach is to include an additional second order filter, $F(s)$, on the control signal which is sent to the plant, Δu_f

$$F(s) = \frac{\Delta u_f(s)}{\Delta u(s)} = \frac{w_{n_u}^2}{s^2 + 2\zeta_u w_{n_u} + w_{n_u}^2} \tag{3.36}$$

The filter states, Δu_f and $\Delta \dot{u}_f$, are then modeled in the system dynamics and used to effectively de-weight the high frequency components of Δu in the LQR design. This is accomplished by selecting an appropriate value corresponding to $\Delta \dot{u}$ in the Q state weighting matrix, as well as control filter constants, ζ_u, and w_{n_u} which are selected to provide fast response with no damping. For the purposes of this application, $\zeta_u = 0.707$ and $w_{n_u} = 1$ rad/s provide fast response on the order of the filter update rate (1 Hz), and with no overshoot. $F(s)$ is then actually implemented as a separate filter running in series with the Kalman estimator, providing the airspeed command to the plant. Figure 3-12 shows the effect of the state penalty in the LQR control design for large and small values on the $\Delta \dot{u}_f$ state. With sufficient weightings, the high frequency content of Δu_f is removed, providing some performance degradation but also a realistic airspeed input signal that can be implemented on the aircraft.

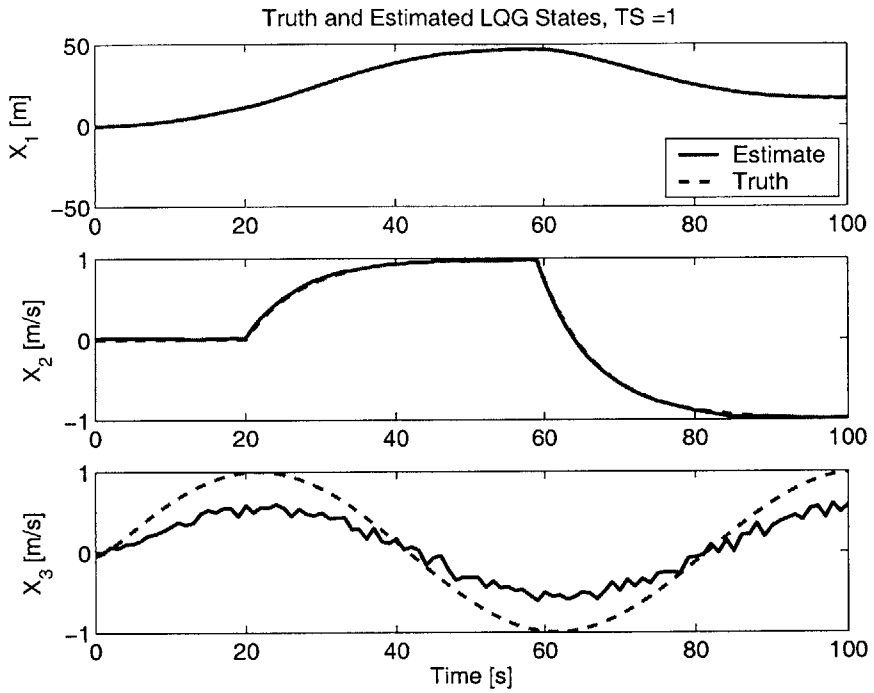
Finally, saturation limits are also applied to the input control signal, such that the airspeed command sent to the vehicle is

$$V_{aCMD} = \text{sat}(\Delta u_f + V_{ref}) \tag{3.37}$$

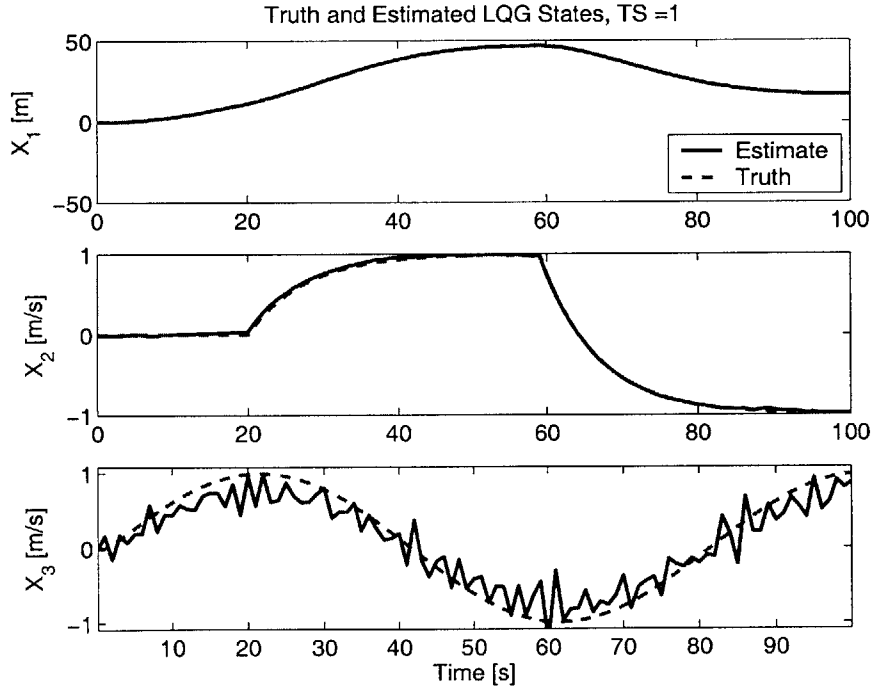
The saturation nonlinearities are passed to the LQE estimator for the state prediction, providing accurate estimates in the nonlinear control regime as well. Measurements of the testbed aircraft in flight have found the operating limits to lie within the range $20 \leq V_{aCMD} \leq 28$ m/s, although this is somewhat variable due to engine settings.

The closed loop system is then depicted in Figure 3-13, including the LQG controller and output control filter, $F(s)$. The complete state and noise input vectors are then

$$X_{e_t} = [\Delta d \ \Delta v \ \delta W \ \delta \dot{W} \ \Delta u_f \ \Delta \dot{u}_f]^T \tag{3.38}$$



(a) $Q_N = 2.5$, $R_N = \text{diag}(.5\ 1)$



(b) $Q_N = 2.5$, $R_N = \text{diag}(.01\ 1)$

Figure 3-11: Discrete Kalman estimator simulation with $\delta W(t_k) = r_1(t_k) = sin(.078t_k)$, moderate velocity measurement noise, v_2, and low measurement noise v_1. The accuracy of the $\hat{X}_3 = \delta W$ estimate is largely dependant on low R_N weightings for good tracking, however increasing noise levels. $X_1 = \Delta d$, $X_2 = \Delta v$

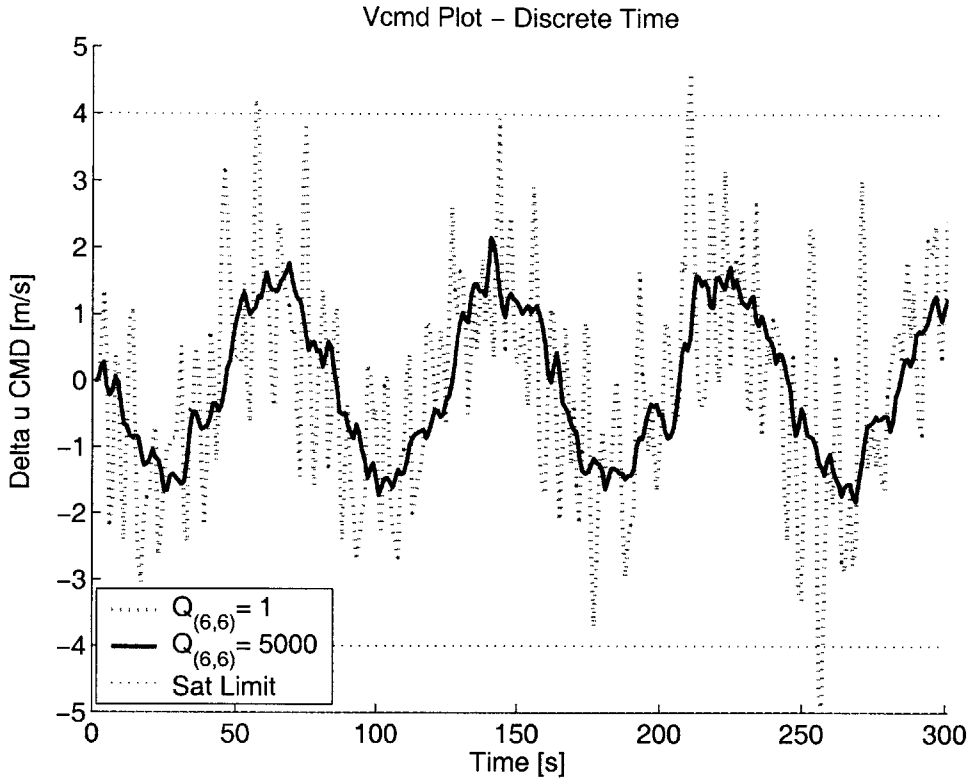


Figure 3-12: The output control filter $F(s)$ is included to penalize the $\Delta\dot{u}_f$ state from large deviations from zero, and limit the noise levels on the control signal Δu. The velocity output command Δu_f is shown for two cases with reasonable values of v_1 and v_2. The state weightings on $\hat{X}_{e_t}$ are $\{100, 1, 10^{-3}, 10^{-3}, 10^{-3}, \ldots\}$ and the control penalty, $R = 1$.

$$w = [w_1 \; w_2]^T \tag{3.39}$$

and the open loop system dynamics are given by Eqs. 3.40 – 3.43

$$\Delta\dot{d} = \Delta v + \sigma_w \sqrt{\frac{L_w}{2\pi V_{ref}}}\delta W + \sqrt{3}\frac{L_w}{V_{ref}}\delta\dot{W} \tag{3.40}$$

$$\Delta\dot{v} = -\frac{1}{T_p}\Delta v + \frac{1}{T_p}\Delta u_f + \frac{\gamma}{T_p}w_2 \tag{3.41}$$

$$\delta\ddot{W} = -\left(\frac{V_{ref}}{L_w}\right)^2 \delta W - 2\frac{V_{ref}}{L_w}\delta\dot{W} + \left(\frac{V_{ref}}{L_w}\right)^2 w_1 \tag{3.42}$$

$$\Delta\ddot{u}_f = -w_{n_u}^2 \Delta u_f - 2\zeta_u w_{n_u}\Delta\dot{u}_f + w_{n_u}^2 \Delta u \tag{3.43}$$

For the purposes of evaluation, this sixth order model is run at a 1 Hz update rate, although in practice much slower update rates, $\approx 0.25Hz$, could be employed.

Calculation of the rank of the observability matrix confirms that all the states are observable, given y_1 and y_2.

In summary, A, G and C are the system dynamics, input disturbance and measurement matrices and determined from Eqs. 3.40 – 3.43 and Eq. 3.35,

$$A = \begin{bmatrix} 0 & 1 & \sigma_w\sqrt{\frac{L_w}{2\pi V_{ref}}} & \sqrt{3}\frac{L_w}{V_{ref}} & 0 & 0 \\ 0 & -\frac{1}{T_p} & 0 & 0 & \frac{1}{T_p} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\left(\frac{V_{ref}}{L_w}\right)^2 & -2\frac{V_{ref}}{L_w} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -w_{n_u}^2 & -2\zeta_u w_{n_u} \end{bmatrix} \tag{3.44}$$

$$G = \begin{bmatrix} 0 & 0 & 0 & \frac{\gamma}{T_p} & 0 & 0 \\ 0 & 0 & \left(\frac{V_{ref}}{L_w}\right)^2 & 0 & 0 & 0 \end{bmatrix}^T \tag{3.45}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.46}$$

3.5.3 Performance Predictions

The closed loop transfer functions derived from the block diagram in Figure 3-13 provide a measure of the expected performance. Neglecting saturation nonlinearities and measurement noise, the LQG controller transfer functions are defined given suitable estimator and controller weighting matrices. Define the numerators and denominators (equivalent) of these two transfer functions as,

$$\frac{u_1(s)}{\Delta d(s)} \equiv \frac{n_{u1}(s)}{d_u(s)} \tag{3.47}$$

$$\frac{u_2(s)}{\Delta v(s)} \equiv \frac{n_{u2}(s)}{d_u(s)} \tag{3.48}$$

where the controller output, Δu, is determined by

$$\Delta u = u_1 + u_2 \tag{3.49}$$

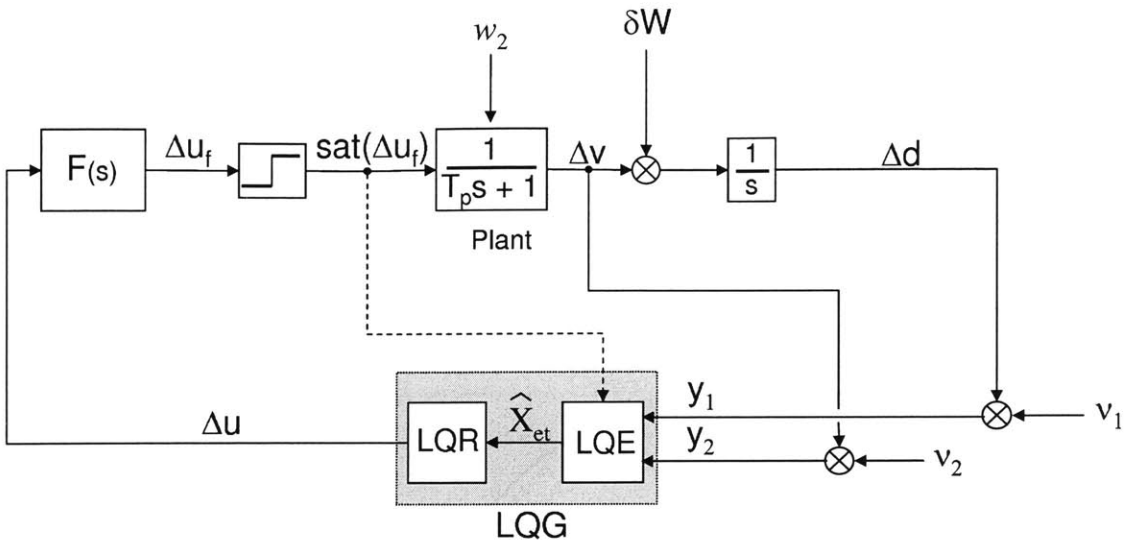


Figure 3-13: Timing control feedback loop with control saturation, disturbance input δW, measurement noise $[v_1\ v_2]^T$, and process noise input w_2. Discrete LQG Controller run with a sample time of 1 second, the dashed line indicates the feedback path for nonlinear saturation effects in the propagation of state estimates $\hat{X}_{e_t}$.

Using Eqs. 3.47 – 3.49 as well as Eq. 3.31 and 3.36, the output-disturbance transfer function can be shown to relate δW to the timing error $e_t = \frac{\Delta d}{V_{ref}}$

$$\begin{aligned} T_{YD}(s) &\equiv \frac{e_t(s)}{\delta W(s)} = \frac{1}{V_{ref}} \frac{d_u(T_p s+1)d_{uf} - n_{u2}w_{n_u}^2}{d_u(T_p s^2+s)d_{uf} - n_{u2}w_{n_u}^2 s - n_{u1}w_{n_u}^2} \\ d_{uf} &= s^2 + 2\zeta_u w_{n_u} + w_{n_u}^2 \end{aligned} \tag{3.50}$$

and the open loop response is

$$T_{YD_{ol}}(s) \equiv \frac{1}{V_{ref}} \frac{1}{s} \tag{3.51}$$

$T_{YD}(s)$ provides a prediction for the disturbance rejection properties of the system, given expected frequency ranges for disturbance signals entering the system. For example, the magnitude and phase of $T_{YD}(s)$ and $T_{YD_{ol}}(s)$ are shown in Figure 3-14, which predicts that for input disturbance frequencies below 10^{-1} rad/s, the amplitude reduction for $T_{YD}(s)$ is about 50%. This is analogous to the situation in Figure 3-10 with $r_1(t) \approx \| \widetilde{W} \| \sin(0.08t)$. The frequency of input disturbance is determined by the time period to complete the circuit. If $\| \widetilde{W} \|$ is 1 m/s, the expected timing error

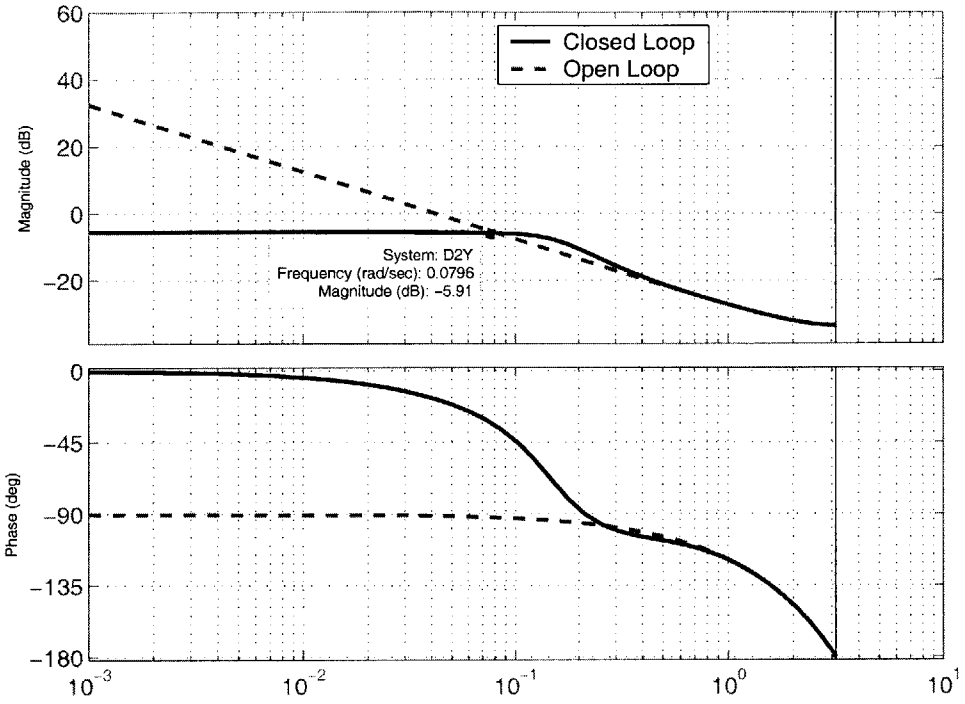


Figure 3-14: Bode plot for $T_{YD}(s)$ with state weightings $\{100, 1, 10^{-3}, 10^{-3}, 10^{-3}, 10^{5}\}$ and control weight $R = 1$. The DC reduction in the timing error, e_t, is -6dB $= 50\%$ amplitude for frequencies $< 10^{-1}$ rad/s, providing a significant reduction from the open loop case.

for these control settings will be ± 0.5 seconds. For most flying situations the heading will not be changing as rapidly as in Figure 3-10, therefore an upper bound on the input disturbance can be assumed to occur at the frequency determined by Figure 3-10. This assumes that the circuit is on the order of the vehicle turn radius, and that the static planning error, $\widetilde{W}$ is not rapidly changing. Note the for both the open and closed loop cases the magnitude plot drops off at -20 dB/dec above 10^{-1} rad/s, so these frequencies will not have a large impact on the timing control of the system.

Figure 3-15 shows the magnitude and phase plot for Eq. 3.50 with varying $\Delta\dot{u}_f$ state weightings. This term penalizes the derivative of the control signal, $\Delta\dot{u}_f$, so that high frequency content is removed from the input signal to the plant. Predictably as $\Delta\dot{u}_f$ is left unconstrained, $T_{YD}(s)$ becomes more effective at reducing the input disturbance magnitude, and this is seen in the reduction of the magnitude plot in

Figure 3-15. Figure 3-12 shows the effect of having too low a penalty on $\Delta\dot{u}_f$, thus there is a tradeoff between the $\frac{Y}{D}$ disturbance rejection performance and the level of noise that can be tolerated in the input command signal, Δu_f. The selection of the $\Delta\dot{u}_f$ control weighting, $Q_{(6,6)}$, is then made to balance these two competing qualities. Note that for reasonably wide saturation limits there is little disadvantage to using the entire range of throttle settings available, thus the control weighting penalty, R, is not heavily weighted relative to the system states Δd and $\Delta\dot{u}_f$. The weighting on Δd is selected to put a large penalty on distance errors, since the purpose of the timing controller is to regulate this value.

As a result, the performance of the LQG compensator in reducing input disturbances is largely dependant on the state control weighting on $\Delta\dot{u}$, and this is selected to tradeoff control noise levels on the filtered control signal, Δu_f. The LQG controller can be tuned to match Dryden model disturbance levels as well as the predicted measurement noise levels.

Because of the saturation limits on the system, it is important to determine what disturbance levels will cause the system to saturate, and if anything can be done to avoid this situation. The output control limitations can be analyzed using the $\frac{U}{D}$ transfer function, which can be determined in a similar fashion to Eq. 3.50

$$T_{UD}(s) \equiv \frac{\Delta u_f(s)}{\delta W(s)} = \frac{n_{u1} w_{n_u}^2 (T_p s + 1)}{d_u(T_p s^2 + s) d_{uf} - n_{u2} w_{n_u}^2 s - n_{u1} w_{n_u}^2} \tag{3.52}$$

Figure 3-16 shows T_{UD} with the variation of the $\Delta\dot{u}_f$ state weighting term. For increasing values of $Q_{(6,6)}$, $T_{UD}(s)$ becomes less sensitive to disturbances around 0.5 rad/s, indicating less control effort exerted at those frequencies. The expected magnitude of the control signal Δu_f can be determined from the disturbance magnitude at a given frequency. For low frequency disturbance signals ($< 10^{-1}$ rad/s), Δu_f will track well with approximately the same magnitude, and opposite phase. This means that for a 1 m/s wind disturbance acting as shown in Figure 3-10, the output control, Δu_f, will be $\pm$ 1 m/s and opposite phase to the input disturbance. The saturation limits will only be reached if the input disturbance magnitude, $\| \widetilde{W} \|$, exceeds the saturation limits of the vehicle airspeed response. Assuming that the planning refer-

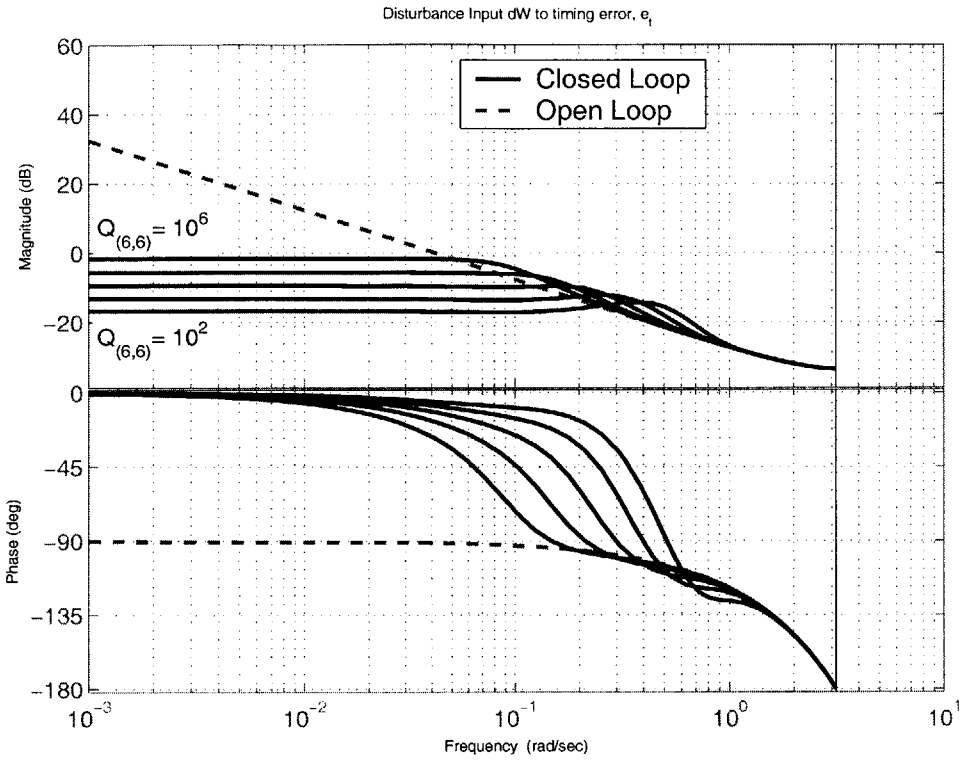


Figure 3-15: The discrete magnitude and phase plots for $T_{YD}(s)$ and $T_{YD_{ol}}(s)$ with varying Δu_f state weightings, control weighting $R = 10$, $T_s = 1$ s and $V_{ref} = 24$ m/s. Results indicate that the controller becomes more effective in reducing disturbance magnitudes as the penalty on Δu_f decreases, but even with high penalties $T_{YD}(s)$ effectively keeps DC timing errors from accumulating. The weightings on the other states in the system are $\{100, 1, 10^{-3}, 10^{-3}, 10^{-3}\}$

ence velocity, V_{ref}, is done at approximately mid-range, the vehicle saturation limits of $\pm$4-5 m/s should be large enough to compensate for these errors.

Above 10^{-1} rad/s, the Δu_f is largely dependant on $Q_{(6,6)}$, which should be selected to provide an appropriate tradeoff between the noise levels entering the plant and the $\frac{Y}{D}$ performance in Figure 3-15. As shown in Figure 3-14, the selection $Q_{(6,6)} = 10^5$ yields a reduction of about 50% in the disturbance rejection at low frequency, and also a low $\frac{U}{D}$ magnitude ratio for frequencies greater than 10$-$1 rad/s. With Figures 3-14 to 3-16 as guides, the LQG controller parameters can be selected to obtain good performance for other selections of the engine plant lag, T_p, measurement noise

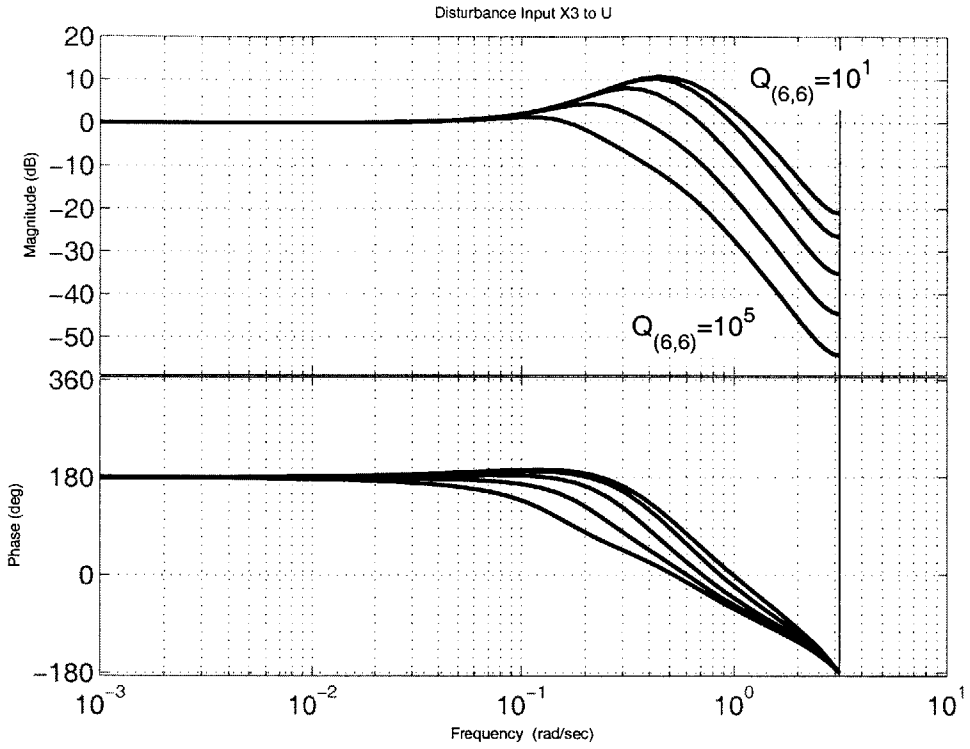


Figure 3-16: The discrete disturbance to control transfer function, $T_{UD}(s)$, is shown for different values of the Δu_f state weighting term with $T_s = 1$ s. Increasing the penalty on Δu_f has the effect of reducing output noise to the plant by decreasing the $\frac{Y}{D}$ magnitude ratio at frequencies $> 10^{-1}$ rad/s. For frequencies $\ll 10^{-1}$, the control signal is able to track the disturbance well, and provide compensation at opposite phase.

levels, ν_1 and ν_2, as well as expected disturbance levels impacting the system. This is the procedure that is usually followed in practice, since many of these parameters are not known until a HWIL or actual flight test experiment is performed. The following section will demonstrate the performance of the LQG controller implemented in several HWIL and flight test experiments.

3.5.4 Timing Control Simulation

Using discretized system dynamics from Eqs. 3.40 – 3.43 and the discrete LQG filter, the closed loop system is simulated in Matlab as well as on the Cloud Cap Hardware-

in-the-loop (HWIL) simulator. This enables representative scenarios to be validated with high accuracy before actually fight testing on the aircraft. Figure 3-10 depicts one such scenario that could be encountered and is used for demonstration purposes in this and the next subsection.

For this scenario, the frequency of the input disturbance will be related to the circular path the vehicle will follow, $\omega = \frac{V_{ref}}{R_g}$. For appropriately scaled values for the vehicle testbed, (R_g = 300 m, and V_{ref} = 24 m/s), ω = 0.08 rad/s and from Figure 3-14 the expected closed loop magnitude ratio, $T_{YD}(s)$, is 50%. This indicates a performance prediction for this scenario, given the same selections for the Q and R matrices. There will also be random variations due to turbulence and measurement errors, and the LQE estimator weighting matrices Q_N and R_N are selected (through trial and error) to mitigate these effects,

$$Q_N = \begin{bmatrix} 0.1 & 0 \\ 0 & \sigma_w^2 \end{bmatrix} \qquad R_N = \begin{bmatrix} 0.5 & 0 \\ 0 & 10 \end{bmatrix} \qquad \sigma_w = 1.2 \tag{3.53}$$

where σ_w is the Dryden turbulence intensity, from Subsection 2.1.4.

Figure 3-17 shows the results of the LQG controller implemented on the Cloud Cap HWIL simulator for the scenario in Figure 3-10. Both the open and closed loop timing errors are shown for an input disturbance $\|\widetilde{W}\|$ = 1 m/s, plus moderate levels of Dryden turbulence ($\sigma_w = 1.2, L_w = 250$). The timing error is shown to be regulated to within ± 0.5 seconds, indicating a $\frac{Y}{D}$ magnitude reduction of roughly 50%, as expected from Figure 3-14. The open loop response has both sinusoidal and random components, as expected from the discrete Matlab simulations with the same input parameters. Figure 3-18 shows the HWIL and Matlab input commands to G_p, as well as the HWIL vehicle speed response, Δv. Confirming simulation results, the higher frequency noise is filtered out of the control command providing a reasonably smooth input signal to $G_p(s)$.

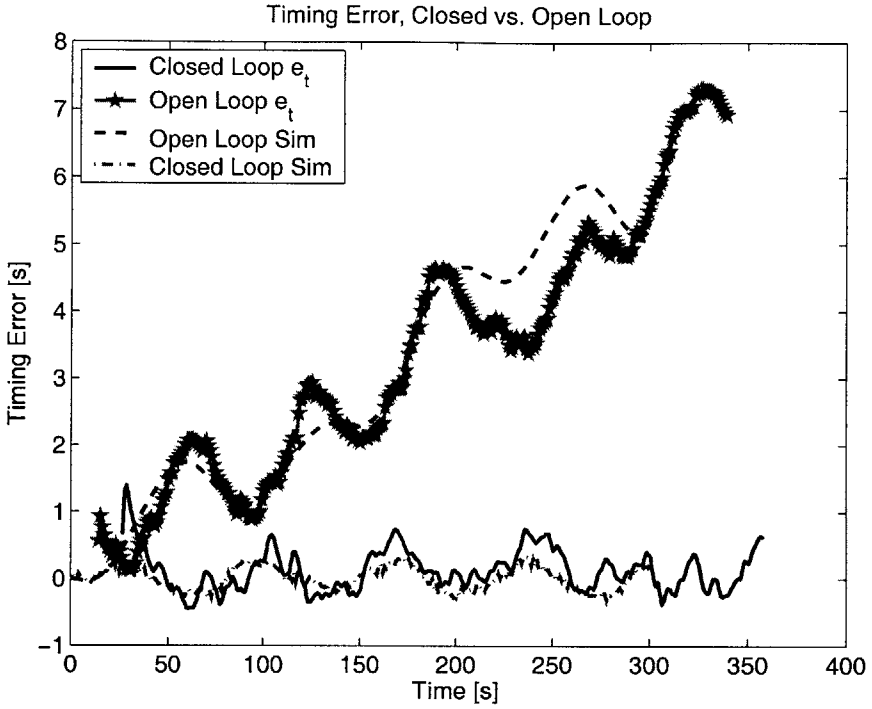


Figure 3-17: Agreement of Matlab and HWIL simulations for open and closed loop timing error responses, $\sigma_w = 1.2$ m/s, $L_w = 150$ m. Results in agreement with analytic performance predictions, regulating timing errors to ± 0.5 seconds for a $\widetilde{W} = 1$ m/s.

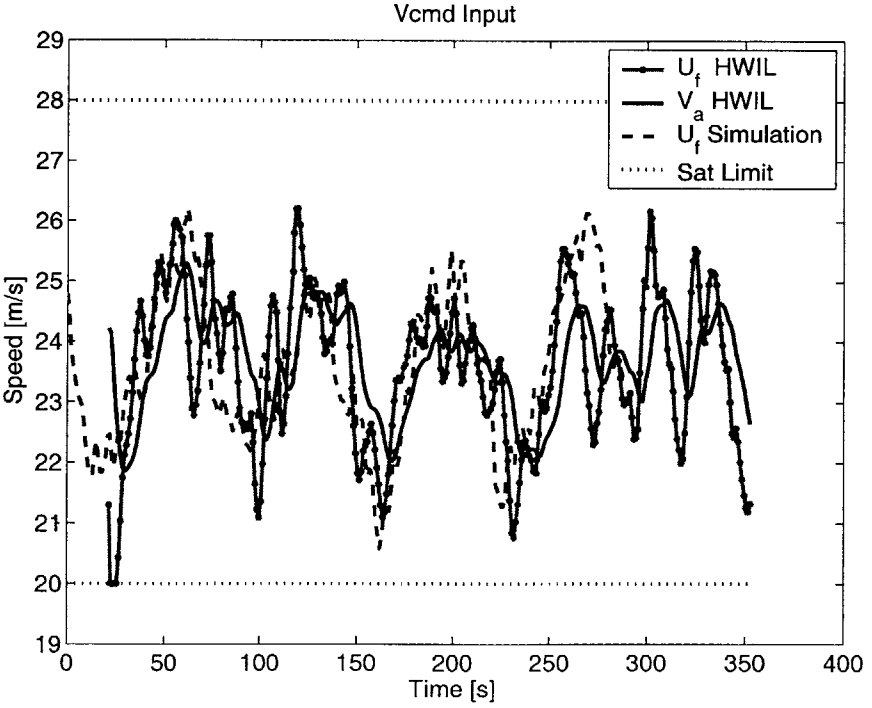


Figure 3-18: Closed loop commands, U_f and aircraft response, V_a for Matlab and HWIL simulations.

3.5.5 Timing Control Flight Test Experiment

As a further demonstration of timing control, the LQG controller was implemented on the UAV testbed under similar flight conditions to those in Figure 3-10. The flight test experiment showed that even when unknown planning level disturbances act on the system, the low level speed controller can compensate by varying the reference speed command, V_{aCMD}. Figure 3-19 shows the groundtrack from one of the circuits with no planning level input disturbance, $\overline{W}$. The timing errors in Figure 3-20 indicate roughly $\pm$ 1 second in timing errors, despite the relatively high input disturbance estimates shown in Figure 3-22. The filtered control commands for the experiment are shown in Figure 3-21, along with the airspeed measurements, and filtered airspeed estimates.

The true airspeed is shown to oscillate more than expected due to the vertical pitch dynamics of the aircraft as altitude variations are made around the circuit. This did not pose a large problem for the estimator in this case because the estimator gains in Eq. 3.53 were designed to weight the model predictions for the Δv state. More consistent measurements could be achieved by modifying the measurement equation, Eq. 3.29, to account for the variation in airspeed due to pitch angle and altitude variation. Overall, good waypoint tracking and timing control performance was achieved, with the vehicle maintaining less that 1 second in timing error for the duration of the flight.

It should also be noted that the scenario shown here (*i.e.*, a tight circuit) probably represents the most difficult situation for the timing controller, and more realistic mission scenarios will have lower frequency input disturbances. Provided the input disturbance magnitude is less than the saturation limits of the airspeed commands, the timing controller will be able to correctly compensate for the disturbance effects and maintain relative timing of the mission.

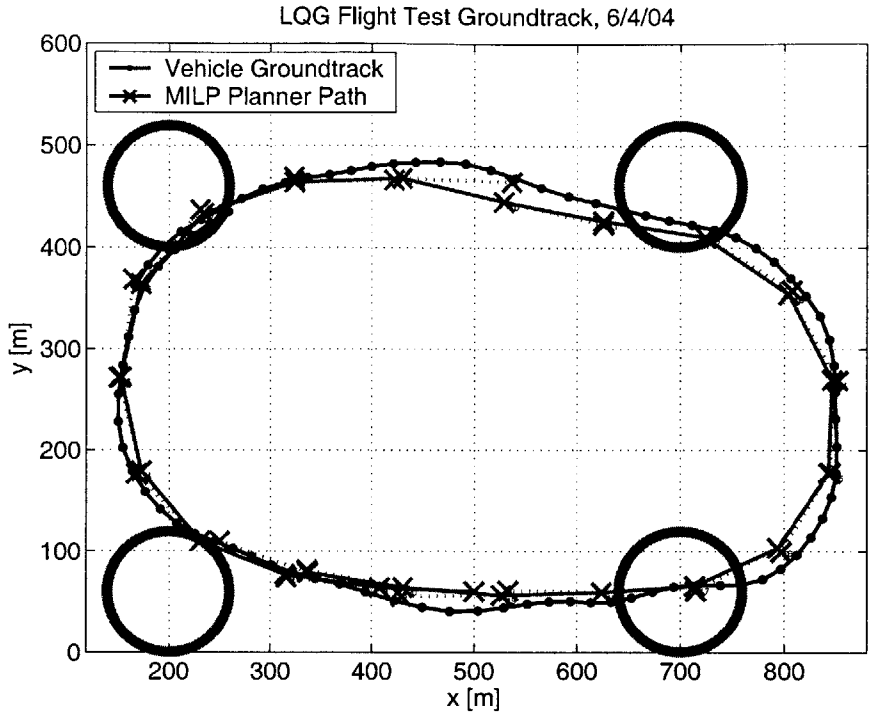


Figure 3-19: The planned and flown paths for 1 of 5 circuits of the LQG flight test. The goal regions are indicated by the circular regions and the vehicle traveled clockwise to each of the targets.

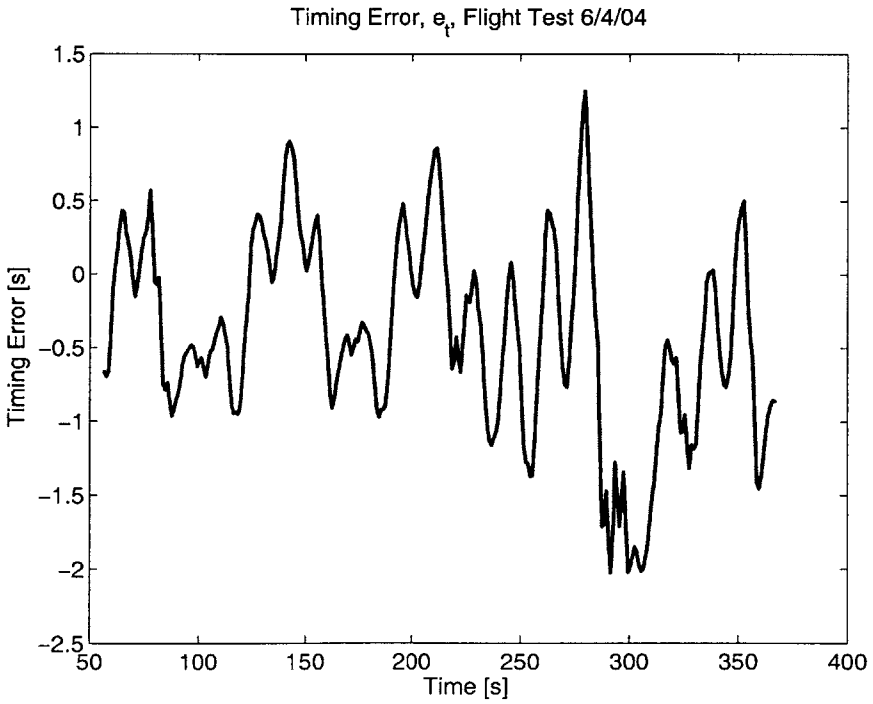


Figure 3-20: Relative timing errors for the LQG flight test, indicating roughly ± 1 second timing accuracy over the course of the flight despite reasonably high input disturbance levels (see Figure 3-22).

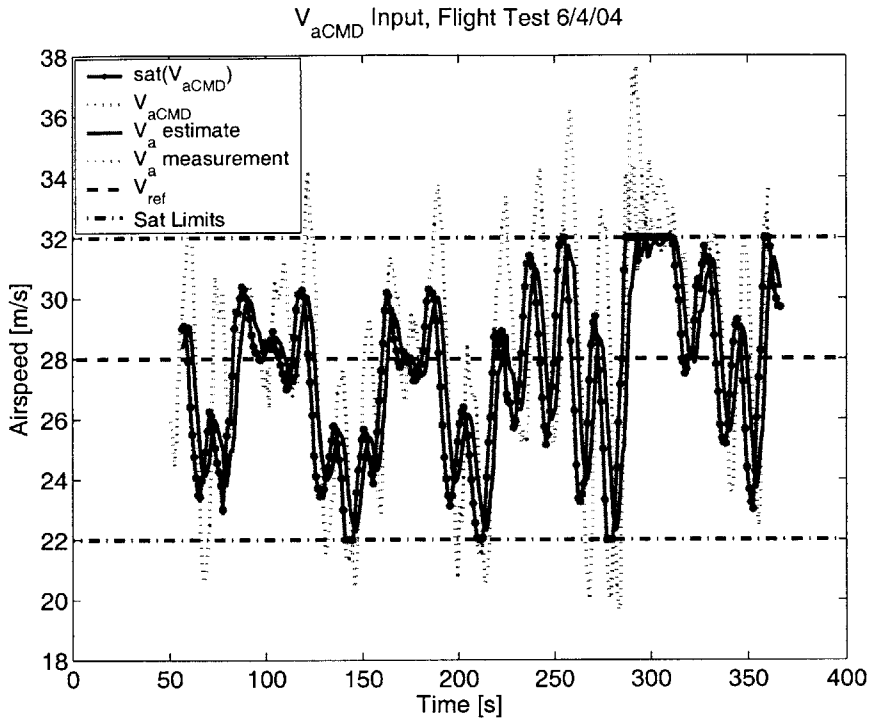


Figure 3-21: The airspeed commands and vehicle response for the LQG flight test experiment. The vehicle reference speed and saturation limits are indicated by the dashed lines.

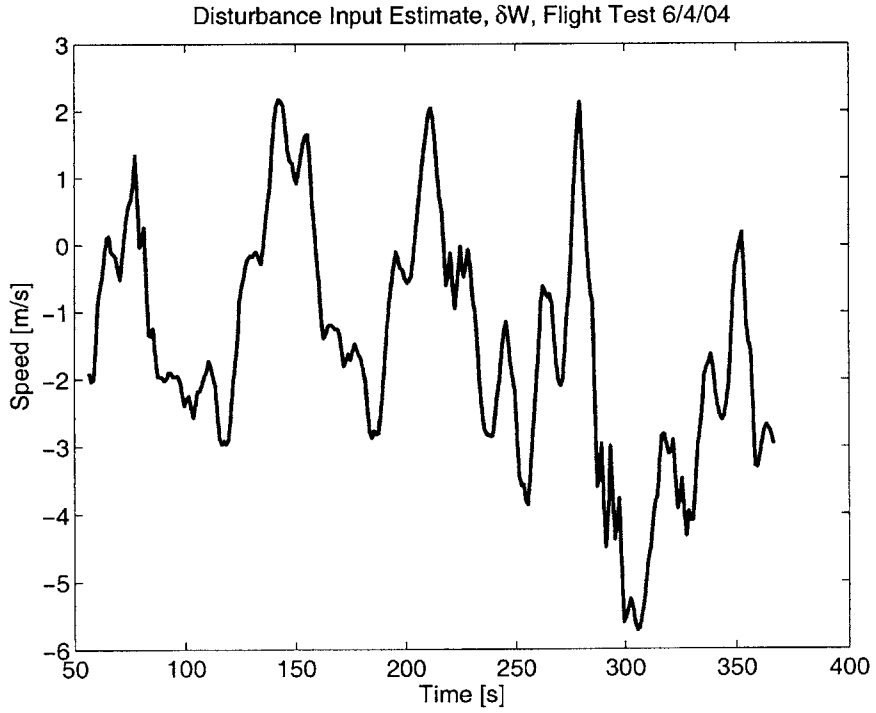


Figure 3-22: The estimates of the disturbance input to the system, δW, over the course of the LQG flight test experiment. Note that the estimates contain both periodic as well as stochastic components, as anticipated by simulation.

3.6 Conclusion

Wind represents a large disturbance source for teams of UAVs, and due to vehicle limitations, it becomes an important task to handle this type of disturbance appropriately on all the levels of control for the system to work effectively. This chapter presented a multi-level approach for dealing with the effect of wind disturbances on the UAV aircraft system. The block diagram for the system is depicted in Figure 3-23, and shows the interactions between each of the levels.

- The aircraft is influenced by wind disturbances that consist of averaged, static components and turbulence that can be modeled using the Dryden model.
- On the task assignment and trajectory design levels, wind estimates found using the algorithms presented in Section 3.2, are used to account for the gross motion of the atmosphere in the design of trajectories, which enables the planner to design paths that are both dynamically feasible and consistent with the timing assignments from the receding horizon task assignment algorithm.
- The wind-robust task assignment algorithm uses wind estimation uncertainty estimates to compute assignments that take into account the static estimation error, as well as finding the reference velocity required to enforce the loitering constraints. The loitering constraints are also shown to be bounded to lie within an interval that is consistent with the vehicle capabilities.
- The system dynamics in the MILP trajectory solver were updated to account for the effect of wind on the ground-relative trajectories, allowing paths to be designed that are consistent with the vehicle motion. The trajectories are uploaded to the vehicles as ordered sets of waypoints at every step of the receding horizon optimization.
- Through the variation of the desired airspeed, the LQG timing compensator is included to mitigate the effect of turbulence, as well as compensate for wind estimation errors used on the planning level to design trajectories. The latter of these two effects represents a persistent global planning error, and can be estimated using the Kalman estimator developed in Subsection 3.5.2.

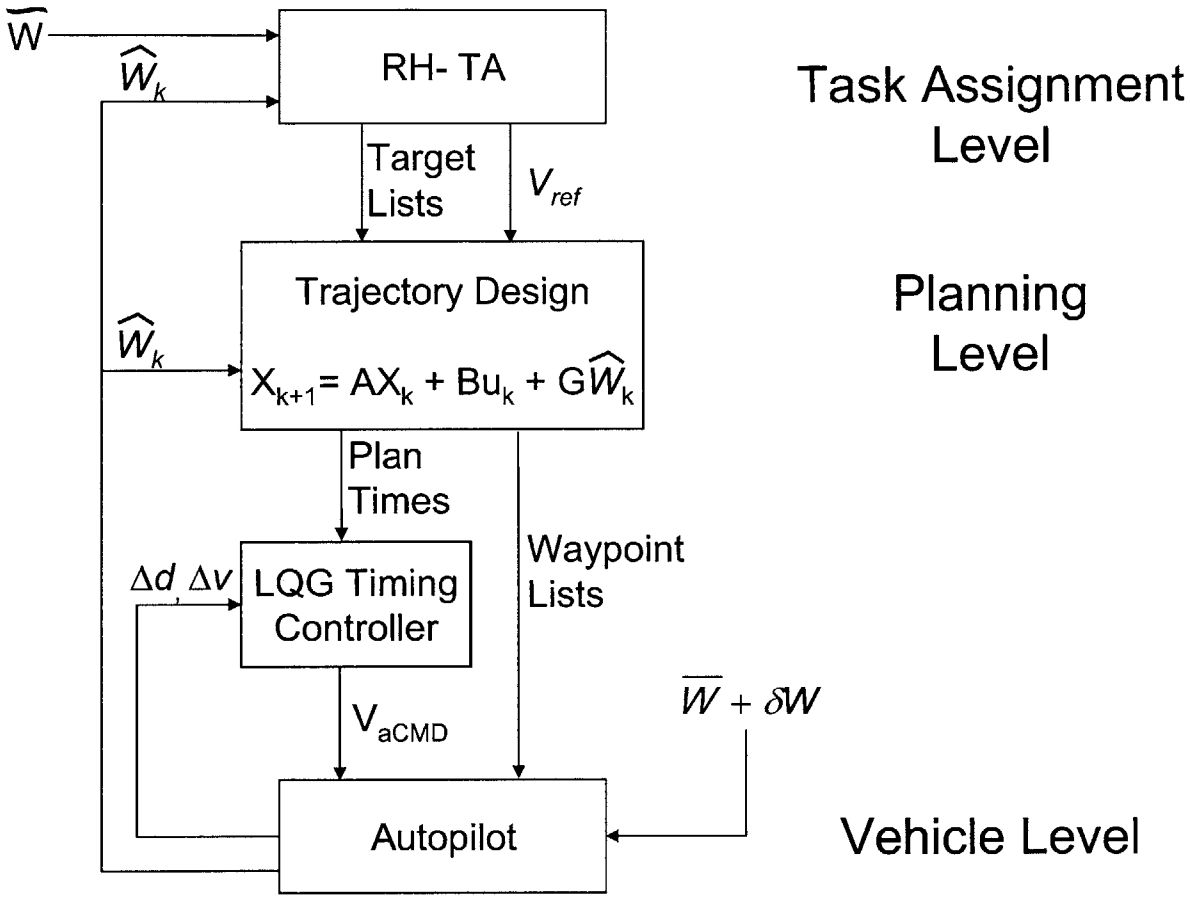


Figure 3-23: Overview of timing control scheme with planning system in the loop. Each level handles the input disturbances differently to make guaranteeing UAV arrival time at targets a multilevel estimation and control process with embedded uncertainty.

The static planning error, $\widetilde{W}$, provides the primary limitation in how well timings from the task assignment will be executed. Since saturation limits on the vehicle airspeed constrain the ability of the vehicle to compensate for errors on the planning level, there is a fundamental limit on how large a planning error the system can tolerate. If accurate $\overline{W}$ estimates exist, the timings and trajectories that are designed on the planning and task assignment levels are consistent with the actual disturbances acting on the system. This allows the timing controller to operate within the saturation limits and reject most other disturbances seen by the vehicle.

Performance improvements can be realized by estimating the direction of the static wind error, $\widetilde{W}$, as well as the magnitude. This provides information that can be used to "anticipate" the disturbance seen by the vehicle as it maneuvers through the environment. This allows the vehicle to compensate for disturbances before they act on the system, theoretically enabling the vehicle to compensate for a much larger percentage of the errors induced by the $r_1(t)$ disturbance.

Chapter 4

Receding Horizon Control with State Feedback

4.1 Introduction

Section 3.4 introduced the notion of disturbance feedback on the planning level in order to correctly account for the motion of the vehicle in a moving airmass. As discussed in Section 3.2, the static wind estimate, $\overline{W}$, will not be perfect, and this disturbance uncertainty will create both timing (along track) errors, and planning errors, which amount to trajectory plans that are inconsistent with the vehicle capabilities (*i.e., dynamically infeasible plans*). Turbulence effects can also be significant, and it is possible that the low level vehicle controller will not have enough authority to completely reject them, making the executed path different from the optimal MILP trajectory. As a result, feedback on the optimization initial conditions is necessary in order to ensure that trajectories designed relate to the actual vehicle state of the aircraft.

The need for RH-feedback is highlighted in Figure 4-1, which plots an initial flight test result employing the MILP based trajectory design techniques in Section 3.4. In this experiment, the planner uses the output of the previous optimization to determine the next initial condition and assumes $\overline{W} = [\ 0 \quad 0\ ]^T$. The deviation from the optimal planned path is evident, indicating the disturbance levels are saturating the

low level controller to its performance limits. Although the optimal path is theoretically feasible for the aircraft to fly, in the presence of real-world disturbances the low level controller has too little authority to track it with the desired precision.

In Section 4.2 the dynamics model is augmented to account for the lag in acceleration associated with the bank angle of the aircraft, which makes the model more realistic to be implemented on an actual vehicle testbed. Section 4.3 introduces a closed loop propagation model to account for the computation delay associated with receding horizon MILP optimization. Section 4.4 discusses the effect of different forms of uncertainty on the planning system, and quantifies the performance expectations due to the prediction error.

4.2 Bank Angle Dynamics

In the preceding chapters and Refs. [8, 18, 21, 15] double integrators with minimum/maximum airspeed and maximum acceleration input constraints are used as discrete MILP models of aircraft motion. This approximation works well under most circumstances, however this model permits the design of physically unrealizable plans in certain situations. Assuming constant speed and coordinated turns, aircraft generate accelerations through the banking as shown in Figure 4-2(a). Because the bank angle, ϕ, is subject to the roll dynamics of the aircraft (see Subsection 2.2.2), the acceleration is also limited by the first order dynamics

$$\frac{a(s)}{a_c(s)} = \frac{1}{\tau_r s + 1} \tag{4.1}$$

where τ_r is the effective time constant of the acceleration response due to a change in bank angle, and a_c is the desired acceleration magnitude. By revolving the accelerations into x and y components and augmenting the dynamics from Section 3.4, the complete discrete system can be expressed in state space form with sample time, T, as

$$\boldsymbol{x}_{k+1} = A_d \boldsymbol{x}_k + B_d \boldsymbol{u}_k + G_d \boldsymbol{w}_k \tag{4.2}$$

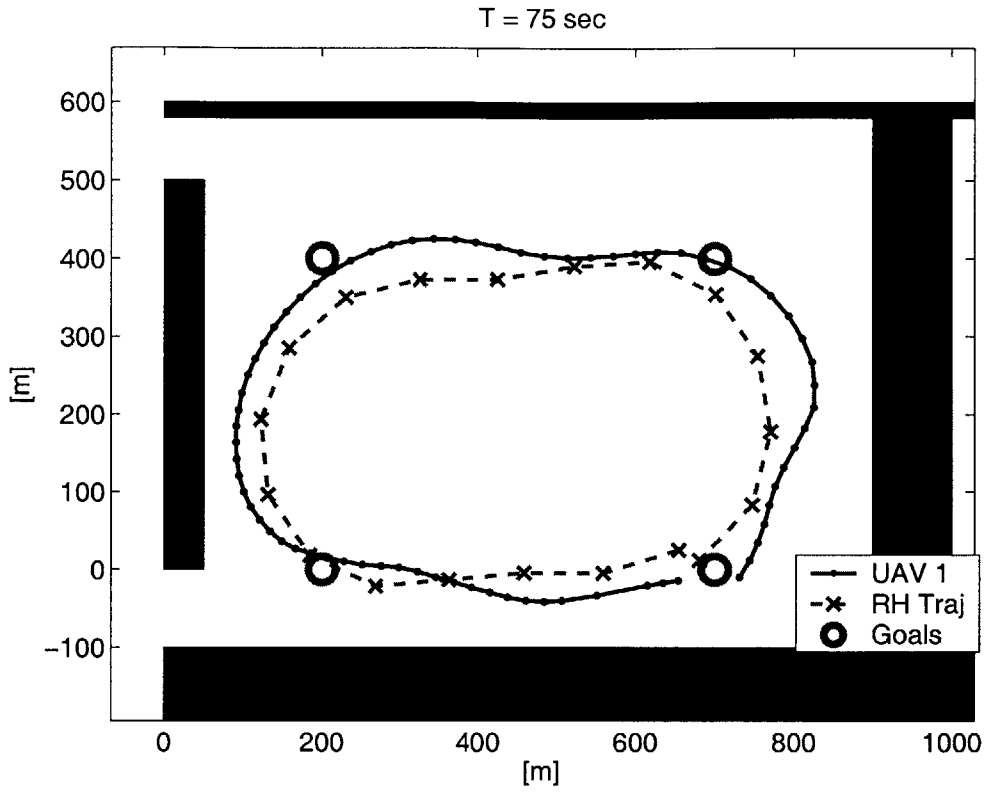


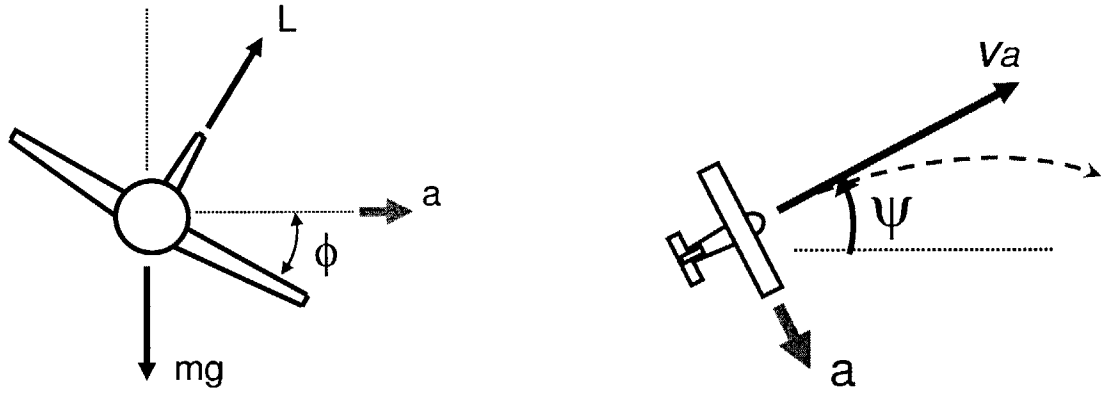
Figure 4-1: Flight test data for RH control without feedback on position or heading, showing the effect of wind disturbance on the closed loop system performance. Obstacle locations shown with blue regions, Planned data (marked with 'x' segments) and vehicle telemetry solid dots, goals marked with circles and ordered in a clockwise sequence.

$$\begin{bmatrix} x \\ y \\ v_{ax} \\ v_{ay} \\ a_x \\ a_y \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & 0 & T & 0 & \Lambda_1 & 0 \\ 0 & 1 & 0 & T & 0 & \Lambda_1 \\ 0 & 0 & 1 & 0 & \Lambda_2 & 0 \\ 0 & 0 & 0 & 1 & 0 & \Lambda_2 \\ 0 & 0 & 0 & 0 & \Lambda_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Lambda_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ v_{ax} \\ v_{ay} \\ a_x \\ a_y \end{bmatrix}_k + \begin{bmatrix} \beta_1 & 0 \\ 0 & \beta_1 \\ \beta_2 & 0 \\ 0 & \beta_2 \\ \beta_3 & 0 \\ 0 & \beta_3 \end{bmatrix} \begin{bmatrix} a_{c_x} \\ a_{c_y} \end{bmatrix}_k + \begin{bmatrix} T & 0 \\ 0 & T \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \end{bmatrix}_k \tag{4.3}$$

with

$$\Lambda_1 = \tau_r^2(e^{-\frac{T}{\tau_r}} - 1) + T\tau_r; \qquad \Lambda_2 = \tau_r(e^{-\frac{T}{\tau_r}} + 1); \qquad \Lambda_3 = e^{-\frac{T}{\tau_r}}$$

$$\beta_1 = \frac{1}{2}T^2 - T\tau_r + \tau_r^2(1 - e^{-\frac{T}{\tau_r}}); \qquad \beta_2 = T + \tau_r(e^{-\frac{T}{\tau_r}} - 1); \qquad \beta_3 = 1 - e^{-\frac{T}{T_c}}$$



(a) Tail view, with the roll angle, ϕ, and lift forces, L, shown.

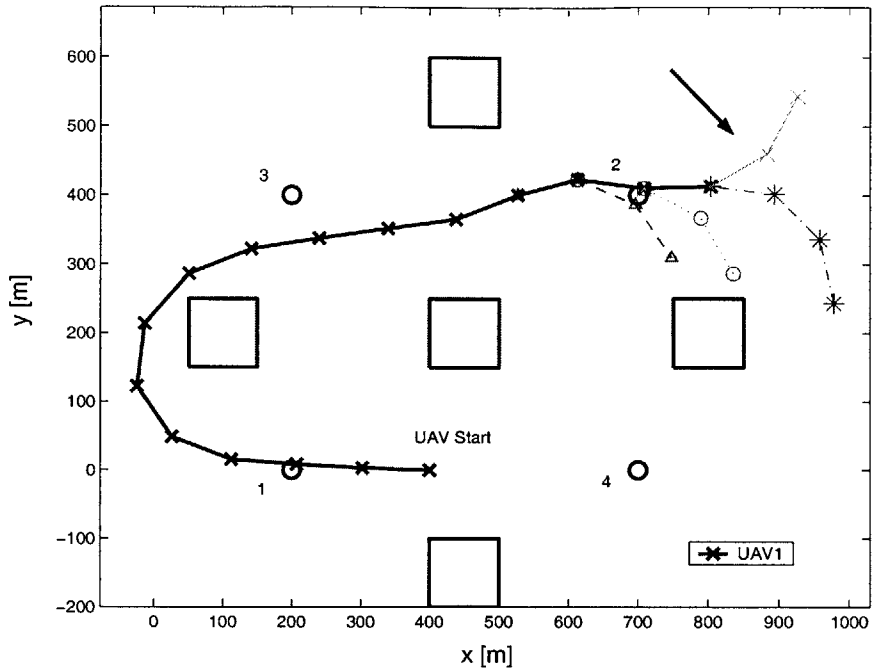
(b) Overhead view, with the airspeed vector, v_a, and yaw angle, ψ, shown.

Figure 4-2: Aircraft in a coordinated turn undergoing lateral acceleration, a. The figure assumes that the airspeed, v_a, is constant.

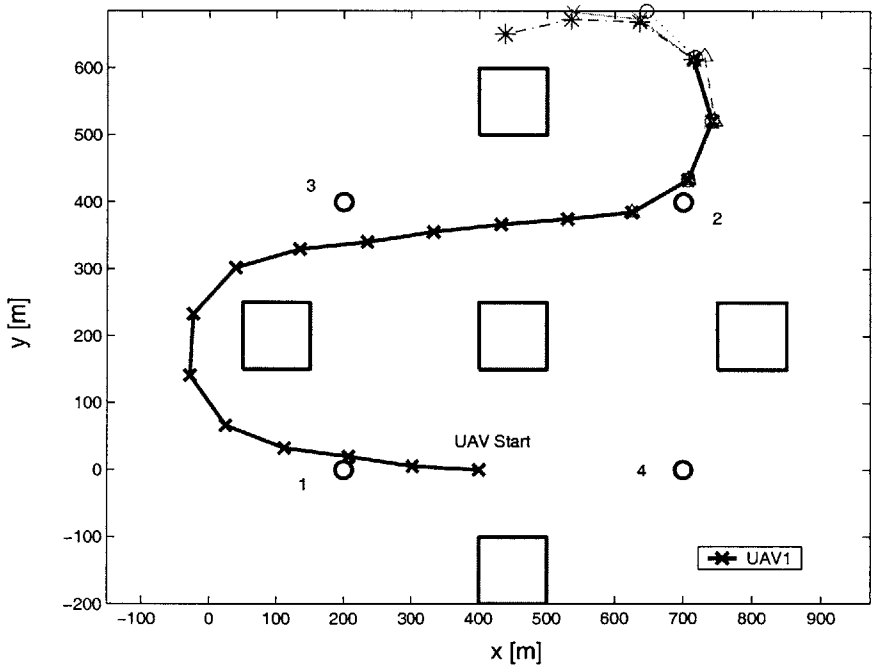
With the modifications to the dynamics in Eq. 4.3, the acceleration used on the trajectory design level more accurately reproduces the bank angle dynamics exhibited by a real aircraft and its effect can be seen in Figure 4-3, depicting a scenario with four targets and five avoidance regions. In Figure 4-3(a) the MILP solver can effectively choose any acceleration vector subject to $\|a\| < a_{\max}$ as a direct input into the dynamics, thereby allowing very rapid acceleration changes as shown by the flip-flop of the plans in the RH design. Figure 4-3(b) shows the same scenario with the bank angle modeled as a first order lag using the discrete dynamics in Eq. 4.3 and the acceleration lag $\tau_r = 0.25$ sec. The added lag smooths the transition between discrete MILP control inputs, allowing the solver to more correctly account for the bank angle dynamics of the aircraft.

4.3 Propagation Model

One of the challenges in utilizing receding horizon control is dealing with the effect of computation and communication in the loop. The delay associated with the time to compute a plan and communicate it can be significant, requiring that it be accounted for in the implementation of the control. The strategy for handling this delay as described in Refs. [40, 41], is implemented here and depicted graphically in Figure 4-



(a) Bank angle dynamics not modeled. The MILP solver initially plans to turn before reaching the right-hand obstacle, but subsequent plans rapidly switch directions causing dynamic instabilities as indicated by the arrow.



(b) Bank angle dynamics modeled with acceleration time constant, $T_c = 0.25$ sec. Slightly more conservatism is exercised to account for the added time to roll the aircraft, allowing for smoother accelerations.

Figure 4-3: Trajectories showing successive RH plans with (b) and without (a) the bank angle modeled in planner. The goal order is indicated by the numbers 1–4, $T = 4$ sec.

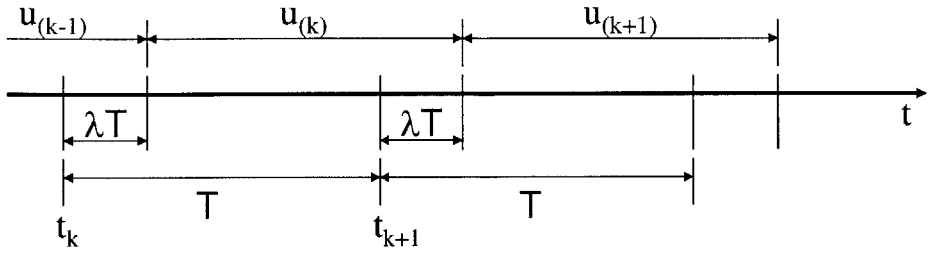


Figure 4-4: Receding Horizon Control timing with prediction step. Measurements of the state $x(k)$ are made at time t_k and closed loop propagation models are used to estimate $x(k + \lambda T)$. The control input, $u(k)$, is calculated using the state prediction during the interval λT and implemented at $(t_k + \lambda T)$.

4. To handle delay, the state $x(k)$ is measured at time t_k, and propagated forward to a future state $x(k + \lambda T)$, where $\lambda \in (0, 1)$ is a parameter chosen to estimate the delay as a function of the discrete sample time, T. The control term $u(k)$ is the new plan that is implemented at predicted step $x(k + \lambda T)$.

In order to do the state propagation in Figure 4-4, a closed loop model of the vehicle dynamics is required. This section develops the propagation model dynamics and validates results obtained utilizing the propagation strategy on the HWIL testbed.

4.3.1 Closed Loop Dynamics

Figure 4-5 depicts the geometry governing the waypoint tracking control law rotated into an intrack reference frame. The propagation model will be used to estimate the vehicle state at the propagated time, $(k + \lambda T)$, given that the autopilot will be applying control inputs to the system to regulate the cross-track error, y_{IT}, to zero. Assuming that the intrack heading, ψ_{IT}, remains small and that there are bank angle dynamics with time constant τ_r governing the roll angle, ϕ (positive for right wing down), the equations of motion governing the closed loop dynamics are as follows:

$$\dot{y}_{IT} = \psi v_a$$

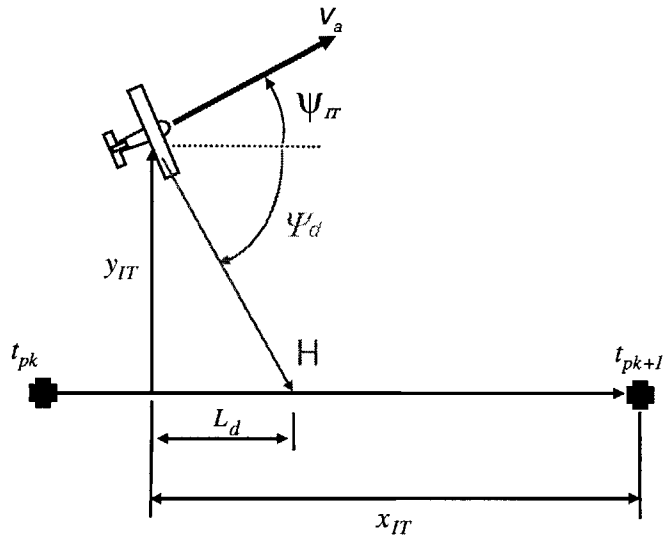


Figure 4-5: Waypoint tracking control law allows the selection of suitable closed loop dynamical relations.

$$\begin{aligned}
\dot{\psi}_{IT} &= -\frac{g\phi}{v_a} \\
\dot{\phi} &= -\frac{1}{\tau_r}\phi + \frac{1}{\tau_r}\phi_c \\
\phi_c &= K_p\left(\frac{y_{IT}}{L_d} + \psi_{IT}\right)
\end{aligned}$$

where ϕ_c is the desired bank angle from the linearized tracker control law in Subsection 2.3.2, and K_p is a proportional gain on the desired heading error. For the short time periods over which this model will be employed, the derivative action of the waypoint tracker will not have a significant effect, however wind disturbances can cause significant errors. Applying an intrack velocity disturbance, $\boldsymbol{w_{IT}}$, to account for the effect of wind on the propagation and converting to state space form, the closed loop dynamics are

$$\begin{aligned}
\dot{\boldsymbol{x}} &= A_{cl}\boldsymbol{x} + G\boldsymbol{w_{IT}} \\
\begin{bmatrix} \dot{y}_{IT} \\ \dot{\psi}_{IT} \\ \dot{\phi} \end{bmatrix} &= \begin{bmatrix} 0 & v_a & 0 \\ 0 & 0 & -\frac{g}{v_a} \\ \frac{K_p}{\tau_r L_d} & \frac{K_p}{\tau_r} & -\frac{1}{\tau_r} \end{bmatrix} \begin{bmatrix} y_{IT} \\ \psi_{IT} \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ \frac{1}{v_a} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w_{IT_x} \\ w_{IT_y} \end{bmatrix}
\end{aligned} \tag{4.4}$$

The linearized closed loop model Eq. 4.4 enables the estimation of the state of the vehicle for over the small time intervals needed to perform the computation. With

2.4GHz Dell laptops running CPLEX^{TM}, the MILP problems take 0.05–1 sec to solve, so $\lambda T = 1$ sec is used as an overbound.

The model parameters, $K_p = 0.25$, $L_d = 120$ and $\tau_r = 1.25$ are selected from the controller gains in Subsection 2.3.2 and the bank angle dynamics in Section 4.2. The crosstrack output for the linearized model is shown for varying selections of the dominant model parameters in Figures 4-6(a) and 4-6(b). The bank angle time constant, τ_r, and the airspeed v_a, were found to have less effect on the crosstrack performance of the model, however they are important in the scaling of the along track and heading dynamics.

4.3.2 MILP Bank Angle Initial Conditions

The added acceleration states in Section 4.2 require initial conditions for the discretized MILP dynamics in Eq. 4.3, which can be estimated from the aircraft bank angle estimate, $\hat{\phi}$. As shown in Figure 4-2(a), for an aircraft in a coordinated turn (*i.e.,* no sideslip motion and approximately level pitch angle), the lift forces contribute to both the vertical and horizontal directions such that an acceleration is generated in the radial direction of the form

$$a = \dot{\psi} v_a = \frac{L}{m} \sin\phi \tag{4.5}$$

where ϕ is the aircraft roll angle, and the lift, L, balances the vehicle weight in the vertical direction

$$L = \frac{mg}{\cos\phi} \tag{4.6}$$

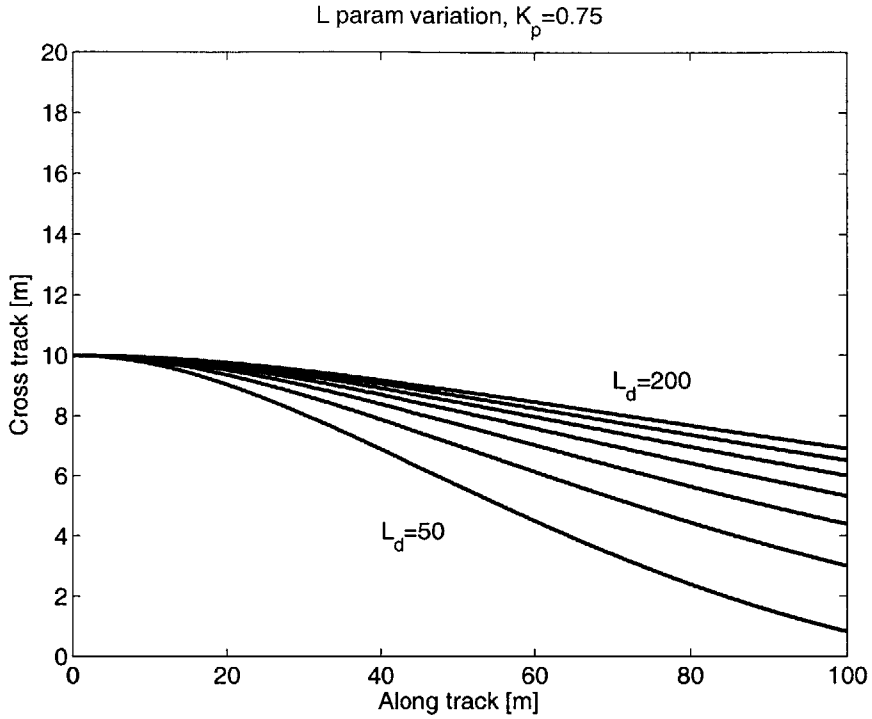
Substituting for L in Eq. 4.5,

$$a|_{W=0} = g \tan\phi \tag{4.7}$$

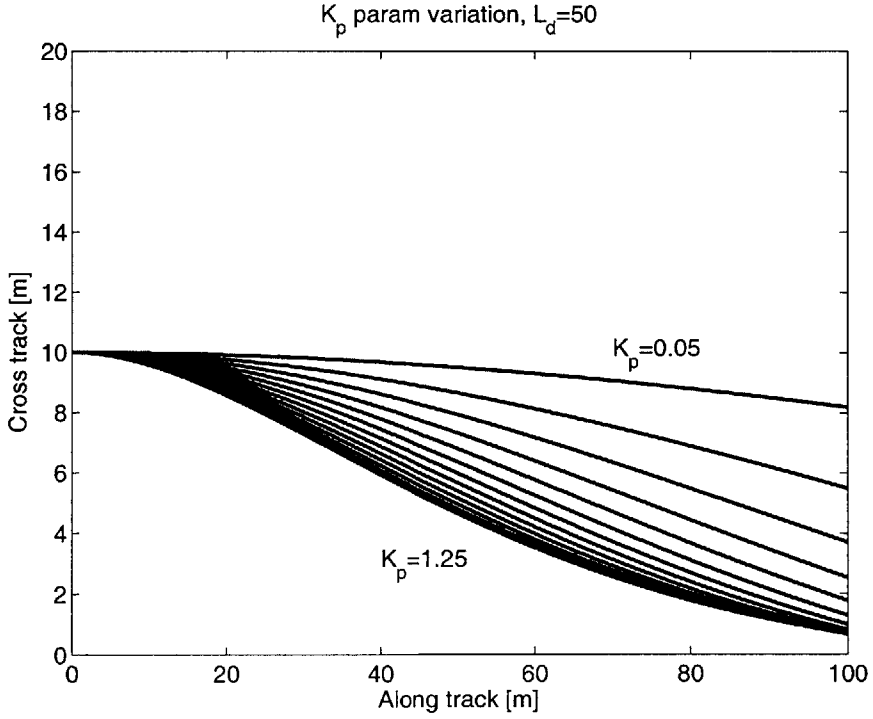
Eq. 4.7 assumes that there is no wind acting on the system, however as shown in Subsection 2.3.2, the Piccolo^{TM}autopilot schedules the tracker convergence parameter, (L_d in Figure 4-5), with the airspeed in order to increase the tracker performance in the ground frame[1].

$$L_d = L_d|_{W=0} \left(\frac{v_a}{v_g}\right)^2 \tag{4.8}$$

[1]Note: one should take care not to confuse the lift L and the tracker parameter L_d.



(a) Variation of the scale length, L_d, on the closed loop model dynamics. $L_d = 75$ most closely resembles the actual tracking performance of the vehicle controller implemented on the HWIL testbed.



(b) Variation of the proportional gain K_p on the closed loop model dynamics. $K_p = 1.0$ is selected as a parameter that most closely matches actual closed loop performance of the HWIL testbed.

Figure 4-6: Variation of the dominant model parameters in the closed loop tracking model, showing the closed loop vehicle path with an initial cross-track error $y_{ITo} = 10$ m, $\psi_0 = 0°$, $v_a = 25$ m/s.

This scheduling effectively scales the turn rate command inversely with the convergence parameter, thereby decreasing turn authority for headwinds ($v_g < v_a$) and increasing turn authority in tailwinds ($v_g > v_a$). The turn rate command, $\dot{\psi}_c$ is given by:

$$\dot{\psi}_c = K_p \left(\psi + \frac{y_{IT}}{L_d} \right) \approx \frac{1}{L_d} K_p y_{IT} = \left(\frac{v_g}{v_a} \right)^2 \cdot \frac{K_p y_{IT}}{L_d|_{W=0}} \tag{4.9}$$

Figure 4-6(a) shows the effect of the airspeed scaling on the closed loop dynamics, through variation in the scale length parameter, L_d. The effect of the reduced turn authority is seen for larger L_d values, while increased tracking is shown as L_d is reduced. Since the plant dynamics, A_{cl}, become time varying in the presence of winds, the trajectory planner must also take into account this scale factor when planning.

From Eq. 4.5, the acceleration magnitude is proportional to the turn rate of the vehicle. The acceleration commands provided to the MILP optimization must remain consistent with the closed loop vehicle dynamics, and as result the acceleration magnitude must also be scaled to account for the change in authority in the presence of winds

$$\begin{aligned} a &= \left(\frac{v_g}{v_a} \right)^2 \dot{\psi} v_a = \left(\frac{v_g}{v_a} \right)^2 a|_{W=0} \\ &\Rightarrow \quad a(\phi, v_g, v_a) = g \tan(\phi) \left(\frac{v_g}{v_a} \right)^2 = g \tan(\phi)(1 + \mathcal{T}_{WV_a})^2 \end{aligned} \tag{4.10}$$

where $\mathcal{T}_{WV_a}$ is the *disturbance magnitude ratio* as seen in Section 3.1, and the groundspeed and airspeeds are available from the onboard aircraft measurements. The state accelerations at time t_k in Eq. 4.3 are

$$a_x = a \sin(\hat{\psi}) \ , \qquad a_y = -a \cos(\hat{\psi}) \tag{4.11}$$

where the *true heading* estimate, $\hat{\psi}$, is the best estimate of the aircraft heading with respect to an inertial axis. Subsection 4.4.1 discusses the effect of uncertainty in this estimate in the model dynamics.

It will also be useful to determine the bank angle ϕ as a function of the model states. Referring to Figure 4-2(b), the heading angle is defined as

$$\psi = \arctan \left(\frac{v_{ay}}{v_{ax}} \right) \tag{4.12}$$

Differentiating gives

$$\dot{\psi} = \frac{1}{1+\left(\frac{v_{ay}}{v_{ax}}\right)^2} \cdot \frac{a_y v_{a_x} - a_x v_{a_y}}{v_{a_x}^2} = \frac{a_y v_{a_x} - a_x v_{a_y}}{v_a^2} \tag{4.13}$$

For small roll angles, $|\phi| \leq 30$ deg, Eqns. 4.5 and 4.7 reduce to

$$\dot{\psi} = -\frac{g}{v_a}\phi \tag{4.14}$$

for positive bank angles (*i.e.,* when the right wing is down). Substituting $\dot{\psi}$ from Eq. 4.13, the bank angle can be written

$$\phi = \frac{a_x v_{ay} - a_y v_{ax}}{g v_a} \tag{4.15}$$

4.4 Prediction Error

Uncertainty in the measured state at time t_k, coupled with unknown disturbances acting on the system during the propagation step, will lead to uncertainty in the vehicle state at time $(t_k + \lambda T)$. Errors in $x(t_k + \lambda T)$, can lead to *dynamical inconsistency* as the plans generated using the RH-MILP algorithm use this estimate as the initial condition for the next optimization step. An example of this type of planning error is shown in Figure 4-7, which was caused by poor wind and aircraft heading estimates on the planning level. Although the planner has a consistent dynamics model for the UAV, imperfect information in the initial condition and disturbances caused inconsistent plans to be repeatedly designed, making the execution of the trajectory impossible for the UAV to perform. To address this issue, a method for constraint tightening will be used to ensure that future plans will be consistent with the vehicle dynamics given bounds on the state uncertainty and input disturbances. This method allows the user to select an appropriate scale factor on the turn rate authority to provide a balance between the authority on the planning level for maximum maneuverability, and the authority on the low level to allow for uncertainty in the vehicle state and disturbances acting on the system.

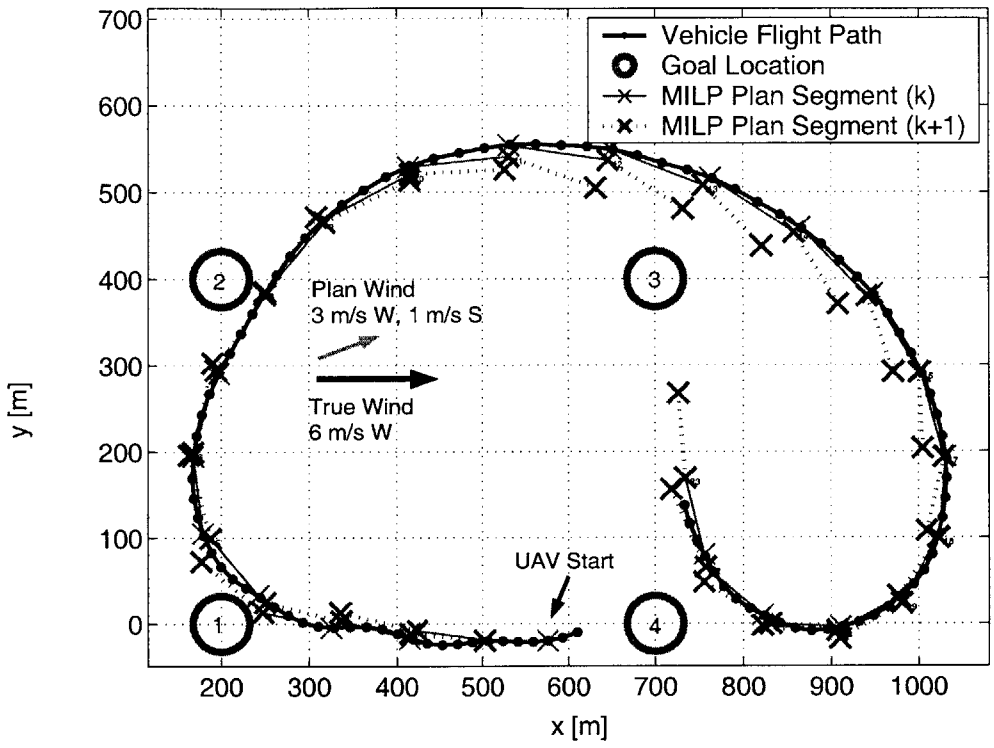


Figure 4-7: Effect of planning with poor wind and true heading estimates. Note discrepancy between planned path marked with '×' segments and actual path flown by the vehicle, indicating the planner is designing plans that are not consistent with the vehicle state at time $(t_k + \lambda T)$.

4.4.1 Measurement Error

Measurement errors at time kT are a source of error for the planning system, and they can be quantified by calculating bounds on the measurement noise levels. The state measurements in the propagation model, Eq. 4.4, consist of cross-track position, heading and bank angle measurements, $Y = X = \begin{bmatrix} y_{IT} & \psi_{IT} & \phi \end{bmatrix}^T$ (*i.e.*, full state feedback), which are obtained from GPS and filtered roll rate measurements, respectively. The aircraft heading, ψ_{IT}, is approximated using the groundspeed heading

$$\psi_{IT} = \arctan\left(\frac{v_{gy}}{v_{gx}}\right) \tag{4.16}$$

which is a valid assumption for small disturbance magnitude ratios, as shown in Section 3.1. While the 1Hz GPS position and velocity measurements are very accurate, the Trimble receiver used onboard the aircraft injects a 1 sec delay on the velocity

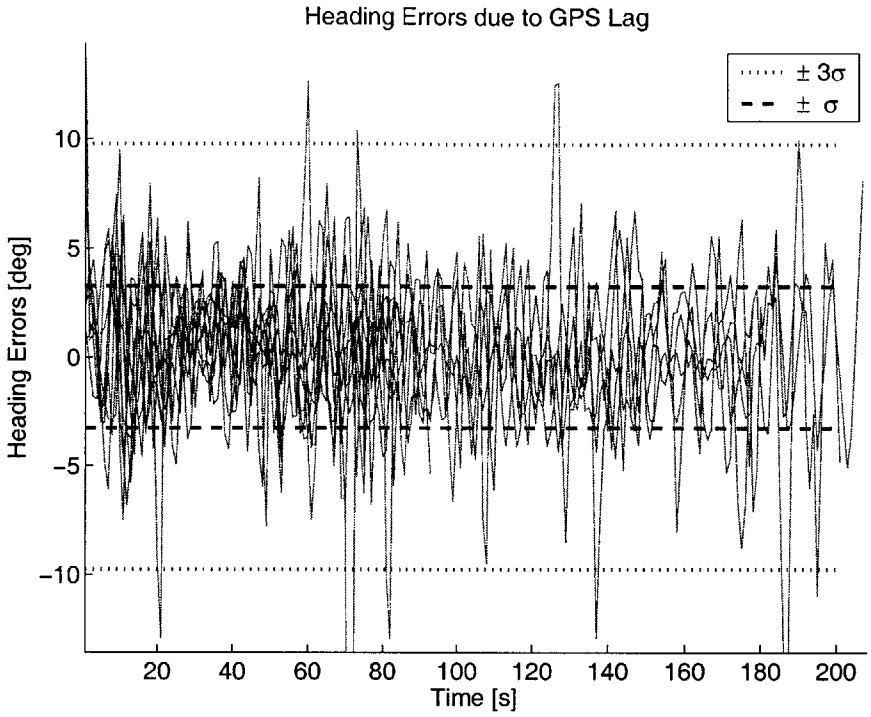


Figure 4-8: The calculated heading errors due to a 1 second lag on the GPS velocity measurements. The $\pm 3\sigma = 9.77$ bounds on the expected uncertainty levels shown are used to approximate the initial heading uncertainty in $\hat{\psi}(kT)$.

measurements, requiring estimation of the velocity at the current time-step. This can be accomplished using a simple discrete time position and velocity model assuming constant acceleration

$$X_k = X_{k-1} + \Delta t V_{k-1} + \frac{a\Delta t^2}{2} \tag{4.17}$$

$$V_k = a\Delta t + V_{k-1} \tag{4.18}$$

Solving for $a\Delta t$ in Eq. 4.17 and substituting into Eq. 4.18 yields the desired expression in terms of available measurements:

$$\hat{V}_k = V_{k-1} + \frac{2(X_k - X_{k-1} - \Delta t V_{k-1})}{\Delta t} \tag{4.19}$$

The constant acceleration assumption in Eq. 4.19 is a source of error since the bank angle dynamics permit the roll angle to vary on the time scale of $\Delta t = 1$ sec. (see

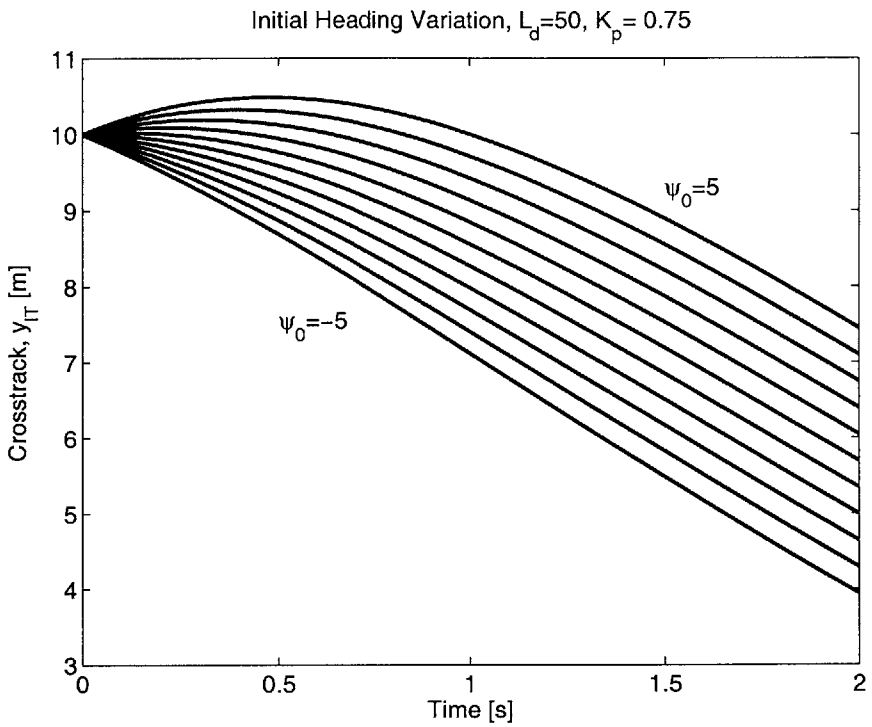


Figure 4-9: Variation of the initial heading angle, $\psi_0 \in (-5, 5)$ deg, in the groundtrack of an aircraft with 25 m/s airspeed, $K_p = 0.75$ and $L_d = 75$ m. The initial heading angle largely determines the estimate of the propagated state estimate at $\lambda T = 1$ sec.

Subsection 2.2.2). The effective heading error as a result of the GPS lag is plotted in Figure 4-8, which shows the heading errors calculated for 10 separate HWIL flights and no wind in the simulation. The $3\sigma = \pm 9.67°$ bounds quantify the expected levels of uncertainty in the initial state, $\hat{\psi}(kT)$.

Aircraft heading measurements, $\hat{\psi}(kT)$, are one of the most significant sources of error for planning, and Figure 4-9 provides some insight into the physical reason for this relatively high sensitivity. Here the vehicle cross-track error, y_{IT}, is shown for various initial conditions in the heading angle, $\hat{\psi}_0$, showing that for even small variations in the initial heading ($\pm 5°$), the closed loop cross-track and heading predictions , $\hat{y}_{IT}$ and $\hat{\psi}_{IT}$ respectively, have large variations over the prediction interval, $\lambda T < 1$ s. Subsection 4.4.2 quantifies the expected propagation errors explicitly as a functions of this initial heading error, and the errors associated with wind disturbances, $\boldsymbol{w}_{IT}$.

Due to the relative accuracy of the GPS position measurements, errors in the cross-track offset, $\tilde{y}_{IT}$, can be neglected. In addition, errors in the roll angle estimate will also be neglected due to high bandwidth roll rate sensors, bias estimation and periodic GPS updates as shown in Subsection 2.1.3 and in Ref. [35]. In the next section the initial condition error on the roll angle, $\tilde{\phi}_0$, will also be shown to contribute relatively little to the uncertainty at future timesteps. From the analysis shown here, the reader should note that the 1 second lag on the GPS velocity states provide the largest source of measurement error, and thus the fundamental limitation on how well the propagated state, $\hat{x}(kT + \lambda T)$, can be known, irrespective of disturbance levels on the system.

4.4.2 Propagation Error

Initial state estimation errors and uncertain disturbances acting on the system cause the propagation estimates, $\hat{x}(kT+\lambda T)$, to have some uncertainty, and the quantification of this uncertainty is important in guaranteeing feasibility of the MILP solution at future timesteps [40]. For the propagation model in Eq. 4.4, define an initial state uncertainty, $\tilde{x}(kT)$, which is bounded by the set $\mathcal{N}$, *i.e.*, $\tilde{x}(kT) \in \mathcal{N} \subset \Re^3$. In addition, uncertain but bounded disturbances will act on the system $\widetilde{W}(kT) \in \mathcal{W} \subset \Re^2$ where $\widetilde{W}$ is analogous to the uncertainty in the wind estimates shown in Subsection 3.1.1, now for the intrack case.

The closed loop dynamics from Eq. 4.4 can be used to form estimates of the propagated state at time $(kT + \lambda T)$

$$x(kT + \lambda T) = e^{A(\lambda T)}x(kT) + \int_{kT}^{kT+\lambda T} e^{A(kT+\lambda T-\tau)}GW(kT)\,\mathrm{d}\tau \tag{4.20}$$

Typical computation times for the MILP trajectory optimization range between 0.1 and 1 sec. Taking $\lambda T = 1$ sec as an overbound and assuming constant wind over the propagation interval, the propagated state an be written as

$$\begin{aligned} x(kT+1) &= e^{A}x(kT) + A^{-1}\left(e^{A} - \mathbf{I}\right)GW \\ &\equiv Fx(kT) + \Gamma W(kT) \end{aligned} \tag{4.21}$$

Solving for the states at the propagated time with the dynamics from Subsection 4.3.1, expressions for the propagated state errors can be formed in terms of the uncertainty in the state variables and input disturbances at time kT. Assuming the state and disturbance uncertainty is constant with time,

$$\begin{bmatrix} \tilde{y}_{IT}(kT+1) \\ \tilde{\psi}_{IT}(kT+1) \\ \tilde{\phi}_{IT}(kT+1) \end{bmatrix} = F\tilde{x} + \Gamma\widetilde{W} \tag{4.22}$$

$$F = \begin{bmatrix} 0.9726 & 25.7349 & -2.6372 \\ -0.0025 & 0.7842 & -0.1336 \\ 0.0102 & 0.9636 & 0.0213 \end{bmatrix} \qquad \Gamma = \begin{bmatrix} 0.1786 & 0.9925 \\ 0.0123 & -0.0010 \\ 0.0082 & 0.0072 \end{bmatrix}$$

where F and Γ are scaled for radian measure of $\tilde{\psi}_{IT}$ and $\tilde{\psi}_{IT}$. From Subsection 4.4.1, the initial condition uncertainty is predominately characterized by uncertainty in the true heading state, $\tilde{\psi}_0$, because the position and bank angle states are relatively well known. The wind direction will not be known a priori, so the worst case error associated with the wind magnitude is found by taking the maximum value for $\Gamma\widetilde{W}$,

$$\tilde{y}_{IT}(kT+1) = |F_{12}\tilde{\psi}_0| + \max_{w_x,w_y\in\mathcal{W}} \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \end{bmatrix} \widetilde{W} \tag{4.23}$$

$$\tilde{\psi}_{IT}(kT+1) = |F_{22}\tilde{\psi}_0| + \max_{w_x,w_y\in\mathcal{W}} \begin{bmatrix} \Gamma_{21} & \Gamma_{22} \end{bmatrix} \widetilde{W} \tag{4.24}$$

$$\tilde{\phi}_{IT}(kT+1) = |F_{32}\tilde{\psi}_0| + \max_{w_x,w_y\in\mathcal{W}} \begin{bmatrix} \Gamma_{31} & \Gamma_{32} \end{bmatrix} \widetilde{W} \tag{4.25}$$

The sets of prediction errors at time $\lambda T = 1$ sec,

$$\tilde{x}(kT+1) \in \mathcal{N} \subset \Re^3 \tag{4.26}$$

are plotted in Figures 4-10(a) – 4-11(a) for varying disturbance bounds, $\mathcal{W}$, and initial heading errors, $\tilde{\psi}_0$. The slope of the planes (defined from the F and Γ matrices in Eq. 4.22) indicate the relative sensitivity of the propagated states to the initial state uncertainty, $\hat{\psi}_0$ and the maximum disturbance bound, $\mathcal{W}$. For the propagation heading errors in Figure 4-10(b) for instance, although the wind disturbances contribute, higher errors are accumulated from the heading error measurements at time (kT). For example, if, as in Subsection 4.4.1, 10° bounds on $\tilde{\psi}_0$ are assumed, the heading

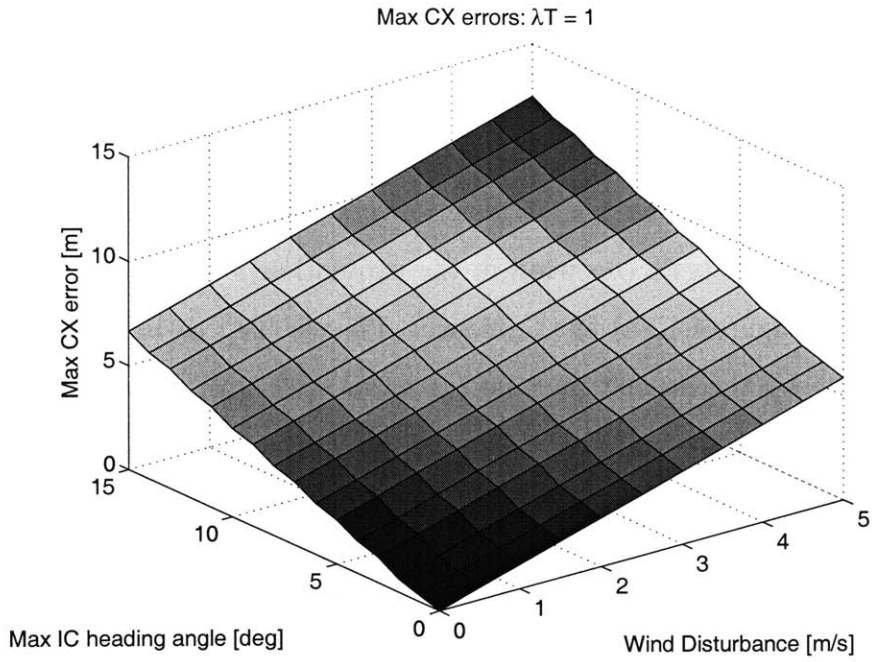
uncertainty in $|\tilde{\psi}(kT+1)|$ can be expected to lie within the range (7.81,11.75), depending on the wind disturbance levels. The sensitivity to roll angle error, $\tilde{\phi}_0$, is also plotted for comparison in Figure 4-11(b). Even large errors in ϕ contribute small errors to the heading and cross-track errors at time $(kT+1)$.

Figure 4-12(a) shows the intrack position prediction errors at time $\lambda T = 1$ sec for a series of HWIL tests (no wind in the simulation). The cross-track bounds as predicted from Figure 4-10(a) at the measured levels (*i.e.*, $\tilde{\phi}_0 = 10$ deg, $\widetilde{W} = [0\ 0]^T$) are also shown by the dashed lines on the cross-track axis. The actual closed loop cross-track errors are shown to lie within the worst case prediction bounds, indicating slightly conservative results in this case. The open loop propagation errors assuming constant heading and velocity and $\lambda T = 1$ sec are also shown for comparison to the closed loop case. The tighter closed loop error distribution about (0,0) indicates much more accurate predictions than the open loop propagations, due to the additional acceleration dynamics in the closed loop model.

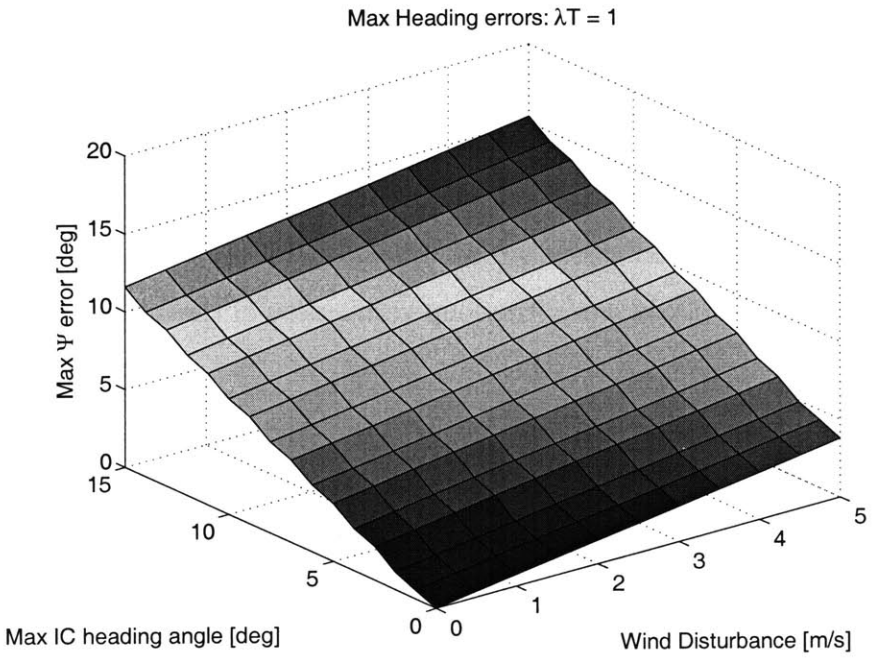
Figure 4-12(b) shows the propagated heading errors measured during several HWIL experiments plotted as a histogram, showing an approximate distribution about $|\tilde{\psi}(kT+\lambda T)|$ with $\sigma = 3.49$ and no wind in the simulation ($\widetilde{W} = [0\ 0]^T$). The bound computed from Eq. 4.21 is also plotted (dashed line), showing the measured prediction errors obtained during simulations are within the $\tilde{\psi}(kT+\lambda T) \in \mathcal{N}|_\psi = 7.15$ bounds. Figure 4-12 provides a hardware-in-the-loop verification of the analytical results found here.

Wind Disturbance Bounds

While initial heading errors contribute some of the uncertainty at state $x(kT+\lambda T)$, as shown in Figures 4-12 - 4-12 the wind disturbances represent a large disturbance source for the trajectory design problem. While in general the wind disturbances acting on the vehicle would not be known, the results from the propagation analysis show the effect given *errors* in the disturbances that can be estimated. Subsection 3.5.2 presented a method for estimating the magnitude of the wind disturbance given velocity and position measurements. With added information about the aircraft

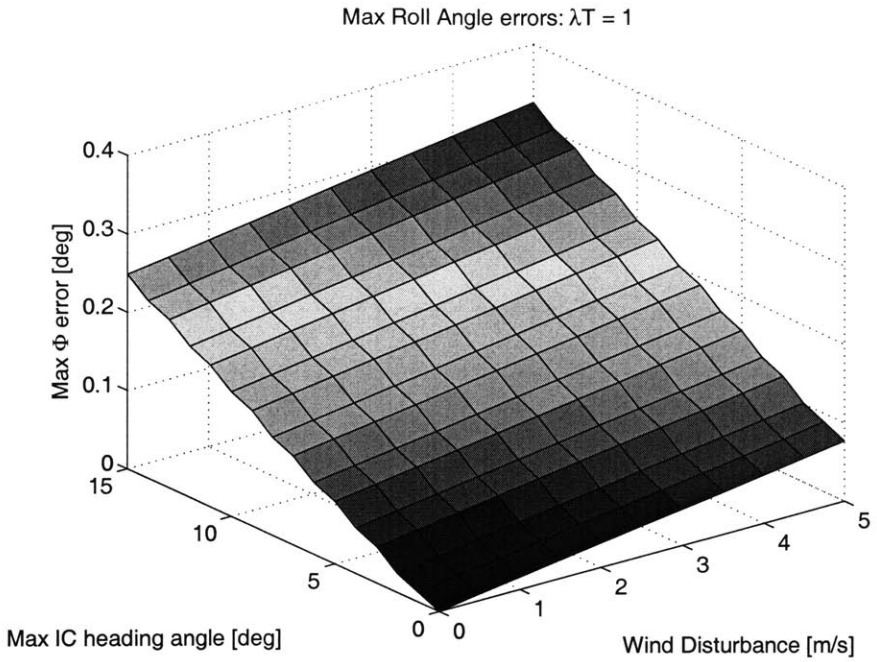


(a) The maximum cross-track error expected at time $(kT + \lambda T)$, for varying bounds on the initial heading error and disturbance magnitude.

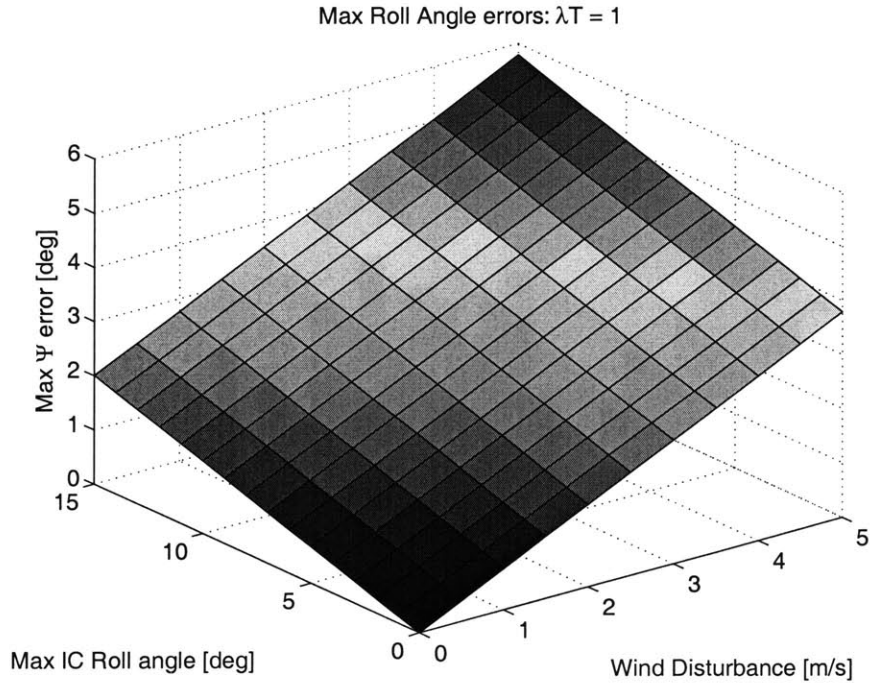


(b) The maximum heading angle error expected at time $(kT + \lambda T)$, for varying bounds on the initial heading error and disturbance magnitude.

Figure 4-10: Calculation of the overbounds on the state prediction errors, $\tilde{x}(kT+\lambda T)$, for the dominant error sources in disturbance levels, $\widetilde{W}$ and initial condition uncertainty $\tilde{\psi}_0$.

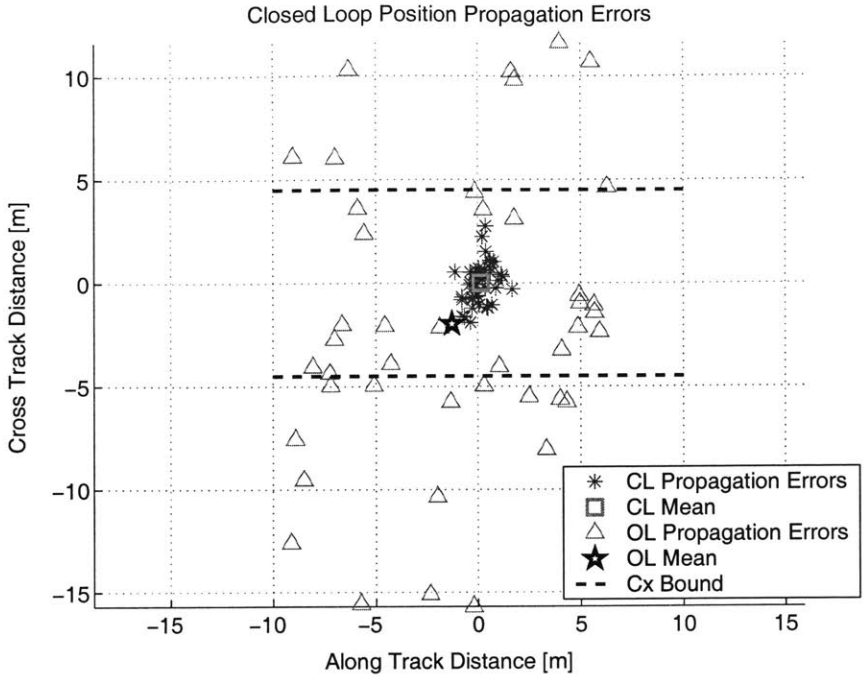


(a) The maximum roll angle error expected at time $(Tk + \lambda T)$, for varying bounds on the initial heading error and disturbance magnitude in m/s. Note the Z axis scale indicates almost no sensitivity.

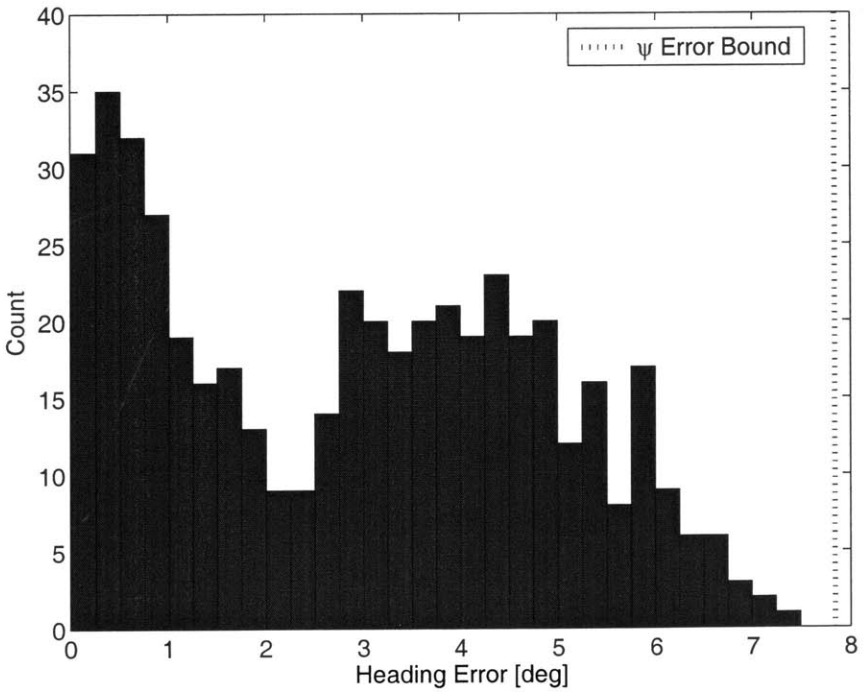


(b) The heading prediction errors as a function of initial roll angle uncertainty, $\tilde{\phi}_0$, and disturbance magnitude, shown for comparison to the case of heading angle uncertainty.

Figure 4-11: Calculation of the overbounds on the state prediction errors for less sensitive parameters, $\tilde{\phi}(kT + \lambda T)$ and $\tilde{\phi}_0$.



(a) The closed loop propagation position errors are shown to remain within the ± 4.5 m bounds (dashed lines) for the initial heading uncertainty, $|\tilde{\psi}_0| \leq 10$ deg, and no wind disturbance $\boldsymbol{W} = [0\ 0]^T$. The open loop state propagation without closed loop dynamics is shown for comparison, $\lambda T = 1$ sec.



(b) Closed loop heading propagation errors plotted as a histogram for an experiment with no wind on the HWIL testbed. The magnitude of the errors are shown to lie below the $|\tilde{\psi}(kT + \lambda T)_{max}| \leq 7.81$ deg bounds from an initial heading uncertainty of $|\tilde{\psi}_0| \leq 10$ degrees.

Figure 4-12: Position and heading propagation errors as measured over 12 HWIL simulations for validation of the analytical bounds computed in Subsection 4.4.2.

heading this could be extended to form estimates of the inertial magnitude components $\boldsymbol{W} = [w_x \; w_y]^T$ and used to potentially reduce the effect of the error due to disturbances on the system.

4.5 Constraint Tightening

Given information about the expected prediction errors, the constraints for the MILP optimization need to be tightened in order to ensure worst case robust feasibility. This amounts to allowing added margin to the account for the state uncertainty at time $(kT + \lambda T)$ so that the low level controller has enough authority to correct for unplanned disturbances acting on the system. The approach here is to compute the added margin required on each of the constraints based on the bounds computed for the prediction step $\hat{x}(kT + \lambda T)$.

4.5.1 Turn Radius Constraints

Given a maximum bank angle, $\phi_{\max} = 30°$, the theoretical minimum turn radius capable for the vehicle at $v_a = 26$ m/s (average airspeed) is

$$\rho_{\min} = \frac{v_a^2}{g \tan(30°)} \approx 120\text{m} \tag{4.27}$$

Note that $\rho_{\min}$ is used to define the tracker convergence parameter, L_d in Figure 4-5, as well. From Figure 4-13, the turn radius is related to the maximum discrete heading deviation, $\Delta\psi$, by the relation

$$\Delta\psi = 90° - \arccos\left(\frac{v_a T}{2\rho}\right) = \arcsin\left(\frac{v_a T}{2\rho}\right) \tag{4.28}$$

where $v_a T$ defines the step length between discrete points in the MILP trajectory optimization. For a plan step length of 100 m ($T \approx 4$ sec), the theoretical maximum discrete heading deviation at any plan point is defined by the minimum turn radius achievable by the vehicle

$$\Delta\psi|_{max} = \arcsin\left(\frac{100\text{m}}{2\rho_{\min}}\right) \approx 25° \tag{4.29}$$

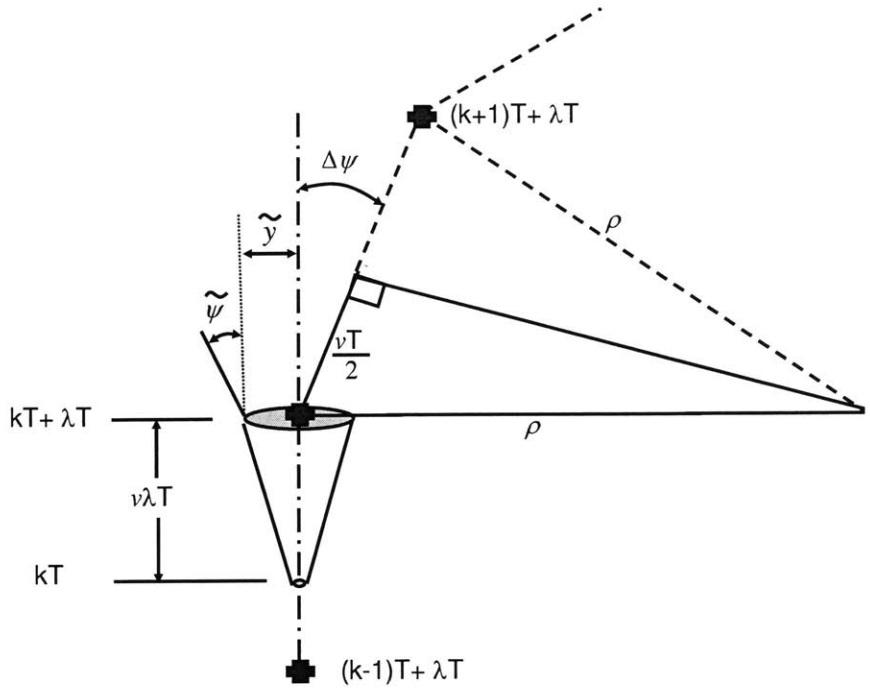


Figure 4-13: Position and heading propagation errors, $(\tilde{y}(kT+\lambda T), \tilde{\psi}(kT+\lambda T))$ generate bounds on the minimum turn radius, ρ, used to calculate the MILP trajectory. Note: propagation and uncertainties are not to scale.

However due to the uncertainty at time $(kT+\lambda T)$, there will be a deviation between the planned and flown paths, and some turn rate authority will be required at the vehicle level to make sure that the plan is flyable. In order to guarantee that the vehicle will reach the plan points at future time-steps, the planner turn radius, ρ_p, must be chosen to reserve some authority for the low level vehicle controller. This amounts to scaling the planner turn radius such that it is consistent with the expected uncertainty at the propagated time $(kT+\lambda T)$

$$\rho_p = \frac{1}{\Upsilon}\rho_{\min}, \qquad \Upsilon \in [0,1) \tag{4.30}$$

where Υ is the turn radius scale factor relating the state uncertainty, $\tilde{x}(kT+\lambda T)$ to the minimum turn rate constraints. As $\Upsilon \to 0$, the planner loses authority to design trajectories, however the low level vehicle controller will gain the maximum turn rate authority to reject disturbances. Conversely, $\Upsilon = 1$ corresponds to the case when the planner is given full authority to design a trajectory for maximum maneuverability,

but no margin is left for uncertainty or disturbances acting on the system.

From Eqns. 4.28 and 4.30, Υ is related geometrically to the minimum turn radius, $\rho_{\min}$, the planning step size, vT, and the maximum discrete heading angle, $\Delta\psi$,

$$\Upsilon = \frac{\rho_{\min}}{\rho_p} = \frac{2\rho_{\min}\sin(\Delta\psi)}{vT} \tag{4.31}$$

$$\Delta\psi = \Delta\psi|_{\max} - \tilde{\psi}(kT + \lambda T) \tag{4.32}$$

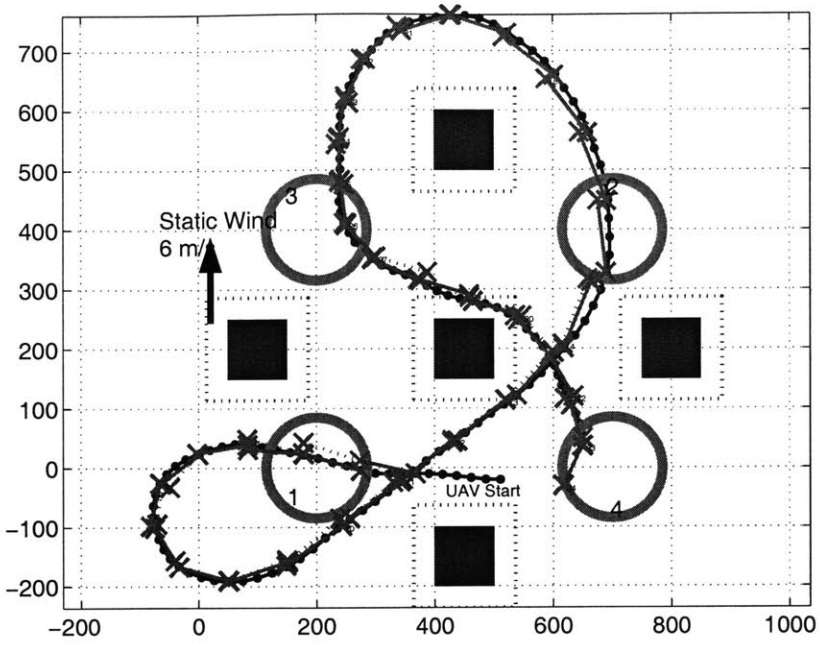
where $\Delta\psi$ is chosen to be the maximum vehicle authority, minus the heading angle uncertainty computed for the propagation time as shown in Figure 4-13. By choosing the planning turn radius consistently with Eqns. 4.30 and 4.31, the vehicle is guaranteed to have enough authority to compensate for the worst case heading errors and disturbance levels at the MILP initial condition, $\hat{x}(kT + \lambda T)$.

As seen in the plots of the heading angle error in Figure 4-11(a), or calculated using Eq. 4.22 with $\tilde{\psi}_0 = 10°$ and 5 m/s wind disturbance, the expected heading uncertainty is computed as $\tilde{\psi}(kT+\lambda T) \approx 10°$. Using Eq. 4.31, the best case planning radius is then

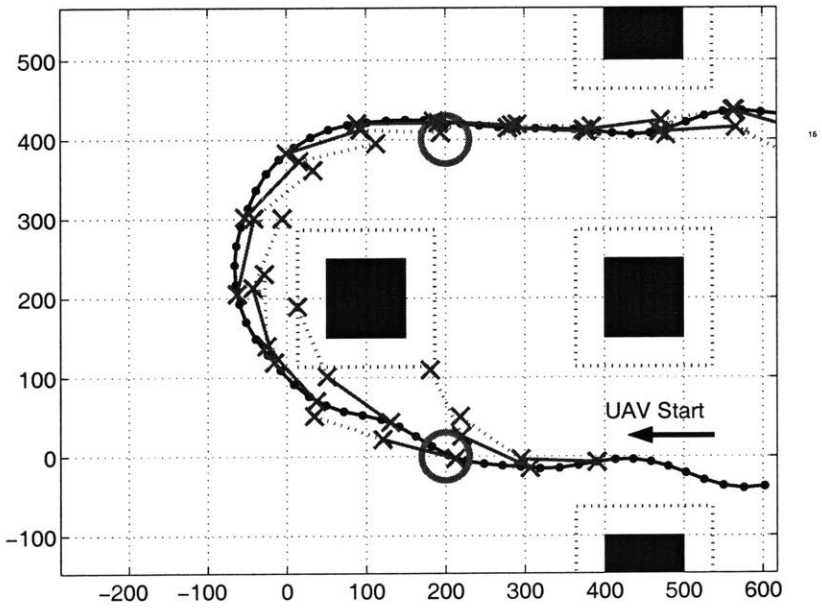
$$\rho_p \approx 200\text{m}, \qquad \Upsilon \approx 0.60 \tag{4.33}$$

The added margin in the selection of the planner turn radius, ρ_p, provides a means to account for uncertainty in the initial state of the MILP optimization, yet is still provides enough authority to the planner to design reasonably maneuverable trajectories. These trajectories utilize complex maneuvers to navigate through a detailed obstacle field, requiring maximum turn rate constraints to be enforced.

Figure 4-14(a) shows the vehicle maneuvering with maximum turn rate constraints, and with wind and turbulence effects producing large disturbance sources ($\mathcal{T}_{WV_a} = 0.25$). With $\rho_p = 200$ m selected as the minimum planning turn radius, the MILP trajectory plans are consistently flown by the vehicle without having to drastically reset the plan position and heading at the start of each optimization. This indicates that the low level vehicle controller is capable of executing the plans that are designed, and that enough authority was provided to it to overcome the uncertainty in the initial conditions of the vehicle state. The true static wind estimate, $\overline{W}$, was provided to the planner (through Eq. 4.3) in this simulation, indicating that dynamically



(a) The vehicle is capable of executing tight maneuvers and obstacle avoidance in the presence of a static wind disturbance at 25% of the aircraft speed. Wind vector 6 m/s from the south added. The plan turn radius, $\rho_p = 200$ m, is used to design the trajectories



(b) The effect of a small a plan turn radius, ρ_p, on the system performance ($\Upsilon \rightarrow 1$). The vehicle is not able to reach the desired states the planner designs, indicating there is too much authority on the planning level. $\rho_p = 160$ m, confirming predictions from Subsection 4.5

Figure 4-14: Receding Horizon Simulation with turbulence, sensor noise, actuator delay acting on the system. Planned data marked with solid 'x' segments for the current plan being flown, and dashed 'x' segments for the next plan step. Vehicle telemetry is also shown, and the goals are marked with circles and ordered in sequence shown by the numbers in Figure 4-14(a).

feasible plans can be designed given suitably accurate information and validating the predictions from Eq. 4.31.

Increasing the value of Υ increases the maneuverability of the plans that can be designed, however due to the uncertain states at the start of the new plan, the maximum authority will rarely be applied and the vehicle will tend to divert from the optimal MILP trajectory. This type of plan is characterized by a "drift" in the plans that are designed as the vehicle continually misses each of the plan points. Figure 4-14(b) shows the effect of too small a planner turn radius, with $\rho_p = 160$ m, again, validating predictions that the minimum turn radius should be set according to Eq. 4.31.

4.6 Conclusion

Planning for UAVs in a real world environment with uncertainty and disturbances creates much more difficulty, and the HWIL testbed has helped to test and validate some of the assumptions about planning for a multi-vehicle system. The MILP dynamics model was extended to account for the bank angle dynamics of the aircraft, allowing for more dynamically consistent trajectories to be generated by the planning system. The effect of uncertainty and disturbances have been classified and analyzed, allowing the limits of the planning system to be determined, with constraint tightening being employed to further increase the robustness of the system.

Although large errors in the wind estimation error have been shown to cause the design of dynamically inconsistent trajectories, estimation techniques such as those presented in Section 3.2 are available to improve these errors. In particular, by utilizing an onboard magnetometer as an added measurement source, the aircraft heading error as well as the wind estimation error can be drastically reduced.