



AUTOMATED CARRIER LANDING OF AN
UNMANNED COMBAT AERIAL VEHICLE
USING DYNAMIC INVERSION

THESIS

Nicholas A. Denison, Ensign, USN

AFIT/GAE/ENY/07-J06

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Navy, the United States Air Force, the Department of Defense, or the United States Government.

AFIT/GAE/ENY/07-J06

AUTOMATED CARRIER LANDING OF AN
UNMANNED COMBAT AERIAL VEHICLE
USING DYNAMIC INVERSION

THESIS

Presented to the Faculty
Department of Aeronautical and Astronautical Engineering
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
In Partial Fulfillment of the Requirements for the
Degree of Master of Science in Aeronautical Engineering

Nicholas A. Denison, B.S.

Ensign, USN

June 2007

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AUTOMATED CARRIER LANDING OF AN
UNMANNED COMBAT AERIAL VEHICLE
USING DYNAMIC INVERSION

Nicholas A. Denison, B.S.
Ensign, USN

Approved:

/signed/

25 May 2007

Maj Chris Shearer, Ph.D. (Chairman)

date

/signed/

25 May 2007

Dr. Brad Liebst (Member)

date

/signed/

25 May 2007

Maj Paul Blue, Ph.D. (Member)

date

Abstract

Dynamic Inversion (DI) is a powerful nonlinear control technique which has been applied to several modern flight control systems. This research utilized concepts of DI in order to develop a controller to land an Unmanned Combat Aerial Vehicle (UCAV) on an aircraft carrier. The Joint Unmanned Combat Air System (J-UCAS) Equivalent Model was used as the test aircraft. An inner-loop DI controller was developed to control the pitch, roll, and yaw rate dynamics of the aircraft, while an outer-loop DI controller was developed to provide flight path commands to the inner-loop.

The controller design and simulation were conducted in the MATLAB[®]/Simulink[®] environment. Simulations were conducted for various starting positions near the carrier and for varying wind, wind turbulence, and sea state conditions. In the absence of wind and sea state turbulence, the controller performed well. After adding wind and sea state turbulence, the controller performance was degraded. Future work in this area should include a more robust disturbance rejection technique to compensate for wind turbulence effects and a method of carrier motion prediction to compensate for sea state effects.

Acknowledgements

There were many individuals which helped to make this research possible. First, I would like to thank the members of the AFRL/VACC directorate for their support on the aircraft model, especially Bill Blake, Jacob Hinchman, and Findlay Barfield. I would also like to thank the members of the NAVAIR N-UCAS Program Office for their support on the ship motion and burble models, especially Glenn Colby and Keith Carter. I am especially grateful for Dr. Brad Liebst and Maj Paul Blue for their helpful suggestions during the process of refining this document. Thanks are also due to Dr. Eric Hallberg of the USNA Aerospace Engineering Department for inspiring me to continue learning about aircraft control and suggesting this project idea. However, no one had more of an impact on my ability to successfully complete this research than my advisor, Maj Chris Shearer. His knowledge, expertise, professionalism, and cheerful prodding for documented results over the past months helped me to stay motivated and on task. To my family and friends whom are too numerous to name, thank you for all of your support and understanding throughout this arduous task.

Nicholas A. Denison

Table of Contents

	Page
Abstract	iv
Acknowledgements	v
List of Figures	ix
List of Tables	xi
List of Symbols	xii
List of Abbreviations	xv
I. Introduction	1
1.1 Naval Unmanned Combat Aerial Vehicles	1
1.2 Aircraft Carrier Landings	2
1.3 Dynamic Inversion	3
1.4 Research Objectives	4
1.5 Thesis Overview	4
II. Theoretical Development	5
2.1 Previous Research and Motivation	5
2.2 Aircraft Carrier Landing Considerations	6
2.3 Dynamic Inversion Control Structure	8
2.3.1 Precompensation	9
2.3.2 Control Variable Definition	9
2.3.3 Desired Dynamics	10
2.3.4 Dynamic Inversion	10
2.4 Environmental Assumptions	11
2.4.1 Horizontal Wind Model	12
2.4.2 Dryden Wind Turbulence Model	12
2.4.3 Burble Model	13
2.4.4 Sea State Model	13
2.5 Equations of Motion	14
2.5.1 Coordinate Systems	15
2.5.2 Aircraft Equations of Motion	16
2.5.3 Aircraft Carrier Equations of Motion	18
2.6 Overview of Dynamic Inversion Controller	18
2.7 Inner-Loop Dynamic Inversion Flight Controller	19

	Page
2.7.1 Relevant Equations of Motion	19
2.7.2 Control Variable Definition	22
2.7.3 Desired Dynamics	23
2.7.4 Inversion	23
2.8 Outer-Loop Dynamic Inversion Homing Controller . . .	24
2.8.1 Relevant Equations of Motion	24
2.8.2 Control Variable Definition	25
2.8.3 “Time” Domain to “Distance” Domain	28
2.8.4 Desired Dynamics	30
2.8.5 Inversion	30
2.9 Transitional Dynamics	33
2.9.1 Heading Angle to Roll Rate	33
2.9.2 Pitch Angle to Pitch Rate	34
III. Modeling and Implementation	35
3.1 Aircraft Model	35
3.1.1 Physical Layout	36
3.1.2 Physical Parameters	37
3.1.3 Actuator Models	37
3.1.4 Engine Model	37
3.1.5 Aerodynamic Model	39
3.2 Aircraft Carrier Model	41
3.2.1 Translational Motion Model	42
3.2.2 Sea State Perturbation Model	42
3.3 Burble Model	44
3.4 Simulator Structure	45
3.4.1 Solver	45
3.4.2 Aircraft Equations of Motion	45
3.4.3 Ship Equations of Motion	46
3.4.4 Sea State Perturbations	46
3.4.5 Aerodynamic Look-Up Tables	47
3.4.6 Engine Model	47
3.4.7 Actuator Models	48
3.4.8 Horizontal Wind Model	49
3.4.9 Wind Turbulence Model	49
3.4.10 Burble Model	50
3.4.11 Sensor Noise	52
3.4.12 Controller Model Look-Up Tables	52
3.4.13 State Estimators	56
3.4.14 Inner-Loop Controller	58
3.4.15 Outer-Loop Controller	62
3.4.16 Transitional Dynamics	69

	Page
IV. Simulation and Results	70
4.1 Monte Carlo Simulation	70
4.2 Sample Simulation Run	72
4.3 Simulation Case 1	76
4.4 Simulation Case 2	77
4.5 Simulation Case 3	79
4.6 Simulation Case 4	80
4.7 Simulation Case 5	81
4.8 Simulation Case 6	82
4.9 Simulation Case 7	84
4.10 Simulation Case 8	85
4.11 Simulation Case 9	86
4.12 Simulation Case 10	88
4.13 Simulation Case 11	89
4.14 Overall Analysis	90
V. Conclusions and Recommendations	95
5.1 Conclusions	95
5.2 Recommendations	96
Bibliography	98
Vita	101

List of Figures

Figure		Page
2.1.	Nimitz Class Carrier Landing Area.	7
2.2.	Generic Dynamic Inversion Control Structure.	9
2.3.	Ship Degrees of Freedom.	14
2.4.	Aircraft Body-Fixed Coordinate Axes.	15
2.5.	Inertial Reference Frames Used in Simulation.	16
2.6.	Overview of Control Structure.	18
2.7.	<i>HCV</i> Figures.	26
2.8.	Definition of θ_{vgd}	27
2.9.	Illustration of “Distance Domain.”	29
3.1.	X-45C, X-47B, and ICE.	35
3.2.	Physical Layout of EQ Model.	36
3.3.	Nimitz Class Aircraft Carrier.	41
3.4.	Conceptual Simulator Block Diagram.	45
3.5.	Layout of Aircraft Equations of Motion Block.	46
3.6.	Layout of Ship Equations of Motion Block.	46
3.7.	Layout of Sea State Perturbation Block.	47
3.8.	Layout of Aerodynamic Look-Up Tables.	48
3.9.	Layout of Horizontal Wind Model Block.	49
3.10.	Turbulence Intensities.	51
3.11.	Layout of Burble Model Block.	52
3.12.	Conceptual Dynamic Inversion Controller Block Diagram. . . .	53
3.13.	Layout of Inner-Loop Controller.	59
3.14.	Illustration of Limited Integral Effects.	61
3.15.	Layout Outer-Loop Controller.	63
3.16.	Layout of Transitional Dynamics.	69

Figure		Page
4.1.	Sample Simulation Trajectory.	73
4.2.	Sample Control Variable Response.	74
4.3.	Sample Control Surface Response.	75
4.4.	Case 1 Landing Dispersion.	77
4.5.	Case 2 Landing Dispersion.	78
4.6.	Case 3 Landing Dispersion.	79
4.7.	Case 4 Landing Dispersion.	81
4.8.	Case 5 Landing Dispersion.	82
4.9.	Case 6 Landing Dispersion.	83
4.10.	Case 7 Landing Dispersion.	85
4.11.	Case 8 Landing Dispersion.	86
4.12.	Case 9 Landing Dispersion.	87
4.13.	Case 10 Landing Dispersion.	88
4.14.	Case 11 Landing Dispersion.	89

List of Tables

Table		Page
3.1.	Summary of Aircraft Physical Parameters.	37
3.2.	Summary of Actuator Parameters.	38
3.3.	Summary of Engine Specifications.	38
3.4.	Summary of Aerodynamic Look-Up Tables.	40
3.5.	Sign Conventions for Aerodynamic Coefficients.	40
3.6.	Summary of Aircraft Carrier Dimensions.	42
3.7.	Summary of Sea State Perturbations.	43
3.8.	Summary of Turbulence Intensities.	51
3.9.	Summary of Sensor Noise Amplitudes.	53
3.10.	Summary of Trim Conditions for Estimator Design.	57
3.11.	Summary of Inner-Loop PID Gains and Integral Limits.	62
4.1.	Summary of Simulation Initial Conditions.	71
4.2.	Summary of Starting Positions Considered.	71
4.3.	Summary of Simulation Environmental Conditions.	71
4.4.	Summary of Landing Dispersion Statistics.	92
4.5.	Summary of Landing Classification Statistics.	92
4.6.	Summary of Total Traps and Misses.	92
4.7.	Summary of USIRD Autonomous Vehicle Landing Requirements.	93

List of Symbols

Symbol		Page
u	x velocity in the body axes	16
v	y velocity in the body axes	16
w	z velocity in the body axes	16
x_n	x position (North-East-Down)	16
y_n	y position (North-East-Down)	16
z_n	z position (North-East-Down)	16
g	acceleration due to gravity	16
W	weight	16
F_{xb}	aerodynamic forces resolved in body axes (etc.)	16
M_{xb}	aerodynamic moments resolved in body axes (etc.)	16
S_θ	$\sin \theta$ (etc.)	16
C_θ	$\cos \theta$ (etc.)	16
p	roll rate	16
q	pitch rate	16
r	yaw rate	16
I_{xxb}	moment of inertia resolved in body axes (etc.)	16
β	sideslip angle	17
ρ	density	17
T	thrust	17
S	wing planform area	17
b	wingspan	17
\bar{c}	mean aerodynamic chord	17
C_N	normal force coefficient	17
C_A	axial force coefficient	17
$C_{Y,\beta}$	partial of sideforce with respect to sideslip (etc.)	17

Symbol		Page
C_ℓ	roll moment coefficient	17
C_m	pitch moment coefficient	17
C_n	yaw moment coefficient	17
V_B	magnitude of ship speed	18
x_{Be}	ship x coordinate (aimpoint)(East-North-Up)	18
y_{Be}	ship y coordinate (aimpoint)(East-North-Up)	18
ψ_B	ship heading angle	18
τ	engine time constant	19
T_{max}	instantaneous maximum thrust	19
k	stabilizing pitch feedback gain	19
δ_f^*	pitch flap command to stabilizing loop	19
δ_e	mean elevon deflection	20
δ_f	pitch flap deflection	20
δ_c	effective clamshell deflection	20
δ_t	throttle setting	20
\bar{C}_m	Taylor series backward difference in pitch moment (etc.) .	20
Q	dynamic pressure	21
LCV	roll-axis control variable	22
MCV	pitch-axis control variable	22
NCV	yaw-axis control variable	22
V_{co}	crossover velocity	22
α	angle of attack	22
XCV	x -axis control variable	23
V_{wxe}	x wind component (East-North-Up)(etc.)	25
ψ_{BF}	flight deck heading angle	25
θ_{hgd}	desired horizontal glidepath angle	25
V_{xe}	aircraft speed over ground in x direction (East-North-Up)	25
V_{HE}	hook engagement velocity	26

Symbol		Page
θ_{hga}	actual horizontal glide angle	26
HCV	heading control variable	27
θ_{vgd}	desired glideslope angle	27
θ_{vga}	actual glideslope angle	27
V_{ze}	aircraft vertical velocity	27
PCV	pitch control variable	28
γ	flight path angle	28
TCV	thrust control variable	28
τ_d	distance constant	29
\bar{x}	distance from aircraft to ship	31
\bar{V}	magnitude of aircraft velocity	34
M	Mach number	40
x_{BeCM}	ship x coordinate (center of mass)(East-North-Up)	42
y_{BeCM}	ship y coordinate (center of mass)(East-North-Up)	42
ϕ_B	ship roll angle	44
θ_B	ship pitch angle	44
Δx_{Be}	Δx between ship center of mass and aimpoint (etc.)	44
$u_{burble,k}$	x burble component resolved in keel axes (etc.)	50
δ_{eR}	right elevon deflection	54
δ_{eL}	left elevon deflection	54
$\bar{\delta}_{eR}$	center of Taylor series expansion (etc.)	54
\hat{x}	state estimate vector	56
L	estimator gain matrix	56
e	state error vector	56
Q	state weighting matrix	65
R	control weighting matrix	65
σ	standard deviation	76

List of Abbreviations

Abbreviation		Page
UCAV	Unmanned Combat Aerial Vehicle	1
N-UCAS	Navy Unmanned Combat Air System	1
NAVAIR	Naval Air Systems Command	1
IOC	Initial Operational Capability	2
ACLS	Automatic Carrier Landing System	2
JPALS	Joint Precision Approach and Landing System	2
DI	Dynamic Inversion	3
STAV	Supersonic Tailless Air Vehicle	3
FACLS	Fuzzy Logic Automatic Carrier Landing System	5
LSO	Landing Signal Officer	6
CV	Control Variable	9
PID	Proportional plus Integral plus Derivative	10
LQR	Linear Quadratic Regulator	10
DCM	Direction Cosine Matrix	12
EOM	Equations of Motion	14
NED	North-East-Down	15
ENU	East-North-Up	15
J-UCAS	Joint Unmanned Combat Air System	35
EQ	J-UCAS Equivalent Model	35
AFRL	Air Force Research Laboratory	35
AAR	Automated Aerial Refueling	35
ICE	Innovative Control Effector	35
WOD	Wind-Over-Deck	44
KTAS	Knots True Airspeed	71
CI	Confidence Interval	76
USIRD	UCAS Shipboard Interface Reference Document	91

AUTOMATED CARRIER LANDING OF AN UNMANNED COMBAT AERIAL VEHICLE USING DYNAMIC INVERSION

I. Introduction

1.1 Naval Unmanned Combat Aerial Vehicles

The future of Naval Aviation includes the integration of Unmanned Combat Aerial Vehicles (UCAVs) into carrier air wings to supplement flight operations. Piloted aircraft will still be used for most missions, especially those which require a human to have eyes-on-target for identification, but the presence of UCAVs will give battle planners more options and resources in order to carry out mission tasking. Because there are no human factors associated with UCAVs and they only require periodic refueling, they can fly missions much longer than a single pilot could fly because of fatigue and other physiological constraints. This makes them ideal for surveillance and reconnaissance missions. The fact that they will incorporate stealth technology makes them ideal for prosecuting hard targets which are heavily defended, such as runways, command and control (C2) facilities, and surface-to-air (SAM) sites. Most importantly, they can be sent on extremely high-risk missions which do not require a human to have eyes-on-target. This accomplishes the mission without putting pilots' lives at risk, allowing them to fight another day.

The Navy Unmanned Combat Air System (N-UCAS) is the Naval Air Systems Command (NAVAIR) program in charge of the development and procurement of the first generation of Navy UCAVs. The two aircraft currently in the competition are the Boeing X-45C and the Northrop Grumman X-47B. These will both be capable of carrying out surveillance and reconnaissance, strike, and suppression of enemy air defenses (SEAD) missions. This will be accomplished in two different variants. The first variant to become operational will be the surveillance and reconnaissance variant,

with an estimated initial operational capability (IOC) of 2015. The strike/SEAD variant will become operational later, with an estimated IOC of 2020 [17, pages 42–43].

1.2 Aircraft Carrier Landings

Landing on an aircraft carrier requires a great deal of pilot skill which is only acquired through rigorous training and procedural mastery. Not only is the aircraft moving, but so is the carrier. The pilot must aim for a single spot on the flight deck with a very small margin for error. If he lands too long, then he must go around and try again, and if he lands too short, then he just barely missed crashing into the ramp, the aft end of the flight deck. The desired landing area of the flight deck is very small, measuring approximately 55 feet wide by 40 feet long. Anything more than a small deviation from this desired landing box can mean hitting other aircraft on the flight deck or worse. To add even more complexity, the landing portion of the flight deck is angled to the port side of the ship. Adding all of this together means that the pilot is aiming for a single spot on a very small “landing strip” in the middle of the ocean that is moving diagonally away from him and to his right.

In daylight with calm seas and light winds, this may not seem like a very complicated issue. However, carrier flight operations are necessary for the support of combat operations and must take place whenever they are needed. This includes at night, during storms, and all other adverse conditions. With all of the effects which complicate landing on an aircraft carrier, designing a system which automates the carrier landing process is extremely complicated and cannot be taken lightly.

While there are already automated carrier landing systems currently in use, such as the Automatic Carrier Landing System (ACLS) and the Joint Precision Approach and Landing System (JPALS), they do not currently meet the specifications for recovery of autonomous aircraft [22]. Also, these systems cannot currently be operated in all conditions. This is not a problem for manned aircraft because the pilot can

still land. However, an automated carrier landing system for UCAVs must be able to operate in all conditions.

1.3 Dynamic Inversion

Dynamic Inversion (DI) is a nonlinear control technique in which a nonlinear model of the plant is used in an attempt to effectively cancel its dynamics. Once the plant dynamics have been cancelled, new desired dynamics are formulated to replace the old ones. The method by which this is most usually accomplished was developed by Honeywell Technology Center in conjunction with Lockheed Martin and Wright Laboratory [15]. The method uses the plant model to mathematically make the dynamics of the plant-inversion combination equal to those of a free integrator. The specification of the desired dynamics then becomes only a matter of choosing an effective time constant by using a common classical linear controller.

The benefit of using a DI controller over a traditional gain-scheduled linear controller is that DI takes changing flight conditions into account in the plant model. This eliminates the need for the time-consuming and complicated process of developing a robust gain-scheduled controller. Also, because the desired dynamics are chosen for the assumption of always attaining a free integrator for the plant-inversion model, the performance of the system will remain robust as long as the model used in the inversion remains sufficiently accurate.

Recent advancements in on-board computational power available to aerospace systems have made the computation-intensive process of DI a reality in today's defense aerospace industry. The F-22A Raptor uses a dynamic inversion flight controller. The F-35 Lightning II, currently in initial production, also utilizes a dynamic inversion flight controller. The U.S Air Force's Supersonic Tailless Air Vehicle (STAV), currently in conceptual development, also uses a proposed dynamic inversion controller.

1.4 Research Objectives

As previously noted, the first operational UCAV has an estimated IOC of 2015. This is a very short time horizon in which to develop a functional, robust, fully automated carrier landing controller. The objective of this research was to design, simulate, and evaluate an automated carrier landing system utilizing a nonlinear Dynamic Inversion control architecture to achieve acceptable performance and robustness for a wide range of sea state and atmospheric conditions. The goal was to improve upon the performance and robustness of ACLS and JPALS by meeting the specifications for recovering autonomous aircraft at sea [22]. This included formulating a controller to control the aircraft as well as determining whether or not the specifications were met, and if not, determining what improvements would make it possible to meet the specifications.

1.5 Thesis Overview

Chapter 2 details the theoretical development for the DI control strategy. The assumptions made by this research are addressed, as well as the specific formulations of the equations of motion and controllers. Chapter 3 expands upon the various models that were used in the simulation, including the UCAV model, ship model, and turbulence model. Discussions of the simulator structure and simulation plan are also included. Chapter 4 includes the results of the various simulation cases and provides a statistical overview of the results. These statistics are then compared to the specifications for carrier landings by autonomous aircraft. Chapter 5 contains the conclusions drawn from the data and the recommendations for future work in this area.

II. Theoretical Development

2.1 *Previous Research and Motivation*

While the topic of automated carrier landing has been around for several decades, most of the research done in the area has focused mainly on classical and robust linear control methods [9, 29, 30]. Most of these have utilized some type of altitude rate and glideslope feedback as their basis. Very little research has been done in the area of applying nonlinear controllers to this problem. Even fewer have attempted to apply dynamic inversion. As aerospace systems are inherently nonlinear, using nonlinear controllers allows for robust stability and performance without the need for gain scheduling. While early embedded shipboard and aircraft computers may not have had the computing power necessary for the implementation of nonlinear controllers, modern computers do. Thus, nonlinear control should be the next logical step for automated carrier landing systems.

One of the few previous research endeavors into nonlinear aircraft carrier landing systems was by Steinberg [27]. He expanded on his investigation into the application of a fuzzy logic carrier landing system. Calling it the Fuzzy Logic Automatic Carrier Landing System (FACLS) [27, page 407], he showed that nonlinear control can yield similar, if not better results than the current ACLS. One of the main reasons he cited for attempting to use a nonlinear controller is to address “key pilot concerns” [27, page 407] that were not addressed by linear controllers. The main concern was the ability of the controller to adapt to changing conditions. While he concluded that only real-time simulation and actual testing could show whether or not pilot concerns were adequately addressed, the initial results were promising [27].

Slightly more relevant to this research was another conference paper written by Steinberg and Page [28] in which they compared the baseline performance of several different nonlinear control schemes applied to automated carrier landing. They investigated multiple schemes, including dynamic inversion, fuzzy logic, and neural networks. Their major findings showed that all of the different control approaches yielded an approximate 90% landing rate, with anywhere from 70–80% of all runs for

each controller classified as excellent landings by their criteria. This showed that it is very likely that nonlinear control may be a very good option for future carrier landing systems [28].

2.2 Aircraft Carrier Landing Considerations

The goal of any carrier landing controller, be it human or computer, is to land an aircraft at a certain point with minimal room for error on a moving ship. As previously addressed, there are many factors which can greatly complicate this process. Between weather, sea state, and any other conditions which may present themselves, just about anything can go wrong. Therefore, the Landing Signal Officers (LSOs), responsible for monitoring the approaches of landing aircraft, are very strict in ensuring that approach procedures are followed very closely.

Before addressing the specific considerations for landing, the layout of the flight deck is presented. Figure 2.1 illustrates the layout of the landing area of the flight deck of a Nimitz Class aircraft carrier. The aft end of the ship is on the left side of the figure. Notice that there are 4 wires spaced 40 feet apart. They are numbered in increasing order from the one most aft, i.e. 1 wire, 2 wire, etc.

It is desired that aircraft catch the 3 wire when landing. This research assumed ± 20 feet from each wire to be a catch on that wire. Therefore, the aimpoint for this research was assumed to be the point where the number 3 wire intersects the centerline of the landing area. Although technically the position of the aircraft in the simulation was the center of mass, the aimpoint can be shifted in real life to match the actual dimensions of the aircraft such that the ideal hook touchdown point is between the 2 wire and 3 wire. The important part is that there is a window 40 feet long for the hook to catch each of the wires.

The width of the runway area is 65 feet. It is desired that the center of the aircraft stay in the middle portion of the runway so that its wings do not interfere with other operations on the flight deck. Since the foul line on the flight deck is

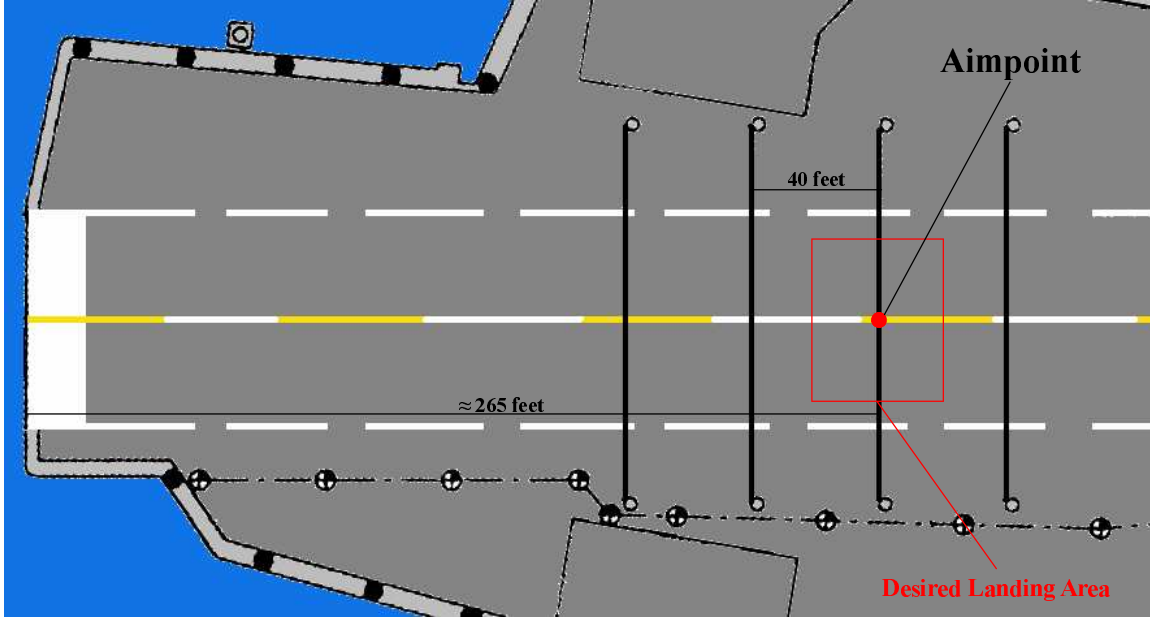


Figure 2.1: Nimitz Class Carrier Landing Area.

at approximately ± 50 feet from the runway centerline, the nominal portion of the landing area was assumed to be ± 22.65 feet from the runway centerline in order to keep the wingtips from crossing the foul line during landing. The red box indicates the desired landing area.

Now that the flight deck layout has been presented, the list of considerations for carrier landings can be addressed. The first consideration is glidepath error. The glidepath is the line extending outward from the desired touchdown point, or aimpoint, and is the path relative to the carrier which the aircraft should fly in order to fly at the correct airspeed and land at the right point on the flight deck. With piloted aircraft, if the aircraft strays too far from the assigned glidepath, either horizontally or vertically, the LSOs will give it a wave off, or abort signal. The same will be true for UCAVs; they must stay within a certain glideslope error or they will be waved off. For the purposes of this research, all approaches were flown to touchdown and it was assumed that most deviations from glidepath occurred as the aircraft was in close to the carrier and were thus reflected in the miss distances.

If the aircraft flies a good approach, the next consideration is to make sure that it is not too low when it comes over the ramp, the aft end of the ship. If it is too low and strikes the ramp, it is called a rampstrike. If any portion of the aircraft hits the ramp, including the hook, it is classified as a rampstrike. However, the most severe case is when the aircraft actually impacts the ramp, resulting in loss of the aircraft and possibly loss of life either by someone in the aircraft or on the deck. Rampstrikes are extremely undesirable, so it is preferred that an aircraft miss long, rather than short. In this research, any approach which resulted in missing short of the aimpoint by greater than 265 feet was considered a ramp strike.

Missing long is classified as a bolter. Bolters, the next consideration, are any instance in which the aircraft touches down on the deck but the hook misses all of the wires, either by landing past them or bouncing over them, commonly called hook skips. This research does not consider hook skips, and only considers bolters which are due to landing past the wires.

The next consideration is to minimize the error between where the aircraft touches down and the desired touchdown point. First consider missing long or short. As previously noted, it is desired that aircraft catch the 3 wire. In this research, any touchdown within 20 feet long or short of the aimpoint qualifies as a 3 wire catch. Statistically, it is desirable that the mean miss distance be as close to zero as possible and that the standard deviation of the miss distances be as small as possible. The further the mean from the aimpoint and the greater the standard deviation, the greater the chance of a bolter or rampstrike. Similarly for missing left or right, small mean miss distances and standard deviations are desired. If they are not small, the risk of a landing aircraft's wingtips striking other aircraft on deck is greatly increased [28, pages 2–3].

2.3 Dynamic Inversion Control Structure

Dynamic Inversion is a nonlinear control structure which uses a mathematical model of the plant to cancel out the original plant dynamics and replace them with

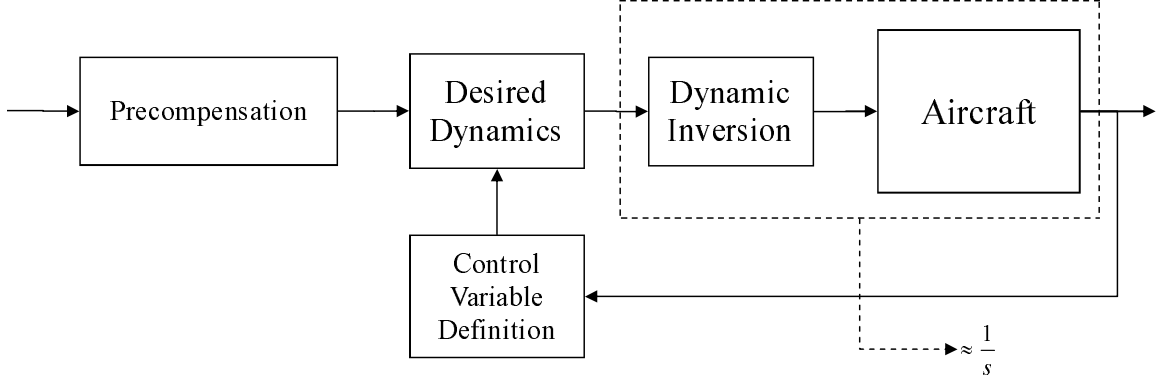


Figure 2.2: Generic Dynamic Inversion Control Structure.

dynamics which are more desirable for the given application. There are four main sections to the DI control architecture, as shown in Figure 2.2. They are precompensation, control variable definition, desired dynamics, and dynamic inversion [15, pages 125–127].

2.3.1 Precompensation. The precompensation portion of the DI control structure accounts for the fine tuning of issues related to pilot interfacing. Handling qualities related issues such as stick and pedal gains and nonlinear stick shaping are tasks which are usually addressed in this area. This research did not appreciably use precompensation, as it dealt with a fully automated aircraft with all necessary inputs coming from sensors and not a human pilot [15, page 127].

2.3.2 Control Variable Definition. The control variable definition section is where the measurements of the states or other sensor outputs are combined into what are called the Control Variables (CVs). The CVs are the various states, sensor outputs, or combinations thereof which the DI structure attempts to control. For simple aircraft, these may be as simple as roll rate, pitch rate, and sideslip angle. However, on more complicated, high performance aircraft, the CVs are usually chosen as a blend of parameters in order to improve performance throughout a wide flight envelope or to account for nonminimum-phase characteristics of some of the dynamic modes of the aircraft. Eliminating nonminimum-phase characteristics from the CVs is

crucial because dynamic inversion essentially aims to produce pole-zero cancellations. Therefore, if the inversion attempts to cancel a right-half-plane zero with a right-half-plane pole and does not cancel it exactly, the system will be unstable [15, page 125].

2.3.3 Desired Dynamics. The desired dynamics portion of the controller is where the new dynamics are specified. Ideally, the result of the dynamic inversion yields perfect integrator dynamics for all of the CVs. If this is the case, then the closed-loop system response will be a first-order response and merely specifying a simple proportional gain for the desired dynamics becomes the time constant. However, since a perfect inversion will rarely be achieved in real world situations, it is suggested that some other desired dynamics be specified. Any simple linear controller can be implemented, such as a Proportional plus Integral plus Derivative (PID) Controller. Also the addition of a Linear Quadratic Regulator (LQR), another H_2 controller, or an H_∞ controller can serve as an excellent way to ensure robust performance. Whatever linear controller is used for the desired dynamics, it is suggested that integral control be used in the controller in order to ensure that the steady-state errors approach zero even when the inversion is not perfect [15, pages 132–133].

2.3.4 Dynamic Inversion. This section of the controller is where the actual nonlinear inversion of the plant dynamics takes place. Consider the following nonlinear state equation for the plant:

$$\dot{x} = f(x) + g(x)u \quad (2.1)$$

This equation is in its generalized vector form, where x is the state vector and u is the input vector. This formulation of DI assumes that the nonlinear equations of motion can be written in the form above as a function of the states f plus a function of the states g times the input vector u . This is commonly referred to as control affine form. This means that the equations of motion must be linear with respect to u . Since the control variables are defined as functions of the states, the control variable vector can

be defined as:

$$V = V(x) \quad (2.2)$$

To get the set of differential equations governing the control variables, differentiate V with respect to time:

$$\dot{V} = \frac{\partial V}{\partial x} \dot{x} = \frac{\partial V}{\partial x} f(x) + \frac{\partial V}{\partial x} g(x)u \quad (2.3)$$

Keeping in mind that the desired dynamics portion of the controller supplies what is essentially \dot{V}_{des} to the inversion, Equation 2.3 can be rewritten as:

$$\dot{V}_{des} = \frac{\partial V}{\partial x} f(x) + \frac{\partial V}{\partial x} g(x)u \quad (2.4)$$

The point of the inversion is to solve for the control input into the plant which will yield the desired dynamics. Thus, solving for the control input u yields:

$$u = \left[\frac{\partial V}{\partial x} g(x) \right]^{-1} \left[\dot{V}_{des} - \frac{\partial V}{\partial x} f(x) \right] \quad (2.5)$$

Note that Equation 2.5 is heavily dependent on the choice of control variables and the plant model. If the control variables that are chosen are not easily controllable, then the inversion may not work very well. More importantly, if the plant model is not of high enough fidelity, a poor inversion may be the result. The inversion is only as good as the plant model. Care should be taken when formulating the model to be used in the inversion to make sure that it is of high enough fidelity for the purpose of the controller [15, page 134].

2.4 *Environmental Assumptions*

As previously mentioned, there are quite a few environmental considerations that affect carrier landings. Although this research did not attempt to estimate the effects of precipitation and fog on landing aircraft, different wind and sea state related factors were taken into account to provide some realism to the simulation. These assumptions will now be addressed.

2.4.1 Horizontal Wind Model. The baseline wind model used was the Horizontal Wind Model that comes as part of the Simulink® Aerospace Blockset. For each simulation case, wind speed and direction were defined. The input to the block was the aircraft Direction Cosine Matrix (DCM), and the output from the block was a vector containing the wind components in the aircraft body axes.

2.4.2 Dryden Wind Turbulence Model. The Simulink® Aerospace Blockset also contained several wind turbulence models. The continuous Dryden wind turbulence model used in this research was chosen because it was more easily implemented than the Von Kármán turbulence model. The general forms for the coloring filters used to create turbulence velocities from white noise inputs using the Dryden and Von Kármán turbulence models, respectively, are:

$$\begin{aligned} G_D(s) &= \frac{a_f + b_f s^2}{(1 + c_f s^2)^2} \\ G_{VK}(s) &= \frac{a_f + b_f s^2}{(1 + c_f s^2)^{\frac{11}{6}}} \end{aligned} \tag{2.6}$$

where a_f , b_f , and c_f are filter constants dependent on the scale lengths for the desired turbulence model [11, page 657]. The $\frac{11}{6}$ power in the denominator of the Von Kármán filter equates to fractional derivatives in the time domain, making it difficult to implement. The squared denominator in the Dryden filter was more easily implemented and closely approximated the Von Kármán filter, so it was used instead.

The inputs to the Simulink® block were altitude, airspeed, and the aircraft DCM, and the output was a vector containing the wind turbulence components in the body axes. These were summed with the wind velocities in the body axes yielded from the horizontal wind model to give the overall wind components in the body axes. These overall wind components were subtracted from the aircraft velocities in the body axes to yield the relative wind velocities in the body axes, used for the calculation of the aircraft aerodynamics.

2.4.3 Burble Model. Like most warships, aircraft carriers are streamlined below the waterline to reduce water drag. Water drag has more of an impact on ships than air drag because water is much more dense. While the lack of streamlining above the waterline does not produce enough drag to affect a carrier’s forward motion, it does provide enough of a disruption in the airflow around the carrier to change the wind conditions for approaching aircraft as they near the ship. In other words, carriers have their own flowfields which are a function of ship speed, wind speed, and wind over deck angle. This flowfield is called the Burble. As the aircraft approaches the carrier, it eventually enters the burble. It has a large effect on landing performance, so it needed to be included.

The burble model used in this research was formulated by NAVAIR [23, pages 102–103]. It is made up of a steady component, an unsteady component, and a periodic component. The periodic component was ignored, as it is due to the ship’s slow periodic motion. Because the times associated with the periodic change of the flowfield, it does not change much during the approach of a single aircraft. The steady component is included, as it is essentially a shift in all three components of the winds as the aircraft approaches the carrier. However, the unsteady component, an adapted Von Kármán model, is ignored in this research due to the Dryden turbulence model being run throughout the simulation. While the unsteady component of this burble model was shaped uniquely for this purpose, the presence of the Dryden turbulence model already serves the purpose of adding a pseudorandom exogenous input to the system. The exact specifications for the burble model can be found in Reference [23].

2.4.4 Sea State Model. Sea State is a term referring to the relative severity of ocean surface turbulence and waves. Sea state 0 refers to calm seas, sea state 6 is extremely heavy seas, and sea states 1–5 are everywhere in between. Different sea states were taken into account during the course of simulation. In addition to the burble model described in Reference [23], a sea state model is also included [23, pages 94–99]. The ship rotational degrees of freedom are termed roll, pitch, and yaw. In

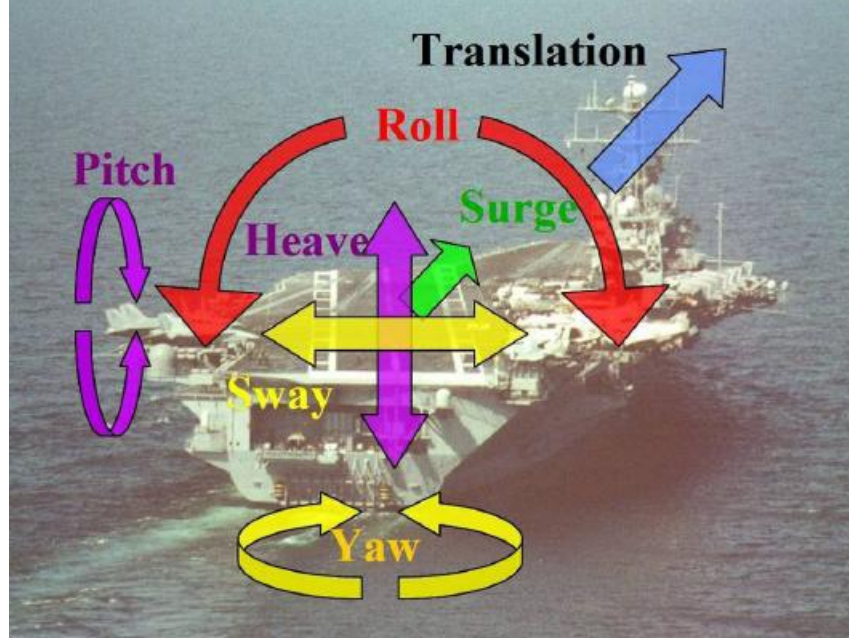


Figure 2.3: Ship Degrees of Freedom [23, page 94].

the translational degrees of freedom, up and down motion is called heave, forward to aft motion is called surge, and port to starboard motion is called sway. These are all illustrated in Figure 2.3.

The NAVAIR sea state model can either be implemented as a sum-of-sines or as a single sine wave for each degree of freedom. For the purposes of this research, only the single sine wave implementation was considered. The specifics of the ship perturbation model can be found in Reference [23].

2.5 Equations of Motion

A full set of nonlinear differential equations was used in the development of both the equations of motion which ran the simulation and the model of the aircraft used by the DI controller. However, a simple set of linear differential equations was used in the simulation of the aircraft carrier's motion. The states yielded by the equations of motion (EOM) for the ship were summed with the perturbations from the sea state model to give the actual ship states. The coordinate systems and equations of motion used in the simulation will now be developed.

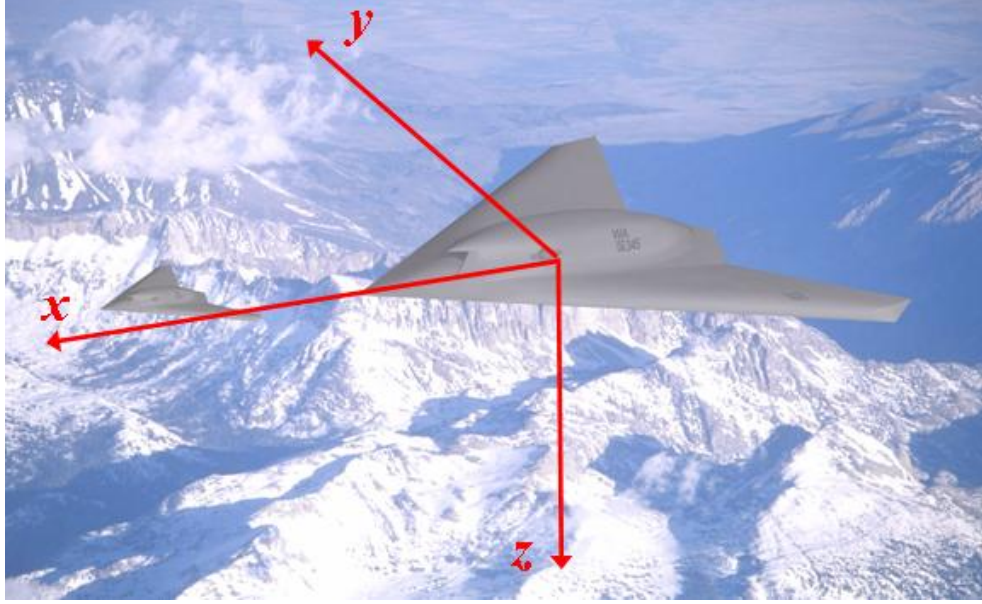


Figure 2.4: Aircraft Body-Fixed Coordinate Axes.

2.5.1 Coordinate Systems. There were three main coordinate systems considered during the development of the simulation. The first was the aircraft body axes system. The traditional formulation of the body axes with x out the nose, y out the right wing, and z out the bottom of the aircraft was used, shown in Figure 2.4. The body velocities, euler angles, and angle rates all deal with the body axes. However, the body axes could not be used to consider the position of the aircraft relative to the Earth.

Therefore, the next reference frame considered was the inertial reference frame of the non-rotating earth, centered about an arbitrary point on the surface of the ocean. This reference frame was used in the computation of the aircraft position during the simulation. The standard North-East-Down (NED) coordinate system in Figure 2.5 was used, with x pointing North, y pointing East, and z pointing into the Earth.

While the NED coordinate system could have worked for actual data analysis, it was inconvenient to have negative numbers for positive altitude. Accordingly, a third East-North-Up (ENU) coordinate system was used, with x pointing East, y

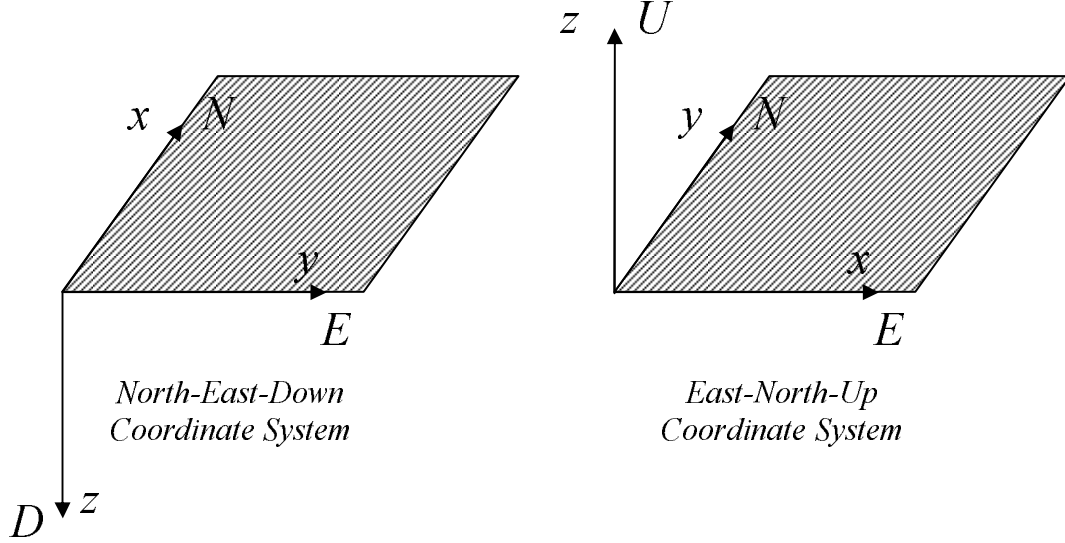


Figure 2.5: Inertial Reference Frames Used in Simulation.

pointing North, and z pointing out of the Earth. The ENU system is also shown in Figure 2.5. The ENU is a valid right-handed coordinate system, so all the usual vector mathematics rules still apply. The ship's EOM and the DI controller both utilized the ENU coordinate system.

2.5.2 Aircraft Equations of Motion. The EOM set used for the aircraft simulation was the twelfth order, nonlinear, fully-coupled, set of differential equations traditionally used to describe aircraft motion in a non-rotating Earth inertial reference frame [24]. Because of the relatively symmetric blended wing-body design, both the xz - and xy -planes were assumed to be planes of symmetry, thus simplifying the inertia tensor. This particular formulation is found in Reference [24, page 637]:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{g}{W} \begin{bmatrix} F_{xb} \\ F_{yb} \\ F_{zb} \end{bmatrix} + g \begin{bmatrix} -S_\theta \\ S_\phi C_\theta \\ C_\phi C_\theta \end{bmatrix} + \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} \quad (2.7)$$

$$\begin{bmatrix} \dot{x}_n \\ \dot{y}_n \\ \dot{z}_n \end{bmatrix} = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.8)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} I_{xxb} & 0 & 0 \\ 0 & I_{yyb} & 0 \\ 0 & 0 & I_{zzb} \end{bmatrix}^{-1} \begin{bmatrix} M_{xb} + (I_{yyb} - I_{zzb})qr \\ M_{yb} + (I_{zzb} - I_{xxb})pr \\ M_{zb} + (I_{xxb} - I_{yyb})pq \end{bmatrix} \quad (2.9)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \frac{S_\phi S_\theta}{C_\theta} & \frac{C_\phi S_\theta}{C_\theta} \\ 0 & C_\phi & -S_\phi \\ 0 & \frac{S_\phi}{C_\theta} & \frac{C_\phi}{C_\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.10)$$

The b subscript indicates the variable is resolved in the body axes, while the n subscript indicates that the variable is resolved in the NED inertial axes. F_{xb} , F_{yb} , F_{zb} , M_{xb} , M_{yb} , and M_{zb} are the aerodynamic forces and moments resolved in the body axes. They are defined as:

$$F_{xb} = T - \frac{1}{2}\rho(u^2 + v^2 + w^2)SC_A \quad (2.11)$$

$$F_{yb} = C_{Y,\beta} \tan^{-1}\left(\frac{v}{u}\right) \quad (2.12)$$

$$F_{zb} = -\frac{1}{2}\rho(u^2 + v^2 + w^2)SC_N \quad (2.13)$$

$$M_{xb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)SbC_\ell \quad (2.14)$$

$$M_{yb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)S\bar{c}C_m \quad (2.15)$$

$$M_{zb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)SbC_n \quad (2.16)$$

A complete list of all the terms appearing in these equations and their definitions can be found in the List of Symbols on page xii of this document. The aircraft model from where these force and moment coefficients come will be discussed in Chapter 3.

2.5.3 Aircraft Carrier Equations of Motion. As previously mentioned, the EOM set used to simulate the carrier's motion was very simple. A constant speed V_B was considered for the ship. The baseline position of the carrier, without the perturbations, is given by:

$$\begin{bmatrix} \dot{x}_{Be} \\ \dot{y}_{Be} \end{bmatrix} = V_B \begin{bmatrix} \sin \psi_B \\ \cos \psi_B \end{bmatrix} \quad (2.17)$$

where x_{Be} and y_{Be} describe the ship's position in the ENU coordinate frame and ψ_B is the ship's heading angle. The e subscript is used throughout this document to denote the ENU coordinate frame. The ship was assumed to have no motion in the vertical plane except for the perturbations due to the sea state model.

2.6 Overview of Dynamic Inversion Controller

There were two main controllers designed for the purposes of this research. The first was an inner-loop DI flight controller similar in function to those found in modern high performance aircraft. Its inputs were desired values of pitch rate, roll rate, and thrust, or $\dot{\theta}$, $\dot{\phi}$, and T respectively. Its outputs were actuator and throttle commands to yield these desired values. The second was the outer-loop, which provided steering commands to the inner-loop. The inputs to the outer-loop were the aircraft and ship states and the outputs were desired values of heading angle, pitch angle, and thrust, or ψ , θ , and T respectively. Additionally, a set of transitional dynamics were formulated to transform the desired values of ψ , θ , and T resulting from the outer-loop to the desired values of $\dot{\theta}$, $\dot{\phi}$, and T required by the inner-loop. Figure 2.6 illustrates this overall concept.

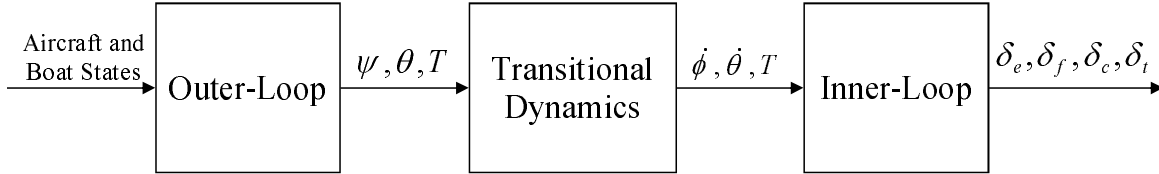


Figure 2.6: Overview of Control Structure.

2.7 Inner-Loop Dynamic Inversion Flight Controller

The inner-loop controller controls the thrust and angular accelerations of the aircraft. This section addresses the specifics of the inner-loop controller formulation.

2.7.1 Relevant Equations of Motion. While the full EOM set described in Equations 2.7–2.10 was necessary to run the actual simulation, not all of those equations were needed for the inner-loop controller. The inner-loop controller was responsible only for controlling thrust and angular accelerations of the aircraft. Therefore, the navigational equations, Equation 2.8, were not needed. Also, the heading angle was not needed, so Equation 2.10 was simplified slightly by removing $\dot{\psi}$ and the bottom row of the 3×3 matrix. However, the inner-loop controller was required to control thrust, so a simple first order equation was added:

$$\dot{T} = \frac{1}{\tau}(\delta_t T_{max} - T) \quad (2.18)$$

where τ is the engine time constant and T_{max} represents the maximum thrust possible at the given flight condition estimated by the engine model of the controller.

One other item of complexity that had to be overcome was that the pitch dynamics of the aircraft were open-loop unstable. Just as an imperfect inversion would drive a nonminimum-phase system unstable, the same is true for an already unstable system. Therefore, to stabilize the pitch dynamics, a simple feedback loop with proportional feedback gain k was added:

$$\delta_f = k(\delta_f^* - q) \quad (2.19)$$

where δ_f^* is the pitch flap command passed from the controller and δ_f is the actual command passed to the actuator. Accordingly, δ_f^* is now the pitch flap control command of interest from the controller.

Before compiling the final form of the EOM set relevant to the inner-loop controller, it should also be noted that the formulation of the aerodynamic forces and moments in Equations 2.11–2.16 were not adequate for the inner-loop control. In order to complete the inversion, the forces and moments need to be in terms of the control inputs. The aircraft configuration will be discussed in more depth in Chapter 3, but for right now, it is sufficient to say that the aircraft has elevons (δ_e) to control roll, pitch flaps (δ_f) to control pitch, clamshells (δ_c) to control yaw, and throttle (δ_t) to control thrust.

In the earlier formulation of the forces and moments, the effects of the control inputs were already summed into the corresponding overall force and moment coefficients. However, a simple Taylor series expansion of the coefficients about the control inputs at every given instant yields the necessary dependence of the forces and moments on the control inputs. The expansion applied to all the force and moment equations with only the first order terms kept yields:

$$F_{xb} = T - \frac{1}{2}\rho(u^2 + v^2 + w^2)S(C_A - \bar{C}_A + C_{A,\delta_e} \delta_e + C_{A,\delta_f} k(\delta_f^* - q) + C_{A,\delta_c} \delta_c) \quad (2.20)$$

$$F_{yb} = C_{Y,\beta} \tan^{-1}\left(\frac{v}{u}\right) \quad (2.21)$$

$$F_{zb} = -\frac{1}{2}\rho(u^2 + v^2 + w^2)S(C_N - \bar{C}_N + C_{N,\delta_e} \delta_e + C_{N,\delta_f} k(\delta_f^* - q) + C_{N,\delta_c} \delta_c) \quad (2.22)$$

$$M_{xb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)Sb(C_\ell - \bar{C}_\ell + C_{\ell,\delta_e} \delta_e + C_{\ell,\delta_c} \delta_c) \quad (2.23)$$

$$M_{yb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)S\bar{c}(C_m - \bar{C}_m + C_{m,\delta_f} k(\delta_f^* - q)) \quad (2.24)$$

$$M_{zb} = \frac{1}{2}\rho(u^2 + v^2 + w^2)Sb(C_n - \bar{C}_n + C_{n,\delta_e} \delta_e + C_{n,\delta_c} \delta_c) \quad (2.25)$$

where the gradients of the form C_{m,δ_f} represent the instantaneous control surface gradients calculated from the lookup tables. Also, the terms of the form \bar{C}_m represent the contribution to the coefficient of the set of control inputs about which the Taylor series is centered.

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{T} \end{bmatrix} = \begin{bmatrix} \frac{g}{W} [T - QS(C_A - \bar{C}_A - C_{A,\delta_f} kq)] - g \sin \theta + rv - qw \\ \frac{g}{W} QSC_{Y,\beta} \tan^{-1}\left(\frac{v}{u}\right) + g \sin \phi \cos \theta + pw - ru \\ -\frac{g}{W} QS(C_N - \bar{C}_N - C_{N,\delta_f} kq) + g \cos \phi \cos \theta + qu - pv \\ \frac{1}{I_{xxb}} [Q Sb(C_\ell - \bar{C}_\ell) + (I_{yyb} - I_{zzb})qr] \\ QS\bar{c}(C_m - \bar{C}_m - C_{m,\delta_f} kq) + (I_{zzb} - I_{xxb})pr \\ \frac{1}{I_{zzb}} [Q Sb(C_n - \bar{C}_n) + (I_{xxb} - I_{yyb})pq] \\ p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ q \cos \phi - r \sin \phi \\ -\frac{T}{\tau} \end{bmatrix} + \begin{bmatrix} -\frac{g}{W} QSC_{A,\delta_e} & -\frac{g}{W} QSC_{A,\delta_f} k & -\frac{g}{W} QSC_{A,\delta_c} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{g}{W} QSC_{N,\delta_e} & -\frac{g}{W} QSC_{N,\delta_f} k & -\frac{g}{W} QSC_{N,\delta_c} & 0 \\ \frac{1}{I_{xxb}} Q SbC_{\ell,\delta_e} & 0 & \frac{1}{I_{xxb}} Q SbC_{\ell,\delta_c} & 0 \\ 0 & \frac{1}{I_{yyb}} QS\bar{c}C_{m,\delta_f} k & 0 & 0 \\ \frac{1}{I_{zzb}} Q SbC_{n,\delta_e} & 0 & \frac{1}{I_{zzb}} Q SbC_{n,\delta_c} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{T_{max}}{\tau} \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_f^* \\ \delta_c \\ \delta_t \end{bmatrix} \quad (2.26)$$

With all of these considerations taken into account, the final form of the relevant EOM set can be formulated. Substituting Equations 2.20–2.25 into Equations 2.7–2.10, removing the unnecessary states as previously described, simplifying, and writing in the form of Equation 2.1 yields Equation 2.26. The Q term is the dynamic pressure:

$$Q = \frac{1}{2} \rho (u^2 + v^2 + w^2) \quad (2.27)$$

Equation 2.26 is the final form of the plant model needed for the inversion of the dynamics for the inner loop controller.

2.7.2 Control Variable Definition. The control variables required for this application were fairly simple. The results of the outer-loop were desired pitch and heading angles and desired thrust, θ , ψ , and T . Through a simple set of transitional dynamics, these were easily transformed into $\dot{\theta}$ and $\dot{\phi}$, and T remained. Both the outer and transitional loops will be discussed later. The most natural choice of control variables given these circumstances were $\dot{\theta}$, $\dot{\phi}$, and T .

For the moment, only consider $\dot{\theta}$ and $\dot{\phi}$. Because these are not states explicitly found in the equations of motion, they need to be expressed as functions of the states. However, noticing that Equation 2.10 gives expressions for $\dot{\theta}$ and $\dot{\phi}$ in terms of the states, the equations for the first two control variables can be written as:

$$LCV = \dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \quad (2.28)$$

$$MCV = \dot{\theta} = q \cos \phi - r \sin \phi \quad (2.29)$$

where LCV is called the roll-axis control variable, and MCV is called the pitch-axis control variable.

Additionally, a control variable was needed to provide directional stability and coordinate turns. Turning again to Reference [15, pages 84,111], a good control variable for these purposes is:

$$NCV = -p \sin \alpha + r \cos \alpha - \frac{\frac{g}{w}QS}{V_{co}}\beta - \frac{g}{\sqrt{u^2 + v^2 + w^2}} \cos \theta \sin \phi \quad (2.30)$$

where NCV is called the yaw-axis control variable and V_{co} is the crossover velocity tuning parameter. Additionally, α and β are the angle of attack and sideslip angles, respectively, defined as:

$$\alpha = \tan^{-1} \left(\frac{w}{u} \right) \quad (2.31)$$

$$\beta = \tan^{-1} \left(\frac{v}{u} \right) \quad (2.32)$$

Finally, thrust is the last control variable, expressed as:

$$XCV = T \quad (2.33)$$

where XCV is the x -axis control variable. Equations 2.28, 2.29, 2.30, and 2.33 are the final forms of the inner-loop control variables. Stacking all of these into a vector yields the control variable vector:

$$V = \begin{bmatrix} LCV \\ MCV \\ NCV \\ XCV \end{bmatrix} \quad (2.34)$$

2.7.3 Desired Dynamics. As previously mentioned, the desired dynamics can take the form of any type of controller. However, linear controllers are most appropriate, as a decent inversion removes most of the nonlinear effects. For the inner-loop, simple PID loops were used for all four control variables. These PID loops are different from most, however, because they utilize a limited integral. The limited integral eliminates the issue of integrator windup while still eliminating steady-state error. The limited integral PID loops provide additional robustness in the event of a less than perfect inversion. These PID loops will be addressed in more detail in Chapter 3.

2.7.4 Inversion. The inversion for the inner-loop is identical to the process explained by Equations 2.1–2.5. Equation 2.26 is the plant model in the form of Equation 2.1, Equation 2.34 is the control variable vector in the form of Equation 2.2, and \dot{V}_{des} is the desired rate of change of the control variable vector from the desired dynamics section of the controller. The only term still needed is the control variable gradient, $\frac{\partial V}{\partial x}$. The control input necessary for inversion is calculated with Equation 2.5.

2.8 Outer-Loop Dynamic Inversion Homing Controller

The outer-loop controller provides steering commands for the inner-loop controller. This section details its design.

2.8.1 Relevant Equations of Motion. As with the inner-loop, not all of the equations in the full EOM set used to run the simulation were needed. The objectives of the outer-loop were to eliminate glidepath error and maintain a constant angle of attack throughout the entire approach. The only states needed to describe this design problem were the aircraft velocities, the aircraft position coordinates, and the ship position coordinates. For the aircraft velocities, only equations for \dot{u} and \dot{w} were needed, as it is assumed that the inner-loop keeps v as close to zero as possible. Also, only the \dot{x} and \dot{y} position equations of the ship were needed, as the z position was assumed to be constant for design purposes.

For the purposes of designing the outer-loop controller, the aircraft was assumed to be a point-mass. Because glidepath error and angle of attack were the variables to be controlled, heading angle, pitch angle, and thrust were logical choices to use as control inputs to the point-mass aircraft. Therefore, the problem that arose in the inner-loop when the force and moment coefficients were not in terms of the control surfaces was not a problem with the outer-loop.

Considering all of this, the set of equations necessary for the inversion of the outer-loop dynamics is:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{x}_e \\ \dot{y}_e \\ \dot{z}_e \\ \dot{x}_{Be} \\ \dot{y}_{Be} \end{bmatrix} = \begin{bmatrix} \frac{g}{W}(T - QSC_A) - g \sin \theta + rv - qw \\ -\frac{g}{W}QSC_N + g \cos \phi \cos \theta + qu - pv \\ uC_\theta S_\psi + v(S_\phi S_\theta S_\psi + C_\phi C_\psi) + w(C_\phi S_\theta S_\psi - S_\phi C_\psi) + V_{wxe} \\ uC_\theta C_\psi + v(S_\phi S_\theta C_\psi - C_\phi S_\psi) + w(C_\phi S_\theta C_\psi + S_\phi S_\psi) + V_{wye} \\ uS_\theta - vS_\phi C_\theta - wC_\phi C_\theta + V_{wze} \\ V_B \sin(\psi_{BF} + 9^\circ) \\ V_B \cos(\psi_{BF} + 9^\circ) \end{bmatrix} \quad (2.35)$$

where V_{wxe} and similar terms are the wind components in the ENU coordinate frame, and ψ_{BF} is the heading angle of the flight deck. Adding nine degrees to ψ_{BF} , as in Equation 2.35 yields the ship heading angle, ψ_B . This is due to the nine degree angled offset of the flight deck to the port side of the ship. The input vector to Equation 2.35 consists of ψ , θ , and T . Notice that Equation 2.35 cannot be written in the control affine form of Equation 2.1. This is because the control inputs θ , ψ , and T cannot be separated out because they are embedded in trigonometric functions. This means that a linearized version of the process described in Section 2.3.4 was needed. It will be discussed in Section 2.8.5.

2.8.2 Control Variable Definition. As previously noted, the main interests in the outer-loop homing control problem were glidepath error and angle of attack. The glidepath error can be separated into two components: horizontal and vertical. The horizontal component will be referred to as horizontal glidepath error, while the vertical component will be referred to as glideslope error.

The desired horizontal glidepath angle is defined as the angle from the aircraft's heading for landing (ψ_{BF}) to the actual heading to the ship. The variable θ_{hgd} is used to describe this angle. If the ship were either not moving or had no angled offset of the flight deck, this angle would be zero. However, because this is not the case, θ_{hgd} is a function of the offset angle, the ship's speed, and the aircraft's speed over ground. Figure 2.7(a) illustrates this relationship. From this geometry:

$$\theta_{hgd} = \tan^{-1} \left(\frac{V_B \sin 9^\circ}{\sqrt{V_{xe}^2 + V_{ye}^2} - V_B \cos 9^\circ} \right) \quad (2.36)$$

where V_{xe} and V_{ye} are the aircraft speeds over ground in the x and y directions, respectively. However, the quantity $\sqrt{V_{xe}^2 + V_{ye}^2} - V_B \cos 9^\circ$ can be simplified, as it is the relative velocity of the aircraft with respect to the ship. This is a defined parameter for the carrier landing problem, and will be referred to as the hook engagement

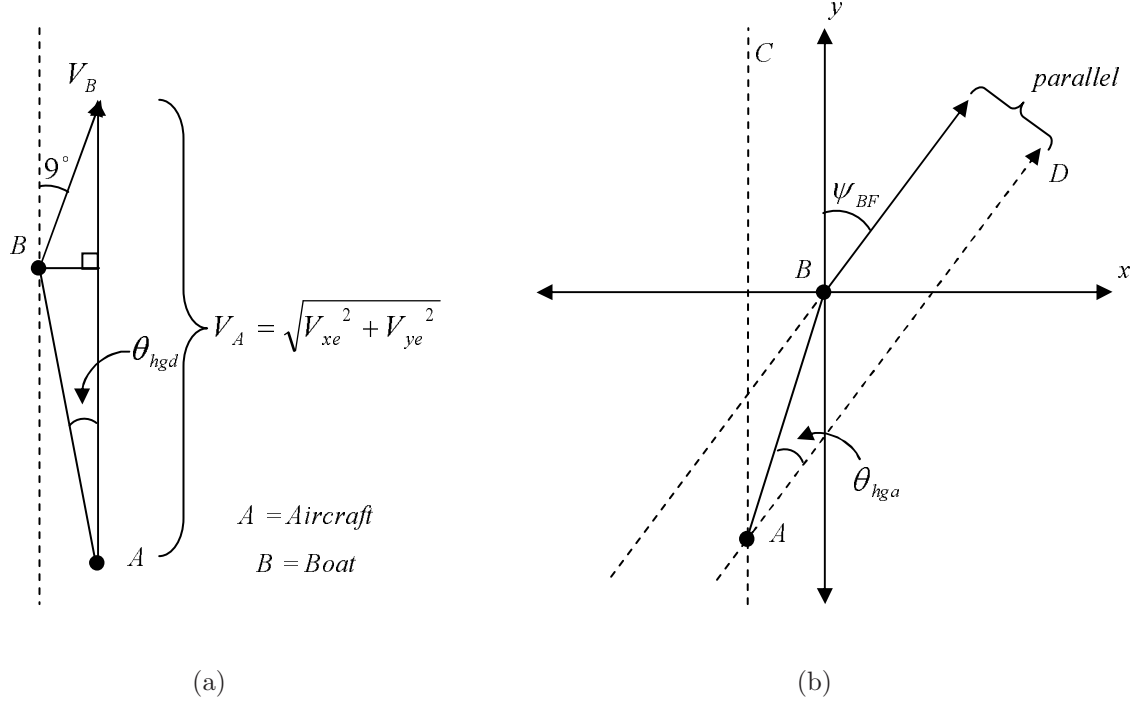


Figure 2.7: (a) Definition of θ_{hgd} . (b) Definition of θ_{hga} .

velocity, V_{HE} . Using this substitution, Equation 2.36 simplifies to:

$$\theta_{hgd} = \tan^{-1} \left(\frac{V_B \sin 9^\circ}{V_{HE}} \right) \quad (2.37)$$

While this is the form of the equation for the desired horizontal glidepath angle, the error between the desired and actual is what is needed for the controller. Figure 2.7(b) illustrates the geometry for determining this error. Note that this research only considered the case where ψ_{BF} was in Quadrant I, but the principle can easily be extended to the other three quadrants. From the figure, the actual horizontal glide angle is angle $\angle BAD$. This will be referred to as θ_{hga} . This is also equal to $\psi_{BF} - \angle CAB$. Through simple trigonometry, the expression for the horizontal glidepath error becomes:

$$HCV = \underbrace{\psi_{BF} - \tan^{-1} \left(\frac{x_{Be} - x_e}{y_{Be} - y_e} \right)}_{\theta_{hga}} - \underbrace{\tan^{-1} \left(\frac{V_B \sin 9^\circ}{V_{HE}} \right)}_{\theta_{hgd}} \quad (2.38)$$

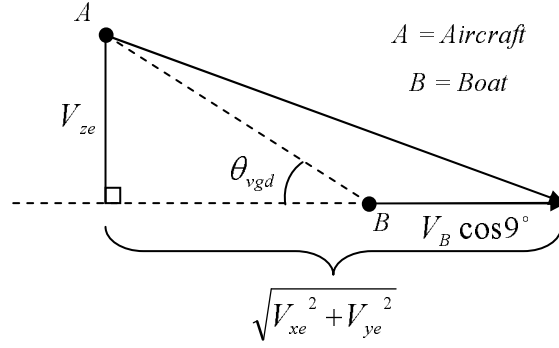


Figure 2.8: Definition of θ_{vgd} .

where HCV is defined as the heading control variable.

The next variable of concern is the glideslope error. The desired glideslope angle is generally defined as the angle between the ground and the desired flight path. However, for the purpose of aircraft carrier landing, it is often more convenient to consider the glideslope angle as the angle above horizontal of the flight path relative to the touchdown point on the flight deck. The desired value of this angle will be referred to as θ_{vgd} , and the actual value of this angle will be referred to as θ_{vga} . This angle is slightly larger than the actual flight path angle due to the ship's forward velocity. As with θ_{hgd} , θ_{vgd} is also a function of the flight deck offset angle, ship speed, and aircraft speed over ground. Figure 2.8 illustrates this relationship. From this geometry, θ_{vgd} can be written as:

$$\theta_{vgd} = \tan^{-1} \left(\frac{V_{ze}}{\sqrt{V_{xe}^2 + V_{ye}^2} - V_B \cos 9^\circ} \right) \quad (2.39)$$

where V_{ze} is the desired vertical velocity of the aircraft, a defined parameter for the carrier landing problem. As before, the quantity $\sqrt{V_{xe}^2 + V_{ye}^2} - V_B \cos 9^\circ$ is equal to V_{HE} . With this substitution, Equation 2.39 becomes:

$$\theta_{vgd} = \tan^{-1} \left(\frac{V_{ze}}{V_{HE}} \right) \quad (2.40)$$

As before, the actual glideslope angle is needed in addition to the desired glideslope angle in order to calculate the glideslope error. Again using Figure 2.8 and substituting θ_{vga} for θ_{vgd} , simple trigonometry shows that θ_{vga} is a function of both the aircraft and ship positions, as shown in Equation 2.41. The final form of the glideslope error is:

$$PCV = \underbrace{\tan^{-1} \left(\frac{-(z_{Be} - z_e)}{\sqrt{(x_{Be} - x_e)^2 + (y_{Be} - y_e)^2}} \right)}_{\theta_{vga}} - \underbrace{\tan^{-1} \left(\frac{V_{ze}}{V_{HE}} \right)}_{\theta_{vgd}} \quad (2.41)$$

where PCV is defined as the pitch control variable.

As previously mentioned, the last of the outer-loop control variables is basically angle of attack. What is really needed in order to drive the glideslope error previously described to zero is flight path control, as opposed to pitch control. However, since the flight path angle γ is merely $\theta - \alpha$ for the small bank angles when glidepath error is small, controlling both θ and α is equivalent to controlling γ . To accomplish this, the outer-loop will use thrust to hold α constant at a specified angle while using θ to control flight path to drive PCV to zero. The value of α used for this research was 8° . Therefore, the expression for the final control variable is:

$$TCV = \alpha - 8^\circ = \tan^{-1} \left(\frac{w}{u} \right) - 8^\circ \quad (2.42)$$

where TCV is defined as the thrust control variable. Driving TCV to zero ensures that the angle of attack is at the desired value.

2.8.3 “Time” Domain to “Distance” Domain. Usually in control systems design, parameters such as time constants and settling times are utilized in order to characterize the speed of response of a system. These parameters most often drive the design of a controller. However, when dealing with a problem such as carrier landings with control variables such as the ones previously defined, time constants and settling times are inadequate in describing the speed of response of a system.

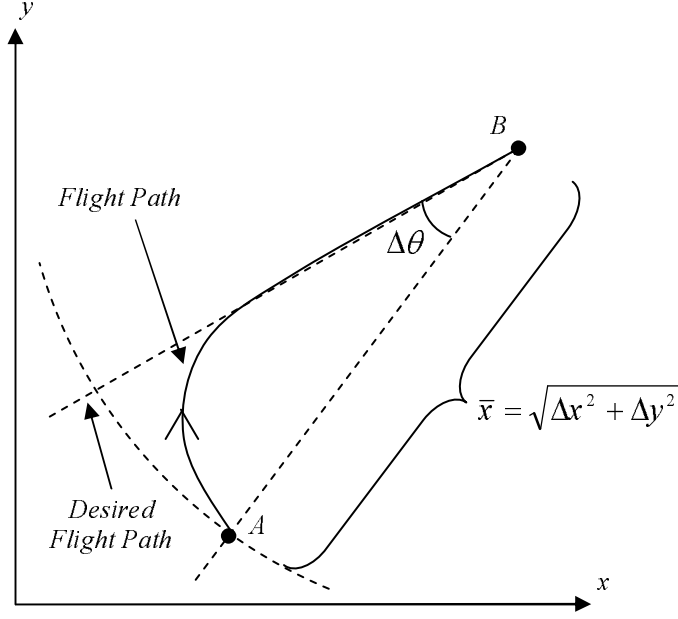


Figure 2.9: Illustration of “Distance Domain.”

This is due to the fact that the angle errors are heavily dependent on the distance from the carrier, and the aircraft speed over ground is relatively constant. It would be more convenient for design purposes to be able to describe the speed of response of the aircraft to glidepath errors in terms of both angle error rates and distance from the ship.

For this reason, a “distance constant” and “distance domain” were devised. Analogous to a time constant, a distance constant describes how quickly the angle errors change relative to how quickly the aircraft is approaching the carrier. This concept is illustrated in Figure 2.9. Suppose an aircraft begins at point A and lands at point B with the solid line as the flight path in between. Assuming a well-behaved first order response, the speed of response can be characterized with a distance constant τ_d in the distance domain as:

$$\Delta\theta(\bar{x}) = \Delta\theta(\bar{x}_A)e^{-(\bar{x}_A - \bar{x})/\tau_d} \quad (2.43)$$

Here, the angle error $\Delta\theta$ is a function of \bar{x} and approaches zero as \bar{x} approaches zero. The distance constant τ_d is a measure of how quickly the error approaches zero. This analysis is much more direct than with a time constant for the given application, so it will be worked into the outer-loop design process. The manner in which this is accomplished will be explained in Section 2.8.5.

2.8.4 Desired Dynamics. The desired dynamics section of the outer-loop controller is somewhat different from that of the inner-loop controller. The desired dynamics for the thrust loop are structured as a PID controller with a limited integral, just as with the inner-loop. However, the difficulty of obtaining a perfect model for the combination of the plant and the inner-loop controller combined with the importance of driving the glidepath errors to zero warranted a more robust controller for the heading and pitch loops. Therefore, a simple LQR design assuming a perfect inversion in both the inner- and outer-loops was utilized, allowing the built-in robustness of LQR controllers to take into account any model imperfections.

2.8.5 Inversion. As previously mentioned, the inversion process for the outer-loop differs in multiple regards from the process used for the inner-loop. The lack of control affine form in the nonlinear equations of motion and the utilization of the distance domain for the heading and pitch loops required a different formulation of the inversion process outlined in Section 2.3.4. The thrust loop, however, is much simpler and will be discussed first.

The plant model for the outer-loop controller design was detailed in Equation 2.35. However, the only two states required for the thrust loop are u and w , as these are the only two states upon which α is dependent. With this in mind and recalling the definition of TCV in Equation 2.42, consider the following:

$$x = \begin{bmatrix} u \\ w \end{bmatrix}, \quad V = TCV, \quad \dot{V} = \frac{\partial V}{\partial x} \dot{x}$$

$$\Rightarrow \dot{V} = \frac{\partial V}{\partial u} \dot{u} + \frac{\partial V}{\partial w} \dot{w} \quad (2.44)$$

Substituting in the expressions for \dot{u} and \dot{w} from Equation 2.35, replacing \dot{V} with \dot{V}_{des} , and solving for T yields:

$$T = \frac{\dot{V}_{des} - \left[\frac{\partial V}{\partial u} \left(-\frac{g}{W} QSC_A - g \sin \theta + rv - qw \right) + \frac{\partial V}{\partial w} \left(-\frac{g}{W} QSC_N + g \cos \phi \cos \theta + qu - pv \right) \right]}{\frac{\partial V}{\partial u} \frac{g}{W}} \quad (2.45)$$

This is the control law for the thrust necessary to achieve the desired angle of attack.

The heading and pitch loops cannot be written in control affine form and cannot be treated in the same manner as the inner-loop inversion process. Therefore, a linearized version of the inversion process must be performed. Both the heading and pitch loops will be treated simultaneously.

First, the problem is simplified slightly by removing u and w from the state vector, as only the aircraft and ship positions are required for these two loops. Next, the EOM set must be transformed into the distance domain. Define the distance from the aircraft to the touchdown point on the flight deck as \bar{x} and consider the following:

$$x = \begin{bmatrix} x_e \\ y_e \\ z_e \\ x_{Be} \\ y_{Be} \end{bmatrix}, \quad V = \begin{bmatrix} HCV \\ PCV \end{bmatrix}, \quad \dot{x} = f(x, u)$$

$$\bar{x} = \sqrt{(x_{Be} - x_e)^2 + (y_{Be} - y_e)^2} \Rightarrow \frac{dx}{d\bar{x}} = \frac{dx}{dt} \frac{dt}{d\bar{x}} = \frac{\dot{x}}{\dot{\bar{x}}} \Rightarrow \frac{\dot{x}}{\dot{\bar{x}}} = f(x, u) \quad (2.46)$$

This set of equations shows how it is possible to express the problem in the distance domain. Expanding on this concept yields:

$$\begin{aligned}
\frac{\dot{x}}{\dot{\bar{x}}} &= \frac{1}{\bar{x}} [(x_e - x_{Be})\dot{x}_e + (y_e - y_{Be})\dot{y}_e + (x_{Be} - x_e)\dot{x}_{Be} + (y_{Be} - y_e)\dot{y}_{Be}] \\
&= \frac{1}{\bar{x}} [(x_{Be} - x_e)(\dot{x}_{Be} - \dot{x}_e) + (y_{Be} - y_e)(\dot{y}_{Be} - \dot{y}_e)] = \frac{\kappa}{\bar{x}}
\end{aligned} \tag{2.47}$$

This expression in terms of the states and their time derivatives allows for the time derivative of the state vector to be written in the distance domain as:

$$\frac{\dot{x}}{\dot{\bar{x}}} = \begin{bmatrix} \frac{\dot{x}_e \bar{x}}{\kappa} \\ \frac{\dot{y}_e \bar{x}}{\kappa} \\ \frac{\dot{z}_e \bar{x}}{\kappa} \\ \frac{\dot{x}_{Be} \bar{x}}{\kappa} \\ \frac{\dot{y}_{Be} \bar{x}}{\kappa} \end{bmatrix} \tag{2.48}$$

Now that the plant is expressed in the distance domain, the linearization and inversion will be performed. Consider the plant model and its approximation with a Taylor series expansion about some arbitrary flight condition:

$$\begin{aligned}
\frac{\dot{x}}{\dot{\bar{x}}} &= f(x, u) \\
\Rightarrow \left. \frac{\dot{x}}{\dot{\bar{x}}} \right|_{(x_0, u_0)} &\approx f(x_0, u_0) + \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial u} \Delta u \\
&\Rightarrow \left. \frac{\dot{x}}{\dot{\bar{x}}} \right|_{(x_0, u_0)} \approx f(x, u_0) + B \Delta u
\end{aligned} \tag{2.49}$$

where $B = \frac{\partial f}{\partial u}$. Also consider the variation of the control variable vector V with \bar{x} :

$$\frac{dV}{d\bar{x}} = \frac{\partial V}{\partial x} \frac{dx}{d\bar{x}} = \frac{\partial V}{\partial x} \frac{\dot{x}}{\dot{\bar{x}}} = C \frac{\dot{x}}{\dot{\bar{x}}} \tag{2.50}$$

using a substitution from Equation 2.46. Substituting Equation 2.49 into Equation 2.50 yields:

$$\frac{dV}{d\bar{x}} = C[f(x, u_0) + B\Delta u] \quad (2.51)$$

where $C = \frac{\partial V}{\partial x}$. Finally, solving for the control input Δu yields:

$$\Delta u = [CB]^{-1} \left[\left(\frac{dV}{d\bar{x}} \right)_{des} - Cf(x, u_0) \right] \quad (2.52)$$

The B and C matrices are symbolically entered into the controller such that they are updated at every time step. Thus, the system is continually re-linearized at every time step. Thus, the final form of the perturbation control input in Equation 2.52 is added to the control input vector for the previous time step to yield the required control input.

2.9 Transitional Dynamics

Recall that the outputs of the outer-loop controller are desired values of ψ , θ , and T . However, the inputs to the inner-loop controller are $\dot{\phi}$, $\dot{\theta}$, and T . While T feeds directly from the outer-loop to the inner-loop, ψ must be changed to $\dot{\phi}$ and θ must be changed to $\dot{\theta}$.

2.9.1 Heading Angle to Roll Rate. Define $\Delta\psi$ as the difference between the desired and actual heading angles, $\psi_{cmd} - \psi$. Then, $\Delta\psi$ can easily be transformed into a desired $\dot{\psi}$ through a desired dynamics loop similar to those used in the inner- and outer-loop controllers. In this case, just a proportional gain was used. See Chapter 3 for more details.

Once $\dot{\psi}$ was obtained, it can be related to a bank angle. The steady, coordinated turn relationships in Reference [24, pages 283,291] yield:

$$\phi_{des} = \tan^{-1} \left(\frac{\bar{V}}{g} \dot{\psi} \right) \quad (2.53)$$

where \bar{V} is the magnitude of the aircraft velocity. With this desired bank angle, another desired dynamics loop, this one also consisting of only a proportional gain, finally yields $\dot{\phi}$.

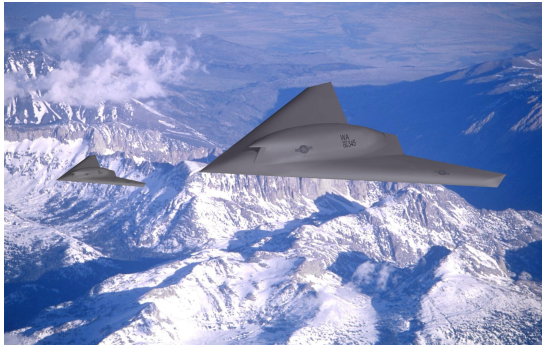
2.9.2 Pitch Angle to Pitch Rate. The transformation of θ to $\dot{\theta}$ is much simpler than the process to convert ψ to $\dot{\phi}$. It only involves one desired dynamics loop. As with the others in the transitional dynamics, it consisted of merely a proportional gain.

III. Modeling and Implementation

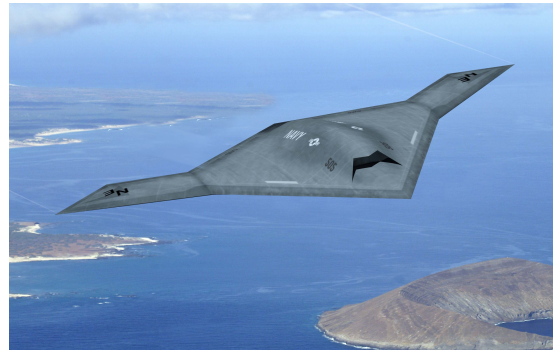
3.1 Aircraft Model

The aircraft which was used for this research was the Joint Unmanned Combat Air System (J-UCAS) Equivalent Model (EQ model) developed by the Air Force Research Laboratory (AFRL). The EQ model was actually developed to facilitate collaboration on research in the area of Automated Aerial Refueling (AAR) [4]. However, this model was easily adapted for use in the carrier landing environment.

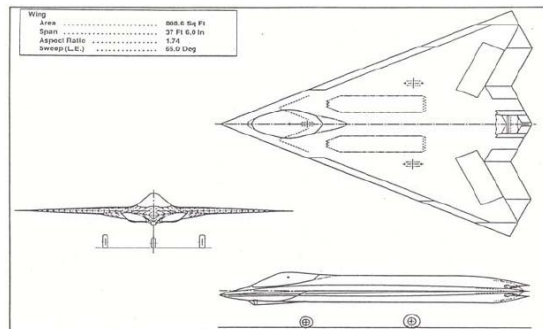
The EQ model was blended from the characteristics of three prototype and research aircraft. The first was the Boeing X-45C, one of the two competitors in the N-UCAS program. The second was the Northrop Grumman X-47B, the other competitor in the N-UCAS program. The final model included was the Innovative Control Effector (ICE) research aircraft developed by AFRL. The EQ model provided by AFRL included a physical layout, weights and other physical parameters, suggested actuator and engine models, and aerodynamic look-up tables [1–3].



(a) X-45C [10]



(b) X-47B [10]



(c) ICE [4, page 2]

Figure 3.1: X-45C, X-47B, and ICE.

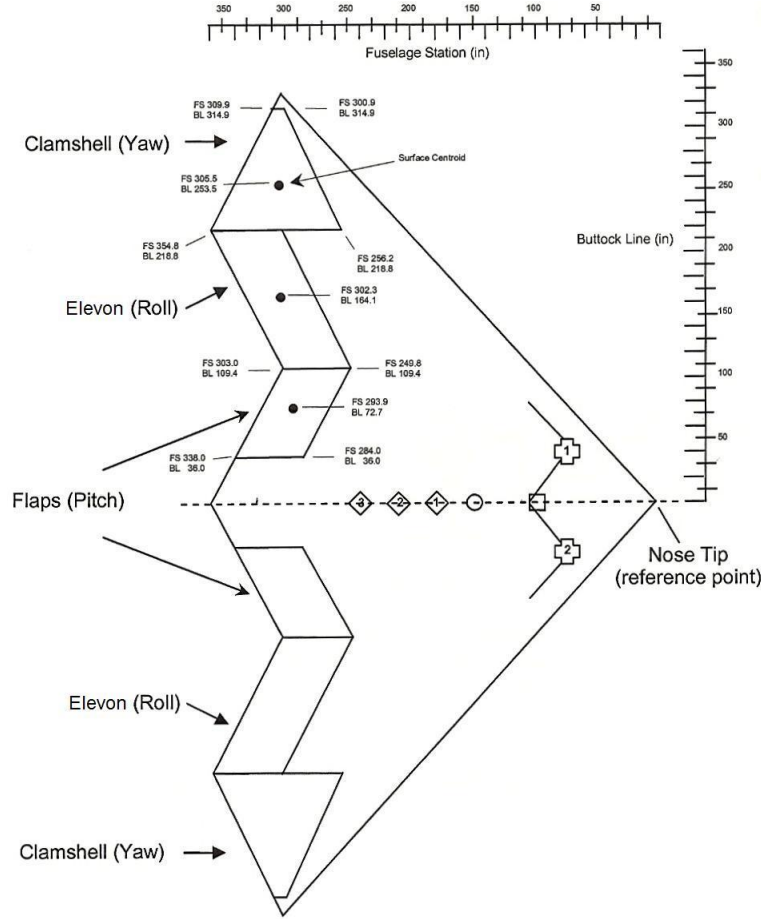


Figure 3.2: Physical Layout of EQ Model [4, page 5].

3.1.1 Physical Layout. Figure 3.2 illustrates the physical layout of the EQ model. It is a tailless, flying wing configuration similar to the B-2. It has three sets of control surfaces available for use: pitch flaps for pitch control, elevons for high lift devices and roll control, and clamshells for speed breaks and yaw control. No speed breaking was employed in this research. The flaps were employed symmetrically, so that they only affected the pitch moment of the aircraft. The elevons were restricted to have deflections opposite in sign, plus or minus a flap setting. Each clamshells consists of surfaces on both the top and bottom of the aircraft which open symmetrically to prevent pitching moments due to clamshell deflections. For the purposes of yaw control, a deflection in the left clamshell was considered analogous to a positive, trailing-edge-left rudder deflection causing a negative yawing moment.

Table 3.1: Summary of Aircraft Physical Parameters.

Parameter	Value
Weight (W)	26,421 lbs
MAC (\bar{c})	18.63 ft
Leading Edge of MAC (from nose)	8.98 ft
CG Location	35.53 % MAC
Wingspan (b)	54.70 ft
Wing Planform Area (S)	808.58 ft ²
Aspect Ratio	3.70
Length at Centerline	29.56 ft
I_{xxb}	48,120 slug·ft ²
I_{yyb}	31,640 slug·ft ²
I_{zzb}	98,475 slug·ft ²
I_{xzb}	0 slug·ft ²

The aircraft has a single engine inlet for a single, centerline mounted turbofan engine. For purposes of this research, the thrust was also assumed to act through the center of gravity of the aircraft, collinear with the x -axis.

3.1.2 Physical Parameters. The various physical parameters for the aircraft which were used during the simulation are summarized in Table 3.1. The values of the weight and moments of inertia were assumed to remain constant for each simulation run and were based on an assumed 10% fuel load with full weapons [2]. The rest of these values were obtained from the EQ model specification [3, pages 3,6].

3.1.3 Actuator Models. The actuators used for this research were all assumed to be simple, second order actuators with both position and rate saturation. Table 3.2 summarizes the characteristics of the actuators used in the simulation. These values were recommended in the EQ model specification [3, page 4]. The specified clamshell deflection limits apply to both the top and bottom surfaces, as they deflect symmetrically.

3.1.4 Engine Model. While the EQ model specification did include recommendations for an engine model [3, page 5], a different set of characteristics were used

Table 3.2: Summary of Actuator Parameters.

Actuator	ω (rad/sec)	ζ	Upper Limit (deg)	Lower Limit (deg)	Max Rate (deg/sec)
Pitch Flap (δ_f)	50	0.8	30	-30	90
Elevon (δ_e)	50	0.8	45	-45	90
Clamshell (δ_c)	50	0.8	60	0	90

Table 3.3: Summary of Engine Specifications.

Model	ω_n (rad/sec)	ζ	Max Thrust (lbs)	Idle Thrust (lbs)
Recommended (2 nd Order)	2.4	0.9	5,600	600
Used (1 st Order)	2.4	N/A	10,000	1,000

for the engine model in this research. The recommended engine characteristics from the EQ model specification are summarized in Table 3.3, along with the values that were actually used.

The reason for the choice of another engine model is that the recommended model did not accurately depict the thrust-to-weight ratio required of carrier-based aircraft. The Northrop Grumman EA-6B Prowler, an aircraft with similar mission profiles and maximum airspeed to the planned Navy UCAV, has a maximum thrust of 22,000 pounds and maximum carrier landing weight (max trap weight) of approximately 45,500 pounds [8]. At max trap weight, the Prowler has a thrust-to-weight ratio of approximately 48%. Compare this to the UCAV model used in this research with a weight of 26,421 pounds. If the recommended engine model with a maximum thrust of only 5,600 pounds were used, this would give the UCAV a thrust-to-weight ratio of only 21.2%. This may not be enough thrust to get the aircraft back into the air in the event of a bolter. Therefore, a model with higher thrust was needed.

The engine currently planned for use on the X-45C is the General Electric F404-102D [12]. It is a variant of the F404 engine which powers the F/A-18, F-117, and other modern aircraft. The maximum thrust on the F404-102, a similar model, is 17,700 pounds [13]. The engine currently planned for use on the X-47B is a variant

of the Pratt and Whitney F100-220E currently used on the F-15E and the F-16 [26]. The maximum thrust on the F100-220E is 23,770 pounds [25]. Therefore, increasing the maximum thrust of the engine for this research to 10,000 pounds is a conservative increase, as it is well within the thrust range of these engines. This increases the thrust-to-weight ratio of the UCAV in this research to 37.8%, which is closer to the 48% of the Prowler.

The other difference between the recommended engine model and the one used for this research was the damping ratio. The turbofan engine model which is built into the Simulink[®] Aerospace Blockset is only a first-order engine model, so the model used was a first-order model. Since the recommended second-order model was nearly critically damped, as shown in Table 3.3, the engine model used assumed the same natural frequency as the recommended model.

3.1.5 Aerodynamic Model. The EQ aerodynamic model used for this research consisted of look-up tables for the force and moment coefficients. Each coefficient had several components which were functions of the states and control surface deflections. The basic structure of the look-up tables which comprised the aerodynamic model is summarized in Table 3.4. The actual tables themselves can be found in Reference [1]. The sign conventions used for the coefficients are summarized in Table 3.5.

Some small modifications were made to the original model in Reference [1]. First, the axial force and pitch moment coefficient values were originally calculated for only a clean configuration (landing gear retracted). However, a landing aircraft would have gear extended, having a significant effect on both axial force and pitching moment. To improve the accuracy of the aircraft model, simple correction factors were calculated for the axial force and pitching moment coefficients based on empirical data [5]. The corrections used were $\Delta C_A = 0.0145$ for axial force and $\Delta C_m = -0.0026$ for pitching moment.

Table 3.4: Summary of Aerodynamic Look-Up Tables.

Coefficient	Component	Function of
Axial Force (C_A)	Basic	α, β, M
	δ_e	α, δ_e, M
	δ_f	α, δ_f, M
	δ_c	α, δ_c, M
Side Force (C_Y)	Basic	β
Normal Force (C_N)	Basic	α, β, M
	δ_e	α, δ_e, M
	δ_f	α, δ_f, M
	δ_c	α, δ_c, M
Rolling Moment (C_ℓ)	Basic	β
	δ_e	α, δ_e, M
	δ_c	α, δ_c, M
	p	α, p, M
	r	α, r, M
Pitching Moment (C_m)	Basic	α, M
	δ_f	α, δ_f, M
	q	α, q, M
Yawing Moment (C_n)	δ_e	α, δ_e, M
	δ_c	α, δ_c, M
	p	α, p, M
	r	α, r, M

Table 3.5: Sign Conventions for Aerodynamic Coefficients.

Coefficient	Positive Direction
Axial Force (C_A)	Aft
Side Force (C_Y)	Right
Normal Force (C_N)	Out Top
Rolling Moment (C_ℓ)	Right Wing Down
Pitching Moment (C_m)	Nose Up
Yawing Moment (C_n)	Nose Right

The other modification to the aerodynamic model was the addition of side force. In the original look-up tables, side force was neglected, as it was assumed to be negligible because of the small side area of the aircraft. However, without the inclusion of a side force term, the wind would have no effect on the aircraft ground track component perpendicular to the direction of travel. Essentially, a crosswind would not effect the aircraft. This would render most of the landing analysis useless, so side force was accounted for. A simple side force model was used [6]. It consisted of a constant slope of side force coefficient with respect to sideslip angle, $C_{Y,\beta} = -0.0258 \frac{1}{\text{rad}}$, where

$$C_Y = C_{Y,\beta} \beta \quad (3.1)$$

3.2 Aircraft Carrier Model

The aircraft carrier considered in this research was a generic Nimitz Class carrier. The basic geometry of the ship is illustrated in Figure 3.3. Point *A* was considered to be the center of mass of the ship for this research. Point *C* was the desired touchdown point. The distances of the line segments defined by points *A*, *B*, *C*, and *D* are shown in Table 3.6. Also included in Table 3.6 are the other pertinent dimensions of the ship [?].

The aircraft carrier motion model which was used for this research consisted of two components. The first was a simple translational model based on a constant ship speed, and the second was a perturbation model based upon sea state. The two components were summed to yield the 6 DOF aircraft carrier motion.

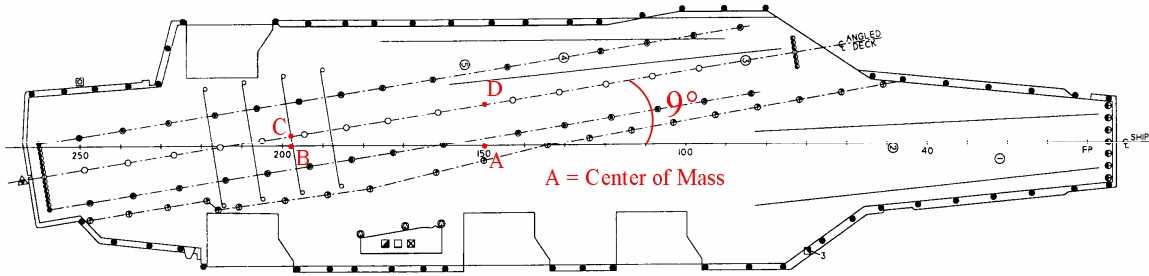


Figure 3.3: Nimitz Class Aircraft Carrier [14].

Table 3.6: Summary of Aircraft Carrier Dimensions.

Measurement	Value
Length	1,092 ft
Width	252 ft
Deck Height (above waterline)	≈ 70 ft
Center of Mass Height (above waterline)	≈ 20 ft
Flight Deck Angle Offset	9 deg
Segment AB	193 ft
Segment BC	10 ft
Segment CD	198 ft
Segment AD	43 ft

3.2.1 Translational Motion Model. The translation of the ship was assumed to be two-dimensional, restricted to the xy -plane in the ENU coordinate frame. The position of the center of mass of the ship did have a z component, but it was assumed to be constant for translational purposes. The equations which describe the translational portion of the ship motion are:

$$\begin{bmatrix} \dot{x}_{Be_{CM}} \\ \dot{y}_{Be_{CM}} \end{bmatrix} = \begin{bmatrix} V_B \sin \psi_B \\ V_B \cos \psi_B \end{bmatrix} \quad (3.2)$$

where $x_{Be_{CM}}$ and $y_{Be_{CM}}$ are the x and y coordinates of the center of mass of the ship.

3.2.2 Sea State Perturbation Model. The sea state perturbation model used was based on the motion model described in Reference [23, pages 94–99] and formulated in Reference [19]. As previously noted, only the simple, single sine wave version of the perturbation model was used. Also, yaw perturbations were neglected, as their magnitudes were significantly smaller than the other perturbations. The values used for the amplitude and frequency for each of the perturbations for the sea states used in this research are summarized in Table 3.7. The values for sea state 0 were all zero.

Table 3.7: Summary of Sea State Perturbations.

Sea State	Perturbation	Amplitude	Frequency
4	Roll	0.6223 deg	0.2856 rad/sec
	Pitch	0.5162 deg	0.5236 rad/sec
	Surge	0.9546 ft	0.3307 rad/sec
	Sway	1.4142 ft	0.3307 rad/sec
	Heave	2.2274 ft	0.3491 rad/sec
5	Roll	0.9829 deg	0.2856 rad/sec
	Pitch	0.8202 deg	0.5236 rad/sec
	Surge	1.5203 ft	0.3307 rad/sec
	Sway	2.2627 ft	0.3307 rad/sec
	Heave	3.5638 ft	0.3491 rad/sec
6	Roll	1.4425 deg	0.2856 rad/sec
	Pitch	1.2374 deg	0.5236 rad/sec
	Surge	2.2840 ft	0.3307 rad/sec
	Sway	3.3941 ft	0.3307 rad/sec
	Heave	5.3528 ft	0.3491 rad/sec

The values for the surge, sway, and heave perturbations were added to the ship center of mass x , y , and z positions from the translational model accordingly:

$$\begin{bmatrix} \Delta x_{BeCM} \\ \Delta y_{BeCM} \\ \Delta z_{BeCM} \end{bmatrix} = \begin{bmatrix} \sin \psi_B & \cos \psi_B & 0 \\ \cos \psi_B & \sin \psi_B & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} surge \\ sway \\ heave \end{bmatrix} \quad (3.3)$$

Since the roll and pitch angles of the ship were assumed to be zero from the translational model, the perturbations are the values of the ship roll and pitch angles. Since the pitch and roll oscillations take place around the assumed center of mass, the geometry shown in Figure 3.3 and Table 3.6 allows for the calculation of how the pitch and roll of the ship affect the position of the desired touchdown point. It can

be shown that:

$$\begin{bmatrix} \Delta x_{Be} \\ \Delta y_{Be} \\ \Delta z_{Be} \end{bmatrix} = \begin{bmatrix} C_{\theta_B} S_{\psi_B} & S_{\phi_B} S_{\theta_B} S_{\psi_B} + C_{\phi_B} C_{\psi_B} & C_{\phi_B} S_{\theta_B} S_{\psi_B} - S_{\phi_B} C_{\psi_B} \\ C_{\theta_B} C_{\psi_B} & S_{\phi_B} S_{\theta_B} C_{\psi_B} - C_{\phi_B} S_{\psi_B} & C_{\phi_B} S_{\theta_B} C_{\psi_B} + S_{\phi_B} S_{\psi_B} \\ S_{\theta_B} & -S_{\phi_B} C_{\theta_B} & -C_{\phi_B} C_{\theta_B} \end{bmatrix} \begin{bmatrix} -AB \\ -BC \\ -\Delta z_{CM} \end{bmatrix} \quad (3.4)$$

where ϕ_B and θ_B are the ship roll and pitch angles, respectively, and Δx_{Be} , Δy_{Be} , and Δz_{Be} are the differences between the ship center of mass and aimpoint coordinates in the ENU frame. Also, AB and BC are dimensions from Table 3.6 and Δz_{CM} is the distance between the center of mass of the ship and the location above it on the flight deck ($\Delta z_{CM} = 50$ ft). The transformation in Equation 3.4 is similar to the use of the DCM to transform the aircraft velocities in the body axis into the ENU coordinate frame.

3.3 Burble Model

The burble model used was based on the model described in Reference [23, pages 102–103] and was formulated in References [18, 20, 21]. As previously noted, only the steady portion of the burble was implemented. The slow periodic portion was neglected, as it would have little effect on an approaching aircraft, and the unsteady portion was neglected because an overall turbulence model was already included.

The steady portion of the burble model was structured such that the wind experienced by the aircraft within the burble region was a function of the aircraft's position behind the ship and altitude. This was implemented with a look-up table on altitude and distance behind the ship. Additionally, this look-up table was dependent on the wind-over-deck (WOD) angle and magnitude. For additional details, refer to References [18, 20, 21].

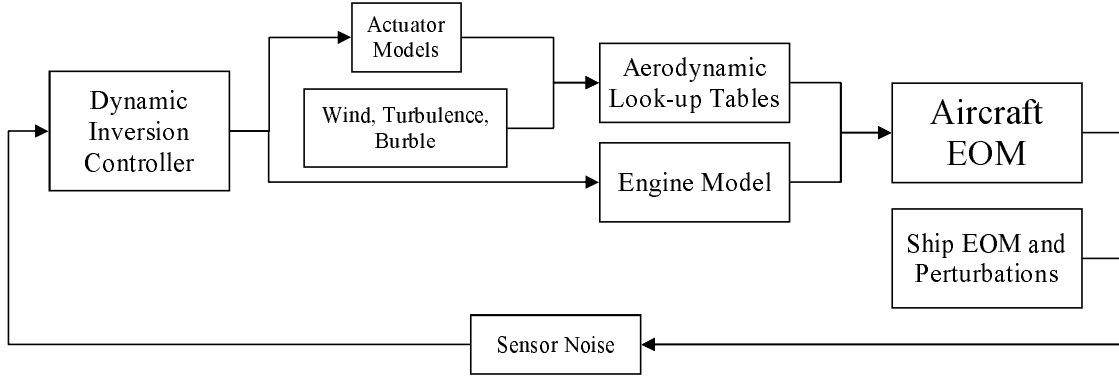


Figure 3.4: Conceptual Simulator Block Diagram.

3.4 Simulator Structure

The simulation for this research was conducted entirely in the MATLAB[®]/Simulink[®] environment. The basic layout of the simulator is illustrated in Figure 3.4. It is intended as an overall conceptual representation. Each of the blocks shown in Figure 3.4 will be addressed in detail in this section.

3.4.1 Solver. The continuous-time real world cannot be exactly replicated by a simulation on a digital computer, so a simulation is only as accurate as its solver. For this research, the Simulink[®] *ode5* fixed-step Dormand-Prince solver was used with a time-step interval of 0.01 seconds. This 100 Hz frequency far exceeded the prevailing frequencies of any of the aircraft, noise, or turbulence dynamics and allowed the simulation to run at a speed faster than real-time. Initial simulations yielded acceptable results, so this solver combination was used for all of the simulation.

3.4.2 Aircraft Equations of Motion. Figure 3.5 illustrates the layout of the aircraft equations of motion block. Its inputs were the forces and moments due to aerodynamics and thrust, the constant weight, and the constant inertia tensor. Its outputs were the twelve aircraft states from Equations 2.7–2.10 and the DCM. It was built from the nonlinear equations of motion in Equations 2.7–2.10. This Euler angle

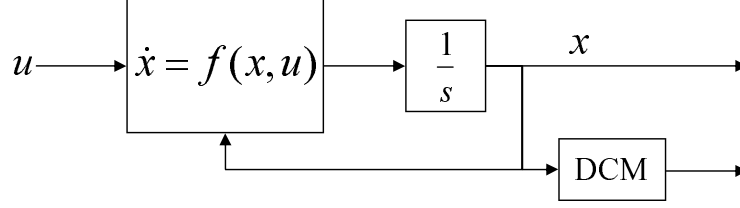


Figure 3.5: Layout of Aircraft Equations of Motion Block.

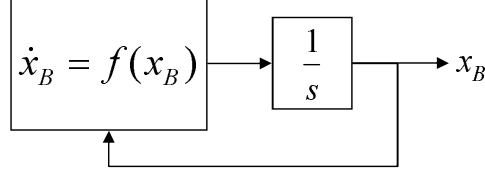


Figure 3.6: Layout of Ship Equations of Motion Block.

formulation was used, rather than a quaternion formulation, because an aircraft on approach will most likely not come close to the singularity points at $\theta = \pm 90^\circ$.

3.4.3 Ship Equations of Motion. Figure 3.6 illustrates the layout of the ship equations of motion block. This particular block was self-contained, so there were no actual inputs to it. However, it did require that a constant velocity magnitude and heading be specified. The outputs were the ship center of mass x , y , and z coordinates in the ENU coordinate system, as well as ϕ_B , θ_B , and ψ_B . Since this block was responsible only for translational motion in the xy -plane, ϕ_B and θ_B were zero, and ψ_B and $z_{Be_{CM}}$ were constant. However, these values did not represent the “actual” position and orientation of the ship, as they still had to be summed with the perturbations due to sea state conditions. The EOM set around which this block was built was introduced in Equation 3.2.

3.4.4 Sea State Perturbations. Figure 3.7 illustrates the layout of the sea state perturbation block. As with the ship equations of motion block, the sea state perturbation block had no inputs. However, contained within the sea state block were the sinusoids for each of the five perturbations considered in this research, outlined

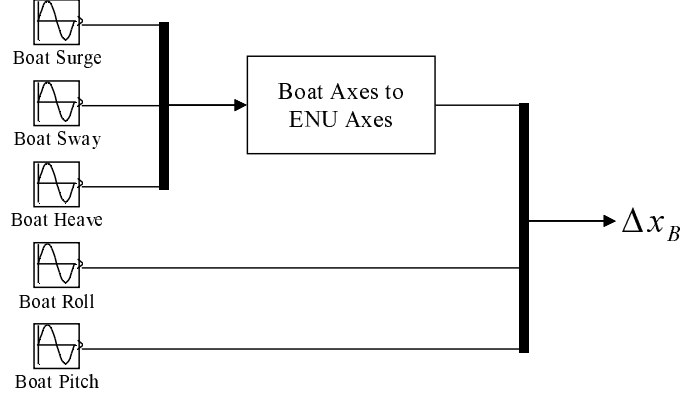


Figure 3.7: Layout of Sea State Perturbation Block.

in Section 3.2.2. Each of the sinusoids required that an amplitude and frequency be specified. These values were summarized in Table 3.7. Also, randomly generated phase angles between 0 and 2π were specified for each sinusoid such that no two simulation runs for a given case had the same perturbation correlations. The outputs of this block were the position and Euler angle perturbations. These perturbations could then be summed with the corresponding values resulting from the translational motion model to yield the actual ship position and orientation. From this, the position of the landing aimpoint on the flight deck could then be calculated using Equation 3.4.

3.4.5 Aerodynamic Look-Up Tables. Figure 3.8 illustrates the layout of the aerodynamic look-up tables. The inputs to the look-up tables were the aircraft relative wind components, angular velocities, control surface deflections, and Mach number. In the calculation of the aerodynamic coefficients, it is the relative wind components that matter, not the aircraft velocity components from the equations of motion. The actual process used to obtain the relative wind components involved subtracting the wind components resolved in the body axes from the aircraft velocities resolved in the body axes. The calculation of the wind components in the body axes will be discussed in Sections 3.4.8–3.4.10.

3.4.6 Engine Model. The engine model which was used for this research was the generic turbofan engine model built into the Simulink® Aerospace Blockset. It

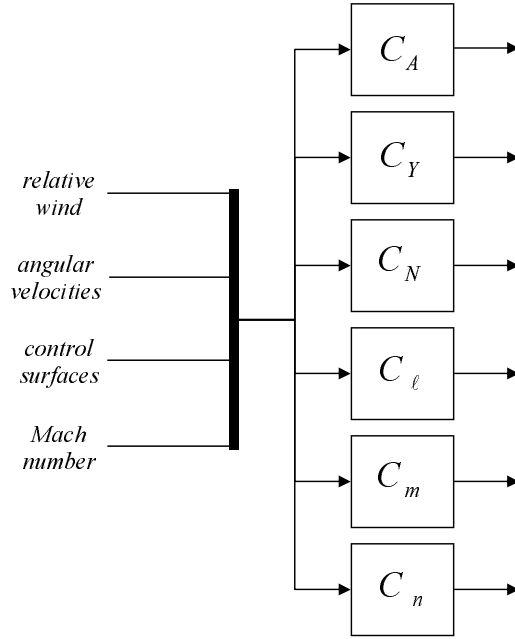


Figure 3.8: Layout of Aerodynamic Look-Up Tables.

was a first-order engine model for which maximum sea-level thrust and engine time constant were specified. The values used for these parameters were summarized in Table 3.3. The inputs to this engine model were throttle position, Mach number, and altitude, and the output was thrust. The engine model block used multiple look-up tables to determine the thrust as a percentage of the sea-level maximum thrust. This value was then multiplied by the specified maximum sea-level thrust to yield thrust.

3.4.7 Actuator Models. The actuator model block which was used for all of the actuators in this research was the generic second order nonlinear actuator model built into the Simulink® Aerospace Blockset. It was a second-order actuator model with upper and lower saturation limits and a rate limit. The parameters which the model required were a natural frequency, a damping ratio, upper and lower saturation limits, and a rate limit. The values which were used for each of the actuator types were summarized in Table 3.2. The input was a commanded actuator deflection, and the output was an actual deflection.

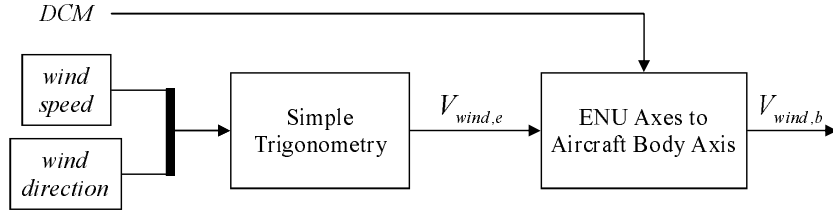


Figure 3.9: Layout of Horizontal Wind Model Block.

3.4.8 Horizontal Wind Model. The wind model used was the horizontal wind model built into the Simulink[®] Aerospace Blockset. The input was the aircraft DCM and the outputs were the wind components resolved in the aircraft body axes. Additionally, wind speed and direction were specified. Figure 3.9 illustrates the layout of the horizontal wind model. The wind speed and direction were converted into a wind vector in the ENU coordinate frame using a coordinate transformation. These wind components were then rotated into the aircraft body axes using the aircraft DCM. The result was a three-component wind vector in the aircraft body axes which could then be summed with the turbulence and burble wind components after they were calculated in order to yield an overall wind vector. Ultimately, the overall wind vector would be subtracted from the aircraft velocities in the body axes to yield the relative wind experienced by the aircraft.

3.4.9 Wind Turbulence Model. The turbulence model used for this research was the Continuous Dryden Turbulence Model built into the Simulink[®] Aerospace Blockset. Its inputs were the aircraft DCM, altitude, and true airspeed, and the outputs were the turbulence components of the wind resolved in the aircraft body axes. In addition to these inputs, there were a number of parameters which had to be specified. The wind speed and direction from the horizontal wind model, the wingspan, noise seeds, and the noise sample time were some of these parameters. The noise seeds were changed for every simulation run, and the sample time used was 0.01 seconds. Also, the scale length for the turbulence was specified, which was the default value for Dryden Turbulence of 1,750 feet. Also, a choice was made regarding which

specification was used for the turbulence formulation. The choices available were MIL-F-8785C and MIL-HDBK-1797 specifications. The MIL-HDBK-1797 specification was chosen for this research, since it was the more recent publication of the two choices. The final parameter involved choosing the turbulence intensity. This was done by choosing the probability of exceedance of the high-altitude intensity. Table 3.8 and Figure 3.10 summarize the definitions of the various turbulence intensities which were used during the course of the simulation.

The actual model included white noise inputs which were passed through coloring filters designed to meet the MIL-HDBK-1797 specifications. The resulting signals were scaled by the turbulence intensities resulting from look-up tables scheduled on altitude. This yielded turbulence components in the NED coordinate frame which were then rotated using the aircraft DCM into the aircraft body axes. These resulting turbulence components could then be summed with the horizontal wind and burble components to yield the overall wind vector.

3.4.10 Burble Model. Figure 3.11 illustrates the layout of the burble model block. The inputs to the burble model were the aircraft distance behind the carrier, aircraft DCM, and ship DCM. The outputs were the burble wind components of the wind resolved in the aircraft body axes, such that summing it with the horizontal wind and wind turbulence components yielded the total wind acting on the aircraft. In addition to the inputs, the ship speed and WOD magnitude were specified. The burble components in the boat keel axes, analogous to the aircraft body axes, are denoted as $u_{burble,k}$, $v_{burble,k}$, and $w_{burble,k}$, respectively. They were determined from look-up tables scheduled on the aircraft distance behind the ship, \bar{x} . The outputs of the look-up tables were the burble components as a fraction of the WOD magnitude. Thus, scaling them by the WOD magnitude yielded the burble components in the boat keel axes. They were then rotated into the NED coordinate frame with the boat DCM and then into the aircraft body axes using the aircraft DCM. These burble components could then be summed with the horizontal wind and wind turbulence

Table 3.8: Summary of Turbulence Intensities.

Intensity	Probability of Exceedance
Light	10^{-2}
Medium	10^{-3}
Heavy	10^{-4}

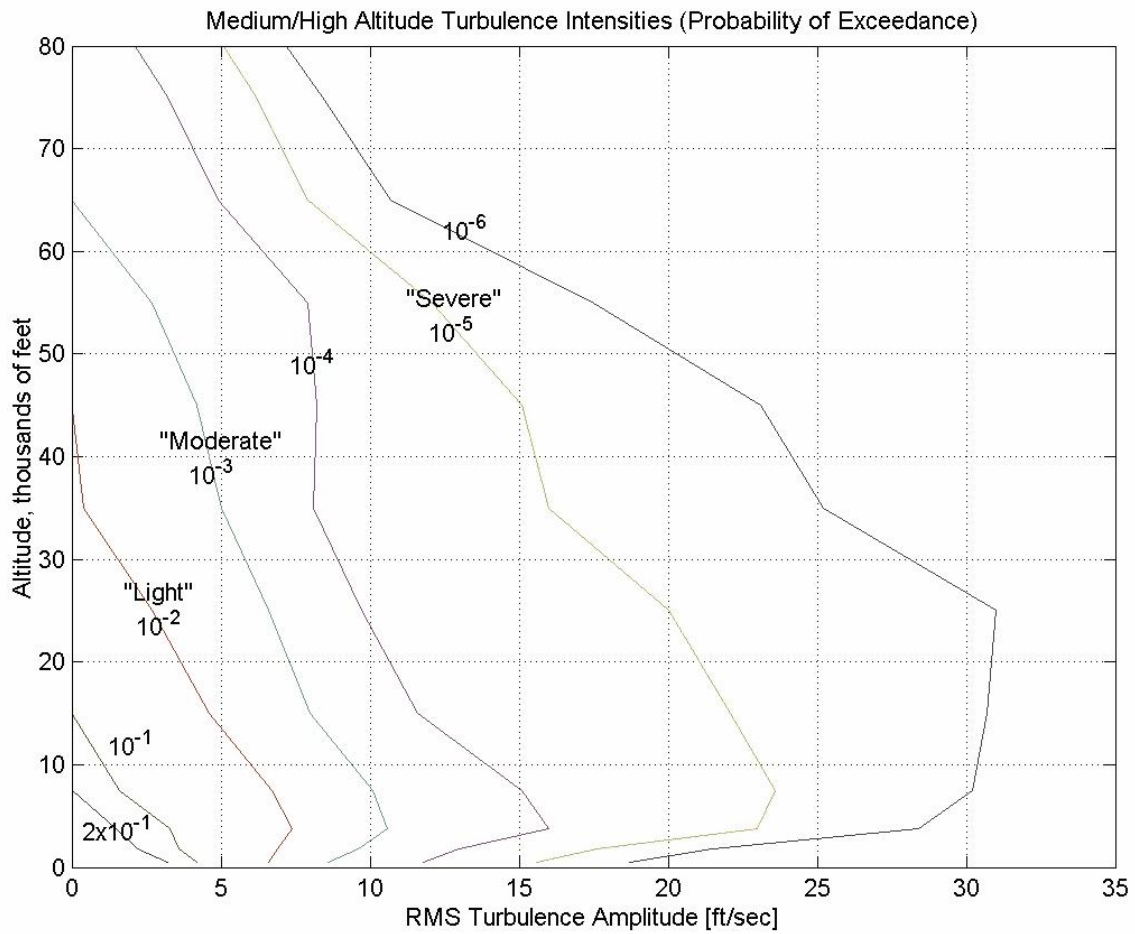


Figure 3.10: Turbulence Intensities [16].

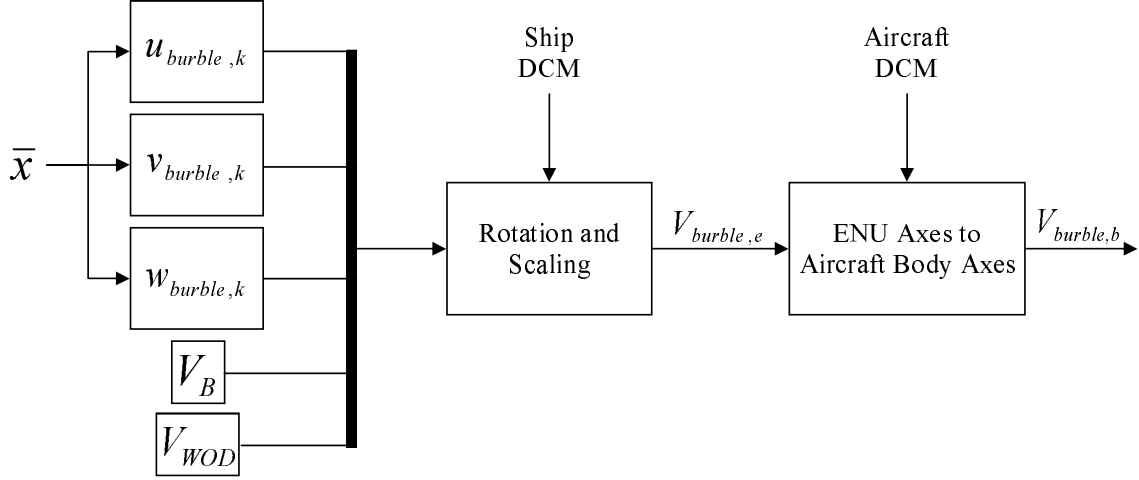


Figure 3.11: Layout of Burble Model Block.

components resolved in the body axes to yield the overall wind vector. The overall wind could then be subtracted from the aircraft velocities in the body axes to yield the relative wind vector resolved in the body axes for calculation of the aerodynamic force and moment coefficients.

3.4.11 Sensor Noise. Sensor noise was simulated by summing pseudorandom noise inputs with selected parameters for the aircraft and ship. This was accomplished with the Simulink® uniform random number block. The parameters to which noise were added are summarized in Table 3.9, along with their corresponding noise amplitudes. Note that noise was not added to the positions of the aircraft and ship. This decision was based on the assumption that a combination of differential GPS and shipboard line-of-sight tracking systems could provide the position of the aircraft relative to the ship with great enough accuracy to support a system similar to the one proposed in this research.

3.4.12 Controller Model Look-Up Tables. The Dynamic Inversion Controller block in Figure 3.4 is expanded in Figure 3.12. Its components are now discussed, starting with the model look-up tables. The aircraft model block which was used for the controller was very similar to the one used to run the simulator. The model used

Table 3.9: Summary of Sensor Noise Amplitudes.

Measurement	Noise Amplitude
u, v, w	± 1.0 fps
p, q, r	± 0.1 deg/sec
ϕ, θ, ψ	± 0.1 deg
V_B	± 0.5 fps
ψ_{BF}	± 0.1 deg

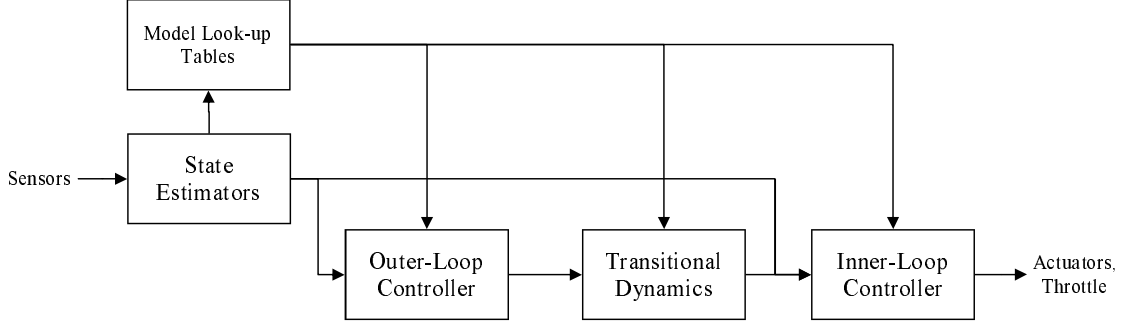


Figure 3.12: Conceptual Dynamic Inversion Controller Block Diagram.

by the controller contained the same look-up tables as shown in Figure 3.8, but with one slight modification. The coefficients were obtained as before, but the coefficient gradients were also needed as functions of each of the control surfaces. Therefore, a backward difference was calculated from an identical set of look-up tables by specifying a small change in the control surface deflections and subtracting this from the values for the actual control surface deflections. For the purposes of this research, a step of 0.1 degrees was used as the small change in control surface deflections. The gradient estimates were then obtained by dividing each of the backward differences by 0.1 degrees. This gradient calculation process is summarized by:

$$\frac{\Delta C_i}{\Delta \delta_j} = \frac{C_i(\delta_j)_{actual} - C_i(\delta_j - 0.1^\circ)}{0.1^\circ} \quad (3.5)$$

Most of this discussion assumes single elevon and clamshell deflections. However, there are actually right and left elevons and right and left clamshells which deflect independently from one another. The right and left elevons deflect independently

because of approach flap settings, while the right and left clamshells deflect independently so only one is open at a time for directional stability. While the controller considered only the symmetric portion of the elevon deflection away from trim and then added in the trim condition for each elevon, it was necessary for each individual surface to be considered in the calculation of the coefficients and gradients.

First, consider the dual elevon problem. Consider the expression for a generic aerodynamic coefficient which is a function of only the elevon deflections:

$$C_i = C_i(\delta_{eR}) + C_i(\delta_{eL}) \quad (3.6)$$

where δ_{eR} and δ_{eL} are the right and left elevon deflections, respectively. Equation 3.6 can be approximated with a Taylor series expansion about the elevon deflections $\bar{\delta}_{eR}$ and $\bar{\delta}_{eL}$ and simplified as follows:

$$C_i \approx C_i(\bar{\delta}_{eR}, \bar{\delta}_{eL}) + C_{i,\delta_{eR}}(\bar{\delta}_{eR})\Delta\delta_{eR} + C_{i,\delta_{eL}}(\bar{\delta}_{eL})\Delta\delta_{eL}$$

$$\Delta\delta_{eR} = \delta_{eR} - \bar{\delta}_{eR}$$

$$\Delta\delta_{eL} = \delta_{eL} - \bar{\delta}_{eL}$$

$$\Rightarrow C_i \approx C_i(\bar{\delta}_{eR}, \bar{\delta}_{eL}) - [C_{i,\delta_{eR}}(\bar{\delta}_{eR})\bar{\delta}_{eR} + C_{i,\delta_{eL}}(\bar{\delta}_{eL})\bar{\delta}_{eL}] + [C_{i,\delta_{eR}}(\bar{\delta}_{eR})\delta_{eR} + C_{i,\delta_{eL}}(\bar{\delta}_{eL})\delta_{eL}] \quad (3.7)$$

The $C_i(\bar{\delta}_{eR}, \bar{\delta}_{eL})$ term is merely the value of the coefficient at these elevon deflections. The next set of terms appearing in Equation 3.7, $C_{i,\delta_{eR}}(\bar{\delta}_{eR})\bar{\delta}_{eR}$ and $C_{i,\delta_{eL}}(\bar{\delta}_{eL})\bar{\delta}_{eL}$, are merely products of the previously computed control gradients and the corresponding elevon deflection. Therefore, they become constants as far as the controller is concerned. These are the same as the backward difference terms of the form \bar{C}_m appearing in Equations 2.20–2.25. It is desired that Equation 3.7 be written as a

function of a single elevon deflection, δ_e . First, making the following substitutions:

$$\begin{aligned}
C_i(\bar{\delta}_e) &= C_i(\bar{\delta}_{eR}, \bar{\delta}_{eL}) \\
C_{i,\delta_e}(\bar{\delta}_e)\bar{\delta}_e &= C_{i,\delta_{eR}}(\bar{\delta}_{eR})\bar{\delta}_{eR} + C_{i,\delta_{eL}}(\bar{\delta}_{eL})\bar{\delta}_{eL} \\
C_{i,\delta_e}(\bar{\delta}_e)\delta_e &= C_{i,\delta_{eR}}(\bar{\delta}_{eR})\delta_{eR} + C_{i,\delta_{eL}}(\bar{\delta}_{eL})\delta_{eL}
\end{aligned} \tag{3.8}$$

Equation 3.7 can be written as:

$$\Rightarrow C_i \approx C_i(\bar{\delta}_e) - C_{i,\delta_e}(\bar{\delta}_e)\bar{\delta}_e + C_{i,\delta_e}(\bar{\delta}_e)\delta_e \tag{3.9}$$

In order to use Equation 3.9, Equation 3.8 must first be written as a function of a single elevon deflection. Using the following substitutions for δ_{eR} and δ_{eL} and simplifying, Equation 3.8 can be written as:

$$\begin{aligned}
\delta_{eR} &= \delta_e + 25^\circ \\
\delta_{eL} &= -\delta_e + 25^\circ \\
\Rightarrow C_{i,\delta_e}(\bar{\delta}_e)\delta_e &= C_{i,\delta_{eR}}(\bar{\delta}_{eR})(\delta_e + 25^\circ) + C_{i,\delta_{eL}}(\bar{\delta}_{eL})(\delta_e + 25^\circ) \\
&= [C_{i,\delta_{eR}}(\bar{\delta}_{eR}) - C_{i,\delta_{eL}}(\bar{\delta}_{eL})]\delta_e + [C_{i,\delta_{eR}}(\bar{\delta}_{eR}) + C_{i,\delta_{eL}}(\bar{\delta}_{eL})]25^\circ
\end{aligned} \tag{3.10}$$

where 25° is the elevon trim condition for use as flaps in this research. Equations 3.9 and 3.10 show how the coefficients and gradients resulting from the multiple elevons are considered by the controller as a single elevon deflection.

Now addressing the multiple clamshell problem, a positive clamshell deflection command corresponded to left clamshell and negative to right clamshell. Since no speedbreak function was included in this research, the only application to consider was directional stability. Acting in place of a rudder, only one clamshell was open at a time, except for the small amount of time necessary for one clamshell to close while

the other was opening if the change in the clamshell command was large. Therefore, if the clamshell deflection was positive, the left clamshell gradient was used, and vice-versa for a negative clamshell deflection.

3.4.13 State Estimators. Once simulation testing was begun, it was necessary that some form of noise rejection be included. The sensor noise added was causing the controller to yield control surface commands with high frequency oscillations due to the high frequency noise in the measurements. A simple linearized state estimator was used for noise rejection. Consider the equations of motion with state vector and its estimate, x and \hat{x} , respectively:

$$\dot{x} = f(x, u) \quad (3.11)$$

$$\dot{\hat{x}} = f(\hat{x}, u) + L(x - \hat{x}) \quad (3.12)$$

where L is the estimator gain matrix. Consider the Taylor series expansion of Equation 3.12 about the actual state vector and the input vector:

$$\dot{\hat{x}} \Big|_{(x,u)} \approx f(x, u) + \frac{\partial f}{\partial x}(\hat{x} - x) + \frac{\partial f}{\partial u} \Delta u + L(x - \hat{x}) \quad (3.13)$$

Realizing that Δu is zero because the estimate is for the current control input, using the following substitutions, and subtracting Equation 3.13 from Equation 3.11 yields the differential equation for the estimator error vector, e :

$$e = x - \hat{x}, \quad \dot{e} = \dot{x} - \dot{\hat{x}} \quad (3.14)$$

$$\begin{aligned} \Rightarrow \dot{e} &= -\frac{\partial f}{\partial x}(\hat{x} - x) - L(x - \hat{x}) \\ &= \frac{\partial f}{\partial x}e - Le \\ &= \left[\frac{\partial f}{\partial x} - L \right] e \end{aligned} \quad (3.15)$$

Table 3.10: Summary of Trim Conditions for Estimator Design.

Parameter	Value
u	192 fps
v	0 fps
w	27 fps
p, q, r	0 deg/sec
ϕ	0 deg
θ	5 deg
ψ	036 deg
V_B	10 kts
ψ_B	045 deg

From linear systems theory, this error in Equation 3.15 is stable if $[\frac{\partial f}{\partial x} - L]$ is negative definite, i.e. if it has negative eigenvalues. In fact, these eigenvalues can be picked via pole placement techniques to yield sufficiently fast filter dynamics.

The form of the estimator equation which was used in the simulation is Equation 3.12, which shows that only the equations of motion, state measurements, state estimates, control surface deflections, and estimator gain matrix are needed to run the estimator. Since x , \hat{x} , and u are all determined during the simulation, the nonlinear equations of motion and L are the only items which must be specified before hand. Since, estimators were needed for both the aircraft and the ship, two sets of equations of motion and two estimator gain matrices were needed. The equations of motion used for the aircraft are the same ones from Equations 2.7–2.10. The equations of motion for the ship were shown in Equation 2.17. The estimator gain matrices were solved for in MATLAB[®]. The *place.m* command was used to place the eigenvalues for $[\frac{\partial f}{\partial x} - L]$ as previously discussed. The eigenvalues were placed in the vicinity of -20 for the aircraft estimator and -10 for the ship estimator.

In order to use the *place.m* command to pick the estimator gain matrices, both the aircraft and ship had to be linearized about their trim conditions for this landing scenario. The trim conditions which were used in the linearization process are

summarized in Table 3.10. The resulting estimator gain matrices were:

$$L_{AC} \approx \begin{bmatrix} 18.9 & 0 & -0.2 & 0 & 0 & 0 & 0 & -27 & 0 & 0 & -32.1 & 0 \\ 0 & 20.0 & 0 & 0 & 0 & 0 & 27 & 0 & -192 & 32.1 & 0 & 0 \\ -6.1 & 0 & 19.2 & 0 & 0 & 0 & 0 & 192 & 0 & 0 & -2.8 & 0 \\ 0.6 & 0.8 & 0.1 & 20.0 & 0 & 0 & 0 & 0 & 0 & -21.8 & 6.0 & 156.6 \\ 0.8 & -0.6 & 0.1 & 0 & 20.0 & 0 & 0 & 0 & 0 & 15.9 & 8.2 & -113.8 \\ 0.1 & 0 & -1.0 & 0 & 0 & 20.1 & 0 & 0 & 0 & 0 & 193.6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 20.1 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0 & 0.0 & 0 & 0 & 0 & 0 & 20.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0.1 & 20.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 20.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 20.1 \end{bmatrix} \quad (3.16)$$

$$L_B \approx \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12.02 & 10.01 & 0 & 0 & 0 & 0 & 7.55 & 0 & 0 \\ -12.02 & 0 & 10.02 & 0 & 0 & 0 & -7.55 & 0 & 0 \\ 0 & 0 & 0 & 10.03 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10.04 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10.05 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10.06 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10.07 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10.08 \end{bmatrix} \quad (3.17)$$

3.4.14 Inner-Loop Controller. Figure 3.13 illustrates the layout of the inner-loop portion of the controller. One input to the inner-loop controller was the control variable command vector resulting from the outer-loop controller and transitional dynamics. The rest of the inputs can be broken into two groups. The first group consisted of inputs that were only used by the inversion process. These inputs were

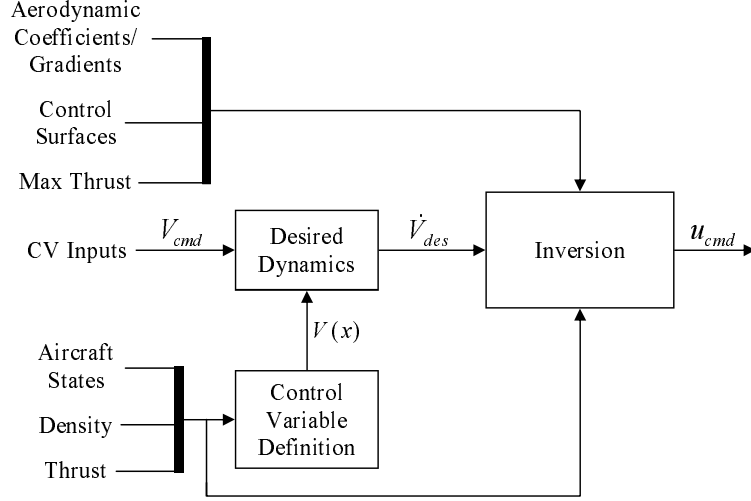


Figure 3.13: Layout of Inner-Loop Controller.

the control surface deflections, the aerodynamic coefficients and gradients, and the maximum thrust possible at the current flight condition. The maximum thrust was provided by the engine model providing thrust to the equations of motion running the simulator. It was assumed that this maximum thrust value could be estimated by an onboard computer with look-up tables scheduled on altitude and Mach number. The remaining inputs were used by both the inversion process and the control variable definition process. These were the aircraft states, the air density at the current altitude, and the current thrust output by the engine. Again, the thrust was provided by the main engine model and it was assumed that this could be estimated by an onboard computer. The outputs of this block were the control surface and throttle commands passed on to the aircraft.

The control variable definition portion of Figure 3.13 combined the aircraft states, air density, and thrust into the inner-loop control variables as defined in Section 2.7.2. In addition to these inputs, the control variables also required the aircraft

weight and wing planform area. For convenience, they are shown again here:

$$\begin{bmatrix} LCV \\ MCV \\ NCV \\ XCV \end{bmatrix} = \begin{bmatrix} p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ q \cos \phi - r \sin \phi \\ -p \sin \alpha + r \cos \alpha - \frac{g}{V_{co}} \beta - \frac{g}{\sqrt{u^2 + v^2 + w^2}} \cos \theta \sin \phi \\ T \end{bmatrix} \quad (3.18)$$

The desired dynamics portion of Figure 3.13 computed the control variable error by subtracting the actual values of the control variables calculated in the control variable definition from the commanded values of the control variables resulting from the outer-loop controller and the transitional dynamics. This error was then used to compute the desired dynamics vector, \dot{V}_{des} . As previously discussed, the desired dynamics for all four of the inner-loop control variables took the form of a PID controller with a limited integral. This control formulation can be written as:

$$\begin{aligned} e_{cv} &= V_{i \text{ cmd}} - V_i(x) \\ P_c &= e_{cv} \\ I_c &= \text{Limited Integral (see Figure 3.14)} \\ D_c &= \frac{de_{cv}}{dt} \\ \dot{V}_{i \text{ des}} &= k_P P_c + k_I I_c + k_D D_c \end{aligned} \quad (3.19)$$

Figure 3.14 illustrates the effects of the limited integral. When the integral of the input signal to the limited integral grows beyond the upper bound, the output of the limited integral is equal to the upper bound. When the input signal crosses zero and becomes negative, the limited integral is able to bleed off its accumulated value much more quickly than the normal integral.

To see how this benefits control theory, consider the input signal as an error between a command and an output. When this error signal crosses zero and becomes

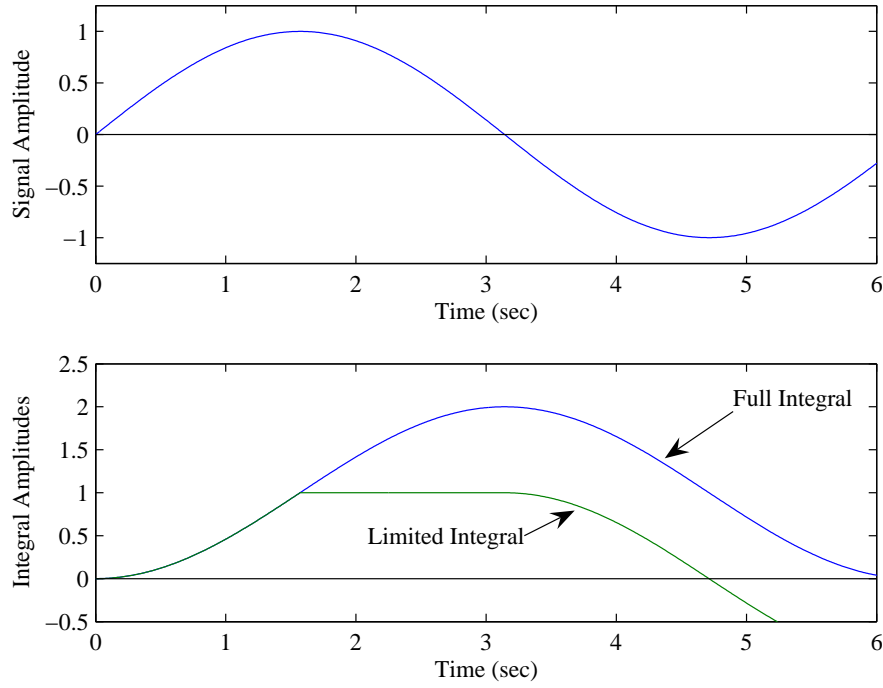


Figure 3.14: Illustration of Limited Integral Effects.

negative, both the normal and limited integrals are commanding a control input which causes overshoot in the error. However, the limited integral bleeds off the accumulation more quickly, decreasing the error overshoot. Also, because the limited integral still maintains some room for changing its value, unlike a pure proportional gain, it can still eliminate steady-state error as long as the value necessary for zero steady-state error is included within the limits. Thus, the limited integral effectively eliminates problems associated with integrator wind-up and is used only to make small corrections to eliminate steady-state error.

Table 3.11 summarizes the gains and integral limits used for each of the PID loops. Note that no integral limits were specified for the NCV loop, as the integral was not utilized.

The inversion portion of Figure 3.13 computed the necessary control surface and throttle commands to achieve \dot{V}_{des} . The inversion process was addressed in Sec-

Table 3.11: Summary of Inner-Loop PID Gains and Integral Limits.

Loop	k_P	k_I	k_D	Integral Limits
LCV	5.5	1	0.25	± 0.1
MCV	5	1	0.25	± 0.05
NCV	5	0	0	N/A
XCV	4	1	0	± 1

tions 2.3.4 and 2.7.4. There are only two portions of the process which were not addressed in these sections. The first was the conversion of the single elevon and clamshell deflection commands returned from the controller into dual elevon and clamshell commands to send to the actuators. The second was the conversion of pitch flap command from the controller into the pitch flap command sent to the actuator, resulting from the pitch-stabilizing proportional feedback loop. These control surface command conversions are summarized as:

$$\begin{aligned}
 \delta_{eR} &= \delta_e + 25^\circ \\
 \delta_{eL} &= -\delta_e + 25^\circ \\
 \delta_{cR} &= \begin{cases} 0 & \forall \delta_c \geq 0 \\ -\delta_c & \forall \delta_c < 0 \end{cases} \\
 \delta_{cL} &= \begin{cases} \delta_c & \forall \delta_c \geq 0 \\ 0 & \forall \delta_c < 0 \end{cases} \\
 \delta_f &= -10(\delta_f^* - q)
 \end{aligned} \tag{3.20}$$

where 25° is the elevon flap trim and -10 is the gain for the pitch-stabilizing loop.

3.4.15 Outer-Loop Controller. Figure 3.15 illustrates the layout of the outer-loop portion of the controller. Note that the control variable command vector consisted of only zeros, as the control variables were already structured as errors. As with the inner-loop, the inputs to the outer-loop controller can be broken into two groups. The first group consisted of inputs that were only used by the inversion process. These inputs were the aerodynamic coefficients and gradients and the estimated

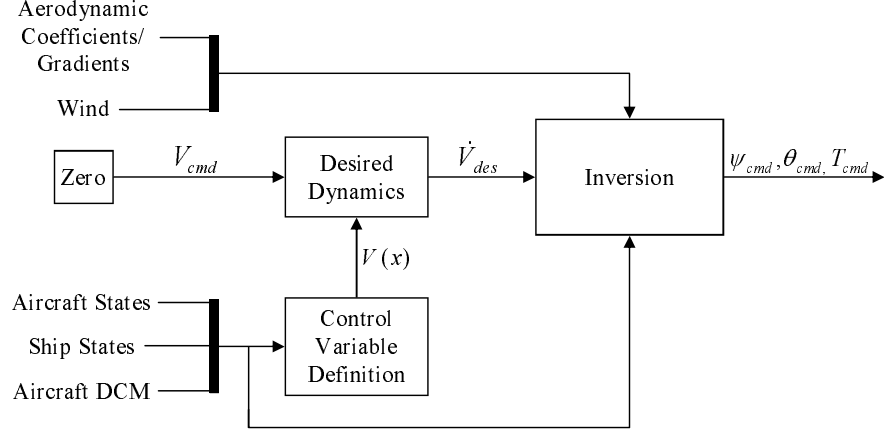


Figure 3.15: Layout Outer-Loop Controller.

wind. It was assumed that the ship could provide the aircraft with a measure of the prevailing winds. Also, since the burble was modeled based on wind tunnel and operational carrier data, it was assumed that an onboard computer could estimate the steady portion of the burble in the same manner that it was determined for this research. Thus, the horizontal and burble wind components were combined in order to give the aircraft an estimate of the winds. Since this estimate did not include the turbulence, the simulation is still running with a higher order of fidelity than the controller. The remaining inputs were used by both the inversion process and the control variable definition process. These were the aircraft states, ship states, and aircraft DCM. The outputs of this block were the desired values for ψ , θ , and T . Each of the inner blocks are now addressed.

The control variable definition portion of Figure 3.15 combined the aircraft states, ship states, and aircraft DCM into the outer-loop control variables as defined in Section 2.8.2:

$$\begin{bmatrix} HCV \\ PCV \\ TCV \end{bmatrix} = \begin{bmatrix} \psi_{BF} - \tan^{-1} \left(\frac{x_{Be} - x_e}{y_{Be} - y_e} \right) - \tan^{-1} \left(\frac{V_B \sin 9^\circ}{V_{HE}} \right) \\ \tan^{-1} \left(\frac{-(z_{Be} - z_e)}{\sqrt{(x_{Be} - x_e)^2 + (y_{Be} - y_e)^2}} \right) - \tan^{-1} \left(\frac{V_{ze}}{V_{HE}} \right) \\ \tan^{-1} \left(\frac{w}{u} \right) - 8^\circ \end{bmatrix} \quad (3.21)$$

The desired dynamics portion of Figure 3.15 was not as simple as the PID loops which were found in the inner-loop. While the *TCV* desired dynamics did utilize a PID with limited integral, the *HCV* and *PCV* desired dynamics were treated simultaneously with a simple LQR controller. First, the *TCV* desired dynamics utilized what was planned as a PID with limited integral control structure similar to the PID loops in the inner-loop. However, the integral and derivative portions were turned off in the final design. This left the *TCV* desired dynamics as only a proportional gain, $k_P = 1$.

The other two outer-loop control variables, *HCV* and *PCV*, greatly affected the landing performance of this controller. These two control variables represented the glideslope and horizontal glidepath error of the aircraft. Even very small values in these variables could have caused large miss distances at touchdown. Therefore, it was important that these control variables be kept as small as possible throughout approach. Also, it was very likely that there would be modeling errors in the outer-loop equations of motion which were considered and the actual dynamics of the system. This was because the inner-loop controller dynamics were not taken into account in the outer-loop model, as they would have been very difficult to model. Therefore, an LQR controller was decided upon for the desired dynamics, as the guaranteed stability margins would provide the robustness necessary to compensate for modeling errors.

For the purpose of selecting LQR gains for the desired dynamics, the system was modeled as having two states, *HCV* and *PCV*. Since both the inner- and outer-loop inversions should each yield dynamics approximately equal to free integrators, $\frac{1}{s}$ in the Laplace domain, they should combine in series to yield dynamics approximately equal to $\frac{1}{s^2}$. Therefore, the transfer functions from the input $\dot{V}_{i\ des}$ to the output $V_i(x)$ should actually approximate $\frac{1}{s^2}$ for both control variables. Replacing $\dot{V}_{i\ des}$ with u_i , this can be written as:

$$\ddot{V}_i = u_i \quad (3.22)$$

Rewriting this relationship in state-space form yields:

$$\underbrace{\begin{bmatrix} \dot{V}_1 \\ \ddot{V}_1 \\ \dot{V}_2 \\ \ddot{V}_2 \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} V_1 \\ \dot{V}_1 \\ V_2 \\ \dot{V}_2 \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_B \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}_u \quad (3.23)$$

This was the form utilized for the original controller design. An improved version of the controller was later formulated with integral control in the outer-loop desired dynamics. The differential equation for the integral term can be written as:

$$\dot{V}_{f,i} = V_i \quad (3.24)$$

Adding this integral term to the state-space formulation in Equation 3.23 yields:

$$\underbrace{\begin{bmatrix} \dot{V}_1 \\ \ddot{V}_1 \\ \dot{V}_{f,1} \\ \dot{V}_2 \\ \ddot{V}_2 \\ \dot{V}_{f,2} \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} V_1 \\ \dot{V}_1 \\ V_{f,1} \\ V_2 \\ \dot{V}_2 \\ V_{f,2} \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}}_B \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}_u \quad (3.25)$$

The cost function minimized was the standard infinite horizon LQR cost function [7, page 196]:

$$J = \frac{1}{2} \int_0^\infty (x^T \mathbf{Q} x + u^T \mathbf{R} u) dt \quad (3.26)$$

where \mathbf{Q} and \mathbf{R} are the state and control weighting matrices, respectively. Two sets of LQR gains were chosen for initial controller design. The first was a low-gain set, used when the aircraft was greater than 2 nm from touchdown. The second was a high-gain set, used when the ship was within 2 nm of touchdown. The transition

from the low-gain set to the high-gain set was a simple switch at 2 nm from the ship. Also, a set of gains for the improved controller design utilizing integral control was computed. Only one gain set was used for the integral controller. The values of \mathbf{Q} and \mathbf{R} used for this research were:

$$\begin{aligned}
\mathbf{Q}_1 &= \begin{bmatrix} 25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 400 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & (low-gain \ case) \\
\mathbf{Q}_2 &= \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 500 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & (high-gain \ case) \\
\mathbf{Q}_3 &= \begin{bmatrix} 100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 500 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} & (integral \ case) \\
\mathbf{R}_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & (low-gain \ case) \\
\mathbf{R}_2 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & (high-gain \ case) \\
\mathbf{R}_3 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & (integral \ case)
\end{aligned} \tag{3.27}$$

The LQR feedback gain which minimizes the cost function in Equation 3.26 is obtained from:

$$K = \mathbf{R}^{-1} B^T \mathbf{P} \tag{3.28}$$

where K is the LQR feedback gain and \mathbf{P} is the solution to the steady-state Riccati Equation [7, page 206]:

$$\mathbf{P}A + A^T\mathbf{P} - \mathbf{P}B\mathbf{R}^{-1}B^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \quad (3.29)$$

To solve the steady-state Riccati Equation, the MATLAB[®] *lqr.m* function was used. The inputs to *lqr.m* were A , B , \mathbf{Q} , and \mathbf{R} . The resulting LQR gain matrices were:

$$K_1 \approx \begin{bmatrix} 5 & 3.32 & 0 & 0 \\ 0 & 0 & 20 & 6.40 \end{bmatrix} \quad (low\text{-}gain\text{ case}) \quad (3.30)$$

$$K_2 \approx \begin{bmatrix} 10 & 4.58 & 0 & 0 \\ 0 & 0 & 22.36 & 6.76 \end{bmatrix} \quad (high\text{-}gain\text{ case}) \quad (3.31)$$

$$K_3 \approx \begin{bmatrix} 10.14 & 4.61 & 0.32 & 0 & 0 & 0 \\ 0 & 0 & 0 & 22.66 & 6.81 & 1 \end{bmatrix} \quad (integral\text{ case}) \quad (3.32)$$

However, recall that the equations of motion for the outer-loop were transformed from the time domain to the distance domain. It was quickly realized due to large overshoots that these gains were optimized for the time domain and were much too high for the distance domain problem formulation used in this research. Since the distances for most of the flight path are on the order of 10^3 and 10^4 , trial simulations were run with scaling the LQR gains by 10^{-3} and 10^{-4} . The scaling factor of 10^{-3} did not work when in close to the ship, as the gains were too high and the errors became unstable just before touchdown. However, the scaling factor of 10^{-4} yielded results similar to what would normally be expected from a system with LQR compensation, so it was used in the control design.

Combining these results with the desired dynamics for the pitch loop, the resulting desired dynamics for the *HCV* and *PCV* loops in the initial controller design

are:

$$\begin{aligned} \begin{bmatrix} H\dot{C}V_{des} \\ P\dot{C}V_{des} \end{bmatrix} &= K_i \begin{bmatrix} HCV \\ H\dot{C}V \\ PCV \\ P\dot{C}V \end{bmatrix} \\ T\dot{C}V_{des} &= -TCV \end{aligned} \quad (3.33)$$

Similarly, desired dynamics for the HCV and PCV loops in the improved controller are:

$$\begin{aligned} \begin{bmatrix} H\dot{C}V_{des} \\ P\dot{C}V_{des} \end{bmatrix} &= K_3 \begin{bmatrix} HCV \\ H\dot{C}V \\ HCV_f \\ PCV \\ P\dot{C}V \\ PCV_f \end{bmatrix} \\ T\dot{C}V_{des} &= -TCV \end{aligned} \quad (3.34)$$

The inversion portion of Figure 3.15 followed the process outlined in Section 2.8.5. The only difference lay in constraints which were placed on the values of ψ_{cmd} and T_{cmd} which resulted from the inversion. When the aircraft was within 530 feet of touchdown, approximately twice the distance from the ramp to the aimpoint, ψ_{cmd} was restricted to $\psi_{BF} \pm 3^\circ$. This was done because a real aircraft must stay in the landing area during rollout, and any angle difference larger than 3° would put the aircraft too close to the edge of the landing area by the time it stopped. Also when within this range, nonlinear thrust commands were imposed. If the glideslope error was greater than 0.085° , then idle thrust was commanded. Conversely, if the glideslope error was less than -0.085° , then maximum thrust was commanded. These values were chosen by observing trial simulations and noting that approaches which

exceeded these bounds near touchdown had a low probability of achieving an arrested landing, commonly called a trap. This was only done in order to get the glideslope error back within these bounds, and normal thrust commands were resumed once this was achieved. However, the time it took to close the 530 feet to touchdown traveling with the hook engagement velocity of 110 kts was just under 3 seconds, so not much time was left to make large adjustments to the glideslope error.

3.4.16 Transitional Dynamics. Figure 3.16 illustrates the layout of the transitional dynamics portion of the controller. This block transformed the desired values for ψ , θ , and T resulting from the outer-loop controller into the desired values of $\dot{\phi}$, $\dot{\theta}$, and T required by the inner-loop controller. The thrust command was a direct feedthrough. For the pitch loop, the actual pitch angle was subtracted from the desired pitch angle resulting from the outer-loop. The resulting error was operated on by a set of desired dynamics, just a proportional gain in this case, resulting in the desired pitch rate required by the inner-loop controller. For the heading loop, the heading error was found and operated on by a proportional desired dynamics gain. This resulted in a desired turn rate. This was operated on by Equation 2.53, yielding a desired bank angle. From this desired bank angle, the bank angle error was calculated and operated on by another proportional desired dynamics gain. This finally resulted in the desired roll rate required by the inner-loop controller.

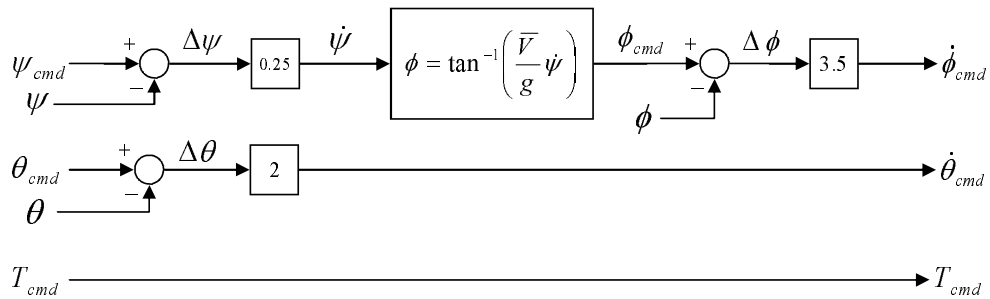


Figure 3.16: Layout of Transitional Dynamics.

IV. Simulation and Results

4.1 Monte Carlo Simulation

The performance of the controller was evaluated through a Monte Carlo simulation. The initial simulation consisted of six different simulation cases, Cases 1–6. After adding integral control to the outer-loop desired dynamics, five more simulation cases were conducted, Cases 7–11. The initial conditions used for the simulation runs are summarized in Tables 4.1–4.3. Table 4.1 contains the initial flight conditions, Table 4.2 contains the various starting positions, and Table 4.3 contains the environmental conditions for each individual case.

The Monte Carlo simulation was structured such that the aircraft would start from multiple locations behind the ship for each of the first six simulation cases. The initial horizontal radial from runway centerline, glideslope error, and distance from the ship were varied. The initial x and y coordinates of the aircraft were determined by the initial horizontal radial and initial range, and the initial altitude was determined from the initial glideslope error and initial range.

Table 4.2 summarizes the starting positions which were considered for this research. An initial positive glideslope error corresponded to starting above glideslope, and an initial positive horizontal radial corresponded to starting to the starboard side of the runway centerline. Note that the starting positions for Cases 1–6 became more restrictive when the aircraft started closer to the ship. Originally, trial simulations were conducted at all of the starting distances for all of the starting horizontal radials and glideslope errors used for the 5 nm starting set. Table 4.2 was developed after analyzing these trial simulations and discarding the starting positions for which the controller was not able to eliminate the glidepath error before the aircraft reached the carrier.

After running Cases 1–6, integral control was added to the outer-loop LQR desired dynamics per Section 3.4.15, and five more simulation cases were conducted, Cases 7–11. Since Cases 1–6 had already shown that the controller was able to capture the desired glidepath from many different starting positions, as will be shown

Table 4.1: Summary of Simulation Initial Conditions.

Variable	Value
u	130 KTAS
v, w	0 KTAS
p, q, r	0 rad/sec
ϕ	0 deg
θ	8 deg
ψ	045 deg
ψ_B	045 deg

Table 4.2: Summary of Starting Positions Considered.

Case	Range (nm)	Glideslope Error (deg)	Horizontal Radial (deg)
1-6	5	$\pm 2.5, 5$	-48, -38, -28, -8, 2, 12
	4	$\pm 2.5, 5$	-38, -28, -8, 2, 12
	3	± 2.5	-38, -28, -8, 2, 12
	2	± 2.5	-8, 2, 12
7-11	3	0	0

Table 4.3: Summary of Simulation Environmental Conditions.

Case	V_B (kts)	V_{wind} (kts)	ψ_{wind} (deg)	Turbulence	Burble	Sea State
1	10	10	027	None	Off	0
2	10	10	027	Light	Off	0
3	10	10	027	Light	Off	4
4	10	10	027	Light	On	4
5	5	20	034	Moderate	On	5
6	5	30	035	Heavy	On	6
7	10	10	027	None	On	0
8	10	10	027	Light	On	0
9	10	10	027	Light	On	4
10	10	10	027	Light	On	5
11	5	20	034	Moderate	On	5

in Sections 4.3–4.8, the focus was shifted to gathering many data points for a single starting position. Thus, the starting position for Cases 7–11 was set to 3 nm behind the ship, on glideslope, and lined up with the runway centerline.

As previously mentioned, Table 4.3 summarizes the environmental conditions considered for each simulation case. It should be noted that the progression of environmental conditions in Cases 7–11 was not structured to replicate the conditions for Cases 1–6. Instead, the aim was to start with the burble already included in Case 7 and gradually add turbulence, sea state intensity, and finally wind and turbulence intensity. However, this natural progression did result in the replication of the conditions for Cases 4 and 5 in Cases 9 and 11, respectively. Comparisons will be made accordingly in later sections.

For Cases 1–6, 10 simulation runs were conducted for each starting position, with a different set of noise seeds for each run. Table 4.2, shows that there were 49 starting positions, totaling 2,940 runs for Cases 1–6. For Cases 7–11, 500 simulation runs were conducted from the single starting point for each simulation case, with a different set of noise seeds for each run. This resulted in 2,500 simulation runs for Cases 7–11, bringing the total number of simulation runs to 5,440. The results presented for each case are statistical in nature, as individual analysis for each simulation run is impractical due to the large number of simulation runs. Therefore, a sample simulation run is first presented in order to provide an illustration of the overall response of the system. Cases 1-11 are then addressed on a case-by-case basis.

4.2 Sample Simulation Run

The conditions for the sample simulation run presented here are representative of the conditions for Simulation Case 1. The wind speed and ship speed were both 10 knots and their headings were such that the relative WOD was straight down the angled flight deck. Also, turbulence, burble, and sea state effects were not included. Sensor noise was the only disturbance included. Although this is the simplest case

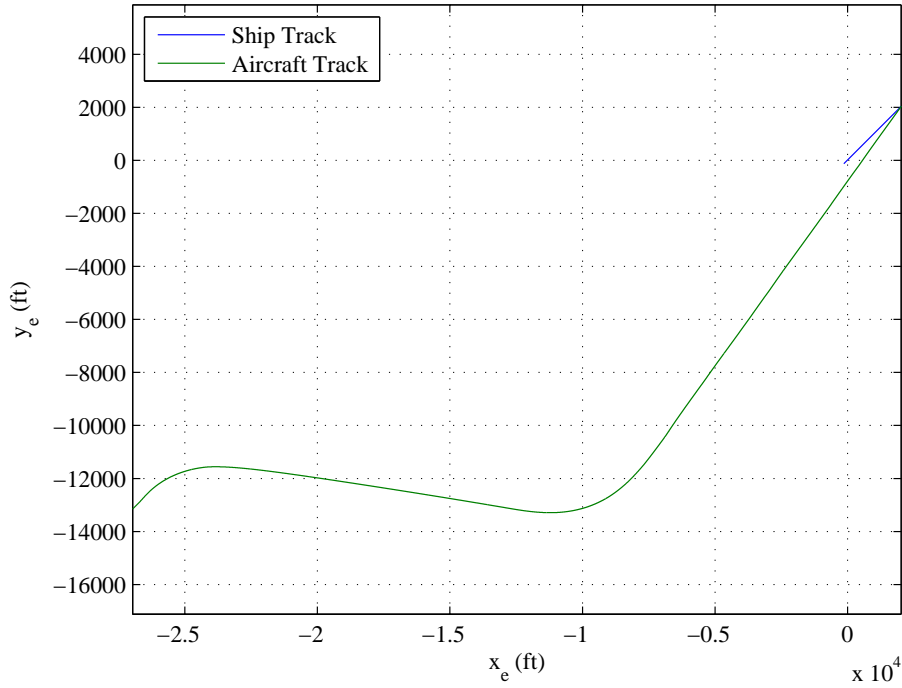


Figure 4.1: Sample Simulation Trajectory.

possible and does not include much realism, it provides a good illustration of the basic response of the controller.

Figure 4.1 depicts the top-down view of the trajectory of both the aircraft and the ship for the sample run. The center of mass of the ship started at the origin, and the aircraft started at a position 5 nautical miles away from the ship at a radial of 28° to the port side of the ship off of the runway centerline at a heading of 045° . When the simulation began, the aircraft immediately began to turn to intercept the glidepath. Notice that it did not fly perpendicular to the prescribed glidepath as it was approaching. This was a result of a limit in the outer-loop control law which did not allow it to command a heading of more than 60° off of the landing heading, ψ_{BF} . As it approached the prescribed glidepath, it began a gradual turn and came out approximately on glidepath at the landing heading with very little overshoot. It then maintained the prescribed glidepath all the way to touchdown.

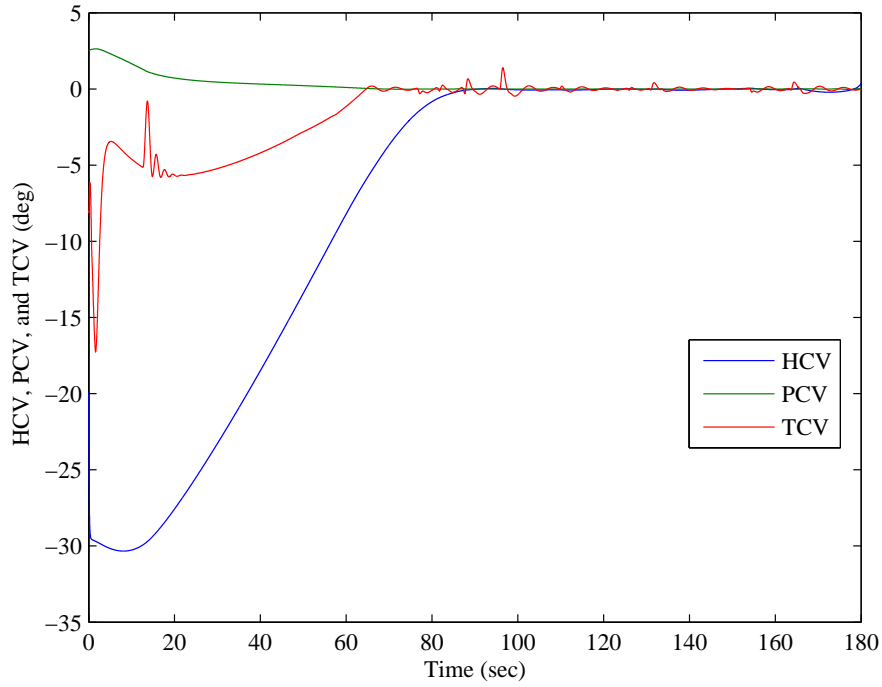


Figure 4.2: Sample Control Variable Response.

Figure 4.2 depicts the control variable response for the sample run. Recall that HCV is the horizontal glidepath error in degrees, PCV is the glideslope error in degrees, and TCV is the angle of attack error in degrees. The PCV plot shows that the aircraft started above glideslope, and the controller drove the glideslope error to zero in a reasonable amount of time with very little overshoot. Similarly, the HCV plot shows that the aircraft started to the left of the prescribed glidepath, and the controller drove the glidepath error to zero with little overshoot. Note that HCV actually becomes slightly more negative at the beginning of the simulation before approaching zero. While this may appear similar to nonminimum-phase behavior, it is actually due to the fact that the aircraft started off flying a course parallel to the ship, but off to its port side. Since this path would have caused an increase in horizontal glidepath error if unchanged, the horizontal glidepath error increased slightly until the aircraft was able to alter its heading enough to start decreasing it.

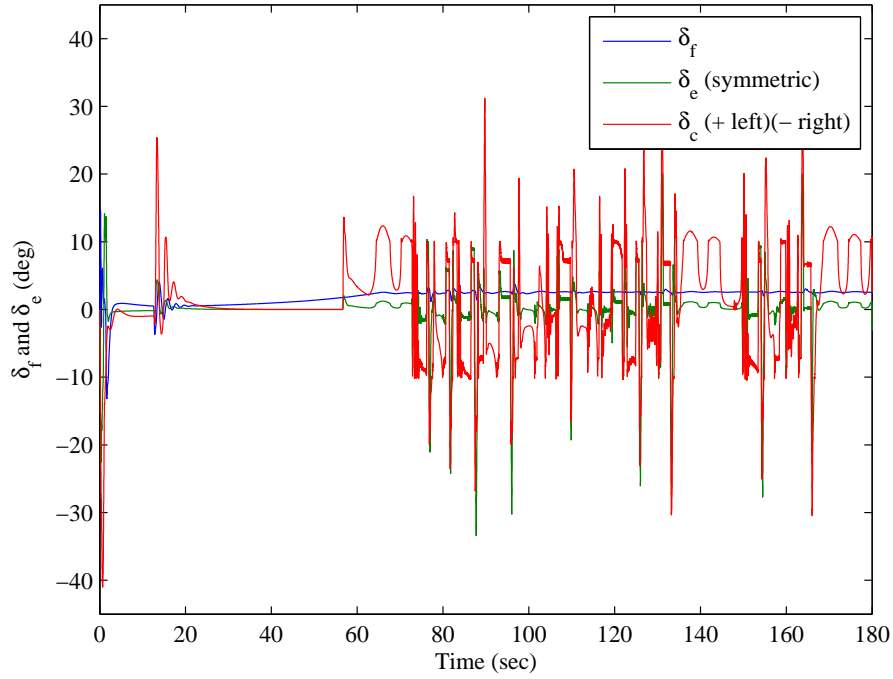


Figure 4.3: Sample Control Surface Response.

Finally, note that while the controller does drive TCV to approximately zero and keep it relatively close to zero for the majority of the approach, it was not a smooth response. Recall that this sample simulation run did not have turbulence included, so keeping a relatively constant angle of attack should not have been a difficult process. This was most likely due to modeling errors in the control law. Much difficulty was encountered while trying to achieve desirable throttle responses, and all of the simulation runs exhibited large variations in the throttle input. The throttle responses essentially appeared similar to bang-bang control schemes, despite many attempts to alleviate this by decreasing the gains. However, since the controller was able to land the aircraft successfully even with these throttle control problems, the decision was made to conduct the Monte Carlo simulation despite the throttle difficulties.

Figure 4.3 depicts the relevant control surface responses for this sample run. The two elevon deflections were simplified into the symmetric component about the

flap deflection, with δ_e considered positive with the right elevon trailing edge down. Also, since only one clamshell was open at a time, the two clamshells were simplified into δ_c considered positive with the left clamshell open. Both the pitch flap and elevon responses were fairly smooth, with little control authority used throughout the approach. However, the clamshells used a large amount of control power. Because this aircraft was a flying wing, its lack of a vertical tail equated to a lack of directional stability. The clamshells had to take over the responsibility of directional stability and damping.

4.3 *Simulation Case 1*

Simulation Case 1 considered the simplest case possible. Table 4.3 shows the environmental conditions which defined this case. Only light wind was considered in this case with both the wind turbulence and burble model turned off. Also, a calm sea state 0 was considered. Sensor noise was the only disturbance considered for this case, and was considered in all simulation cases for this research. This case provided the most docile set of conditions which a landing aircraft could ever experience.

Figure 4.4 depicts the results from the first simulation case. All 490 simulation runs resulted in a successful trap. In fact, every run resulted in catching the 3 wire. For this case, the mean miss distance was only 2.87 feet and the standard deviation from this was 1.13 feet. While these are the mean and standard deviation of the absolute value of the miss distance, the average approach was only 2.70 feet long of the aimpoint with a standard deviation of only 1.13 feet. Also, the average approach missed only 0.20 feet to the left of the aimpoint with a standard deviation of only 0.97 feet. With the $\pm 2\sigma$ interval equal to approximately a 95% confidence interval (CI), the 95% CI for this case is an area only 4.52 feet long and 3.88 feet wide. These statistical measures are summarized in Tables 4.4–4.6, along with the corresponding values for the other simulation cases.

These results show that the controller performs well when the system model is accurate. The results of the later cases will show that the performance of the

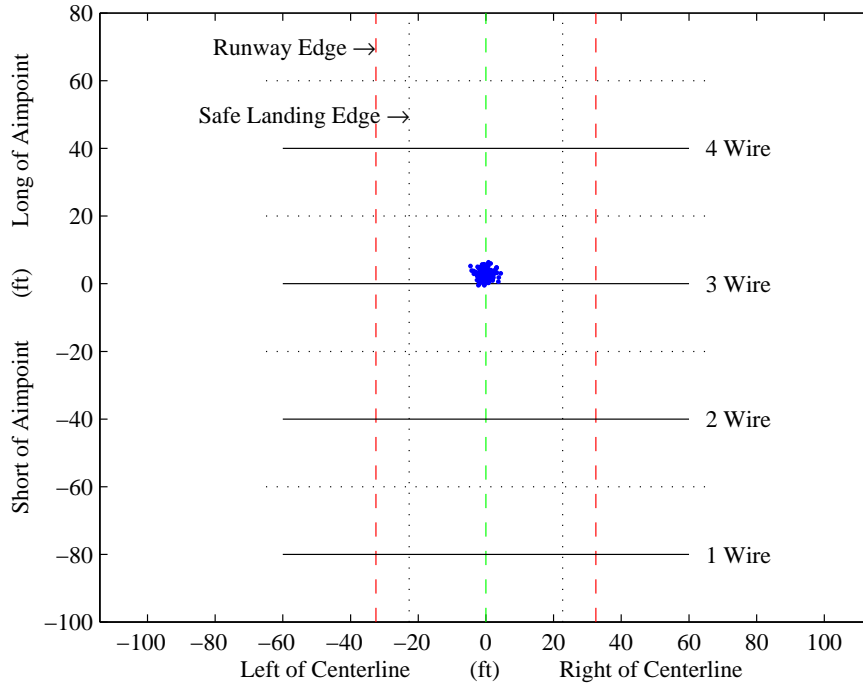


Figure 4.4: Case 1 Landing Dispersion.

controller degrades as more disturbances are added which cannot be predicted by the controller. This simulation case effectively showed that with proper modeling, the DI controller concept can work for automated aircraft carrier landing applications.

4.4 *Simulation Case 2*

Simulation Case 2 added a little more realism to the previous case by including light wind turbulence. The burble model was not included and a sea state of 0 was considered. See Table 4.3 for more information on the conditions.

Figure 4.5 depicts the results for Case 2. Unlike the first simulation case in which every approach resulted in successful traps, only 446 out of the 490 approaches in this case, 91.0%, resulted in traps. Of these, 13 (2.7%) caught the 2 wire, 131 (26.7%) caught the 3 wire, and 302 (61.6%) caught the 4 wire. The remaining 44 (9.0%) missed long, resulting in bolters. Figure 4.5 also shows that the addition of turbulence caused the aircraft to miss well long of the aimpoint the majority of the

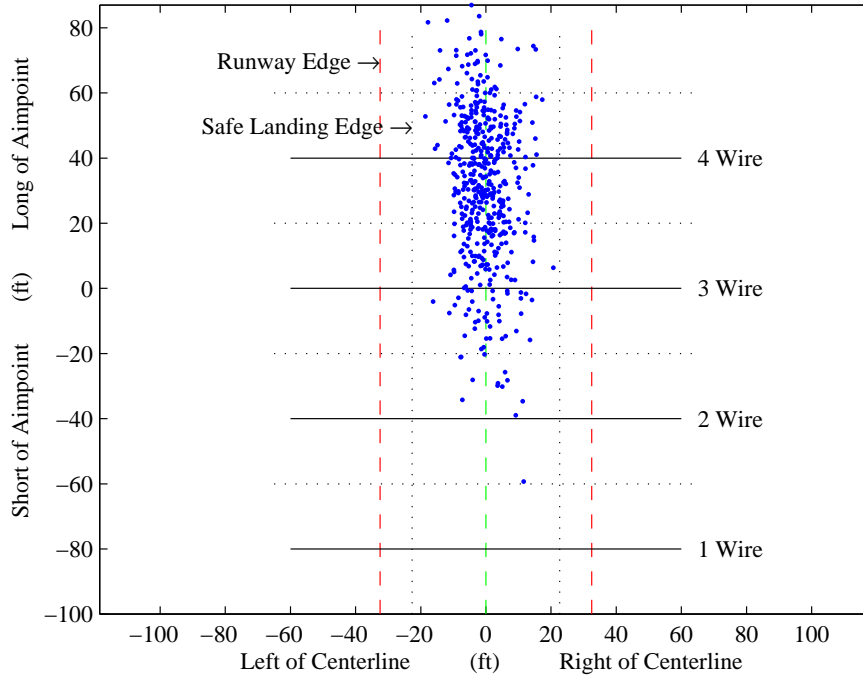


Figure 4.5: Case 2 Landing Dispersion.

time. Statistically, the average approach missed 31.14 feet long of the aimpoint with a standard deviation of 23.45 feet. While the longitudinal touchdown dispersion was degraded substantially, the lateral dispersion was only degraded slightly. The average approach missed left of the aimpoint by only 0.45 feet with a standard deviation of 6.27 feet. See Tables 4.4–4.6 for more details.

The addition of wind turbulence effects to the simulation constituted a large increase in the model uncertainty. While the lateral dispersion characteristics were degraded somewhat, the longitudinal dispersion characteristics were degraded significantly. However, despite the increase in touchdown dispersion from the first case, this data set shows potential for improvement. Adding a more robust form of disturbance rejection may tighten the dispersion, and adding integral control to the LQR in the outer-loop desired dynamics may bring the mean of the dispersion back toward the aimpoint. As previously mentioned, the addition of integral control is explored in Cases 7–11.

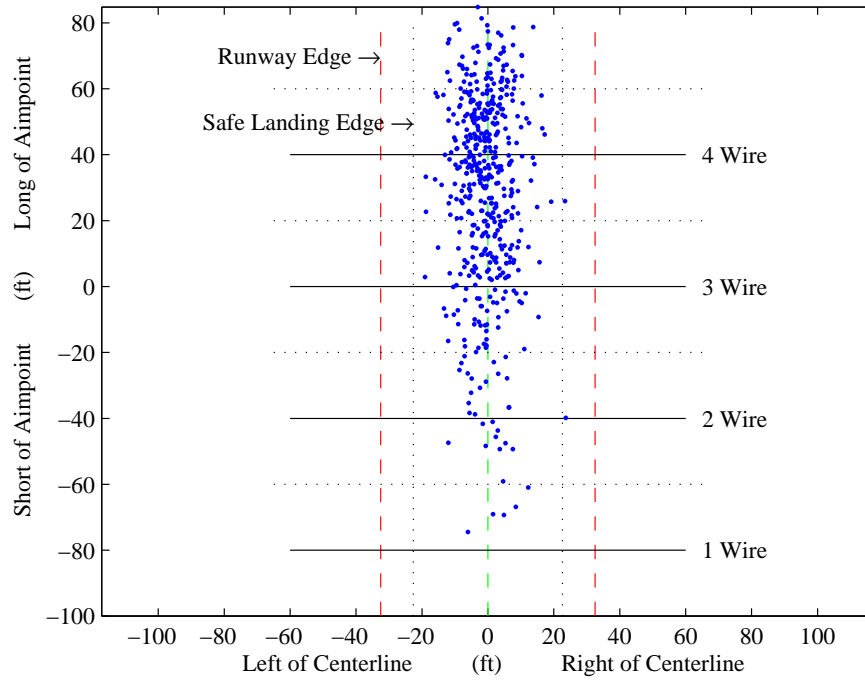


Figure 4.6: Case 3 Landing Dispersion.

4.5 *Simulation Case 3*

Simulation Case 3 added yet more realism to the previous cases by including significant sea state effects by using the sea state 4 conditions outlined in Table 3.7. The burble model was not included in this case. See Table 4.3 for more information on the conditions.

Figure 4.6 depicts the results for Case 3. Again, the touchdown dispersion and landing statistics were degraded from the previous case. This time, only 424 out of the 490 runs, 86.5%, resulted in traps. Of these, 5 (1.0%) caught the 1 wire, 27 (5.5%) caught the 2 wire, 130 (26.5%) caught the 3 wire, and 262 (53.5%) caught the 4 wire. Of the remaining, 64 (13.1%) missed long, resulting in bolters, and 2 (0.4%) missed left or right. As before in Case 2, the controller had a tendency to miss long of the aimpoint. Longitudinally, the average approach missed long of the aimpoint by 29.31 feet with a standard deviation of 30.27 feet. Laterally, the average approach missed 0.56 feet left of the aimpoint with a standard deviation of 6.50 feet. This case

resulted in a 29.1% increase in the standard deviation of the longitudinal dispersion and a 3.7% increase in the standard deviation of the lateral dispersion from Case 2. See Tables 4.4–4.6 for more details.

As with Case 2, these results still show potential for improvement. Better disturbance rejection and the addition of integral control may significantly improve dispersion characteristics. However, the increase in standard deviation with the addition of sea state effects shows that controller becomes increasingly less accurate as more disturbances are added.

4.6 Simulation Case 4

In Simulation Case 4, the burble model was included in the simulation. It still utilized the same light winds, light wind turbulence, and sea state 4 conditions as the previous simulation case. See Table 4.3 for more information on the conditions.

Figure 4.7 depicts the results for Case 4. Despite the addition of the burble, the results of this case were improved statistically. This time, 453 out of the 490 runs, 92.4%, resulted in traps. This actually increased by 5.5% from Case 3. As Figure 4.7 shows, this is due to the fact that the addition of the burble moved the longitudinal average of the landing dispersion back toward the aimpoint, resulting in less bolters. Of the total traps, 14 (2.9%) caught the 1 wire, 65 (13.3%) caught the 2 wire, 181 (36.9%) caught the 3 wire, and 193 (39.4%) caught the 4 wire. The remaining 37 (7.6%) missed long, resulting in bolters. Longitudinally, the average approach missed long of the aimpoint by 14.15 feet with a standard deviation of 33.09 feet. Laterally, the average approach missed 0.84 feet left of the aimpoint with a standard deviation of 6.29 feet. See Tables 4.4–4.6 for more details.

While the results are slightly better for this simulation case, this is only because the burble had a tendency to cause the average approach to land further aft than previously because of the downdraft associated with the burble. As the sea state

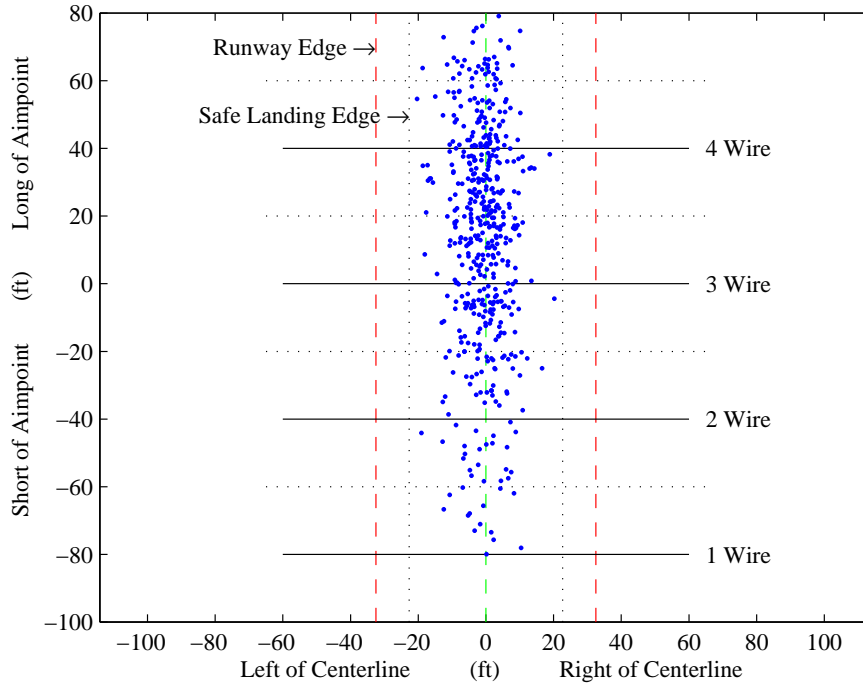


Figure 4.7: Case 4 Landing Dispersion.

is increased, the standard deviations associated with the touchdown dispersion will most likely increase.

4.7 Simulation Case 5

Simulation Case 5 increased the severity of all the disturbances. With the burble on, the wind and wind turbulence were increased to moderate levels of intensity, and the sea state was increased to 5. See Tables 3.7 and 4.3 for more information on the sea state and other conditions.

Figure 4.8 depicts the results for Case 5. The increase in wind turbulence significantly degraded the landing performance. Only 322 out of the 490 runs, 65.7%, resulted in traps. This decreased by 26.7% from Case 4. Of the total traps, 25 (5.1%) caught the 1 wire, 54 (11.0%) caught the 2 wire, 92 (18.8%) caught the 3 wire, and 151 (30.8%) caught the 4 wire. Of the 168 misses, 13 (2.7%) missed more than 20 feet short of the 1 wire, 105 (21.4%) missed long and boltered, and 50 (10.2%) missed to

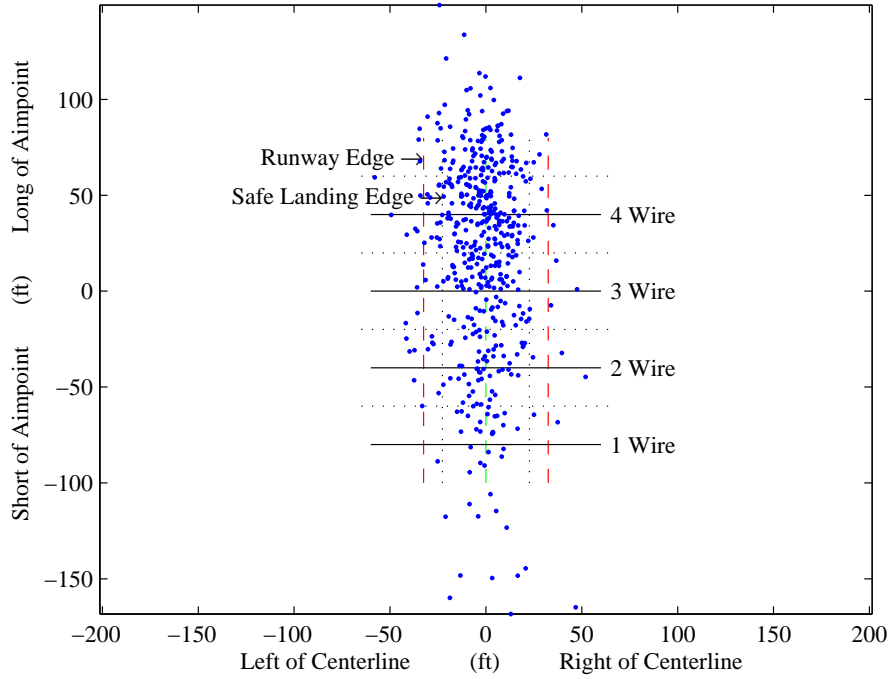


Figure 4.8: Case 5 Landing Dispersion.

the left or right of the safe landing area. Missing more than 20 feet short of the 1 wire was considered as a miss because LSOs would most likely give an aircraft the wave-off signal if it were that low below glideslope. Longitudinally, the average approach missed long of the aimpoint by 19.08 feet with a standard deviation of 52.16 feet. Laterally, the average approach missed 2.16 feet left of the aimpoint with a standard deviation of 15.21 feet. This case resulted in a 57.6% increase in the standard deviation of the longitudinal dispersion and a 141.8% increase in the standard deviation of the lateral dispersion from Case 4. See Tables 4.4–4.6 for more details.

4.8 *Simulation Case 6*

Simulation Case 6 increased disturbance intensities to near maximum levels. With the burble on, heavy wind and wind turbulence intensities, and sea state 6 conditions were included. This represents the worst-case scenario for landing on an aircraft carrier. In fact, conditions present during sea state 6 are usually out of the

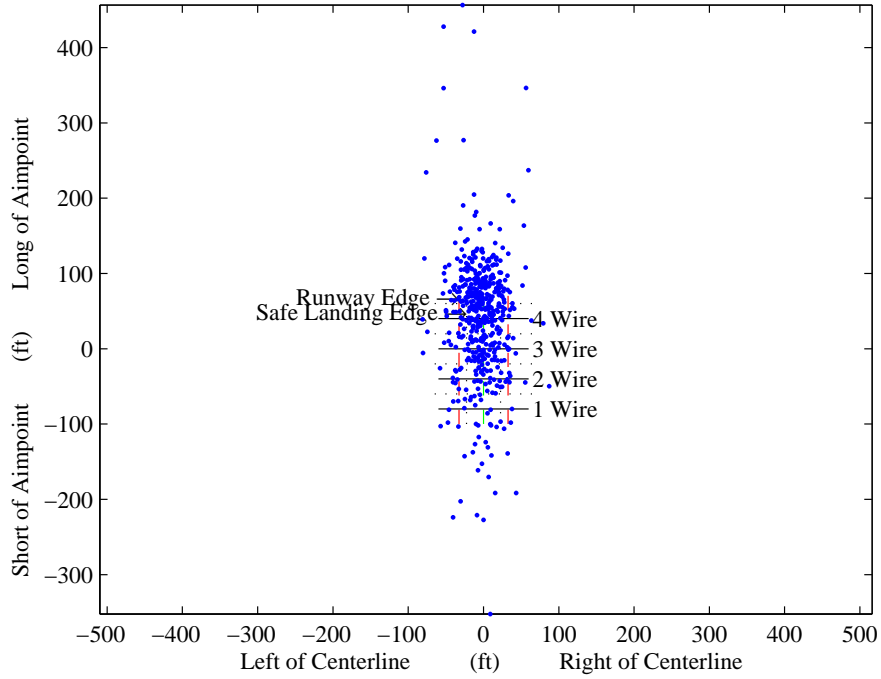


Figure 4.9: Case 6 Landing Dispersion.

range of unrestricted carrier flight operations [23, page 95]. See Tables 3.7 and 4.3 for more information on the sea state and other conditions.

Figure 4.9 depicts the results for Case 6. Again, the increase in wind turbulence significantly degraded the landing performance. Only 171 out of the 490 runs, 34.9%, resulted in traps. This decreased by 30.8% from Case 5, and 57.5% from Case 4. Of the total traps, 10 (2.0%) caught the 1 wire, 26 (5.3%) caught the 2 wire, 55 (11.2%) caught the 3 wire, and 80 (16.3%) caught the 4 wire. Of the 319 misses, 25 (5.1%) missed more than 20 feet short of the 1 wire, 219 (44.7%) missed long and boltered, and 75 (15.3%) missed to the left or right of the safe landing area. Longitudinally, the average approach missed long of the aimpoint by 41.79 feet with a standard deviation of 82.86 feet. Laterally, the average approach missed 3.42 feet left of the aimpoint with a standard deviation of 23.78 feet. This case resulted in a 58.9% increase in the standard deviation of the longitudinal dispersion and a 56.3% increase in the standard deviation of the lateral dispersion from Case 5. When referenced back to Case 4 with

light wind turbulence, this is a 150.4% increase longitudinally and a 278.1% increase laterally. See Tables 4.4–4.6 for more details.

Clearly, the touchdown dispersion for this case is unacceptable. Even though flight operations may not be conducted for safety reasons when heavy wind turbulence is present, the yielded boarding rate of 34.9% is much lower than desired even for poor conditions. The addition of integral control to the outer-loop LQR desired dynamics was simulated to see if it yielded any improvements to the results for Cases 1–6.

4.9 *Simulation Case 7*

Simulation Case 7 was the first case run with the improved controller. As previously addressed, the improved controller utilized integral control in the outer-loop LQR desired dynamics. This simulation case and the ones that follow all considered the same, single starting position 5 nm behind the ship, on glideslope and lined up with the runway centerline. The environmental conditions for this case were considered as a best-case scenario for the conditions an aircraft landing on a carrier could expect: light wind, no turbulence, and sea state 0. The burble was included, but the burble lacks realism without the wind turbulence.

Figure 4.10 depicts the results for Case 7. All of the 500 simulation runs resulted in traps via catching the 3 wire. Longitudinally, the average approach missed short of the aimpoint by 10.23 feet with a standard deviation of only 0.98 feet. Laterally, the average approach missed 0.38 feet right of the aimpoint with a standard deviation of only 0.49 feet. See Tables 4.4–4.6 for more details.

When the burble was first added in Case 4, a shift in the average longitudinal miss distance toward the aft end of the ship was noted. This was most likely due to the downdraft associated with the burble. Although this is the first simulation case run with this controller, the fact that all of the approaches missed short of the aimpoint in a very tight grouping, despite the integral control, suggests that this is a viable explanation for the tendency for missing short of the aimpoint in this case.

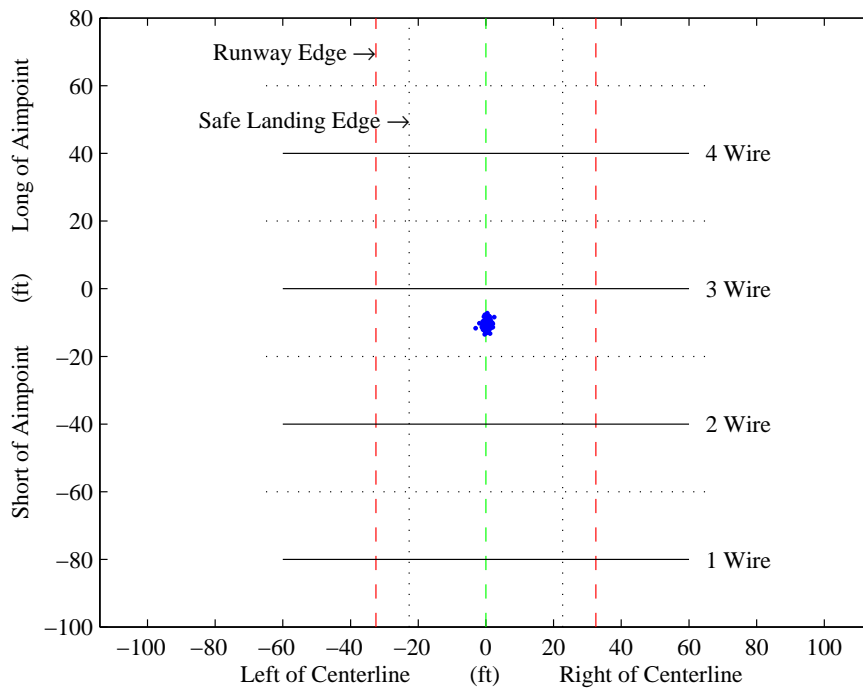


Figure 4.10: Case 7 Landing Dispersion.

4.10 *Simulation Case 8*

Simulation Case 8 added light wind turbulence to the conditions for the previous case. With the turbulence turned on in this case, the full effect of the burble was acting on the aircraft. Figure 4.11 depicts the results for Case 8. Out of the 500 simulation runs, all but two of them resulted in successful traps. Of the 498 traps, 5 (1.0%) caught the 1 wire, 113 (22.6%) caught the 2 wire, 276 (55.2%) caught the 3 wire, and 104 (20.8%) caught the 4 wire. Longitudinally, the average approach missed short of the aimpoint by only 2.04 feet with a standard deviation of 25.89 feet. Laterally, the average approach missed 0.55 feet left of the aimpoint with a standard deviation of 6.55 feet. See Tables 4.4–4.6 for more details.

The addition of wind turbulence in Case 8 had the effect of greatly increasing the standard deviations of both the longitudinal and lateral miss distances from those in Case 7, as well as shifting the mean longitudinal miss further forward in the landing area. A similar trend was noticed between the results of Cases 1 and 2 when turbulence

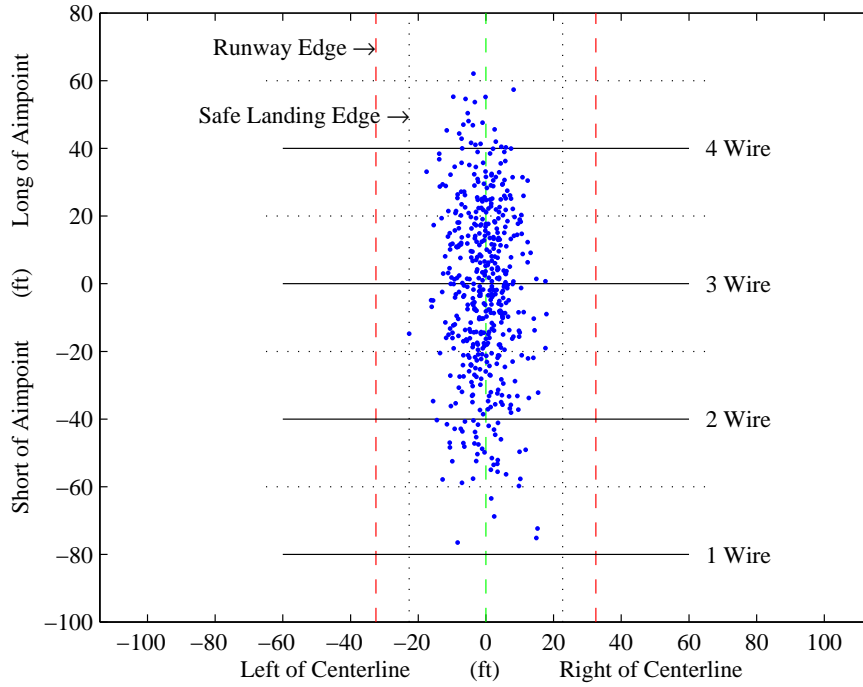


Figure 4.11: Case 8 Landing Dispersion.

was added in Case 2. These facts show that wind turbulence tends to cause this controller to have a more loosely spread touchdown dispersion and to land further forward on the flight deck.

4.11 *Simulation Case 9*

Simulation Case 9 had environmental conditions of light wind, light wind turbulence, and the burble turned on just as in Case 8, but also considered sea state 4. The simulation conditions for this case were identical to those for Case 4, so the results for Case 9 will be compared to the results of Case 4.

Figure 4.12 depicts the results for Case 9. All 500 of the simulation runs resulted in successful traps. The traps were distributed similarly to Case 8, as 13 (2.6%) caught the 1 wire, 123 (24.6%) caught the 2 wire, 250 (50.0%) caught the 3 wire, and 114 (22.8%) caught the 4 wire. Longitudinally, the average approach missed short of the aimpoint by only 2.15 feet with a standard deviation of 28.16 feet. Laterally, the

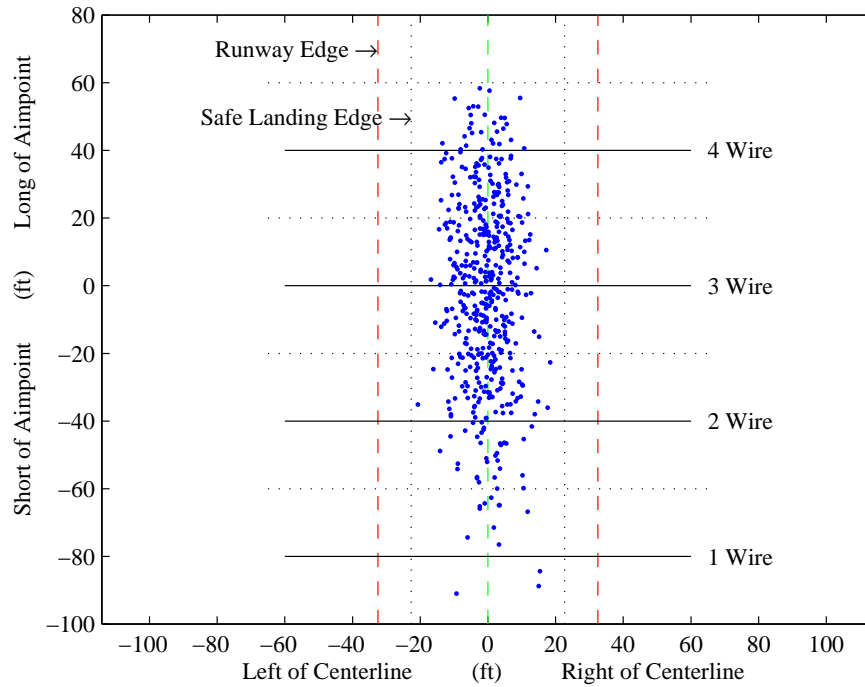


Figure 4.12: Case 9 Landing Dispersion.

average approach missed 0.51 feet left of the aimpoint with a standard deviation of 6.52 feet. See Tables 4.4–4.6 for more details.

The boarding rate for Case 9 was 100.0%, compared to 92.4% for Case 4. Also the dispersion characteristics were similar for Cases 9 and 4 except for the mean longitudinal miss distance. The average approach in Case 9 landed approximately 16 feet further aft than the average approach in Case 4. Recall that the Case 4 considered multiple starting positions, while Case 9 did not. However, also recall that the initial starting positions in Case 4 only included positions which had ample time to capture the desired glidepath, as previously discussed. Therefore, starting position was not a factor in the touchdown dispersion. In fact, Figures 4.7 and 4.12 show similar touchdown dispersions, except for the longitudinal center of the dispersion. Since all of the environmental conditions for these two cases were identical, this difference in the mean longitudinal miss distance can be attributed solely to the integral control. As alluded to earlier, the integral control appears to have the effect of moving the center

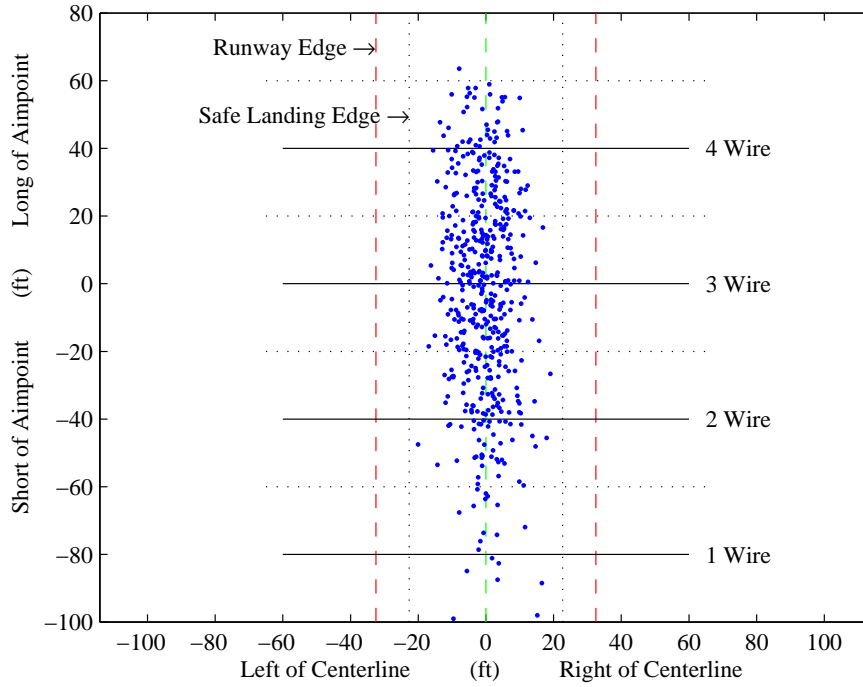


Figure 4.13: Case 10 Landing Dispersion.

of the touchdown dispersion close to the aimpoint. However, since the integral control did not appreciably reduce the standard deviations associated with the touchdown dispersion, a form of disturbance rejection is still needed to tighten the touchdown dispersion.

4.12 *Simulation Case 10*

Simulation Case 10 had environmental conditions consisting of light wind, light wind turbulence, the burble turned on, and sea state 5. Figure 4.13 depicts the results for Case 10. All but one of the 500 simulation runs resulted in successful traps. Of the 499 traps, 19 (3.8%) caught the 1 wire, 122 (24.4%) caught the 2 wire, 233 (46.6%) caught the 3 wire, and 125 (25.0%) caught the 4 wire. Longitudinally, the average approach missed short of the aimpoint by only 2.41 feet with a standard deviation of 31.17 feet. Laterally, the average approach missed 0.49 feet left of the aimpoint with a standard deviation of 6.54 feet. See Tables 4.4–4.6 for more details.

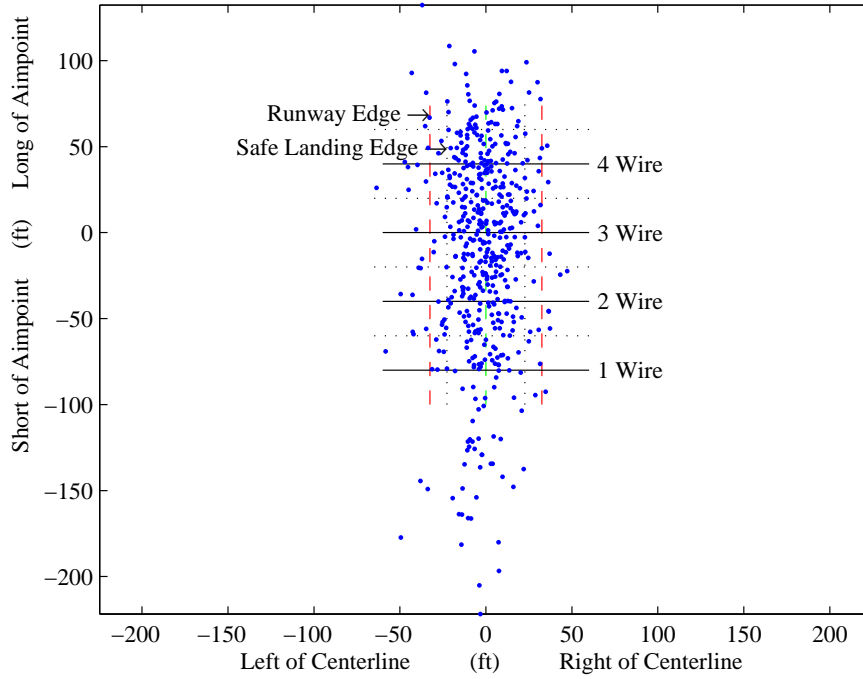


Figure 4.14: Case 11 Landing Dispersion.

Figures 4.12 and 4.13 show very little difference between the touchdown dispersions of Cases 9 and 10, despite the increase in sea state intensity. Table 4.4 shows that the main effect of the increase in sea state was a 10.7% increase in the longitudinal standard deviation. In terms of the number of approaches which caught each wire, this effectively resulted in 17 less 3 wire catches and dispersed them among the other wires, with 1 additional bolter. Thus, sea state does play a small role in the accuracy of the controller, but not enough to appreciably decrease the boarding rate if the integral control is included.

4.13 *Simulation Case 11*

Simulation Case 11, the last case considered for this research, had environmental conditions consisting of moderate wind, moderate wind turbulence, the burble turned on, and sea state 5. These conditions were identical to those in Case 5, so a comparison will be made accordingly.

Figure 4.14 depicts the results for Case 11. The increase in wind turbulence greatly degraded the performance of the controller from the previous case. The boarding rate for this case was 69.6%, compared to 99.8% for the previous case. Out of the 348 traps, 39 (7.8%) caught the 1 wire, 87 (17.4%) caught the 2 wire, 117 (23.4%) caught the 3 wire, and 105 (21.0%) caught the 4 wire. Of the 152 misses, 37 (7.4%) missed more than 20 feet short of the 1 wire, 46 (9.2%) missed long and bolted, and 69 (13.8%) missed to the left or right of the safe landing area. Longitudinally, the average approach missed short of the aimpoint by only 9.98 feet with a standard deviation of 58.82 feet. Laterally, the average approach missed 2.08 feet left of the aimpoint with a standard deviation of 16.76 feet. See Tables 4.4–4.6 for more details.

The results of this case show that wind turbulence has the greatest effect on the controller, out of the various environmental conditions considered in this research. While the controller could compensate somewhat for light turbulence, it could not compensate as well for moderate turbulence. However, this case does show some improvement over Case 5. The difference between Cases 5 and 11 was that integral control was included in Case 11. The boarding rate for Case 11 was 69.6%, compared to that of 65.7% for Case 5. Both the longitudinal and lateral standard deviations were similar for the two cases. The main difference appeared in the mean longitudinal miss distance. The average approach in Case 11 missed approximately 52 feet further aft on the flight deck than the average approach in Case 5, much closer to the aimpoint. This difference can be solely attributed to the integral control. As hoped, the integral control had the effect of forcing the center of the touchdown dispersion closer to the aimpoint. However, better disturbance rejection is needed in order to tighten the dispersion and account for wind turbulence.

4.14 Overall Analysis

Tables 4.4–4.6 contain the statistics for each of the eleven simulation cases. The data has been grouped in order to best facilitate comparison between the different cases. Table 4.4 contains the dispersion statistics for each case. Table 4.5 contains the

breakdown of the number of times each wire was caught and how approaches missed for each case. Finally, Table 4.6 tabulates the total number of traps and misses for each case.

The results for Cases 1–6 show that every time a disturbance was added or increased in intensity, the landing performance of the controller was degraded. However, Cases 7–10, in which integral control was utilized in the outer-loop, showed very little degradation in the boarding rate. The only degradation in performance was the increase in the standard deviations of the touchdowns due to the light wind turbulence considered in Cases 8–10. Thus, the integral control action gave the controller the ability to compensate for the burble and the increase in sea state intensities experienced in Cases 7–10. However, Case 11 yielded a very pronounced degradation in the landing performance due to the increase in wind and wind turbulence intensities. As with the addition of the light wind turbulence, the integral control action was not able to provide enough compensation to reduce the effect of the medium intensity wind turbulence on the touchdown dispersion.

While these results are all important, what is more important is how they compare to existing specifications for autonomous carrier landings. The Unmanned Combat Air System Shipboard Interface Reference Document (USIRD) sets forth guidelines for the integration of UCAVs into fleet operations. There are two versions of the document, References [22, 23]. Reference [22] contains the touchdown dispersion specifications for the mean longitudinal and lateral miss distances and the corresponding standard deviations. These values, along with the target boarding rate, are summarized in Table 4.7, and apply at conditions for sea states 5 and below.

Table 4.7 shows that the only simulation case for which all the requirements were met was Case 1. However, this does not provide any useful information, as Case 1 did not include any disturbances other than sensor noise. Cases 2–6 included disturbances, but did not come close to meeting any of the requirements. Cases 7–10 all came closer to meeting the specifications, but each of them failed at least one of the

Table 4.4: Summary of Landing Dispersion Statistics.

Case	Mean Miss (ft)	Mean Miss Long (ft)	Mean Miss Right (ft)	σ -Miss (ft)	σ -Long (ft)	σ -Right (ft)
1	2.87	2.70	-0.20	1.13	1.13	0.97
2	34.91	31.14	-0.45	18.43	23.45	6.27
3	37.52	29.31	-0.56	20.23	30.27	6.50
4	31.08	14.15	-0.84	19.17	33.09	6.29
5	49.79	19.08	-2.16	28.93	52.16	15.21
6	77.23	41.79	-3.42	56.71	82.86	23.78
7	10.25	-10.23	0.38	0.98	0.98	0.49
8	22.34	-2.04	-0.55	14.74	25.89	6.55
9	24.17	-2.15	-0.51	15.96	28.16	6.52
10	26.55	-2.41	-0.49	17.72	31.17	6.54
11	50.23	-9.98	-2.08	36.29	58.82	16.76

Table 4.5: Summary of Landing Classification Statistics.

Case	<i>Traps</i>				<i>Misses</i>		
	1 Wire	2 Wire	3 Wire	4 Wire	Short	Long	Left/Right
1	0 (0.0%)	0 (0.0%)	490 (100.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
2	0 (0.0%)	13 (2.7%)	131 (26.7%)	302 (61.6%)	0 (0.0%)	44 (9.0%)	0 (0.0%)
3	5 (1.0%)	27 (5.5%)	130 (26.5%)	262 (53.5%)	0 (0.0%)	64 (13.1%)	2 (0.4%)
4	14 (2.9%)	65 (13.3%)	181 (36.9%)	193 (39.4%)	0 (0.0%)	37 (7.6%)	0 (0.0%)
5	25 (5.1%)	54 (11.0%)	92 (18.8%)	151 (30.8%)	13 (2.7%)	105 (21.4%)	50 (10.2%)
6	10 (2.0%)	26 (5.3%)	55 (11.2%)	80 (16.3%)	25 (5.1%)	219 (44.7%)	75 (15.3%)
7	0 (0.0%)	0 (0.0%)	500 (100.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
8	5 (1.0%)	113 (22.6%)	276 (55.2%)	104 (20.8%)	0 (0.0%)	1 (0.2%)	1 (0.2%)
9	13 (2.6%)	123 (24.6%)	250 (50.0%)	114 (22.8%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
10	19 (3.8%)	122 (24.4%)	233 (46.6%)	125 (25.0%)	0 (0.0%)	1 (0.2%)	0 (0.0%)
11	39 (7.8%)	87 (17.4%)	117 (23.4%)	105 (21.0%)	37 (7.4%)	46 (9.2%)	69 (13.8%)

Table 4.6: Summary of Total Traps and Misses.

Case	Total Traps	Total Misses
1	490 (100.0%)	0 (0.0%)
2	446 (91.0%)	44 (9.0%)
3	424 (86.5%)	66 (13.5%)
4	453 (92.4%)	37 (7.6%)
5	322 (65.7%)	168 (34.3%)
6	171 (34.9%)	319 (65.1%)
7	500 (100.0%)	0 (0.0%)
8	498 (99.6%)	2 (0.4%)
9	500 (100.0%)	0 (0.0%)
10	499 (99.8%)	1 (0.2%)
11	348 (69.6%)	152 (30.4%)

Table 4.7: Summary of USIRD Autonomous Vehicle Landing Requirements.

Parameter	Specification
Mean Miss Long (absolute value)	≤ 10.0 ft
σ -Long	≤ 17.2 ft
Mean Miss Right (absolute value)	≤ 2.0 ft
σ -Right	≤ 2.5 ft
Target Boarding Rate*	99 %

*The target boarding rate allows for 1% missed approaches due to bolters.

requirements. Case 7 met the standard deviation, mean lateral miss, and boarding rate specifications, but failed the mean longitudinal miss requirement. Cases 8–10 all met the mean miss requirements and the boarding rate specification, but failed the standard deviation requirements. Finally, Case 11 failed the boarding rate and standard deviation requirements, as the increase in turbulence intensity was too great a disturbance for the controller to handle.

However, just because the specifications were failed does not mean that the concept of Dynamic Inversion, or nonlinear control in general, is not feasible for automated carrier landing applications. Especially for the cases which were run with the integral control in the outer-loop, the results showed that a robust form of noise rejection may be able to bring the range of the touchdown dispersion back within the required standard deviations. Out of the 2,000 simulation runs conducted with light wind turbulence and with integral control in place, only 3 resulted in missed approaches. This fact alone shows that a control strategy such as this has potential to meet the specifications if coupled with the proper disturbance rejection techniques.

However, these results also show that the specifications in Reference [22] may require reformulation. The specifications in Table 4.7 were formulated for the geometry of the newest aircraft carrier, which only has 3 arresting wires. However, most of the carrier fleet still has 4 wires. Consider the main desired result as a 99% boarding rate, with the 1% missed approaches due to bolters. Cases 7–10 of this research exceeded this requirement without meeting the standard deviation specification. Therefore, if

the hard constraint is the boarding rate specification, the rather tight standard deviation requirements summarized in Table 4.7 need not be so tight at this point in time if the majority of early UCAV carrier flight operations will be on ships with 4 arresting wires.

V. Conclusions and Recommendations

5.1 *Conclusions*

The objective of this research was to design, simulate, and evaluate an automated carrier landing system utilizing a nonlinear Dynamic Inversion control architecture to achieve acceptable performance and robustness for a wide range of sea state and atmospheric conditions. There are multiple levels upon which the achievement of this objective can be measured. Indeed, an automated carrier landing system was successfully designed, implemented, and simulated. To this end, this research was a success. However, acceptable performance and robustness were not attained, according to Reference [22]. Neither the performance of the initial controller design nor the improved version utilizing integral control completely met the specifications for integration of automated landing systems into fleet operations, as set forth by the USIRD [22].

Early simulation results were promising, as the results of Case 1 were nearly perfect. However, with each disturbance that was added to the simulation, the landing performance of the controller was degraded. When the disturbance intensities were increased, the performance was degraded even more. After the first six simulation cases, an improved version of the controller utilizing integral control was devised. Its early simulation results were also promising, but it also experienced problems as wind turbulence was added. While the boarding rates yielded by the improved controller were nearly 100% under manageable wind turbulence conditions, some of the USIRD specifications were not met. While these specifications were not met by any of the cases considering disturbances, the results of this research show that proper disturbance rejection techniques may yield the acceptable performance and robustness which were not obtained as a result of this research. On the other hand, if the high boarding rates yielded by the controller met the boarding rate specification, but not the standard deviation specifications, it may be a sign that the standard deviation requirements may need to be relaxed.

While not all of the objectives were met, this research shows that Dynamic Inversion is a viable option around which an automated carrier landing system can be constructed. Additionally, this is not true of only Dynamic Inversion, but other nonlinear control techniques, as well. Nonlinear methods such as Dynamic Inversion allow for detailed modelling of the carrier landing environment that cannot always be accounted for in linear control techniques. While only nonlinear aircraft and burble models were used by the controller in this research, the addition of nonlinear sea state models to the controller holds a great deal of potential for improvement in the results presented here.

5.2 *Recommendations*

In order for the Dynamic Inversion control concept presented here to be able to achieve the required performance and robustness specifications, a proper form of disturbance rejection is needed. This research showed that while integral control was capable of handling much of the model uncertainty presented by the steady portion of the burble and sea state conditions, it was unable to reduce the effect of wind turbulence on the system. Therefore, it is recommended that this research be built upon with the addition of some form of disturbance rejection capable of reducing the effect of wind turbulence on the touchdown dispersion of the controller.

It is also recommended that the formulation of a ship motion prediction model be explored. While the effects of the sea state on the touchdown dispersion were not as pronounced as the effects of the wind turbulence, any improvement that can be made to the system is another degree of reliability which the system will achieve. An increase in the reliability may even allow for an expansion of the sea state envelope for unrestricted flight operations.

As previously alluded to, the specifications regarding the recovery of autonomous aircraft on aircraft carriers may have room for adjustment. If a specific boarding rate is the desired end, then why disregard a possible solution because it does not meet a set of far more stringent touchdown dispersion requirements. As with an optimization

problem, placing too many constraints on a problem may eliminate a large portion of the solution set. While this is not to suggest that trying to set a higher standard is not desirable, it is merely to suggest that some relaxation of the specifications may allow for more solutions.

The results of this research may also serve as a foundation upon which future explorations may be undertaken into the areas of Dynamic Inversion or other nonlinear-based automated carrier landing systems. The world is inherently nonlinear, and there are many nonlinear control techniques which may prove useful in this application. Dynamic Inversion in particular is a very powerful nonlinear control technique, and its worth to the application of automated aircraft carrier landing systems warrants continued exploration.

Bibliography

1. AFRL/VACC. “JUCAS-EQ-Aero-Final.xls”. Microsoft® Excel file with aerodynamic data.
2. AFRL/VACC. “JUCAS-EQ-inertias.xls”. Microsoft® Excel file with inertia data.
3. AFRL/VACC. *Automated Aerial Refueling J-UCAS Equivalent Model Specification*. Technical report, Air Force Research Laboratory, May 2005.
4. Barfield, A. Finley and Jacob L. Hinchman. “An Equivalent Model for UAV Automated Aerial Refueling Research”. *Proceedings of the AIAA Modeling and Simulation Technologies Conference*, (AIAA-2005-6006), 2005.
5. Blake, William B. “Estimate of J-UCAS-EQ Landing Gear Aerodynamic Increments”. Electronic Message. 30 October 2006.
6. Blake, William B. “J-UCAS Sideforce”. Electronic Message. 29 November 2006.
7. Burl, Jeffrey B. *Linear Optimal Control*. Addison Wesley Longman, Menlo Park, CA, 1999.
8. Costello, Donald H. LT, USN, EA-6B Prowler Pilot. Interview on 12 April 2007.
9. Crassidis, John L. and D. Joseph Mook. “Robust Control Design of an Automatic Carrier Landing System”. *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, (AIAA-1992-4619):1471–1481, 1992.
10. DARPA J-UCAS Program Office. “Joint Unmanned Combat Air Systems”. Overview of the J-UCAS Program. “<http://www.darpa.mil/j-ucas/index.htm>”. Accessed on 01 May 2007.
11. Department of Defense. *Flying Qualities of Piloted Aircraft*. Technical Report MIL-STD-1797A, Washington, D.C., January 1990.
12. General Electric Aviation. “GE F110 and F404 Fighter Engines Expand Capability and Global Presence”. Press Release from 19 July 2004. “http://www.geae.com/aboutgeae/presscenter/military/military_20040719d.html”. Accessed on 12 April 2007.
13. General Electric Aviation. “Model F404-GE-102”. Overview of F404-102 engine. “<http://www.geae.com/engines/military/f404/f404-102.html>”. Accessed on 12 April 2007.
14. GlobalSecurity.org. “CVN-68 Nimitz-Class”. Nimitz Class description from GlobalSecurity.org. “<http://www.globalsecurity.org/military/systems/ship/cvn-68-schem.htm>”. Accessed 14 April 2007.

15. Honeywell Technology Center, Lockheed Martin Tactical Aircraft Systems, Lockheed Martin Skunk Works. *Application of Multivariable Control Theory to Aircraft Control Laws*. Technical Report WL-TR-96-3099, Wright Laboratory, May 1996.
16. The MathWorks. “Dryden Wind Turbulence Model (Continuous)”. Overview of the Simulink® Dryden Wind Turbulence Model. “<http://www.mathworks.ch/products/aeroblks/blocklist.html>”. Accessed on 10 May 2007.
17. Naval Aviation Enterprise. *Naval Aviation Vision 2020*. United States Navy, 2005.
18. N-UCAS Advanced Development Program Office. “CG_CVN73_Winds.m”. MATLAB® m-file for determining burble effects.
19. N-UCAS Advanced Development Program Office. “CVN71dynamics.m”. MATLAB® m-file for determining sea state effects.
20. N-UCAS Advanced Development Program Office. “CVN73_25_351.m”. MATLAB® m-file for determining burble effects.
21. N-UCAS Advanced Development Program Office. “CVN73_SBproc.m”. MATLAB® m-file for determining burble effects.
22. N-UCAS Advanced Development Program Office. *Unmanned Combat Air System Shipboard Interface Reference Document*. Technical Report NA-4580-UCAS-1003, Naval Air Systems Command, February 2007.
23. N-UCAS Advanced Development Program Office. *Unmanned Combat Air System Shipboard Interface Reference Document*. Technical Report NA-4150-USRD-1005, Naval Air Systems Command, February 2007.
24. Phillips, Warren F. *Mechanics of Flight*. John Wiley & Sons, Hoboken, NJ, 2004.
25. Pratt & Whitney. “F100”. Overview of F100 engine. “<http://www.pw.utc.com/vgn-ext-templating/v/index.jsp?vgnextrefresh=1&vgnextoid=227f07b06f5eb010VgnVCM1000000881000aRCRD>”. Accessed on 12 April 2007.
26. Pratt & Whitney. “United Pratt & Whitney, Northrop Grumman Begin Ground Tests for X-47B J-UCAS Engine”. Press Release from 12 August 2005. “http://www.pw.utc.com/vgn-ext-templating/v/index.jsp?vgnextoid=2e35288d1c83c010VgnVCM1000000881000aRCRD&prid=633dcc2190a3c010VgnVCM1000000881000a_”. Accessed on 12 April 2007.
27. Steinberg, Marc L. “A Fuzzy Logic Based F/A-18 Automatic Carrier Landing System”. *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, (AIAA-1992-4392):407–417, 1992.

28. Steinberg, Marc L. and Anthony B. Page. "A Comparison of Neural, Fuzzy, Evolutionary, and Adaptive Approaches for Carrier Landing". *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, (AIAA-2001-4085), 2001.
29. Urnes, J.M. and R.K. Hess. "Development of the F/A-18A Automatic Carrier Landing System". *Journal of Guidance, Control, and Dynamics*, 8(3):289–295, 1985.
30. Urnes, J.M., R.K. Hess, R.F. Moomaw, and R.W. Huff. "H-Dot Automatic Carrier Landing System for Approach Control in Turbulence". *Journal of Guidance, Control, and Dynamics*, 4(2):177–183, 1981.

Vita

ENS Nick Denison was born and raised near Akron, Ohio. He graduated from Green High School in Green, Ohio in June of 2002. He promptly reported to Annapolis, Maryland and was appointed a Midshipman at the United States Naval Academy. After four long years as an Aerospace Engineering major at the Academy, he graduated with distinction as a member of the great Naval Academy Class of 2006 and was commissioned as an Ensign and Student Naval Flight Officer in the United States Navy. He then reported to the Air Force Institute of Technology as part of the Immediate Graduate Education Program and continued his aeronautical education, specializing in Aircraft Stability and Control and Optimal Control Theory. Upon graduating from AFIT, he reported to Pensacola, Florida for flight training and the beginning of his adventure in Naval Aviation.

Permanent address: 2950 Hobson Way
Air Force Institute of Technology
Wright-Patterson AFB, OH 45433

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 074-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to an penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 14-06-2007		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From – To) September 2006-June 2007	
4. TITLE AND SUBTITLE Automated Carrier Landing of an Unmanned Combat Aerial Vehicle Using Dynamic Inversion				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Denison, Nicholas A., Ensign, USN				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765 DSN: 785-3636				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GAE/ENY/07-J06	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <p>Dynamic Inversion (DI) is a powerful nonlinear control technique which has been applied to several modern flight control systems. This research utilized concepts of DI in order to develop a controller to land an Unmanned Combat Aerial Vehicle (UCAV) on an aircraft carrier. The Joint Unmanned Combat Air System (J-UCAS) Equivalent Model was used as the test aircraft. An inner-loop DI controller was developed to control the pitch, roll, and yaw rate dynamics of the aircraft, while an outer-loop DI controller was developed to provide flight path commands to the inner-loop. The controller design and simulation were conducted in the MATLAB/Simulink environment. Simulations were conducted for various starting positions near the carrier and for varying wind, wind turbulence, and sea state conditions. In the absence of wind and sea state turbulence, the controller performed well. After adding wind and sea state turbulence, the controller performance was degraded. Future work in this area should include a more robust disturbance rejection technique to compensate for wind turbulence effects and a method of carrier motion prediction to compensate for sea state effects.</p>					
15. SUBJECT TERMS Carrier Landings, Naval Aircraft, Flight Control Systems, Control Systems, Nonlinear Systems					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
REPORT U	ABSTRACT U	c. THIS PAGE U			Maj Christopher M. Shearer, PhD (ENY)
				118	19b. TELEPHONE NUMBER (Include area code) (937) 255-3636, ext. 4643; email: christopher.shearer@afit.edu

Standard Form 298 (Rev. 8-98)

Prescribed by ANSI Std. Z39-18