Design and Analysis of Lunar Lander Control System Architectures

by

Joseph M. Morrow

B.S. Aerospace Engineering
North Carolina State University, 2010

Submitted to the Department of Aeronautics and Astronautics
in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Aeronautics and Astronautics
at the
Massachusetts Institute of Technology

June 2012

Signature of Author: _____
Department of Aeronautics and Astronautics
May 24, 2012

Certified by: _____
Jeffrey A. Hoffman
Professor of the Practice of Aerospace Engineering
Thesis Supervisor

Certified by: _____
Brett J. Streetman
Senior Member of the Technical Staff, Draper Laboratory
Thesis Supervisor

Accepted by: _____
Eytan H. Modiano
Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Design and Analysis of Lunar Lander Control System Architectures

by

Joseph M. Morrow

Submitted to the Department of Aeronautics and Astronautics
on May 24, 2012, in Partial Fulfillment of the
Requirements for the Degree of Master of Science in
Aeronautics and Astronautics

Abstract

Although a great deal of separate work exists on the development of spacecraft actuators and control algorithm design, less work exists which examines the connections between the selection of specific actuator types and placements, how this affects control algorithm design, and how these combined factors affect the overall vehicle performance of a lunar lander. This thesis attempts to address these issues by combining a functionality-oriented approach to actuator type/placement with a controls-oriented approach to algorithm design and performance analysis. Three example control system architectures are examined for a generic autonomous 350kg lunar lander during the terminal descent flight phase. Results indicate that stability and control can be achieved using a wide variety of actuator types/placements and algorithms given that a set of 'common sense' subsystem functionality and robustness metrics are met; however, algorithm development was often heavily influenced/restricted by actuator system capabilities. It is therefore recommended that future designers of lunar lander vehicles consider the impact of their control system architectures from both a functionality-oriented and a controls-oriented approach to gain a more complete understanding of the effects of their choices on overall performance.

Thesis Supervisor: Jeffrey A. Hoffman
Title: Professor of the Practice of Aerospace Engineering

Thesis Supervisor: Brett J. Streetman
Title: Senior Member of the Technical Staff, Draper Laboratory

Acknowledgments

I would like to thank my advisors, Jeff and Brett, for their guidance over the past two years as well as my parents, my sister, and Jessica for their love and support. I would also like to thank the numerous MIT students and faculty, Draper employees, and friends that made my graduate experience worthwhile.

"I was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate (NDSEG) Fellowship Program."

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

Table of Contents

List of Figures

List of Tables

Acronyms and Abbreviations

ACS	Attitude Control System
CEV	Crew Exploration Vehicle
CM	Center of Mass
CNSA	China National Space Administration
COTS	Commercial-Off-The-Shelf
DC (gain)	Direct Current (zero frequency)
DOF	Degree of Freedom
DOFB	Dynamic Output Feedback
DPS	Descent Propulsion System
ESA	European Space Agency
FSFB	Full State Feedback
GLXP	Google Lunar X PRIZE
GNC	Guidance Navigation and Controls
ISRO	Indian Space Research Organization
LEO	Low Earth Orbit
LH2	Liquid Hydrogen
LHP	Left Hand Plane
LM	Lunar Module
LOI	Lunar Orbit Insertion
LOR	Lunar Orbit Rendezvous
LOX	Liquid Oxygen
LQE	Linear Quadratic Estimator
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
LTV	Lander Test Vehicle
MIMO	Multi Input Multi Output
MIT	Massachusetts Institute of Technology
MOI	Moment of Inertia
NASA	National Aeronautics and Space Administration
PD	Proportional Derivative
PDI	Powered Descent Initiation
PID	Proportional Integral Derivative
PWM	Pulse Width Modulation
PWPFM	Pulse Width Pulse Frequency Modulation
RCS	Reaction Control System
RHP	Right Hand Plane
SISO	Single Input Single Output
TALARIS	Terrestrial Artificial Lunar And Reduced gravity Simulator
TLI	Trans Lunar Injection
TVC	Thrust Vector Control
TWD	Tail Wags Dog
US	United States
USSR	Union of Soviet Socialist Republics

Variables and Constants

A	System dynamics matrix in the state space equations
B	Actuator matrix in the state space equations
C	Measurement matrix in the state space equations
Co	Controllability matrix
D	Feed-forward matrix in the state space equations
d	Generic distance (m)
F	Force (N)
G	Generic transfer function
g	Gravity or vertical acceleration (m/sec^2)
h	Generic height (m)
I	Moment of inertia (kg*m^2)
I_{sp}	Specific impulse of propellant (sec)
J	Cost function
K	Controller feedback matrix
k	Generic spring constant
L	Estimator feedback matrix, or generic moment arm
M	Moment (N*m)
Ob	Observability matrix
Q	State penalty matrix
R	Control penalty matrix
s	Laplace variable ($j\omega$)
T	Thrust (N)
t	Time (sec)
T_d	Time delay (sec)
u	Control signal in state space equations
W	Weight (N)
x	Generic state
X	Generic state vector
δ	Gimbal angle (rad)
ΔV	Change in velocity (m/sec)
ζ	Damping ratio of a second order system
θ	Generic rotational state (rad)
σ	Minimum nonzero on time of a thruster (sec)
τ	Time constant of a first order system (sec)
ω	Frequency (rad/sec)

1 Introduction

Lunar lander guidance, navigation, and control (GNC) is the branch of engineering responsible for the design and analysis of closed loop logical and mathematical systems which ensure the delivery of the spacecraft from a given initial state to the lunar surface in a safe and timely manner. Although GNC is often referred to in a strictly algorithmic sense, this thesis will adopt a more holistic viewpoint to the GNC framework by including hardware components in the discussion. The guidance system is then primarily concerned with general vehicle properties (e.g. propellant mass, vehicle mass, ΔV required) and the generation of the desired flight path; the navigation system includes the sensor suite (e.g. inertial measurement unit, Doppler radar) and the algorithms used to estimate the real-time state; and the control system is responsible for using real-time information from the guidance and navigation systems to compute desired linear and angular accelerations and command multiple actuators (e.g. engines, thrusters) in order to reach and maintain a set of desired states.

As the scientific and engineering challenges associated with the lunar lander GNC problem are varied and multidisciplinary in nature, it is difficult to attempt to solve any one of these problems in isolation. The collective problem is typically approached by teams of experienced professionals with many specialties and is much beyond the scope of this thesis. We will narrow our focus towards one specific area within the general lunar lander GNC architecture while doing our best to be cognizant of cross-disciplinary interactions which may invalidate or interfere with our approach. This topic of interest is the design and analysis of control algorithms and actuator selection with special emphasis on their interaction with one another. In colloquial language, this thesis will try to answer the following question:

> "When choosing lunar lander thrusters/engines, what combinations do you choose, where do you put them, and how will your choices affect control algorithm design, stability, and overall performance in the context of the terminal descent flight phase?"

A more formal problem statement is presented in the following sections, including the specific scenario for which design and analysis will be performed, the approach to the problem taken in the remainder of the thesis, and the definition of several useful terms which will help in the understanding and discussion of the 'control system architecture' design and analysis paradigm.

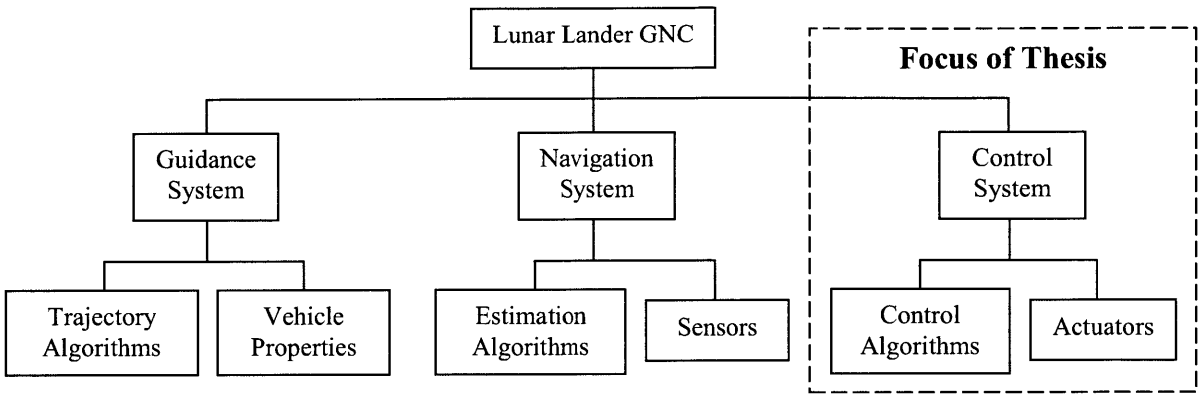


Figure 1-1: Lunar lander guidance, navigation, and control (GNC) framework

1.1 Nominal Mission Descent Trajectory

We begin by defining the type of mission trajectory which the spacecraft must accomplish. Different lunar missions have taken several different approaches in the past, but they tend to include the same general components: launch, a trans lunar injection burn, correction burn(s), an optional lunar orbit insertion burn, and the landing sequence. As the landing phase is typically (only) the responsibility of the landing craft, and thus the primary operational time of the landing craft's control algorithms/actuators, it shall be the focus of this thesis.

The landing phase profile can vary greatly depending on the design of the landing craft, but the general goal tends to be the same: null out the velocity accumulated during the previous portion of the mission, reorient the spacecraft so the landing legs are pointing towards the landing surface, and then touchdown with near-zero velocity. As noted in [1], the larger the magnitude of the average deceleration, the more efficient the maneuver; the optimal descent sequence involves an impulsive burn to null all horizontal velocity and another impulsive burn just before touchdown to null any vertical velocity. However, many other factors besides fuel consumption must be considered by the spacecraft designers, including actuator performance limits, maximum allowable g-loads on the vehicle (and/or passengers), and the ability to modify the trajectory path during descent (e.g. contingency plans). The nominal trajectory considered in this thesis should therefore be similar to those used by previously successful lunar missions, and is outlined in Figure 1-2.

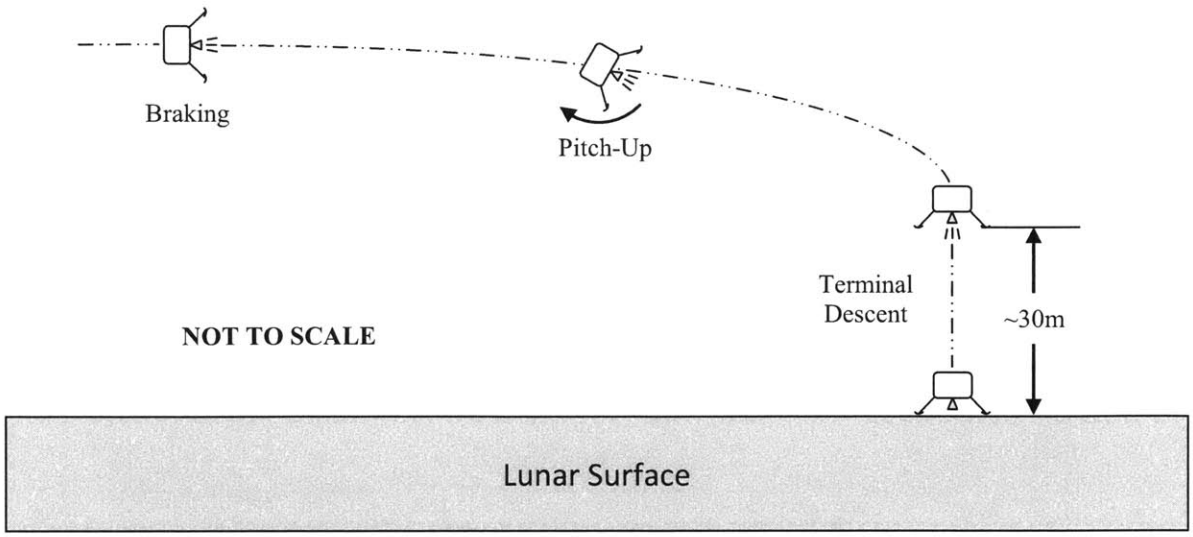


Figure 1-2: Nominal descent trajectory

Within this generic descent profile, the actuators and control algorithms are responsible for commanding and imparting linear and angular accelerations. If we assume no disturbances, the actuators and algorithms are only responsible for performing open loop braking, pitch-up, and descent burns as functions of time in two dimensional geometric space. But as errors are bound to accumulate, it is safe to assume the lander must be able to perform the three aforementioned maneuvers as functions of time in three dimensional space in a closed loop feedback architecture. In other words, they will need to have controllability over all six degrees of freedom (DOF's) of the rigid body spacecraft (three degrees of rotation, and three degrees of translation) in the lunar coordinate frame. Simply put, the control algorithms and actuators will need to be capable of commanding and imparting specific and time-varying amounts of linear and angular impulse in all directions in a 'real-time' manner.

1.2 Defining the Problem

This thesis will address control algorithm design and actuator selection for a generic unmanned lunar lander during the terminal descent mission phase and will highlight key design and analysis challenges. The unmanned lunar lander problem was chosen because it allows us to closely examine the impact of actuator type/placement on control design and performance in a reduced-disturbance environment (e.g. no atmosphere) without concerns for pilot interaction, and the terminal descent phase was chosen because control performance requirements are likely to be highest (and therefore most challenging to meet) during the final vertical descent maneuver [2]. Addressing the terminal descent phase directly

also largely decouples the analysis from guidance and navigation, which are instrumental in mission design but are not the focus of this thesis. Some specific lander/scenario properties which will be used in the remainder of all design and analysis are shown in the tables and figures below. These properties will be expanded upon in Chapter 5 when specific case studies are examined. Note that **CM** refers to the vehicle's center of mass, and the coordinate systems are concurrent with 'aircraft' convention (for simulation and analysis purposes). Therefore, 'X' and 'Y' refer to lateral motion, 'Z' refers to vertical motion, and 'roll,' 'pitch,' and 'yaw' refer to rotations about the X, Y, and Z axes, respectively.

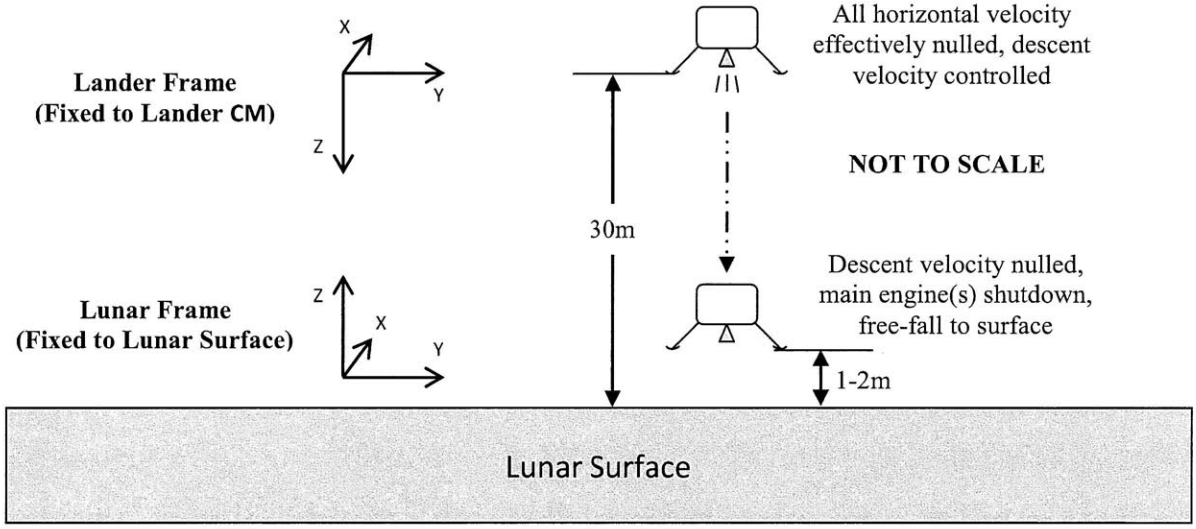


Figure 1-3: Scenario of interest for this thesis – Terminal Descent

Table 1-1: Generic lunar lander vehicle properties (initial, unless specified)

Property	Value	Comments
Mass	350.0kg	Similar to Surveyor
Moment of Inertia (MOI)	$\begin{bmatrix} 45.0 & 0.05 & 0.05 \\ 0.05 & 45.0 & 0.05 \\ 0.05 & 0.05 & 60.0 \end{bmatrix}$ kg*m^2	Similar to comparable Earth-based testbeds; decreases linearly (proportional to current-over-initial mass ratio)
Propellant	(2600/I_{sp})kg	Enough for 45sec of $T/W = 1.0$

Table 1-2: Initial conditions for the terminal descent problem

Property	Value	Comments
Position	[0.0 0.0 30.0]m	30.0m above lunar surface
Velocity	[0.0 0.0 1.0]m/sec	1.0m/sec descent rate in lander body frame coordinate system
Attitude	[5.0 5.0 0.0]deg	Thrust vector nearly perpendicular to lunar surface
Attitude Rate	[0.0 0.0 0.0]deg/sec	

Table 1-3: Touchdown condition limits for the terminal descent problem

Property	Value	Comments
Attitude	$\pm$[6.0 6.0 N/A]deg	Adapted from [2]
Attitude Rate	$\pm$[6.0 6.0 6.0]deg/sec	Adapted from [2]
Max Distance to Target	6.0m	Adapted from [2]
Max Horizontal Speed	1.5m/sec	Adapted from [2]
Max Descent Rate	2.5m/sec	Adapted from [2]

1.3 Defining the Approach

Now that we have defined the operating conditions for which we must design the actuator and control systems, it will be helpful to present the general process for design and analysis. The remainder of the thesis will follow this path forward:

1. Overview of known previous and planned soft-landing lunar programs **(Chapter 2)**
2. Functional decomposition of actuators in the mission context **(Chapter 3)**
3. Identification and discussion of possible actuator types and arrangements **(Chapter 3)**
4. Presentation of useful control paradigms and algorithms **(Chapter 4)**
5. Discussion of design and performance verification techniques **(Chapter 4)**
6. Definition and analysis of three case studies which demonstrate the application of the design process with a focus on actuator-algorithm interaction **(Chapter 5)**
7. Presentation of key results **(Chapter 5)**

Chapters 3 and 4 will attempt to identify key motivations, trends, and sensitivities in general actuator and control system design for the lunar lander, while Chapter 5 will examine three carefully selected actuator arrangements and associated control algorithms to highlight the advantages and importance of understanding and respecting their relationship.

1.4 Important Definitions

Before proceeding any further it will also be useful to define several important terms which will help to classify the areas of interest addressed in the following chapters. This terminology is meant to convey the interconnectedness between algorithms and actuators as well as provide a basis for their discussion.

Actuator	Device used for imparting forces and/or moments on the spacecraft (e.g. thruster)
Actuator Architecture	Describes the number, types, and placements of actuators on the spacecraft (e.g. single gimbaled engine)
Control Algorithm	Mathematical equation which uses information from sensors to command the spacecraft's actuators (e.g. PID, LQG)
Control Algorithm Architecture	Describes the number and types of control algorithms used on the spacecraft (e.g. minimum-time phase plane controller for roll)
Control System Architecture	Refers to both the actuator architecture and the corresponding control algorithm architecture

This thesis is predicated on the belief that control algorithm design and actuator selection/placement cannot be separated, and that neither should be done independently of the other. The term 'control system architecture' captures this relationship and alludes to a design and analysis paradigm that includes actuators, control algorithms, actuator architectures, and control algorithm architectures. This relationship is displayed in Figure 1-4. Chapters 3 and 4 will elaborate on each of the individual branches of the figure, and Chapter 5 will use three carefully chosen case studies to apply the comprehensive paradigm.

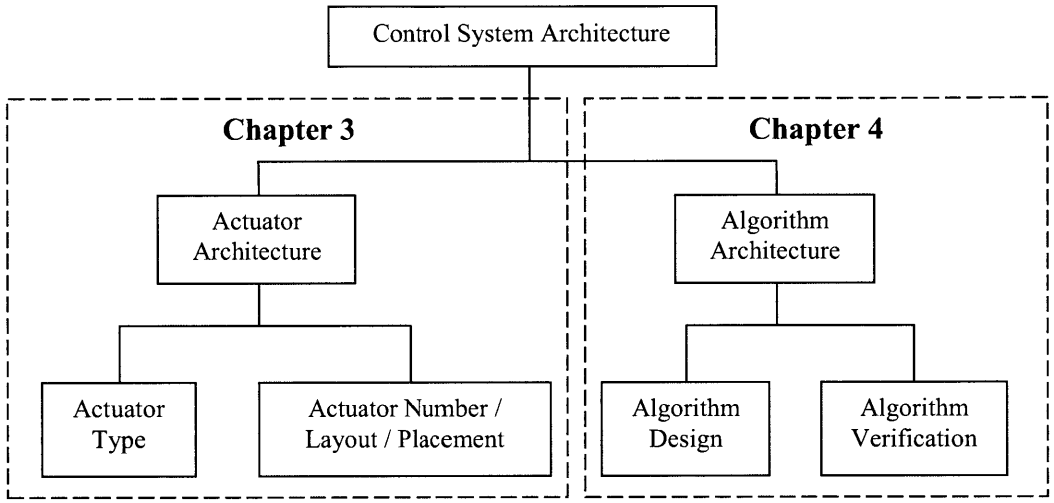


Figure 1-4: Control system architecture breakdown

2 A Historical Perspective

The challenges associated with manned and unmanned lunar landing are not new to the scientific and engineering community; successful 'soft landing' missions were completed by the United States (US) and the Soviet Union (USSR) in the 1960's and '70's, and numerous others have been planned since then. However, advances over the last several decades in the areas of guidance, navigation, and control, and commercial-off-the-shelf (COTS) actuator availability, reliability, and performance, coupled with recent interest in commercial space endeavors make this problem worth revisiting.

The remainder of this chapter will review relevant past lunar lander programs as well as current missions/programs still under development. General landing-phase vehicle, actuator, and control properties will be highlighted (when available).

2.1 Surveyor (US)

The Surveyor program consisted of seven unmanned lunar missions which launched between May 1966 and January 1968 [1] [3]. This was the second generation of lunar exploration vehicles following the Ranger program from 1961 to 1965. Five of these spacecraft (Surveyor 1,3,5,6, and 7) successfully soft-landed on the lunar surface, making them the first US vehicles to make a controlled touchdown on another celestial body. In general, the program is still considered remarkably successful, and much of the scientific and technological information/experience was used to enable the success of the Apollo program.

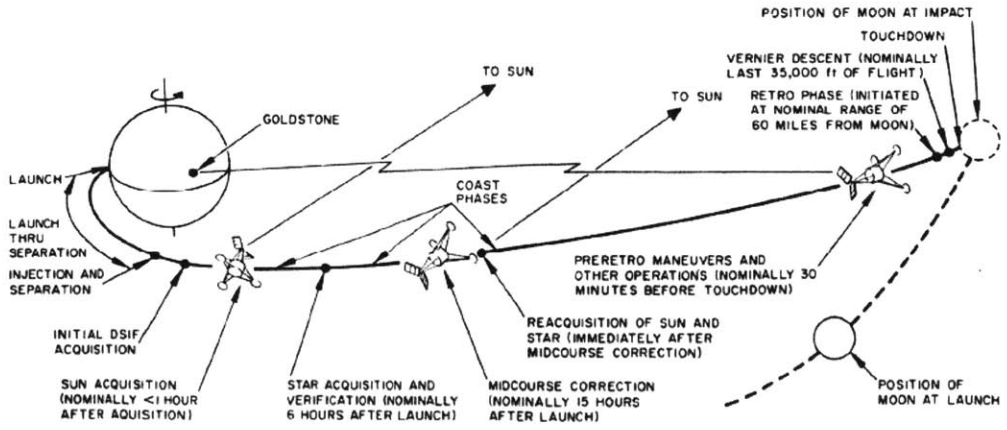


Figure 2-1: Surveyor trajectory [4]

All seven spacecraft launched during the Surveyor program were designed to follow the same general direct-ascent and -descent trajectory outlined in Figure 2-1. All vehicles (initially about 1000kg) were launched on Atlas Centaur rockets directly into a lunar-intercept trajectory without entering a parking orbit around the Earth. Five hours after launch the spacecraft would perform a midcourse correction maneuver using its three throttleable vernier (mid-to-low thrust) engines. Approximately 30 minutes prior to touchdown (96km lunar altitude) the vehicle would reorient itself and ignite the main retro motor to remove nearly 95% of its velocity magnitude. At 7.6km lunar altitude the retro motor would be jettisoned, and the vernier engines would stabilize attitude and remove the final 100m/sec of velocity prior to touchdown. Mass at touchdown was nominally 300kg. Total flight time was approximately 65hrs.

The actual Surveyor lander was 3.0m tall and nearly 5.0m across. A generic schematic can be seen in Figure 2-2. It consisted of an open aluminum structure with three equally spaced landing legs, a solar panel, several antennas, a television camera, a large retro motor, and various other thrusters and sensors. The retro motor was a fixed nonthrottleable solid rocket responsible for removing the majority of the velocity incurred during launch and trans lunar injection. It made up nearly 2/3 of the spacecraft's launch weight and was jettisoned prior to landing [1]. Attitude was controlled by three throttleable engines and a cold gas thruster system. The cold gas system was only responsible for yaw control during the midcourse correction and coast phases; the throttleable engines were responsible for roll, pitch, and yaw control during the entire descent maneuver (one engine was gimbaled ±6.0deg for yaw control).

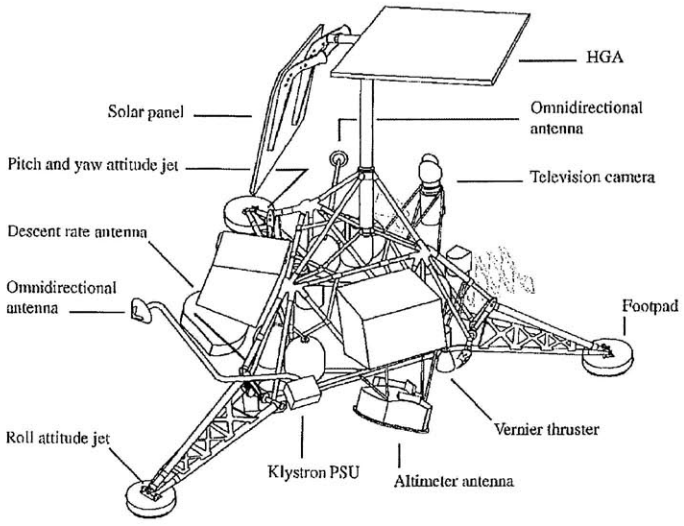


Figure 2-2: Surveyor lander [1]

Development of the throttleable vernier engines was one of the most difficult undertakings of the Surveyor program [5] because they were required to operate over a wide throttle range and perform reliably enough to adequately control multiple degrees of freedom simultaneously. The engines burned hypergolic propellants (meaning the propellants would ignite on contact, so no separate ignition system was required) and used a pressurized blow-down feed system. This eliminated the need for ignition systems and turbopumps but did pose additional challenges such as propellant saturation with Helium (the pressurant), which altered engine performance [6]. The engines were throttleable between 130.0N and 460.0N. Nominal maximum control powers were 233.0N*m in roll and pitch, and at least 3.4N*m in yaw [7].

Guidance was accomplished using linear circuit elements to close a proportional derivative (PD) feedback loop around the velocity error as a function of altitude without specific regard to lateral position. Once the spacecraft reached a predetermined altitude, the logic would switch to a constant velocity command, followed by engine cutoff several meters above the lunar surface [5]. No information was found on the attitude control algorithms.

2.2 Apollo (US)

The Apollo lunar landing missions were the culmination of nearly a decade of effort by NASA and its partners, and they resulted in the first and last instances in which humans have ever visited another celestial body. Seven total missions launched between 1969 and 1972 resulted in six successful Moon landings (Apollo 11, 12, 14, 15, 16, 17) and one aborted mission (Apollo 13); twelve astronauts walked on the surface of the Moon, and all 21 astronauts returned safely to earth.

The mission trajectory was designed to utilize lunar orbit rendezvous (LOR). This meant that the crew and associated spacecraft(s) would enter Earth orbit directly from launch, a separate trans lunar injection (TLI) burn would put them on course to intercept the Moon, and another lunar orbit insertion (LOI) burn would place them into lunar orbit. The lunar module (LM) would then detach from the command module, depart from orbit, and proceed to land on the lunar surface. The descent trajectory was designed as a three-phase maneuver. The first phase, termed the 'braking' phase, was designed to reduce the lunar orbit from 111.0km to 15.0km for more efficient propellant usage; it began with powered descent initiation (PDI), which required the continuous thrust of the descent propulsion system (DPS). The second and third phases, termed 'approach' and 'landing' (terminal descent), were designed for crew visibility and manual controllability. Nominal DPS burn time was 676.0sec (constraint was

910.0sec) and vertical descent distance was 30.0m [8]. After completion of the lunar surface mission, the ascent stage of the LM would detach from the descent stage and reenter lunar orbit where it would rendezvous with the command module before returning to Earth. The basic Apollo mission profile can be found in Figure 2-3.

Design of the actual LM was largely motivated by the decision to attempt the lunar orbit rendezvous technique, which allowed the trajectory planning to be optimized for lunar orbit descent/ascent. The vehicle weighed approximately 15000kg (full), stood 5.5m tall, and was 9.0m wide. The LM was also divided into an ascent stage and a descent stage. The descent stage contained the DPS and four landing legs; the ascent stage housed the two astronauts and included sixteen RCS (reaction control system) thrusters, an ascent propulsion system, and the communications equipment. A schematic can be seen in Figure 2-4. More information on the motivations and challenges behind the design can be found in [9].

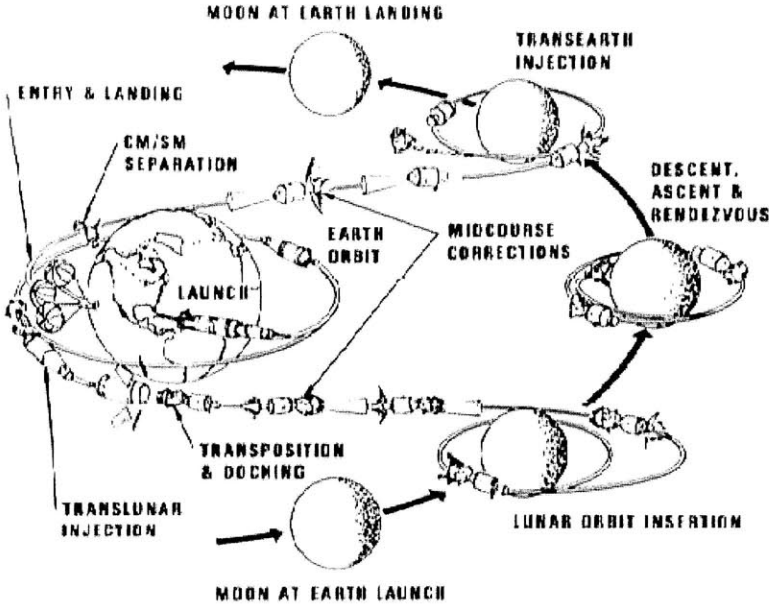


Figure 2-3: Apollo mission profile [10]

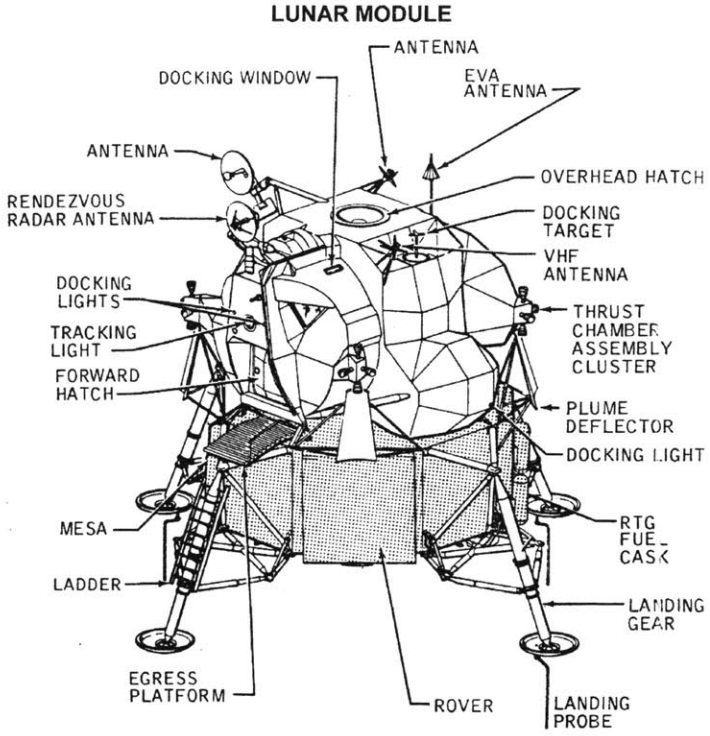


Figure 2-4: Apollo Lunar Module (LM) [11]

The DPS, which was responsible for controlling roll and pitch in addition to imparting ΔV, consisted of a single gimbaled throttleable engine which used hypergolic propellants [12]. The DPS had a maximum thrust of 44500N, was capable of ±6.0deg of gimbal motion and a maximum gimbal slew rate of 0.2deg/sec, and was throttleable to as low as 10%. The RCS, which was solely responsible for yaw during the landing sequence, consisted of sixteen fixed nonthrottleable hypergolic thrusters. Each thruster produced 445.0N and exhibited an equivalent time delay of 15.0ms. Note that the RCS was initially designed to also control roll and pitch during descent, but tight RCS propellant budget requirements led the designers to use the gimbaled descent engine to control these degrees of freedom [13].

Guidance and control logic was processed using a digital computer/autopilot. Guidance was defined by an acceleration function which was a quadratic function of time; it used position and velocity vector errors to compute the required acceleration (DPS throttle) and direction (DPS gimbal angle). Roll and pitch control were defined by a third-order nonlinear minimum-time controller (largely motivated by the low angular rate limit of the gimbal) [14]. Yaw control utilized phase plane logic. For more information on the Apollo guidance computer and the digital autopilot please see [15], [16], and [17].

2.3 Luna (USSR)

The Luna program consisted of a series of robotic spacecraft sent to the Moon between 1959 and 1976. Mission objectives ranged from planned lunar impact to sample return. The program achieved many milestones in lunar (and space) exploration, including the first man-made object to impact the Moon (Luna 2), the first spacecraft to achieve a soft landing on the Moon (Luna 9), the first artificial satellite of the Moon (Luna 10), and the first robotic sample return mission from the Moon (Luna 16). In total, eight spacecraft achieved soft landings (Luna 9, 13, 16, 17, 20, 21, 23, 24); two of these landers deployed rovers (Luna 17 and 21); and three of these landers returned samples of lunar soil to Earth (Luna 16, 20, and 24) [18].

Given the diverse nature of these missions and the lack of information sharing between the US and USSR in the 1960's and 70's, specifics of the actual Luna spacecraft are difficult to find. Sources indicate that landing mass tended to be similar to the Surveyors (somewhat less for the lander-only missions, more for the rover and sample-return missions) and less than the Apollo LM's. Accounts also indicate that different missions used a variety of propulsion and landing techniques, including multiple throttleable engines and airbag systems [19] [20] [21] [22]. An image of a sample-return model is shown in Figure 2-5.

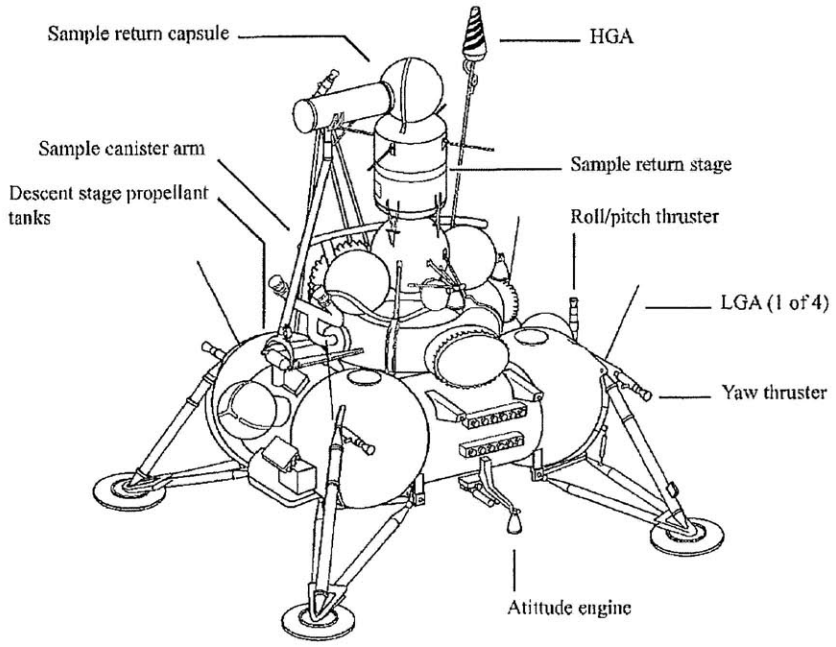


Figure 2-5: Luna 16, 20 [23]

2.4 Constellation (US)

The Constellation program existed from 2005 to 2010; its primary mission was to enable NASA (and partners) to gain significant experience in manned spaceflight operating away from Earth orbit, and to develop the technologies necessary to explore other planets (e.g. Mars). The program featured development programs for a variety of next-generation hardware, including the Crew Exploration Vehicle (CEV), the Altair Lunar Lander, the ARES-I launch vehicle, the ARES-V launch vehicle, and numerous other systems. Initial planned missions included repeated lunar landings and the establishment of a manned lunar outpost which would provide a stepping-stone for more far-reaching manned exploration. The program was canceled in 2010 due to budget cuts while the specifics of the lunar lander mission were still largely in the development phase.

The planned mission trajectory called for two separate launches into low Earth orbit (LEO): the manned CEV on an ARES-I, and Altair/Earth Departure Stage on an ARES-V. After docking and checkout the 'stack' would be placed on a lunar intercept trajectory with a single impulsive burn. A second burn would be initiated after approximately 90hrs to place the stack into a 100.0km lunar orbit. From this point the descent profile for Altair would be quite similar to that of the Apollo LM and would include PDI, braking, pitch-up, and terminal descent. Mass at landing would be approximately 20 metric tons. After mission completion the manned ascent stage of Altair would depart the lunar surface, redock with the CEV, and return to Earth.

The actual lander was much larger (10m tall and 15m wide) than the Apollo LM but had the same general form. It included a manned ascent stage, an unmanned descent stage, four landing legs, a single main gimbaled engine, and multiple RCS thrusters. However, Altair did include several notable differences, including more consumables to accommodate weeklong surface missions, more fuel to perform the LOI burn (which was performed by the Service Module on Apollo), the inclusion of an airlock, and the addition of many advanced sensors. An image of the designed Altair lander can be seen in Figure 2-6.

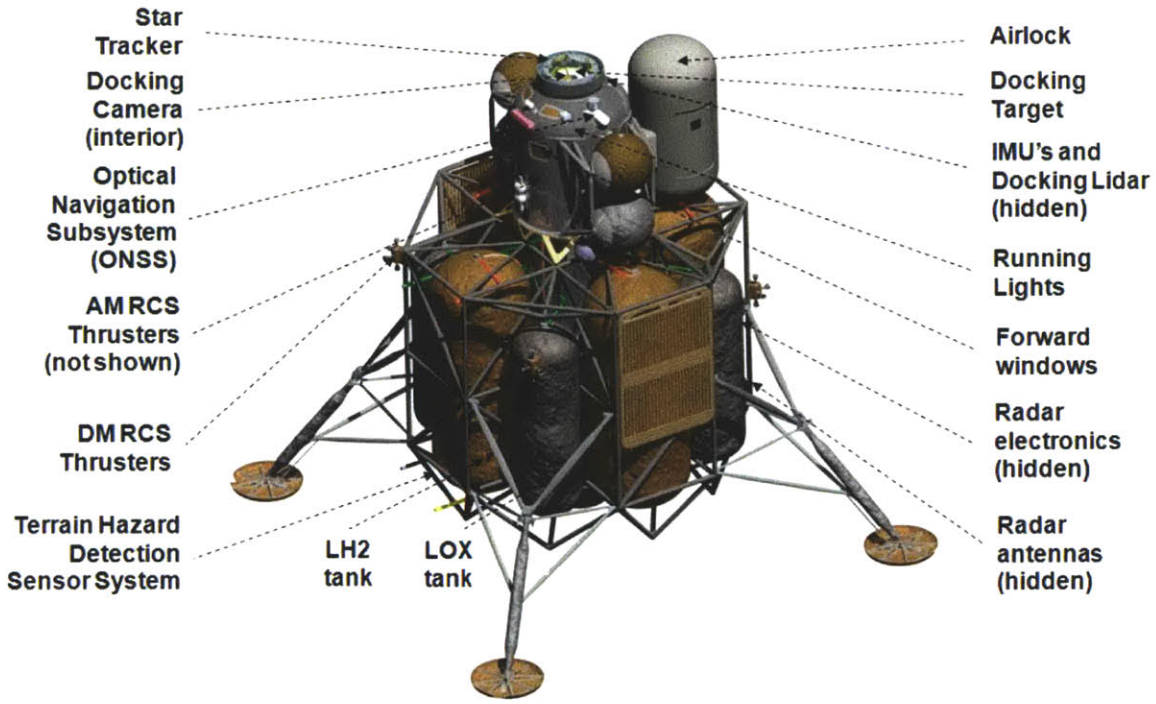


Figure 2-6: Altair Lunar Lander [24]

The descent stage was designed to use a single throttleable gimbaled engine derived from the RL-10 and capable of 66.7kN of thrust. The engine would be responsible for roll and pitch control during the entire descent phase of the mission. Propellants were cryogenic pressure-fed liquid oxygen and liquid hydrogen (LOX and LH2). The sixteen 445.0N RCS thrusters located on the descent stage were responsible for yaw control and small correction burns. The ascent stage contained a single throttleable fixed engine which required four additional fixed nonthrottleable 890.0N thrusters to counter any torque produced by a center of mass (CM) offset from the main thrust vector. The ascent stage also contained 20 RCS thrusters primarily for attitude and position control during docking [24].

Preliminary guidance logic was based on the LM trajectory and restricted acceleration to be a quadratic equation of time. Landing position was required to be within 1.0km of the desired location [25]. Initial control algorithm design called for proportional derivative (PD) or proportional integral derivative (PID) for the descent stage thrust vector control (TVC) and phase plane logic for yaw [26]. Research also indicates that the terminal descent phase would require the tightest performance requirements on all degrees of freedom.

2.5 European Lunar Lander (ESA)

The European Space Agency (ESA) Lunar Lander Mission, also known as MoonNext, is a program designed to soft-land a robotic spacecraft on the lunar South Pole. The specific objectives call for a successful autonomous soft precision landing (with autonomous hazard avoidance), deployment of a scientific payload or rover, and one calendar year of operations on the lunar surface [27]. The mission is currently in Phase B1 and is not expected to reach operational phases for several years [28].

The planned mission trajectory calls for a launch on a Soyuz 2-1B rocket and later injection into polar lunar orbit. PDI will begin at 15.0km lunar altitude followed by coasting at 2-3km altitude and vertical descent soon thereafter. Hazard avoidance and precision landing techniques will be used to place the spacecraft within a 200.0m landing radius. The entire descent and landing phase will be autonomous.

Preliminary designs are calling for a landing mass near 700kg, multiple fixed nonthrottleable main engines to impart ΔV, and pairs of fixed nonthrottleable pulsed thrusters for attitude control. Designers have expressed general concerns over propellant slosh mode interaction with attitude limit cycling as well as with clustered main engine hydraulic cross-talk and thermal issues. At this point it is unclear which control system architecture will be selected for the final design [28].

2.6 Chinese Lunar Exploration Program (CNSA)

This program, also known as Chang'e, is part of a robotic lunar exploration program organized by the China National Space Administration (CNSA). It calls for two lunar orbiters (Chang'e 1 and 2, which have already been launched) as well as multiple planned lunar landers, rovers, and sample-return missions. A 2009 article has confirmed 2013 has the launch date for the first lander/rover mission (Chang'e 3) and reports that the intended landing region is near the Moon's equatorial region [29] [30]. Little else is known about the status or specific design of the mission.

2.7 Chandrayaan-2 (ISRO/Roskosmos)

The Chandrayaan-2 mission is a joint venture between the Indian Space Research Organization (ISRO) and Russia's Federal Space Agency (Roskosmos) designed to soft-land a robotic spacecraft on the surface of the Moon [31]. The mission was originally planned for 2013, but recent news indicates it may be delayed several years due to mass overruns and challenges associated with cryogenic rocket propulsion [32] [33]. Reports indicate that the spacecraft will include a 1400kg orbiter, a 1250kg lander,

and a 15kg rover. Little information has been released on the actual lander design or the control system architecture.

2.8 Commercial Efforts

Although all major missions to date have been conducted by government-run agencies, the commercial sector has also made great strides in recent years in addressing the lunar landing problem. The following examples are those that were most influential/relevant to this thesis.

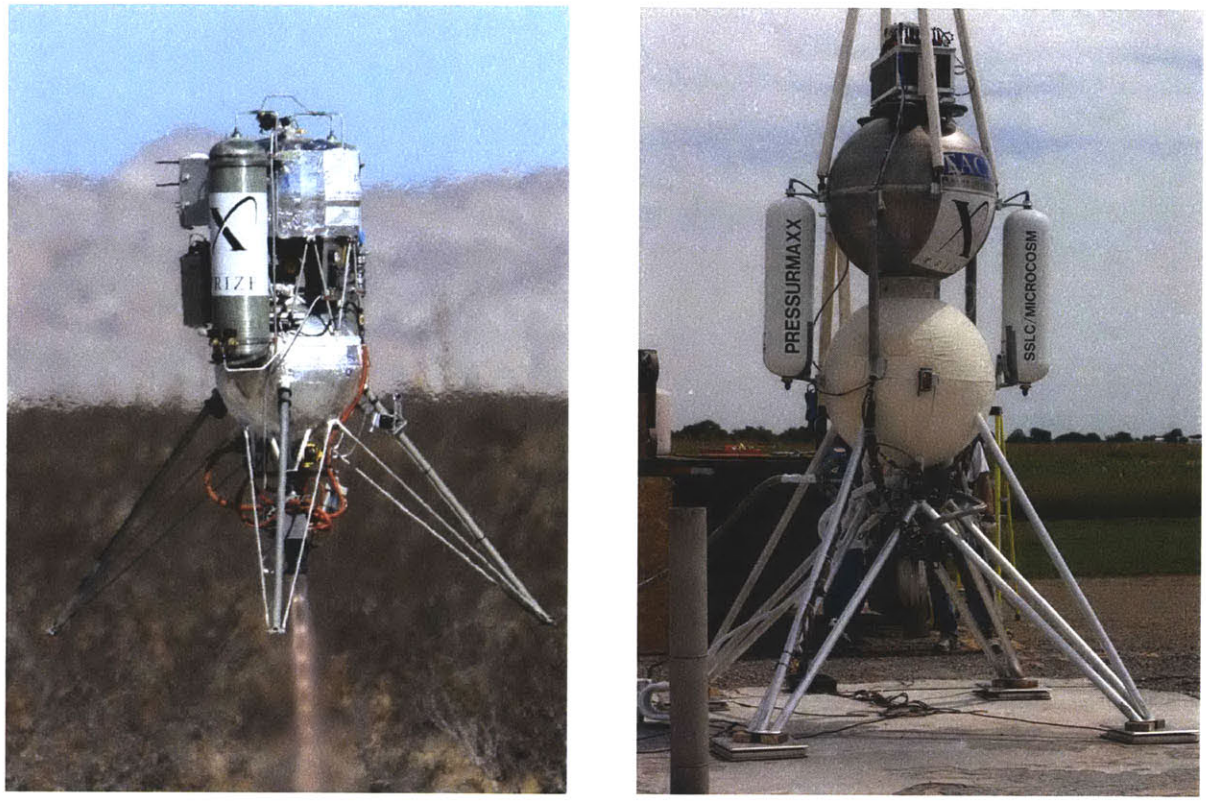
2.8.1 Northrop Grumman Lunar Lander Challenge (US)

The Lunar Lander Challenge [34] [35] was a competition designed to foster innovation and commercial interest. Although the competition took place on Earth and at low altitudes, the engineering challenges were remarkably similar to those faced in the terminal descent phase of an actual lunar lander. It was funded by NASA's Centennial Challenges program that offered up to $1.0 million in prize money to the first team able to achieve the following goals with vertical takeoff and landing rocket-powered vehicles:

- Level 1: Takeoff from a concrete pad, gain 50.0m altitude, fly laterally 100.0m, descend and land on another concrete pad. Repeat the flight in reverse before less than 150.0 minutes have elapsed. Both flights must last longer than 90.0sec.
- Level 2: Same as Level 1, except landing pad contains 'lunar surface-like' obstacles (e.g. boulders) and minimum flight time is 180.0sec.

The competition was conducted by the X PRIZE Foundation with sponsorship from Northrop Grumman. In 2009 Masten Space Systems was awarded the $1.0 million first place prize and Armadillo Aerospace was awarded the $500,000 second place prize [36].

Both the Masten and the Armadillo vehicles (shown in Figure 2-7) have the same basic structural layout and actuator architecture. Both use a single gimbaled throttleable engine for altitude, roll, and pitch control (approximately 4kN max thrust) and a blow-down RCS for yaw control. Unfortunately, the specifics of the designs are unpublished.



(a) (b)

Figure 2-7: Lunar Lander Challenge winners; (a) Xoie [34], (b) Scorpius Super Mod [37]

2.8.2 Google Lunar X Prize (International)

The Google Lunar X Prize (GLXP) is organized by the X PRIZE Foundation and is sponsored by Google. It offers a total of $30 million in prizes to the first privately funded teams to land a robot on the Moon, travel more than 500.0m, and transmit high definition images and video back to the Earth. Similar to the other X PRIZE competitions, this challenge is intended to inspire private investment in hopes of developing more cost-effective technologies [38] [39].

Although there are numerous registered teams, very few have published results applicable to the challenges associated with the descent and landing of a vehicle on the lunar surface. Team Next Giant Leap [40] [41] [42] [43] and Team Moon Express (in partnership with NASA) [44] are two exceptions. Their vehicles, which are designed to validate lunar lander systems on Earth-based testbeds, are shown in Figure 2-8. As there are many differences between the planned final lunar lander design(s) and these vehicles, hardware specifics will not be discussed.

(a) (b)

Figure 2-8: GLXP testbeds; (a) Next Giant Leap's TALARIS [45], (b) Moon Express's LTV [44]

3 Actuators and Actuator Architectures

In Chapter 1 we identified that actuators must be able to apply necessary (and changing) angular and linear impulse to the spacecraft about all three rotational degrees of freedom and all three translational degrees of freedom. They must provide force and torque, and be able to respond to 'real-time' GNC commands to alter the collective forces and torques over time. In this section we will investigate the various types of actuators and actuator placements applicable to lunar lander control system architecture design.

3.1 Actuators

This section will first address the various ways in which actuators can provide force and torque; it will then address the ways in which actuators can vary linear and angular impulse.

3.1.1 Ways of Providing Force and Torque

Historically, actuation has been accomplished through the use of chemical and cold gas thrusters/engines. Other 'low-thrust/-moment' solutions (e.g. reaction wheels, electric thrusters) may be feasible under some mission profiles, but they tend to lack the ability to initiate the timely maneuvers required by our specified landing profile; these devices will therefore not be included in this discussion. Solid propellant rockets are also ignored as they are unable to generate the non-predetermined time-varying thrust levels required by the selected mission.[1] For a more complete discussion on the details of the following propulsion systems, please see [46] [47] and [48].

Cold Gas

Cold gas systems are the simplest and lowest-performing actuator. They operate on the same principles of isentropic expansion as a hot-fire actuator but use stored high pressure (typically several thousand lb/in^2) gas (often non-reactive, such as N_2) instead of chemical combustion to create the pressure differential necessary to obtain supersonic flow through a convergent-divergent nozzle. These systems are extremely simple, flight-proven, non-toxic, and relatively easy to make. However, these systems are

[1] Surveyor did use a solid rocket 'retro' motor to reduce velocity from 2700m/s to 100m/s, but only liquid propellant throttleable thrusters were used during the final 7600m of the descent [5].

difficult to throttle and are not well suited for high-thrust or high-impulse applications due to the non-negligible weight penalty associated with upscaling.

Monopropellant

Monopropellant systems react a single propellant (such as Hydrazine, Hydrogen Peroxide, Hydroxyl Ammonium Nitrate) with a catalyst (typically solid) to create high pressure and temperature in the combustion chamber. These systems are more complex than cold gas systems, as they require specialized catalyst equipment and a separate propellant pressurization system. However, they offer the potential of throttleability and a better mass fraction for mid-range applications.

Bipropellant

Bipropellant systems react a fuel and oxidizer inside the combustion chamber to create high pressure and temperature. These systems offer the highest theoretical efficiency but at the cost of complexity and support system overhead. They are therefore typically reserved for high-thrust high-impulse applications. They are most often seen as attitude control system (ACS) actuators when high I_{sp} (specific impulse) is a driving requirement, and when the ACS can share the same propellant feed lines as a larger onboard propulsion system. Note that hypergolic propellants have been used on multiple lunar landing vehicles, as they eliminate the need for a separate ignition system.

Comparison

The results are summarized in Figure 3-1. Cold-gas tends to be best for low-impulse applications, monopropellants tend to be best for mid-impulse applications, and bipropellants tend to be best for high-impulse applications. A few simple calculations indicate that the impulse necessary for attitude control during a single landing profile of a Surveyor-sized vehicle will fall in the mid-to-low impulse range, while any maneuver requiring non-negligible ΔV will be in the high-impulse range. This is an important realization, and it helps explain why spacecraft often have separate actuators for attitude control and trajectory modification. This discussion will be continued in detail later.

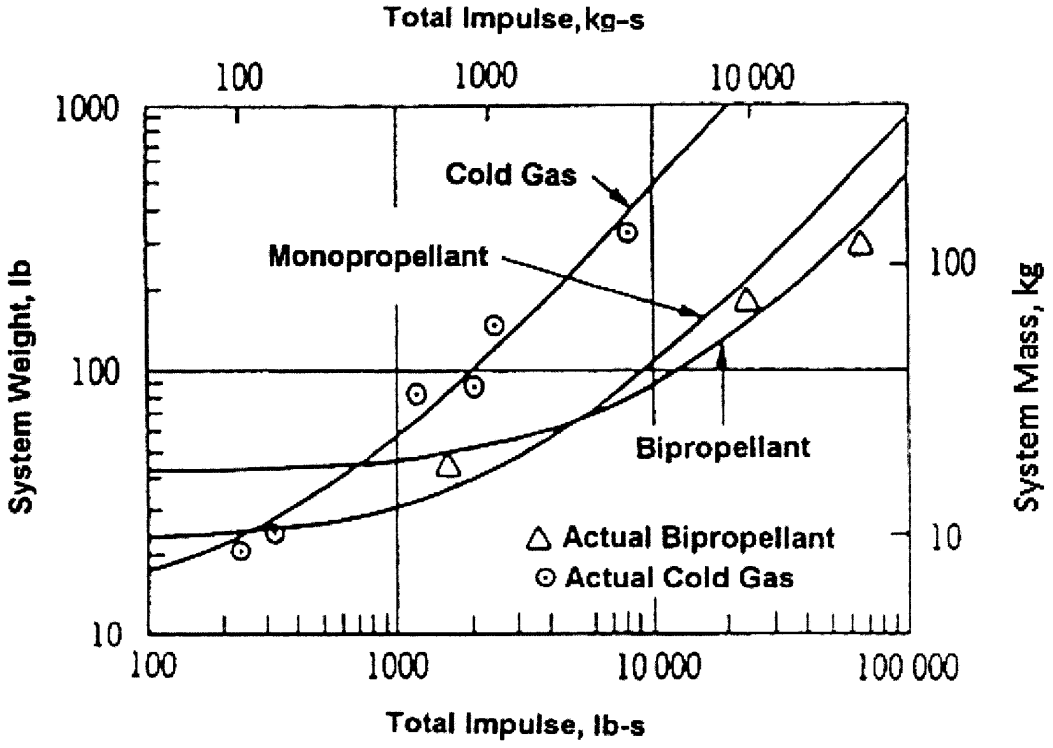


Figure 3-1: Thruster system mass vs. total impulse comparison [23]

3.1.2 Ways of Changing Linear and Angular Impulse

The nature of closed loop control specified by our flight profile requires that actuators also be able to vary the linear and angular impulse imparted to the spacecraft on a 'real-time' basis. As impulse is a function of force (a vector) and time, solutions tend to involve changing one or both of these variables. This section will provide the background for the 'fixed vs. gimbaled' and 'throttleable vs. pulsed' discussion.

Changing Impulse Magnitude

Changes in impulse magnitude can be accomplished by enabling the actuators to be throttleable (change impulse by controlling thrust) or by enabling actuators to pulse on and off quickly (change impulse by controlling on-off time). Throttleability offers the potential for flexibility in control algorithm design and better 'control performance' but at the cost of complexity and combustion efficiency. Liquid propellant rocket engine throttling is a difficult problem that should not be taken lightly [49] [50] [51]. Pulsed actuators (often referred to as 'bang-bang' actuators) grant less flexibility in control algorithm

design and generally induce sustained oscillation about a set point, but the advantages in hardware simplicity and reliability mean they have been used frequently for spacecraft ACS systems. Research also indicates that pulsed actuators also suffer from a decrease in combustion efficiency and an increase in impulse variation [52] [53] [54].

Changing Impulse Direction

Changing impulse direction can be accomplished by changing the direction (for linear and angular control) and/or origin (for angular control) of the thrust vector. The direction is typically changed by modifying the physical nozzle direction of an actuator (gimbaling) or by firing multiple nonparallel actuators simultaneously; the origin is typically changed by firing multiple parallel actuators at different locations on the spacecraft. Note that these two methods are often combined to simultaneously change impulse linear and angular impulse simultaneously. The first method requires the use of at least one gimbaled actuator, while the second method typically requires the use of multiple fixed actuators. As gimbaling is more complex and requires additional moving hardware, it is often reserved for high-impulse applications where precise thrust vector control is necessary. The use of multiple fixed actuators favors simplicity and is widely used in ACS systems.

3.1.3 'Gimbaled vs. Fixed' and 'Throttleable vs. Pulsed'

Choosing whether to use gimbaled, fixed, throttleable, and/or pulsed actuators on a lunar lander is important because it has a significant impact on how the remainder of the control system architecture will be designed. While all four combinations are theoretically capable of providing the necessary angular and linear impulse required to complete the mission, each one offers some clear advantages and disadvantages. Table 3-1 summarizes these qualitative tradeoffs from the perspective of a control system architect.

Table 3-1: Tradeoffs associated with gimbaled, fixed, throttleable, and pulsed actuators

Actuator Type/Combo	Advantage	Disadvantage	Comments/Example
Gimbaled	Continuous controllability of angular impulse	Gimbal system adds mass and complexity	Typically used with high-impulse engines
Fixed	Simple and robust	Multiple often required to achieve full controllability	Typically used for low-impulse ACS
Throttleable	Continuous controllability of thrust magnitude	Complex; throttleable engines can be difficult to develop	Typically used for high-impulse engines
Pulsed	Simple and robust	Discrete controllability; unavoidable limit cycle; time lags increase with thrust magnitude	Typically used for low-impulse ACS
Gimbaled and Throttleable	Continuous controllability of both linear and angular impulse	See above	Apollo LM main engine responsible for ΔV and roll/pitch control
Gimbaled and Pulsed	Discrete controllability of both linear and angular impulse	See above	Rarely used
Fixed and Throttleable	See above	See above	Surveyor engines responsible for ΔV and roll/pitch control
Fixed and Pulsed	See above	See above	Widely available; used by almost all ACS systems

3.1.4 Choosing Thrust Levels

Another important step in selecting actuators is choosing thrust magnitude. Although the basic thrust equation in rocket propulsion indicates that thrusters/engines should be quite scalable, there are some important 'real-world' challenges and consequences associated with designing actuators of different sizes (e.g. throttleability limits, combustion stability, timing characteristics). When in doubt, it may be helpful to compile a list of actuator types that are commercially available and/or flight-proven. If a particular design calls for a type of actuator not on the list, think twice before committing to the design.

Recall that the mission profile calls for a high-impulse deceleration maneuver, a low-impulse maneuver to change attitude, and a short terminal descent phase requiring both high linear impulse (gravity-cancellation burn) and low angular impulse (attitude stabilization) maneuvers. Using an Apollo-like landing profile [8], a 350kg lander should expect to impart on the order of 5e5N*sec during the high-

impulse maneuvers, while the low-impulse maneuvers will be several orders of magnitude lower. This is an important observation because it means that under nominal conditions there is a natural decoupling between actuators required for the two types of maneuvers: we either need to design a single system capable of both high linear impulse and low angular impulse, or we need to design two separate systems for each type of maneuver. This tradeoff will be revisited in the following sections.

Based on the conclusions of the previous paragraph, we know that the thrust levels required by the high linear impulse actuator(s) will be primarily determined by the trajectory design of the landing profile. Historical data and simulations indicate that the maximum 'thrust' should be between 1.2 and 5.0 times the vehicle's empty lunar surface weight, and that the minimum 'thrust' should be less than the vehicle's empty lunar surface weight [8] [12].

We also know that the thrust levels required by the low angular impulse actuator(s) will be less dependent on trajectory and more dependent on desired closed loop attitude system bandwidth (e.g. maximum and minimum angular acceleration desired by GNC algorithm designers). Data from the Apollo program and more recent unmanned Earth-based lunar lander analog systems indicate that acceptable minimum acceleration tends to be in the vicinity of $0.02 rad/sec^2$ and maximum acceleration tends to be in the vicinity of $1.0 rad/sec^2$ [40] [55] [56] [57] [58]. Additional factors that affect the chosen ranges include closed loop performance characteristics (e.g. rise time, slew rate), magnitude of anticipated disturbances, acceptable limit cycle characteristics, and propellant consumption.

3.2 Actuator Architectures

We have identified the different types of actuators and the various methods of changing linear and angular impulse. We have also concluded that under nominal operating conditions there is a natural decoupling between the systems required for high linear impulse and low angular impulse. This section will discuss the various actuator layouts/configurations/types which may be suited to meet the needs of these decoupled systems as well as introduce the notion of intentional coupling.

3.2.1 High Linear Impulse Actuation System

As observed in the previous sections, the high linear impulse actuation system must be capable of changing its linear impulse in response to closed loop GNC commands. This leaves an infinite number of potential architectures involving medium-to-high thrust single/multiple, gimbaled/fixed, and throttleable/pulsed engines. As it would be unnecessary and impractical to include multiple high-thrust

engines with nominally opposing thrust vector directions, let us assume that all remaining configurations require the engine(s) to be nominally directed perpendicular to the plane formed by the landing legs. This requires that all large changes in acceleration direction require reorientation of the spacecraft via the second low angular impulse actuation system, which is not an unreasonable requirement given the importance of low system mass and relatively low required system position control bandwidth during descent. This is, in fact, the same general requirement placed on the landing systems for all known successful manned and unmanned lunar landers.

If we assume that the high linear impulse actuation system is fully decoupled from the low angular impulse actuation system, then the preferred choices for actuator architectures tend to involve fixed single/multiple (if multiple, symmetrically placed around the vehicle's center of mass and commanded as one larger actuator) pulsed/throttleable engines capable of collectively providing the impulse range specified by the preselected trajectory. (As noted in Table 3-1, gimbaled actuators offer the advantage of continuously varying angular impulse, which is not necessarily beneficial if we intend to decouple the linear and angular impulse actuation systems. High impulse gimbaled actuators will be discussed in the section reserved for intentional coupling between linear and angular control systems.) From the perspective of linear impulse efficacy, these options all seem comparable; only differences in system mass can be used to rank options under nominal operating conditions, and this comparison must be done on a hardware-specific basis.

3.2.2 Low Angular Impulse Actuation System

By the nature of the maneuvers required in the nominal flight profile, the low angular impulse actuation system should be required to exert on the order of tens of N*m*sec. If we again choose to decouple the low angular impulse actuation system with the high linear impulse actuation system, then the preferred choices for actuator architectures involve multiple fixed/gimbaled, throttleable/pulsed thrusters capable of controlling all three degrees of rotational freedom. (A single thruster cannot directly control rotation about all three orthogonal axes in a decoupled manner.) We also expect an even number of thrusters situated such that multiple thrusters can be fired simultaneously in order to induce a rotational acceleration without a coupled translation acceleration.

A configuration with multiple gimbaled throttleable ACS thrusters offers continuous and linear actuation capability, and this often means remarkable control performance and flexibility in algorithm design. However, research and experience strongly indicates that the design/cost/mass overhead associated

with multiple low-impulse throttleable gimbaled thrusters render these configurations nearly implausible [46] [47]. The only remaining advisable ACS architectures involve multiple low-impulse fixed pulsed (bang-bang) thrusters, and the remainder of the section will be devoted to this topic.

The use of fixed 'bang-bang' actuators to control attitude is a popular choice among spacecraft designers; it has a long heritage of flight-proven hardware and success. Although few papers detail the exact factors affecting the decision of actuator number and placement, most lunar landers tend to have similar configurations. To avoid preconceived notions about which configuration is 'best,' we will use a simple thought experiment to demonstrate the tradeoffs associated with various decisions. As actuator number and placement on an actual spacecraft will no doubt be affected by additional factors than discussed here, the intention is to alert the reader to sensitivities and trends rather than to find an optimal design. Table 3-2 shows the basic factors which should be considered.

Table 3-2: Important factors to be considered during pulsed actuator architecture selection

Factor	Comments
Angular acceleration (maximum and minimum)	This value should be determined by the GNC team based on desired performance characteristics. It will be a factor of thrust magnitude and distance from the vehicle's center of mass. Increasing the distance will decrease the required thrust, which may decrease thruster mass.
Full controllability of rotational degrees of freedom	Required for all missions studied in this thesis.
Controllability of translational degrees of freedom	Optional, depending on mission profile. For example, ACS may be responsible for 'fuel-settling' ullage burns and/or direct translation during docking.
Decoupling between rotation about different axes, and decoupling between rotation and translation	This will greatly reduce control complexity, particularly if actuators are fixed and nonthrottleable.
Redundancy	As full controllability of all six DOF's is required for mission success, the penalty for adding some additional thrusters is relatively small. Nearly all spacecraft have redundant ACS thrusters.
Plume impingement	High velocity/temperature gases can damage the spacecraft and reduce effective thrust. Plume impingement must be kept to a minimum, or necessary shielding measures must be taken.
Reducing system mass and propellant usage	Desired for all missions.

Let us begin by considering a rigid body capable of rotation about one DOF and translation about two DOF. Knowing that we must have rotational controllability of the body using fixed nonthrottleable thrusters (we assume translational control and ΔV is handled by a separate system), we add a minimum of two thrusters to enable angular acceleration in both directions. We place the thrusters on a corner of the body (furthest from the center of mass to minimize thrust requirements), 180deg apart from one another and perpendicular to the line connecting the corner to the center of mass (to maximize moment). However, this configuration does not decouple rotation from translation. We therefore add two more thrusters opposite the first cluster. Opposite thrusters can now be fired simultaneously to enable rotation without inducing translation. However, if one thruster fails the vehicle can no longer decouple translation from rotation during maneuvers nominally requiring the failed thruster. We therefore add two more thrusters (each) on the two remaining corners. Now all single failures (and some double failures) can be tolerated without a loss of controllability or decoupling. Under nominal conditions this configuration also allows for the actuators to impart two separate levels of angular impulse magnitude. This configuration also allows the thrusters to impart some linear impulse without a coupled rotational impulse (although linear impulse here is not redundant). The thought process is outlined in Figure 3-2. The main conclusion is that each degree of rotational freedom requires four actuators to achieve decoupled rotational control and eight actuators to maintain full redundancy.

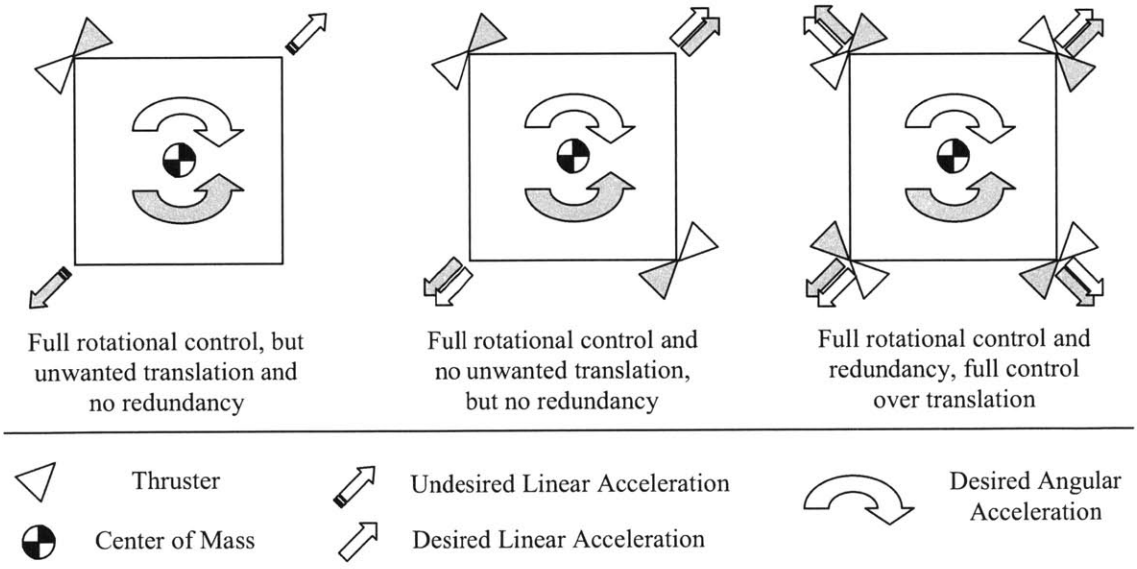


Figure 3-2: Thought process for choosing thruster number and position for a 2D rigid body

Now we extend the thought experiment to a 3D rigid body capable of three degrees of rotational freedom and three degrees of translation freedom. As before, we assume that we can select the number and placement of fixed nonthrottleable actuators to control the three degrees of rotational freedom. It is assumed that the spacecraft has a second actuation system responsible for high-impulse translational control (ΔV). Continuing to imagine the spacecraft as a cube, we can now place clusters of thrusters on each face, edge, or corner. Placing them on the face offers the fewest clusters (which may reduce the number of propellant feed lines), but it decreases distance to the vehicle center of mass and increases the chances of plume impingement. Placing them on the edges decreases the chance of plume impingement, but it does not maximize distance to the vehicle center of mass and has the most clusters. Placing them at the corners (sets of three thrusters, each thruster parallel to an edge at each vertex) offers a combination of the aforementioned tradeoffs. Regardless, the number of thrusters remains 24 if we wish to maintain full redundancy, full decoupling, and the benefit of non-redundant translational controllability.

Finally, it is worth noting that there is in fact an infinite combination of thruster numbers and placements which provide the benefits discussed in the previous paragraph. More often than not, other important factors will influence these decisions, such as a non-symmetric vehicle planform, a non-negligible shift in the center of mass and/or moments of inertia, and the requirement for redundant translational controllability by the ACS (as perhaps is required during docking). Although the final configuration is likely to vary from the architecture presented in this thesis, the factors presented in Table 3-2 should remain guiding factors throughout the design.

3.2.3 Designed Coupling between Actuation Systems

The two previous sections discussed the design of two separate actuation systems for high linear impulse maneuvers and for low angular impulse maneuvers. This section will discuss actuator architectures with intentional coupled functionality.

Intentional coupling should generally be viewed as a subset of the high linear impulse architectures (because low angular impulse architectures are unable to produce non-negligible linear accelerations). Recalling that the high linear impulse architectures discussed in the previous sections involved fixed single/multiple, pulsed/throttleable engines all nominally directed towards the same plane, there are two straight-forward ways of imparting an angular impulse: differential impulse and thrust vector control (TVC). Differential impulse is typically accomplished by placing multiple throttleable/pulsed

engines symmetrically around the center of mass, and TVC is typically accomplished by placing one or more throttleable engines symmetrically around the center of mass. Note that if one wishes to control all three degrees of rotational freedom, the planar architecture must include multiple TVC engines or multiple fixed engines and a single TVC engine.

It may now be evident why designed coupling between actuation systems is often a popular choice: relatively small modifications to a high linear impulse system offer the potential for controllability of all necessary degrees of freedom. This allows the separate small angular impulse actuator system to be reduced or even eliminated all together. In addition, non-gimbaled architectures with more than four engines and gimbaled architectures with more than one engine offer the benefit of redundancy. However, as one might have already guessed, this means increased design requirements are placed on the actuator architecture, and complexity and mass are often increased. For more information on the role of actuator redundancy in control system architecture design, see the Delta Clipper program [59] and the following Martian lander case study [60].

Unfortunately, it is difficult to compare all possible actuator architectures which utilize the aforementioned coupling, but we can parameterize the key variables in the design space to create a basis for a case-by-case comparison. For example, consider the new basic design requirements: linear impulse range specified by the trajectory, maximum and minimum angular acceleration, and controllability of roll and pitch. Assuming that all engines are mounted on a plane parallel to the plane formed by the landing legs, we choose our design parameters to be number of engines, fixed or gimbaled, and the placement of each actuator with respect to the vehicle center of mass. Engine number must be a nonzero integer; engine thrust must be the collective thrust specified by the trajectory and divided by the engine number; and collective engine thrust must go through the vehicle center of mass. Note that pulsed engines are considered identical to throttleable engines because they are assumed to be able to control total impulse within a reasonable time period. An example comparison is shown for a single TVC engine (Figure 3-3), three differential-impulse engines (Figure 3-4), and four differential-impulse engines (Figure 3-5). The parametric specifications mean that all three cases are equal from a nominal controllability perspective.

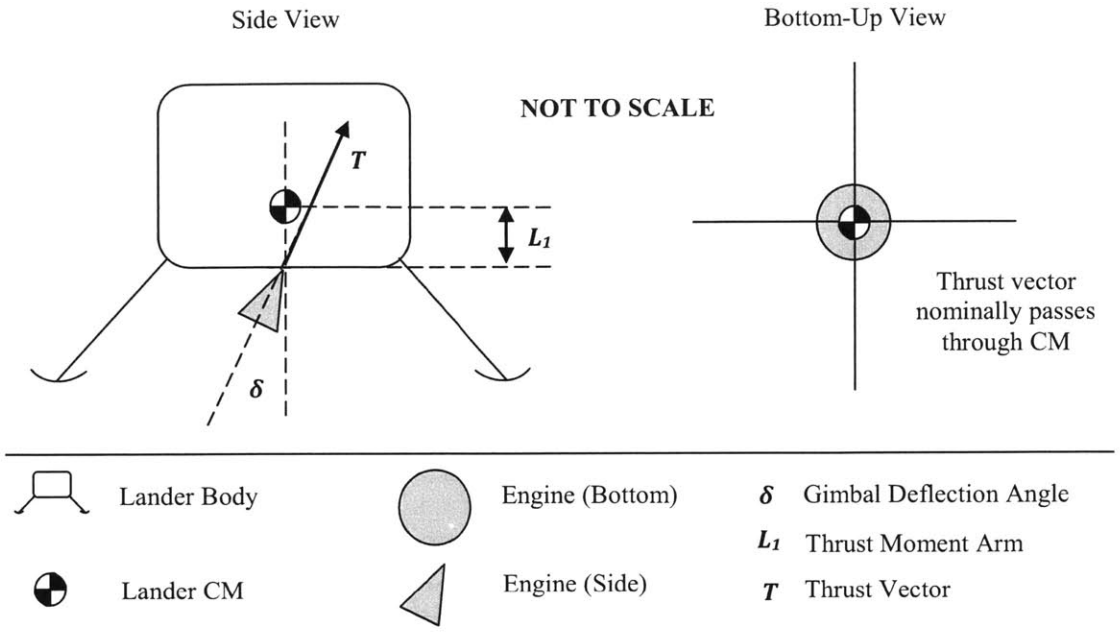


Figure 3-3: Example architecture 1 – single engine with thrust vectoring

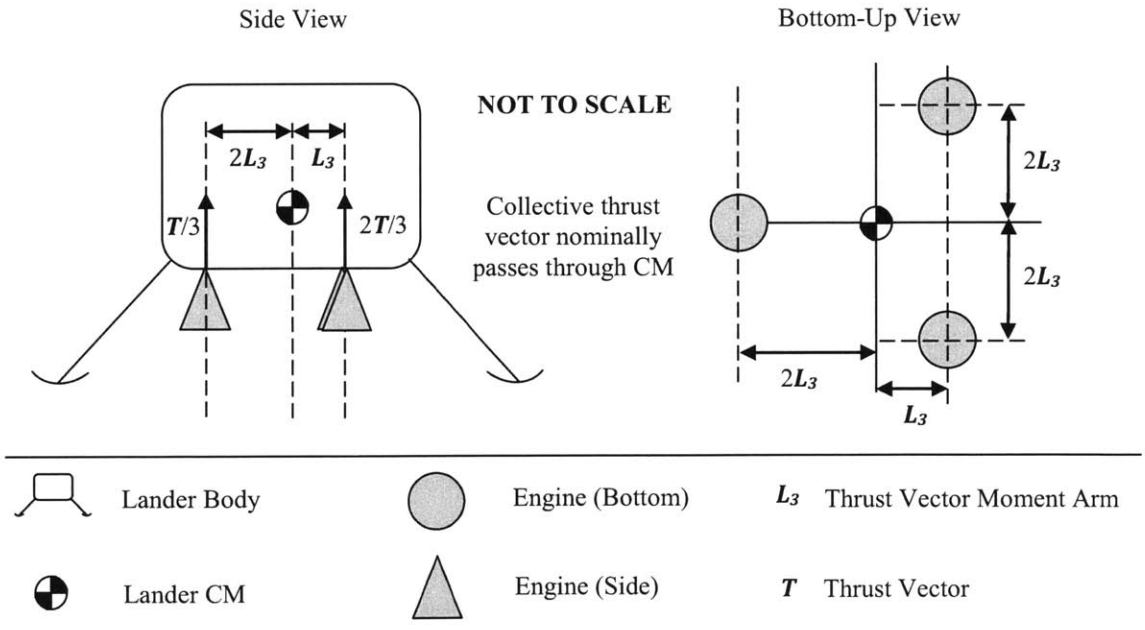


Figure 3-4: Example architecture 2 – three engines with differential impulse

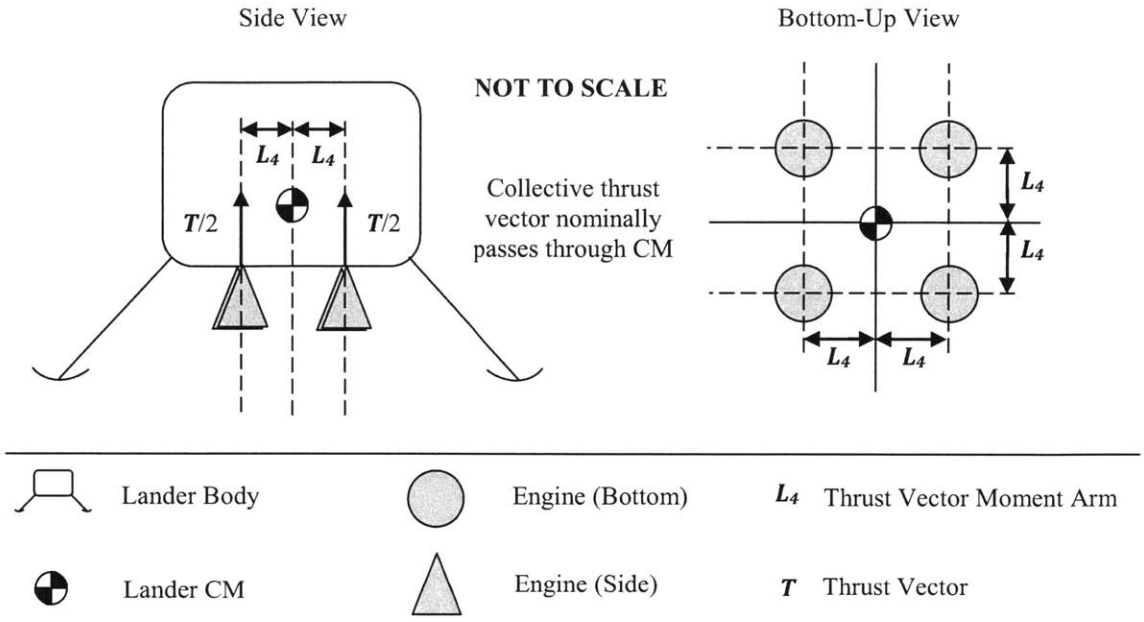


Figure 3-5: Example architecture 3 – four engines with differential impulse

If the total required nominal thrust as specified by the mission trajectory is $\boldsymbol{T}$, then the single engine in architecture 1 must produce $\boldsymbol{T}$, each engine in architecture 2 must produce $\boldsymbol{T}/3$, and each engine in architecture 3 must produce $\boldsymbol{T}/4$. The engine placements as specified in the previous figures ensure that the collective thrust vector passes through the vehicle center of mass when all engines are at the same throttle level (or zero gimbal angle). Simple moment equations can now be derived for each architecture as a function of gimbal angle ($\boldsymbol{\delta}$), total thrust ($\boldsymbol{T}$), change in thrust for each engine ($\boldsymbol{\Delta T}$), and moment arms ($\boldsymbol{L_x}$). The mathematical expression which now connects the three architectures is shown in Equation 3-1.

$$\mathbf{TL_1 \sin \delta_{max} = 4\Delta T_{max} L_3 = 4\Delta T_{max} L_4} \qquad \mathbf{3\text{-}1}$$

If this equation is satisfied, the three architectures have identical control power limits. Notice that the control power in the gimbaled architecture is a function of gimbal angle and total thrust, while the control power in the differential-impulse architecture is a function of throttleability. As indicated before, this process can be extended to any similar architecture, allowing the designer to investigate the basic tradeoffs of different actuator placements without requiring explicit knowledge of the control algorithms.

3.3 Summary

This research shows that no clear 'winner' emerges from the tradespace of potential actuator architectures. However, four conclusions can be drawn:

- The vehicle must have direct controllability of three degrees of rotational freedom and one degree of translational freedom in the body frame during landing (the other two degrees of translational freedom can be indirectly controlled by changing vehicle attitude)
- Linear and angular maneuvers tend to have several orders of magnitude difference in required impulse levels, resulting in a natural decoupling between actuator systems
- Some coupling between actuator systems may be beneficial and should be considered
- There are an infinite number of actuator architectures which satisfy the aforementioned requirements; parametric modeling and common sense can help a designer choose initial designs

4 Control Algorithms and Algorithm Architectures

Control algorithms and algorithm architectures are the second primary topic of investigation within this thesis. When combined with a chosen actuator architecture (Chapter 3), a full control system architecture is realized. In terms of the standard feedback loop block structure, control algorithm design explicitly deals with the design and verification of the logical (typically mathematical) expression which transforms the estimated state error signal into desired actuation commands. As the algorithm design is highly dependent on spacecraft, actuator, disturbance, and estimator characteristics, these will also be discussed in various levels of detail.

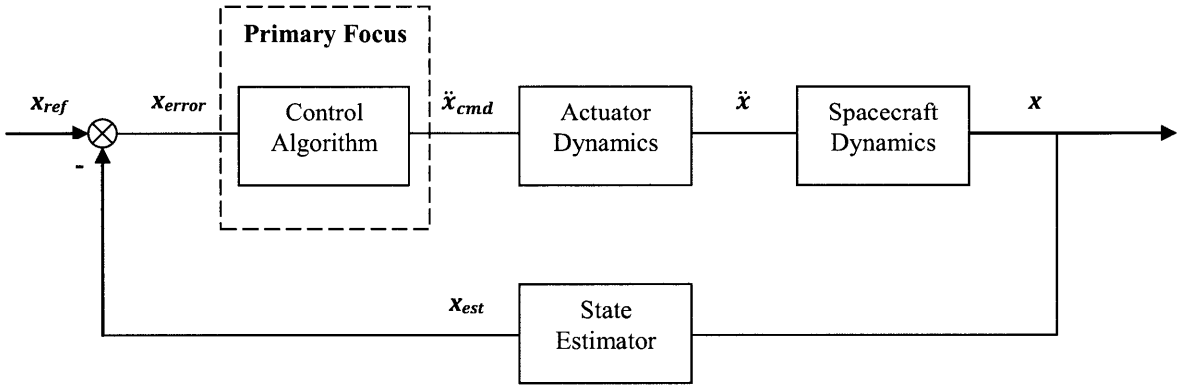


Figure 4-1: Standard feedback structure with relevant block indicated

Figure 4-1 shows the standard feedback structure and highlights the Control Algorithm block as the primary focus. In this case, x refers to a generic state (e.g. angle) and $\ddot{x}$ refers to the acceleration of that state. This chapter will outline common mathematical spacecraft and actuator models, present various types of control algorithms and architectures, discuss the importance of algorithm verification, and briefly discuss additional factors which may be of importance to the designer.

It should be noted that this chapter is in no way meant to be an exhaustive presentation of all available control algorithms and design methods; rather, the purpose of this chapter is to present methods and ideas in control algorithm design which have been shown to be successful on hardware and/or in simulation. The ideas and process presented here should be considered adequate for 'first-cut' algorithm architecture designs. For a more complete discussion, please see the following references and corresponding topics: spacecraft dynamics [61]; linear control [62]; nonlinear control [63]; optimal

control and estimation [64] [65]; robust control [66]; and describing functions [67]. The lecture notes from MIT course 16.30/31 are useful [68].

4.1 Nominal Plant Model

It is first necessary to define the mathematical model of the dynamics on which our actuators will be acting. This model should include all 'blocks' which the control signal must pass through before completing a full loop. We will initially assume perfect state knowledge (no estimator) and present basic spacecraft and actuator dynamics models. It is recommended that these models be iteratively refined and augmented as the design matures.

4.1.1 Spacecraft

If we assume that our coordinate system is fixed to the spacecraft, and that the spacecraft is free to rotate about the three orthogonal axes, then the dynamics are governed by the following equation:

$$\sum \vec{M} = I\dot{\vec{\omega}} + \vec{\omega} \times (I\vec{\omega}) \qquad \textbf{4-1}$$

In this case, $\vec{\omega}$ represents angular velocity, $\vec{M}$ represents moments, and I is a matrix containing the moments of inertia. Expanding this equation to show the scalar components yields:

$$\sum M_x = I_{xx}\dot{\omega}_x - (I_{yy} - I_{zz})\omega_y\omega_z - I_{xy}(\dot{\omega}_y - \omega_x\omega_z) - I_{yz}(\omega_y^2 - \omega_z^2) - I_{zx}(\dot{\omega}_z - \omega_x\omega_y) \qquad \textbf{4-2}$$

$$\sum M_y = I_{yy}\dot{\omega}_y - (I_{zz} - I_{xx})\omega_x\omega_z - I_{zy}(\dot{\omega}_z - \omega_x\omega_y) - I_{xz}(\omega_z^2 - \omega_x^2) - I_{yx}(\dot{\omega}_x - \omega_z\omega_y) \qquad \textbf{4-3}$$

$$\sum M_z = I_{zz}\dot{\omega}_z - (I_{xx} - I_{yy})\omega_y\omega_x - I_{zx}(\dot{\omega}_x - \omega_y\omega_z) - I_{yx}(\omega_x^2 - \omega_y^2) - I_{yz}(\dot{\omega}_y - \omega_z\omega_x) \qquad \textbf{4-4}$$

One will note that there is significant coupling between the three degrees of freedom, but if we are able to assume that angular velocities and accelerations are low, and that the off-diagonal moments of inertia are small, the majority of the right hand side of the equation(s) becomes negligible. These assumptions make analysis simpler (as the resulting equations are linear) but are not always valid.

4.1.2 Actuators

Generic actuator thrust models are typically characterized by a linear first or second order system in which the natural frequency of the poles tend to be inversely proportional to thrust magnitude (meaning that larger engines will have 'slower' dynamics). Gimbal dynamics are typically modeled as linear second order systems. Gimbaled actuators are therefore modeled as third or fourth order systems. As it is common to consider dynamics as signals in the frequency domain, the following equations will be expressed as transfer functions where $\boldsymbol{x}$ represents a generic signal (e.g. gimbal angle or thrust). Recall that transfer functions placed in series can be multiplied; therefore, a first order thrust model can be combined with a second order gimbal model to create a third order actuator model.

$$\frac{\boldsymbol{x}}{\boldsymbol{x_{cmd}}} = \frac{\mathbf{1}}{\boldsymbol{s\tau + 1}} \tag{4-5}$$

$$\frac{\boldsymbol{x}}{\boldsymbol{x_{cmd}}} = \frac{\boldsymbol{\omega_n^2}}{\boldsymbol{s^2 + 2\zeta\omega_n s + \omega_n^2}} \tag{4-6}$$

Here $\boldsymbol{\tau}$ represents the time constant of the first order system, $\boldsymbol{\omega_n}$ represents the natural frequency of the poles in the second order system, and $\boldsymbol{\zeta}$ represents the damping factor (0-1) of the second order system.

4.2 Control Algorithm Architectures

While it is typically relatively simple to stabilize an ordinary differential equation, real-world systems often deviate from the mathematical equations by which we choose to model them. As a result, control algorithm design can be just as much about understanding the implications of the simplifications used to generate the mathematical model as it is about choosing feedback gains; as a result, understanding the nature of parametric and dynamic uncertainty, sources of noise, and the disturbance environment is crucial in algorithm architecture design. No single algorithm architecture or algorithm type is effective in controlling all systems; rather, it is useful to gain proficiency in a variety of algorithm types. We can think of these different algorithm types/architectures (e.g. Linear Quadratic Regulator, Lead/Lag, Phase Plane) as 'tools' which are useful in solving certain types of control problems. Collectively, these tools form a versatile 'toolset' which can be used to solve a wide variety of control problems. This section will present several tools which have been shown to be effective in the soft lunar landing problem.

4.2.1 *Feedback Structures*

Feedback lies at the heart of most control architectures. It is based on the notion that all real system states will incur errors over time due to modeling uncertainty, sensor errors, external disturbances, and numerous other factors. Feedback requires the dynamic states of the system to be observed real-time; the feedback signal is then subtracted from the desired state(s) and used to calculate the required control input to return the system to the desired state(s). Design and analysis is generally performed under the assumption of linear time invariance (LTI) when possible.

The Classical Paradigm

The classical paradigm for control algorithm design typically involves single signal 'error feedback.' This means that the output of the control algorithm is a function of the difference between a measured variable and a separate reference command. It is particularly useful for design and analysis of single-input-single-output (SISO) systems in the frequency domain. Typical control algorithms used with the classical paradigm are Lead/Lag and PID.

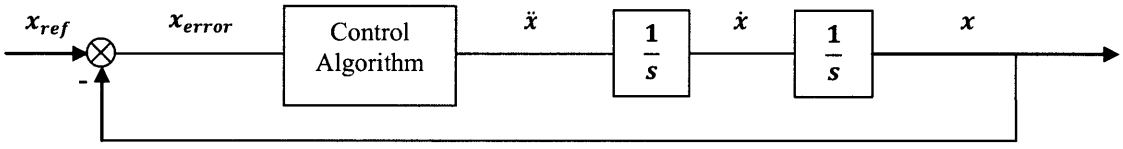


Figure 4-2: Example error feedback control architecture for double integrator system

This approach also easily accommodates multiple nested loops; this gives the control algorithm designer more flexibility and can even allow outer loops to be designed without complete knowledge of the properties of the inner loop as long as bandwidths are reasonably separated [62]. Now the designer can use multiple algorithms to control a system. However, this does introduce more complexity into the design and analysis process.

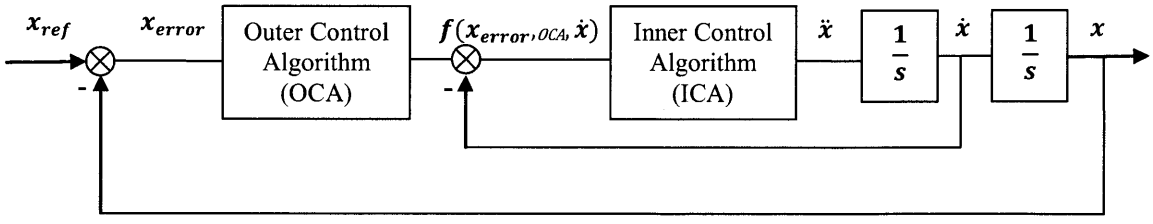


Figure 4-3: Example error feedback control architecture with multiple loops

The Modern Paradigm

The modern paradigm typically involves design and analysis using matrices and linear algebra; as a result, these architectures are more suited for multi-input-multi-output (MIMO) systems. Systems are typically expressed in the following format:

$$\dot{\vec{x}} = A\vec{x} + B\vec{u} \quad \text{4-7}$$

$$\vec{y} = C\vec{x} \quad \text{4-8}$$

In this case $\boldsymbol{x}$ represents the state vector, $\boldsymbol{u}$ represents the control signal, $\boldsymbol{A}$ represents the system dynamics, $\boldsymbol{B}$ represents the actuator(s), $\boldsymbol{y}$ represents the measured state(s), and $\boldsymbol{C}$ represents the sensor(s).

Now the control signal is simply a gain matrix ($\boldsymbol{K}$) multiplied by the state vector. Often referred to as full state feedback (FSFB), this allows the designer to essentially 'weight' the importance of controlling individual states. If the system happens to be LTI, the designer can now use the gain matrix to precisely place the eigenvalues of the system and therefore explicitly specify the behavior of the closed loop dynamics.

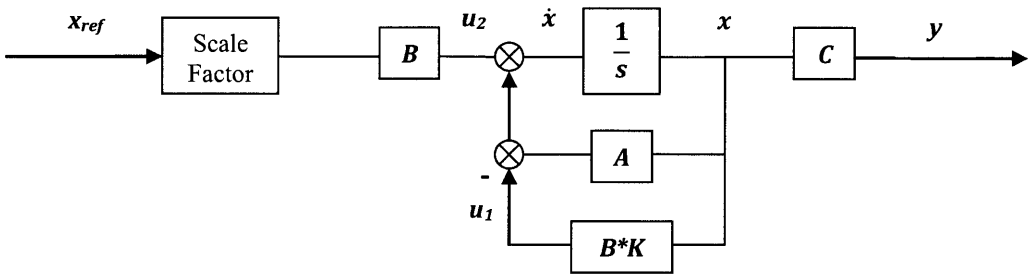


Figure 4-4: Example state feedback control architecture

4.2.2 Specific Algorithms

Listed below are several types of algorithms which are useful when addressing the challenges associated with lunar lander control system architecture design. Please see the references listed at the beginning of the chapter for more details.

Proportional, Integral, and Derivative

This type of algorithm, often referred to as PID, can be implemented in single loop, multiple loop, or state feedback architectures. The three letters in the name denote gains which are multiplied by the corresponding state dynamics to produce a feedback control signal. This type of algorithm is simple, intuitive, tunable, and effective on many linear systems. Unfortunately, the algorithm can be sensitive to noise and tends to be less effective in controlling systems with higher order dynamics or nonlinearities.

Lead/Lag

Lead/Lag is a more flexible version of PID: a 'Lead' is similar to a derivative term which is gain- and phase-limited at higher frequencies; a 'Lag' is similar to an integral term which can be adjusted to balance steady state error and disturbance rejection time. These algorithms heavily favor frequency domain design and analysis, and they give the designer greater control over the system's frequency domain characteristics. While this approach can be highly effective, it has the reputation of being less intuitive, difficult to tune, and requiring extensive experience to master.

Pole Placement

Pole placement refers to the designer's ability to relocate the eigenvalues of the LTI closed loop dynamics to any specified position (mathematically, at least) using FSFB. This approach is simple, intuitive, and highly attractive from a mathematical viewpoint. However, the possibility of control signal saturation must be thoroughly accounted for, and direct state availability is often an issue. The nature of this algorithm also means that it is difficult to form a single 'open loop' transfer function to express the multi-input-multi-output (MIMO) system, meaning classical robustness measures are difficult to evaluate (e.g. gain margin, phase margin).

Linear Quadratic Regulator

The linear quadratic regulator (LQR) approach is a popular option because it applies optimal control to FSFB. It allows the designer to simply specify weights for the state variables and for control effort; given that the system is LTI and controllable (in the mathematical sense), the algorithm guarantees a stable closed loop system. Commonly available computer programs simplify the optimization calculations, allowing the designer to focus on balancing the importance between state variables and control effort rather than on the math involved. However, state availability and robustness evaluation remain

challenges. For clarification, the optimization cost function $\boldsymbol{J}$ is shown below; $\boldsymbol{Q}$ denotes the penalty associated with the state vector, and $\boldsymbol{R}$ is the penalty associated with control effort. Note that the control signal is $\boldsymbol{-BKx}$, which is the same as in the pole placement approach.

$$\boldsymbol{J = \int_0^\infty [x(t)^T Q x(t) + u(t)^T R u(t)] dt} \qquad \textbf{4-9}$$

Linear Quadratic Gaussian

The linear quadratic Gaussian (LQG) approach combines LQR with a linear quadratic estimator (LQE) designed to estimate the full state vector with only partial state knowledge. Assuming full controllability and observability (in the mathematical sense), this approach also guarantees stable closed loop dynamics for an LTI system. As stability and control of real-world systems is often affected by sensor errors and state knowledge, LQG has been shown to be remarkably robust to both sensor and process noise. The LQG controller can also be conveniently expressed as a single transfer function, which allows classical robustness evaluation techniques to be easily incorporated into the design. However, this approach is only mathematically optimal for LTI systems with unbiased, white, Gaussian, uncorrelated sensor/process noise; time-varying and nonlinear system properties void the guarantee of stability. These controllers are also generally of higher order and are therefore prone to pole-zero cancellations. Those unfamiliar with the process may also find the underlying math and theory complex and difficult to understand. An excellent overview of the advantages of LQG controllers can be found in [69].

Phase Plane

Phase plane controllers are commonly used when actuators are unable to apply a variable control signal (such as for fixed, nonthrottleable thrusters). The control signal is therefore given the value of ±1 or 0, and is typically a function of a state and its derivative (e.g. angle and angular rate). The designer is then responsible for dividing the relevant state space (or phase plane) into two regions: an upper region where the control signal is -1, and a lower region where the control signal is +1. Assuming 'perfect' actuators, the state trajectory will converge to the origin of the phase plane in a finite amount of time. To account for real-world characteristics (e.g. time delays, actuator dynamics) a third region is often added between the upper and lower region where the control signal is 0. This 'deadband' adds robustness but also results in a sustained stable state oscillation in the vicinity of the phase plane origin. An example phase plane controller is shown in Figure 4-5.

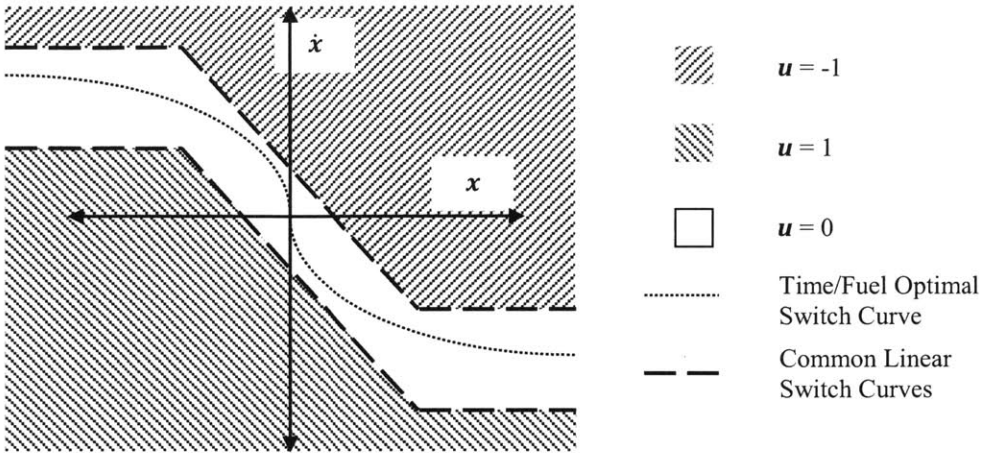


Figure 4-5: Example phase plane controller

As these controllers are well suited for bang-bang actuators, they have been used on numerous spacecraft. They are very simple, intuitive, and robust to modeling errors, but they can be very difficult to analyze.

Summary

Each one of the aforementioned algorithms is useful for solving different types of problems. PID is simple to design/analyze and has been widely used in industry on low order systems; lead/Lag networks offer more control of the open loop frequency domain characteristics; pole placement and LQR make gain selection simple for higher order systems when the full system state is available; LQG offers optimality characteristics which make it remarkably robust to noise; and phase plane controllers can be specifically designed for signed binary actuators. Ultimately, it is up to the designer to select which algorithm is best suited to a particular problem. This process will be demonstrated in Chapter 5.

4.2.3 Algorithm Verification

The previous section outlined various types of algorithms and briefly discussed advantages and disadvantages of each approach. This section will discuss the second important aspect of control architecture design: verifying that the selected algorithm will meet stability and performance criteria on the actual system. Because spacecraft rarely undergo modifications after launch, the control algorithm designer must be certain that the selected algorithm(s) will 'work.' If the designer is fortunate enough to have access to the working spacecraft and the necessary testing equipment during the algorithm design

phase, he or she can use hardware testing methods to select and tune an algorithm. However, as these opportunities are not common, we will discuss analytical and simulation-based verification methods.

As in selecting control algorithm architectures, there is no single test which guarantees performance on the actual spacecraft. Instead, it is useful to be proficient in a variety of mathematical and experience-based testing 'tools;' these can be applied collectively to different problems to help alert the designer to sensitivities which may adversely affect performance. Whenever possible, multiple methods should be applied to any given problem.

Analytical Methods

System performance can often be expressed in closed form mathematical solutions which require no simulation or complex computation to understand. These are an excellent 'starting place' in algorithm verification and can also be used as a 'sanity check' further in the design process as models become more complicated.

Linear Systems

If it can be shown that it is reasonable to approximate a system with linear expressions, there are a wide variety of tools which allow a designer to confidently assess stability and robustness. The open loop transfer function (typically the product of the controller and plant models) can be used to generate the Root Locus, Bode, and Nyquist plots. The Root Locus displays the order of the system and provides clues as to the progression of the closed loop poles as a function of DC gain. The Bode plot is useful for evaluating general response characteristics (e.g. bandwidth, DC gain) in the frequency domain and for evaluating robustness in terms of gain and phase margin. The Nyquist plot displays similar information as the Bode plot, but it provides the only true test for stability of closed loop system as it includes right-hand-plane dynamics. If the closed loop transfer function is available, one can use the Root Locus to show the location of the closed loop system poles and a Bode plot to show general frequency domain characteristics for both the sensitivity (r to y) and complimentary sensitivity transfer functions (dy to y).

Nonlinear Systems

It is not uncommon for systems to contain nonlinear characteristics (in fact, it is quite *uncommon* for real-world systems to be perfectly linear). If these nonlinearities have a significant impact on the system's performance, the designer should find a way to analyze/understand the nature of the

dynamics. Ideally, the designer would be able to apply the same set of tools used in the analysis of linear systems to obtain equivalent stability and performance characteristics. Unfortunately, this is not the case, and the designer must rely on a different set of methods.

If the system dynamics can be represented as a function with continuous slope, the designer can use a Lyapunov candidate function to characterize stability. However, there is no known way of automatically generating Lyapunov candidate functions, and the results do not explicitly indicate actual performance characteristics.

If the system dynamics contain discontinuities (e.g. bang-bang actuators), the designer can approximate the discontinuity as a describing function, which is a function of the amplitude of the input signal. While this has been shown to be effective for simple systems/nonlinearities, it is much more difficult to apply to complex systems with multiple nested feedback loops.

Simulation

Simulation is a useful tool at all stages of the design and analysis process. In its simplest form, it can be used to verify analytical results; in more complicated forms it can help answer the 'what if...' questions about performance for time-varying nonlinear systems. Often these are incorporated into Monte Carlo simulations. As computational power during the design phase is typically widely available (along with numerous easy-to-use simulation environments), it is now rare for a designer not to utilize simulation capability. However, one should be very careful to thoroughly verify the accuracy of the simulation (or at least understand its limitations) before drawing conclusions from the results.

4.2.4 Other Factors to Consider

Control system architecture design for a well developed mission/vehicle will inevitably require myriad other considerations in addition to those presented in this chapter. Some of these can be easily anticipated (e.g. fuel slosh, time-varying mass properties) while others may not become evident until later in the design process (e.g. actuator performance variation due to propellant saturation with dissolved Helium [6]). Anticipating such additional considerations (and knowing how to modify the design to account for them) is one of the most important jobs of a control system architect. The purpose of this section is to present common challenges not covered in the nominal design case.

General Uncertainty

Examining the general types of uncertainty is helpful in understanding how the uncertainty will impact system dynamics and how the controls engineer can design/redesign his or her algorithm. Uncertainty (which we can think of as modeling errors) tends to manifest itself in four different ways: incorrectly chosen DC gain of the open loop system (e.g. actuator thrust magnitude, mass, moment of inertia), incorrectly modeled dynamics (e.g. damping ratio), unmodeled dynamics (e.g. higher order terms), and stochastic disturbances (e.g. process noise, sensor noise). Anticipation and analysis of an incorrectly chosen open loop DC gain is relatively straightforward, particularly in the classical feedback architecture; the Root Locus can be used to understand trends in system dynamics as a function of the proportional feedback gain. Incorrectly modeled dynamics are somewhat more difficult to account for; it is the responsibility of the designer to approximate reasonable variations in existing dynamics and corresponding coefficients. Specific attention must be given to lightly damped and unstable dynamics (e.g. structural bending, fuel slosh) to reduce their impact on performance sensitivity. Unmodeled dynamics are arguably the most difficult to anticipate and are often not discovered until hardware tests are conducted. Instances are best predicted by examining previous attempts/results, and problems are generally solved on a case-by-case basis. Lastly, the effects of stochastic disturbances (which are typically sensor-, actuator-, and environment-specific) are crucial in predicting actual hardware performance. As these are more widely studied in the navigation context, they will only be briefly addressed in this thesis.

Time-Varying Properties

Most landing profiles require that a vehicle consume a total propellant weight equal to the vehicle's dry weight during descent and landing. As a result, mass, moments of inertia, and center of mass location can be expected to change significantly. These effects alone are not difficult to account for, as they generally equate to changing the DC gain of the open loop system. However, coupling with more complex dynamics such as fuel slosh and bias moments makes it difficult to form closed loop solutions for the entire flight profile. If in doubt, discretize the landing profile and utilize some form of gain scheduling.

State Knowledge Availability/Quality and Sensor Placement

State knowledge availability and quality are of key importance to the control system architect. At a bare minimum the lander must have reasonable knowledge of lunar position, which can be used to generate all other states; however, it is customary to provide direct knowledge of multiple other states to reduce estimation error covariance. Most landers are equipped with sensors which provide direct information on position, velocity, acceleration, angular velocity, and altitude relative to the lunar surface, but the quality of these measurements are highly dependent on sensor selection and sensing methodology. Recalling that high bandwidth control requires low-noise, high-accuracy state knowledge, the control system engineer will almost always prefer 'better' sensors and the capability to measure more states.

Sensor placement should also not be overlooked. This stems from the fact that most sensors are body-mounted, but few sensors are mounted exactly at the vehicle's center of mass. As each point on the body experiences slightly different accelerations, point dynamics must be included in the sensing equations. An excellent example is provided in [62] on page 497 which shows how an inertial measurement unit (IMU) mounted on a flexible structure decreases overall performance margins and stability; the problem is solved by simply repositioning the IMU to a more stable location. If this is not an option available to the control system architect, he or she must typically lower system bandwidth and/or design and implement a notch filter. This type of problem can be very challenging, especially if high bandwidth is required and/or if the resonant characteristics are time-varying and difficult to predict. Therefore, designers should take great care to ensure that unnecessary (avoidable) resonances are not included in the nominal system.

Time Delays and Digitalization of Signals

Time delays are one of the most common destabilizing factors of a closed loop system. They have no effect on frequency domain magnitude but do induce a phase lag. Fortunately, delays can be modeled in the frequency domain with a Pade approximation [62]. The output signal is therefore shifted in phase proportional to frequency (ω) multiplied by the time delay (T_d). As we expect, time delays induce more phase lag for high frequency signals.

$$\angle G_{delay}(j\omega) = -\omega T_d \qquad \textbf{4-10}$$

As digitalization of signals is also fairly common (e.g. sensor signals, actuator command signals), the control system architect must find a way to account for digitization effects. One approach is to design and analyze the control system in the digital 'z' domain. However, it is often reasonable to simply use continuous dynamics and approximate the digital signal (assuming zero-order-hold) as an equivalent time delay approximately equal to the digital period; experience has shown this to be an effective approach for systems with relatively high sampling and control rates.

Propellant Slosh

Propellant dynamics will be one of the most difficult factors to master because it is a function of multiple fixed and changing variables (e.g. tank geometry, fuel level, linear and angular acceleration) and because the control system architect often has little direct control over the 'slosh' dynamics. While this thesis will analyze the effects of a simple mass-spring analogy, more in-depth discussion and modeling methods can be found in [70] and [71].

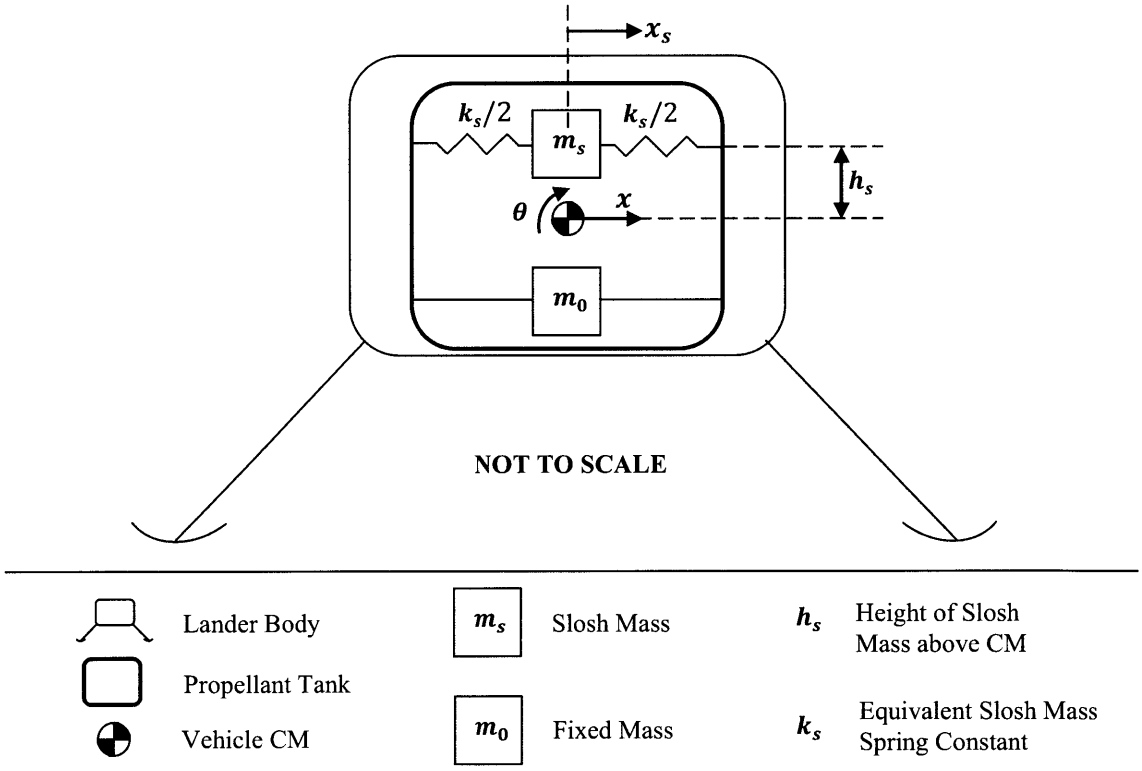


Figure 4-6: Generic 'single slosh mass' model

The basic single-tank slosh model used in this thesis is shown in Figure 4-6. It approximates the fluid as two masses: one fixed to the bottom of the tank, and one free to oscillate side-to-side (in two degrees of

freedom). This is a popular modeling technique because mass and frequency of the slosh mass (which are functions of up/downward acceleration and fill level) can be found using simple experimental procedures. However, this model does make several broad assumptions:

- The slosh mass does not change height, and vertical acceleration does not change appreciatively (pendulum modeling method may be preferred if these are concerns)
- The slosh mass is characterized by a single dominant (lowest) frequency; higher slosh frequencies are well above controller bandwidth and are ignored
- The fixed mass does not move and only contributes to overall moment of inertia
- Neither the slosh mass nor the fixed mass rotates with respect to the vehicle body

If these are found to be reasonable assumptions then the dynamics of the slosh mass can be modeled by the following equation:

$$\ddot{x}_s = (x - x_s + \theta h_s)k_s/m_s + g\theta \qquad \textbf{4-11}$$

In this case θ represents vehicle rotation about its CM, x represents translation of the vehicle CM, x_s represents translation of the slosh mass, m represents the vehicle mass, and g represents the equivalent acceleration seen by the vehicle in the up-down direction. Notice that slosh mass acceleration is a function of both angular and linear parameters, and that h_s may change sign as propellant depletes. The transfer function from u to θ can now be derived:

$$\frac{\theta(s)}{u(s)} = \frac{mm_s s^2 + k_s(m + m_s)}{s^2\left(mm_s I s^2 + Ik_s m + Ik_s m_s + gh_s mm_s^2 + h_s^2 k_s mm_s\right)} \qquad \textbf{4-12}$$

One will now notice that the resultant transfer function contains a double integrator (from the rigid body) and a pair of purely oscillatory poles and zeros. As some damping inside the tank will exist (due to wall friction, baffles, etc.), it is customary to include low ζ values in the equation, which pulls the poles and zeros slightly into the left hand plane. The designer will also generally prefer the poles to have a higher natural frequency than the zeros to reduce phase lag in a system which already begins with (negative) 180deg at zero rad/sec. If the zeros adequately contain the two poles throughout the duration of the landing profile, the slosh dynamics can generally be left alone. However, if the poles are not contained, the designer may wish to consider gain-scheduled notch filters. Note that a similar analysis example exists in [61] for a single-engine TVC system with a pendulum slosh mass

approximation, and that an example characterization of Altair's slosh dynamics can be found in [72]. For simulation purposes, we will assume a nominal ζ of 0.015 and slosh saturation limits of ±15.0cm.

4.3 Summary

In this chapter we elaborated on the specifics of the second primary focus of the designer: the algorithm architecture. Various generic models of spacecraft and actuators were presented, along with specific control algorithms and verification methods. As no single model or approach was regarded as 'best,' the reader was encouraged to view the options as 'tools' which may be better suited for different types of problems. Lastly, additional factors were mentioned that are likely to affect controller design. The following chapter will use three carefully chosen case studies to demonstrate how these different approaches might be applied.

5 Case Studies

In this chapter we will use three carefully selected examples of actuator and control algorithm architectures to demonstrate the importance of the control system architecture paradigm. These case studies are meant to serve as concrete examples of the ideas presented in the previous chapters and were selected to demonstrate both the subtle and the obvious relationships between the actuators and controllers in real-world systems. All case studies are performed for the terminal descent stage of a generic 350kg lunar lander as was presented in Chapter 1 and Figure 1-3. As a reminder, this thesis uses 'aircraft' coordinates for the lunar surface and lander coordinate frames, and the terms 'roll,' 'pitch,' and 'yaw' refer to rotations about the X, Y, and Z lander frame axes, respectively.

5.1 Preliminary Discussion

Prior to the presentation of the three case studies, it will be helpful to elaborate upon the specifics of the terminal descent phase of the lunar lander problem and provide a background for the selection of trajectory, actuator, and algorithm requirements. In general, this section can be thought of as a primer to the challenges that will be faced during the actuator and control architecture design process as well as a presentation of common sense functionality/performance metrics that should be met by all lunar lander control system architecture designs (not just these three case studies). This will allow us to reserve the specifics of the actuator and algorithm architecture design for the individual Case Studies sections.

5.1.1 Terminal Descent Trajectory

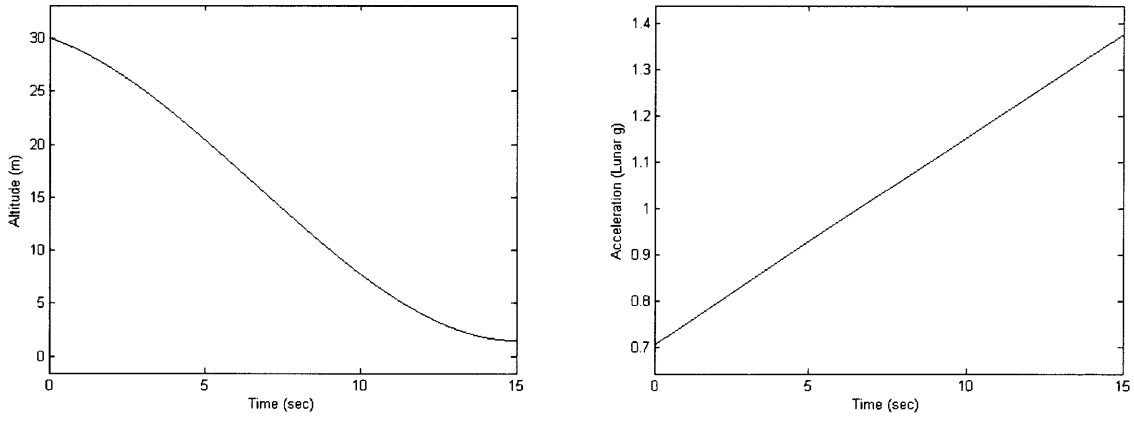
As stated in the earlier chapters of this thesis, the particular portion of the soft lunar landing problem on which we have chosen to focus is the final terminal descent to the lunar surface. We will assume that the lander has successfully completed the previous portions of its mission and has arrived 30.0m above the lunar surface with its thrust vector nearly perpendicular to the landing plane, all acceleration and horizontal velocity effectively nulled, and with a vertical descent velocity of 1.0m/sec in the body coordinate frame. The lander is assumed to have perfect knowledge of basic states (e.g. position, velocity), enough propellant for 45sec of hovering flight, and a control system architecture responsible for regulating three dimensional position in the lunar frame. We will therefore generate an 'open loop' descent trajectory as a function of time, which will serve as the reference input to the position 'regulation' controllers.

Let us begin by noting that trajectory generation typically falls in the realm of guidance algorithms within the GNC architecture, and, as guidance algorithms are not the focus of this thesis, this process should not be more complicated than it needs to be. In fact, we have already assumed that lateral body velocity and acceleration have been reduced to zero at the end of the previous trajectory phase; as this is also a desired landing condition we do not need to impose any additional changes to these states. The only three states in the trajectory which must be considered are altitude, descent velocity, and descent acceleration, and we can generate the following guidelines to help shape the nominal flight path. Note that engine cutoff should occur at the nonzero final altitude to reduce adverse interaction with the lunar surface.

- Boundary conditions: $\boldsymbol{Z}(0) = 30.0$, $\dot{\boldsymbol{Z}}(0) = -1.0$, $\boldsymbol{Z}(\boldsymbol{t_f}) = 1.5$, $\dot{\boldsymbol{Z}}(\boldsymbol{t_f}) = 0.0$
- Continuous acceleration (to account for actuation dynamics)
- No 'overshoot' in $\boldsymbol{Z}$
- No net acceleration less than 0.7 lunar g's or above 5.0 lunar g's (to account for throttle range)
- Reduce propellant

These guidelines naturally lend themselves to a trajectory for which altitude is a cubic function of time and can be adjusted to satisfy a final time ($\boldsymbol{t_f}$) boundary condition. Trial and error indicate that a final time of 15.0sec produces a satisfactory nominal descent trajectory. The cubic function along with the resultant trajectory is shown in Equation 5-1 and Figure 5-1.

$$\boldsymbol{Z = 0.0121t^3 - 0.24t^2 - t + 30} \qquad \textbf{5-1}$$



(a) **(b)**

Figure 5-1: Reference descent trajectory as a function of time; (a) altitude, (b) acceleration

It should be noted that using fixed initial conditions for trajectory generation (in both vertical and lateral axes) is not recommended for an actual mission because it means that any initial state deviations will present themselves as step errors to the control algorithms, which is undesirable for a variety of reasons. A more robust choice is to use the estimated initial conditions at the start of terminal descent to generate a real-time trajectory. However, as we are most interested in the design and analysis of the control system (rather than trajectory generation), we will continue to use the trajectory generated from fixed initial conditions and focus more on relative performance and sensitivity. The specific results presented in the upcoming case studies should be viewed accordingly.

As the first four guidelines listed at the beginning of this section are satisfied by the cubic trajectory, we will now turn our attention to the fifth guideline: conserving propellant. As propellant consumption is partially dependent on mass properties and specific impulse, it is more convenient to examine the problem in terms of total impulse required, which can be calculated using Equation 5-2.

$$\boldsymbol{Impulse} = \int_{t_0}^{t_f} \boldsymbol{F}\boldsymbol{dt} \qquad \textbf{5-2}$$

In this case, $\boldsymbol{F}$ is the magnitude of the force exerted. Intuition suggests that the minimum impulse solution (assuming we do not use the actuators to actively accelerate towards the lunar surface) should also be the minimum time solution and involves a free-fall from the initial altitude followed by an impulsive burn to arrest the spacecraft at the terminal altitude. This can be found using Equation 5-3.

$$\boldsymbol{Impulse_{free\_fall}} = \boldsymbol{m}\sqrt{\boldsymbol{\dot{Z}_0}^2 + \boldsymbol{2gZ_0}} \qquad \textbf{5-3}$$

In this case, $\boldsymbol{m}$ represents vehicle mass (assumed constant), $\boldsymbol{g}$ represents lunar gravity (also assumed constant), and $\boldsymbol{Z_0}$ represents distance of descent. The total impulse required for the suggested nominal trajectory is then given by the integral of the actuation acceleration profile (the second derivative of Equation 5-1) plus lunar gravity and is shown in Equation 5-4.

$$\boldsymbol{Impulse_{cubic\_traj}} = \boldsymbol{m}\int_0^{15} (\boldsymbol{6(0.0121)t - 2(0.24) + g})\boldsymbol{dt} \qquad \textbf{5-4}$$

By comparing the two results for our given scenario we see that the cubic trajectory requires just more than 2.6 times the total impulse of the free-fall trajectory; this seems like reasonable performance given the acceleration constraints imposed in our guidelines. The remainder of this chapter will therefore use the cubic solution as the reference descent trajectory for the terminal descent phase of the lunar lander mission.

5.1.2 Actuator Architecture Requirements

It is now necessary to state the actual requirements for possible actuator architectures. This will help narrow the range of admissible architectures as well as establish a ball-park range for functionality/performance characteristics (e.g. control powers, impulse range). We begin by restating the requirement that the actuator architecture have full controllability over all three translation degrees of freedom (and rotational degrees of freedom) in the lunar coordinate frame. This is formalized by Equation 5-5, which states that full controllability is obtained as long as the rank of $\boldsymbol{Co}$ is equal to the number of rows or columns in the state dynamics matrix $\boldsymbol{A}$. Note that this does not necessarily mean that the actuator architecture has full controllability over all three translational degrees of freedom in the lander coordinate frame, as translation in the lunar frame can be obtained 'pointing' the collective thrust vector via changing vehicle attitude.

$$\boldsymbol{rank(Co) = rank[B \quad AB \quad A^2B \quad \cdots \quad A^{n-1}B] = number\ of\ states} \qquad \textbf{5-5}$$

Next, we set a guideline for effective throttleability of the main descent propulsion system. As this system will, at a minimum, be responsible for initiating the ΔV required for braking and descent, it is desirable that it be able to exhibit a wide range of effective throttleability. Ideally, this range would be infinite (to enable the fuel-optimal free-fall and impulsive-arrest descent trajectory), but historical hardware limitations suggest that this range fall between 50% and 500% of the vehicle's landing weight [47] [46]. Note that this range may also need to be adjusted depending on how effective throttleability is obtained. Engines with throttleable thrust ranges will probably want to avoid multiple restarts and throttling below 10% maximum thrust, while engines which obtain effective throttleability via pulsing will be amenable to multiple restarts but will have 'deadband' regions in the upper- and lower-most throttle ranges (due to finite valve actuation times and effective hysteresis). It is therefore recommended that effective throttle range be selected such that the minimum range be less than the landing weight and the maximum range be greater than the landing weight without violating the

aforementioned constraints and while still allowing for reasonable controllability of any coupled degrees of freedom (e.g. allow adequate headroom for pulsed actuators to also control attitude).

On a similar note, it will also be beneficial to establish guidelines for attitude control power ranges in the form of angular acceleration. Once again we would prefer an infinite and continuous control power range, but hardware does not enable this capability. Historical evidence indicates that a reasonable range for roll and pitch is 0.02 to 1.0rad/sec^2 while a reasonable range for yaw is half an order of magnitude less [40] [55] [56] [57] [58]. This discrepancy is largely due to the tendency of lunar landers to encounter much higher disturbance torques about the roll and pitch axes than the yaw axis. We will therefore include a guideline which requires the roll and pitch actuators to have sufficient control power to null-out the angular acceleration induced by a 5.0cm CM offset from the main descent propulsion system thrust vector at 100% throttle while maintaining adequate additional control power for attitude maneuvers.

Lastly, it will be helpful to define a uniform type of propulsion system and efficiency level for ease of comparison in the case studies. We therefore decide that all actuators will use bipropellant liquid pressure-fed hypergolic fuel/oxidizer with an I_{sp} of 270sec. The overall system is chosen for simplicity and flight-proven success, and the specific propellants are chosen for storability and low-to-mid efficiency characteristics. Changes in specific impulse due to pulsing/throttling will not be included because of their close dependence on actual hardware selection.

5.1.3 Control Algorithm Architecture Requirements

The control algorithm architecture requirements, which should be thought of differently than the control algorithm performance requirements (and will be discussed individually for each case study), deal more with the general selection and characteristics of the feedback paradigm than the determination of specific gains. The following requirements are therefore intentionally subjective in nature.

The first requirement is that the control algorithm architecture have full observability of the lander's actual states. From a hardware perspective this means that the vehicle is equipped with sufficient sensors which are of high enough accuracy and bandwidth that they do not interfere with the design and performance of the control system architecture. From an algorithm perspective, this is known as

'perfect navigation' and also assumes that the rank of $\boldsymbol{Ob}$ is equal to the number of rows or columns of the state dynamics matrix $\boldsymbol{A}$. This final statement is formalized in Equation 5-6.

$$\boldsymbol{rank(Ob) = rank}\begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \boldsymbol{CA^2} \\ \vdots \\ \boldsymbol{CA^{n-1}} \end{bmatrix} = \boldsymbol{number\ of\ states} \qquad \textbf{5-6}$$

Secondly, the control algorithm architecture should not be more complicated than necessary; it should be simple to understand, implement on hardware, and debug. Thirdly, we must be able to make a strong intuitive and mathematical case for stability, and this must be verified by nonlinear simulation. Fourthly, the control algorithm architecture must be robust. This is not necessarily meant in a strict mathematical sense; rather it is meant in the sense that we are primarily concerned with developing a control system architecture that will function adequately on hardware. This means that the architecture must be able to tolerate reasonable variations in spacecraft parameters, process noise, time delays, etc., and that the analysis must support this. Lastly, the control algorithm architecture must reduce propellant usage whenever possible.

5.1.4 Common Simulation Architecture

The common architecture shown in Figure 5-2 displays the layout of the nonlinear simulation which will be used to help verify and compare the performance of the control system architectures in each of the case studies. As each case study requires some modifications to the specific blocks shown in the following figures, one should interpret the following figures as logic-flow outlines rather than as compiled code.

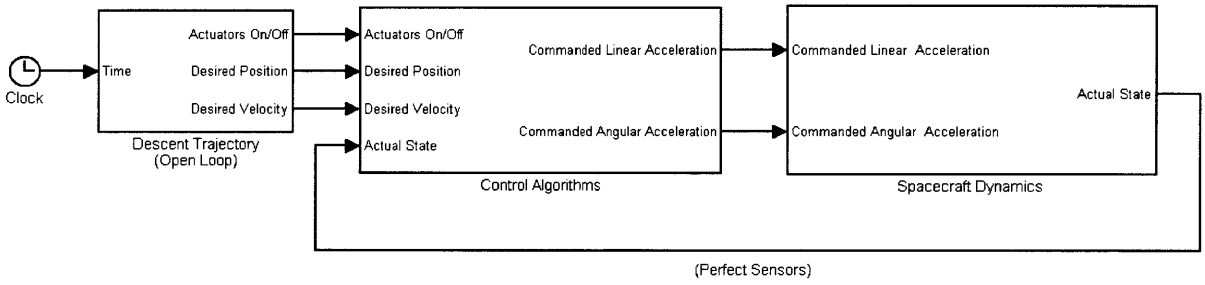


Figure 5-2: Common simulation block diagram

The simulation begins with the initiation of a time signal (dt = 0.001sec), which is used by the Descent Trajectory Block (Figure 5-3) to generate the Boolean engine on/off signal as well as the continuous desired vertical position and velocity as cubic and quadratic functions of time, respectively. These signals are then given to the Control Algorithms Block (Figure 5-4), which compares the actual spacecraft states to the desired states and generates commanded angular and linear acceleration signals in the lander body frame. The commanded accelerations are then passed to the Spacecraft Dynamics Block (Figure 5-5) where they are transformed into body forces and moments via an actuator selection function and an actuator dynamics function. The 6DOF dynamics block then transforms the body forces and moments into vehicle states which can be used by the control algorithms at the next time step. Additional dynamics and forces such as gravity and fuel slosh are implicitly included inside the 6DOF block.

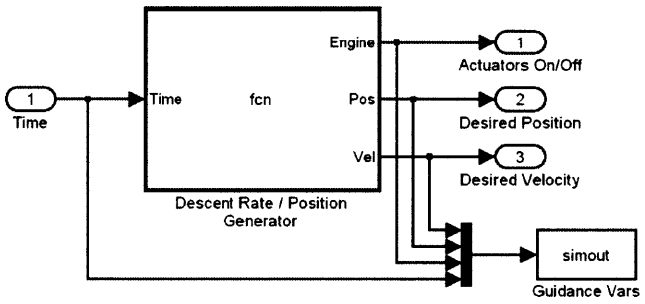


Figure 5-3: Descent trajectory block diagram

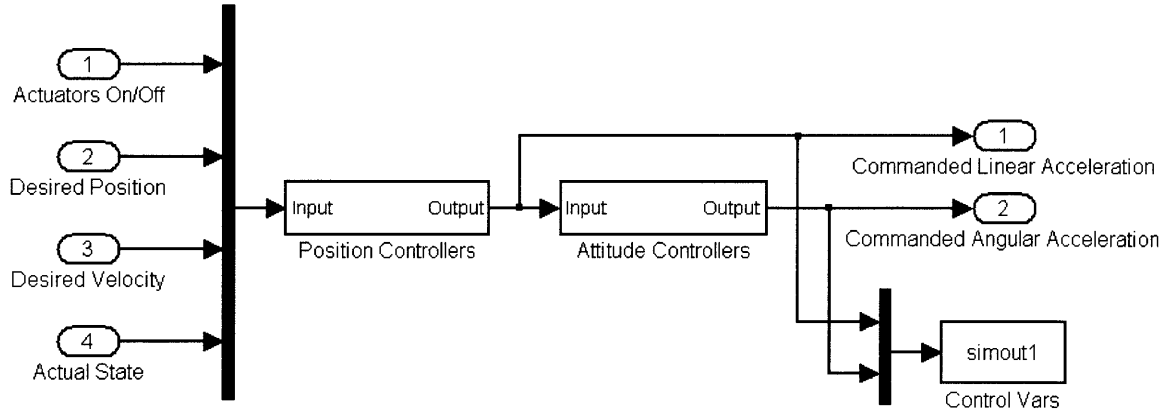


Figure 5-4: Control algorithms block diagram

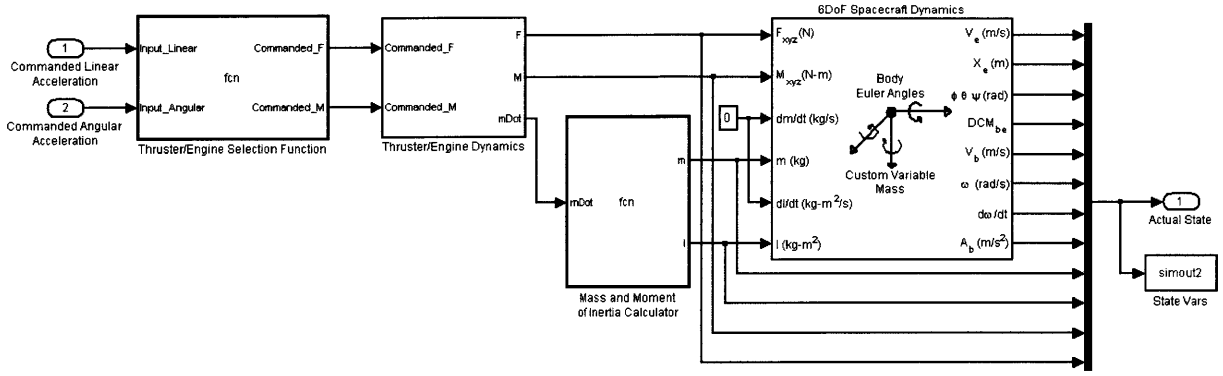


Figure 5-5: Spacecraft dynamics block diagram

As the primary purpose of the common simulation is to verify the performance of the complete nonlinear 6DOF system, it will incorporate many of the factors discussed in previous chapters of this thesis. These factors include linear and nonlinear actuation dynamics, the full nonlinear equations of motion, a single mass-spring propellant slosh model, time delays, time-varying mass and moment of inertia properties, CM offset, and initial condition/parameter variation. Numerical values will be given for each case study. Additional dynamics not explicitly listed here (or in each case study discussion) are not included in the simulations, and results should be interpreted accordingly.

5.2 Case 1: Multiple Fixed Nonthrottleable Thrusters for Attitude Control; Single Fixed Throttleable Engine for Altitude Control

The actuator architecture under consideration in the first case study is motivated by the natural decoupling under nominal conditions that exists between the low angular impulse and the high linear impulse systems as was identified in Chapter 3. This decoupling has the potential to allow the two actuator systems to be designed separately and optimized for their respective performance requirements: the throttleable high-thrust main engine can be designed to provide the linear impulse necessary to control altitude; and the 'bang-bang' low-thrust attitude control system (ACS) thrusters can be designed to control attitude. The baseline actuator architecture is shown in Figure 5-6.

The key advantage of this design is its potential for overall simplicity and lack of complex gimbaling equipment. The throttleable engine should be able to provide precise linear control over a wide throttle range and therefore enable a smooth and efficient descent trajectory, while the fixed-axis ACS thrusters can be selected such that they can control all three rotational DOF's. The only planned coupling occurs in the control of the X and Y position parallel to the lunar surface, where translation is enabled by using the ACS to change vehicle attitude and tilt the main engine thrust vector to achieve translational acceleration in the lunar frame.

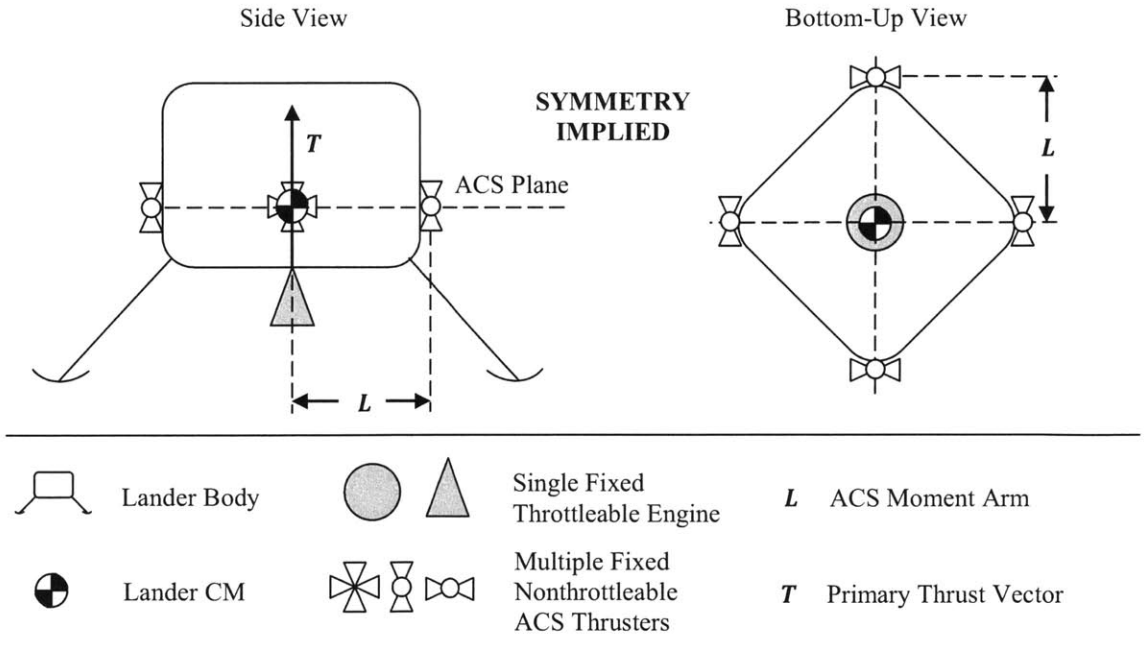


Figure 5-6: Case 1 baseline actuator architecture

5.2.1 Designing the Actuator Architecture

The full actuator architecture includes specific actuator characteristics such as type, thrust levels, dynamic properties, and placement. The following sections will provide this information for the baseline architecture shown in Figure 5-6.

Main Engine Parameters

As the I_{sp} and engine location have already been selected, the main characteristic of interest is the thrust limit(s). Knowing that higher thrust allows for a more efficient descent trajectory (including the portions prior to the terminal descent phase), and that higher thrust reduces the chances of control signal saturation, we choose the upper limit to be approximately three times the vehicle's landing weight. We also assume 'deep throttling' down to 10% of maximum thrust but that multiple restarts are not permitted. This allows the lander to hover at 33% throttle, which provides adequate throttle range for closed loop altitude control. These characteristics, along with several others, are included in Table 5-1. 'Time Constant' refers to the rise time of a first order differential equation used to model the engine (Equation 4-5), and 'Time Delay' refers to the time lag between commanded and actual thrust increase (Equation 4-10).

Table 5-1: Single throttleable main engine characteristics

Characteristic	Value
Maximum Thrust	1700.0N (100% Throttle)
Minimum Thrust	170.0N (10% Throttle)
Time Constant (τ)	0.20sec
Time Delay (T_d)	0.05sec
Thrust Quantization	5.0N

ACS Parameters

The attitude control system must be capable of controlling all three rotational DOF's, and the choice to use fixed-axis thrusters means we should expect multiple symmetric clusters of actuators on the sides and/or edges of the lander. Historical data indicates that control authority should range from 0.02 to 1.0rad/sec^2, but the selected actuators are not capable of directly applying continuously varying thrust. We must therefore choose a single angular acceleration for each DOF, and this decision will involve

compromise. On the one hand, lower thrust enables more precise pointing control and less propellant usage. This is particularly important when we consider that all bang-bang actuators are limited in minimum impulse capability, meaning the spacecraft will never settle to a fixed attitude without experiencing sustained oscillation from thruster firings. This phenomenon is known as a 'limit cycle' and is not uncommon in systems with discontinuous nonlinearities. Smaller thrusters tend to have smaller minimum impulse capabilities and allow for lower frequency and amplitude oscillations. On the other hand, thrusters which are only capable of inducing low angular acceleration severely limit the bandwidth of the control system and reduce disturbance rejection capability. It is therefore necessary to anticipate likely disturbance sources and estimate their magnitudes. Fortunately, the atmosphere-free lunar environment does not impose any significant disturbance forces and torques, and it is reasonable to assume that the majority of disturbances will be induced by other actuators on the vehicle. It now makes sense to recall the requirement stated earlier in Section 5.1.2 that the actuator architecture be capable of handling a 5.0cm CM offset from the main thrust vector at 100% throttle (which is primarily a function of the main throttleable engine). This value of approximately 86N*m vastly outweighs all other anticipated disturbances and becomes the driving factor in the roll/pitch ACS design. If we assume that each signed DOF has two thrusters capable of applying equal moments but opposite forces about the vehicle CM (as was discussed in Chapter 3) and that each thruster has a moment arm of 1.0m (which is a reasonably large value), then each thruster must be capable of approximately 50N to offset the anticipated disturbance while still allowing for adequate bidirectional control authority. This means that the roll and pitch designs will have a minimum control authority larger than 1.0rad/sec^2, which is highly undesirable from efficiency and pointing requirements perspectives. It is beginning to become clear that the intention to separate the design of the two actuator systems may not be as realizable (or advantageous) as it originally seemed.

However, there is another option which allows for a wider range of control authority in roll and pitch without adding too much complexity to the baseline architecture. By including multiple additional low-thrust vertically oriented thrusters to the roll and pitch actuator clusters, we can achieve a form of quantized throttleability in control authority about the respective axes. Under nominal conditions the control architecture can select one or two actuators to achieve reasonable control authority; under biased acceleration conditions (e.g. CM offset from main thrust vector) additional actuators can be fired for the sole purpose of countering the bias torque while the original one or two actuators can still be used to achieve reasonable control authority. Note that this design may raise additional concerns of

thrust cross-coupling effects due to high numbers of shared propellant feed lines, but these will not be included here as they would be highly hardware-dependent.

The specifics of the chosen ACS actuator architecture are shown in Table 5-2. The thrust levels, moment arms, and number of actuators were chosen from initial guesses and refined during the control architecture design and verification process discussed in the following sections. The remaining characteristics are functions of thrust levels and based on historical data and experience.

Table 5-2: Multiple fixed nonthrottleable ACS thruster characteristics

Characteristics	Value for Roll/Pitch Actuators	Value for Yaw Actuators
Thrust (T)	22.50N (~0.50rad/sec^2)	4.50N (~0.30rad/sec^2)
Thrusters Per Signed DOF	6 (5 used for bias compensation)	4
Moment Arm (L)	1.00m	1.00m
Time Constant (τ)	0.01sec	0.01sec
Minimum On/Off Time (σ)	0.04sec	0.02sec
Time Delay (T_d)	0.02sec	0.01sec

5.2.2 Designing the Control Algorithm Architecture

The control algorithm architecture is responsible for commanding the associated actuator architecture to impart a desired linear and/or angular acceleration as a function of the errors between the actual and desired vehicle states. As was previously discussed, the architecture will address the stability and control of each DOF individually; separate algorithms will be designed for roll, pitch, yaw, X, Y, and Z. Assuming vehicle symmetry for simplicity, the roll/pitch and the X/Y controllers will be treated as identical. Note that the choice of actuators means that rotational and altitude DOF's can be controlled directly, while X and Y must be controlled indirectly by changing vehicle attitude. This architecture naturally places the attitude dynamics in an inner logic loop, and their controller design will be discussed first.

Attitude Controllers

Attitude controller design will be heavily influenced by the decision to use bang-bang actuators which are unable to directly vary thrust continuously. Observe the thrust profile for a typical ACS actuator in Figure 5-7. There will be a finite time lag between the commanded firing and actual firing ('Open Lag') due to valve actuation times; this will be followed by a short period of increasing thrust ('Rise') while

flow/combustion conditions reach equilibrium; there will be another time lag between the commanded firing termination and actual valve closing ('Close Lag'); and finally this will be followed by a short period of decreasing thrust ('Fall'). It is also important to note that each actuator has an associated minimum on and off time which is typically a function of these four variables and effectively limits the rate at which the thruster can be fired, as well as the minimum impulse it can impart. Collectively, this means that the actuator can be approximated as a constant signed control signal ($\boldsymbol{u}$ = -1,0,1), which has an associated time delay and/or is limited by effective hysteresis.

We are now faced with an important decision: do we design a control algorithm architecture which directly assumes constant signed control inputs with the aforementioned delay constraints, or do we assume a continuously variable control signal which utilizes the actuators' pulsing capability to impart a variable impulse at a predetermined control rate? A control scheme with a binary signed control signal has the advantage of design and implementation simplicity but also tends to make analysis and verification challenging, as many traditional stability and robustness metrics cannot be analytically determined. A linear control scheme, which assumes a continuously variable control signal, may be much easier to analyze, but one must still be careful to account for the inherent characteristics exhibited by the hardware. As these may not be common knowledge and have not been previously presented, they will be briefly discussed here.

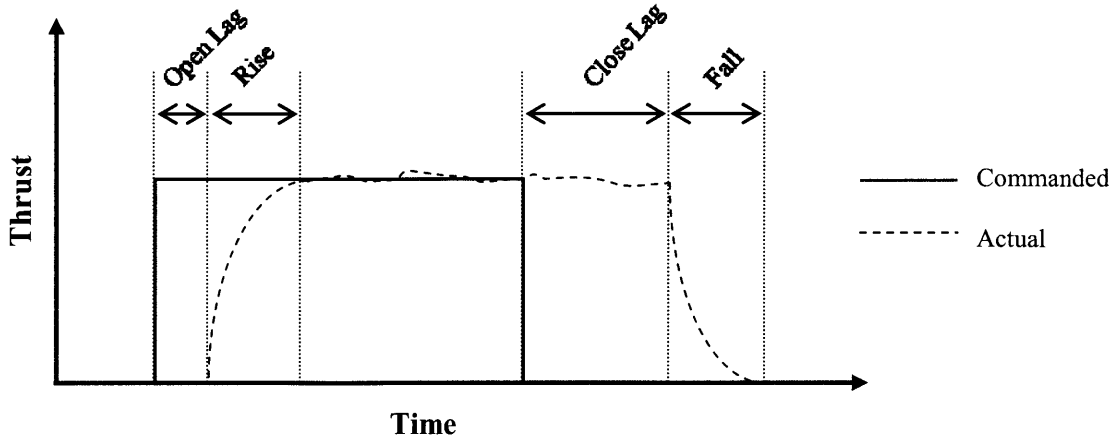


Figure 5-7: Pulsed actuator thrust profile

Pulsed control schemes, often called pulse width modulation (PWM) or pulse width pulse frequency modulation (PWPFM), require that the actuator operate at a constant or variable frequency (usually between 2 and 20Hz). Within the time period specified by the selected frequency the actuator must exhibit a ratio of on-to-off time proportional to the fraction of the total impulse commanded by the control scheme. For example, a 5Hz PWM scheme with a 40% total impulse command would require the actuator to repeat a sequence of 80ms on followed by 120ms off every 200ms, and the effective impulse would be 0.4*thrust for any given period. By changing the % impulse (throttle) command at the beginning of each period, the actuator and control scheme can approximate a continuously variable (albeit discrete) closed loop signal. This signal will also exhibit deadband characteristics in the upper and lower throttle regions due to minimum on/off time actuator limitations. For example, the roll and pitch actuators shown in Table 5-2 have a minimum on/off time of 40ms, which means that the actuators must be commanded on for the first 40ms and off for the last 40ms of each period to avoid nonlinearities (e.g. hysteresis, saturation). Note that this means limit cycling will still occur regardless of the control scheme. The minimum time period must therefore be selected such that it is not less than the minimum on/off time of the actuator and also large enough that it allows for adequate throttle range to achieve continuous impulse variability. However, there is also a clear time delay penalty associated with increasing the period. Recalling that phase lag of a linear system is proportional to frequency multiplied by time delay, and given that a digital control signal with a given period 'T' incurs between 0.5T and 1.5T effective time delay (depending on how the control scheme is implemented), then a PWM scheme with a 200ms control period essentially limits the open loop crossover frequency to below 1.5rad/sec. Given this information, the designer must weigh the importance of algorithm simplicity against attitude loop bandwidth limitations in order to decide which control scheme to adopt.

This case study will use phase plane attitude controllers, which are designed directly for the nonlinear and signed characteristics of the ACS actuators for two primary reasons. First, as steady state dynamics will be dominated by a limit cycle, there is reasonable concern that the propellant slosh mode will become excited. Phase plane controllers can directly address this because limit cycle requirements can be built into the design process. Secondly, additional phase lag in the inner attitude loop will necessitate low bandwidth for both the attitude and lateral position controllers, and this raises robustness concerns with regard to recovery time from bias moments. Phase plane controllers can help minimize this phase lag and increase bandwidth. This method is also well studied and flight proven for attitude control in low-disturbance environments [73] [74] [16], and we hope that it will also perform adequately for the

lunar lander terminal descent problem. As phase plane controllers are not commonly taught in university courses, the following paragraphs will provide a brief overview and alert the reader to important sensitivities and design motivations.

Phase plane controllers are most commonly applied to systems with low order dynamics which can be characterized by a primary state (e.g. angle) and its derivative (e.g. angular rate). The progression of these states are plotted along the x and y axes, respectively, as implicit functions of time. If constant signed control inputs are applied, the states typically follow a parabolic trajectory from any given initial condition. Trajectories which intersect the origin form the well known time-optimal switch curves, which form the boundaries between the regions that indicate the sign of the control input which must be applied to drive a system from an initial condition to the origin in the minimum time [64]. These switch-curves, which can also be modified in accordance with a minimum-time minimum-fuel optimization problem formulation, are the basis for most phase plane controller designs. An example is shown in Figure 5-8a.

Unfortunately, the time delays and hysteresis exhibited by real-world bang-bang actuators result in an inability to smoothly follow the parabolic switch curves, and the trajectory will typically 'chatter' back and forth about the switch curve until settling into a stable limit cycle about the origin. These oscillations waste propellant, and they can be reduced/controlled by splitting the switch curve into two separate boundaries with a deadband region in between where the control command is zero. The simplified phase plane controller which will be used in this thesis is based on this principle and approximates the nonlinear switch curves as discontinuous linear boundaries. Now the controller (or switch curves) can be designed to explicitly control the limit cycle characteristics, and the design variables essentially reduce to the selection of the deadband, the slope, and the height of the drift channel. The deadband is typically derived from pointing requirements, the drift channel height is selected to limit excessive angular velocity (which causes more inertial cross-coupling disturbance but may improve recovery characteristics to bias moments), and the slope is chosen to tailor the limit cycle and step response characteristics. This architecture also allows the controller to be expressed in an intuitive linear feedback structure. The simplified controller is shown in Figure 5-8b, and the feedback structure is shown in Figure 5-9.

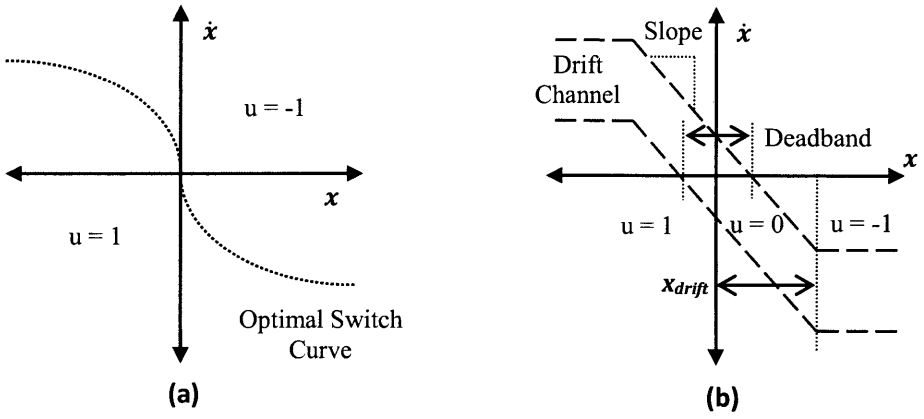


Figure 5-8: (a) Time-fuel optimal controller, (b) Simplified controller with linear switch curves

A typical trajectory for a step command can be seen in Figure 5-10 for a double integrator system with realistic bang-bang actuators. The system begins with an angle error and its state clearly in the u=1 region. The control input forces the state to adopt a parabolic trajectory until it enters the vertical deadband known as the drift channel (this limits the maximum rate of the vehicle under normal conditions) and the control input becomes 0. The trajectory then drifts towards the sloped switch curve boundaries and enters the u=-1 region, where the control input returns the state to within the vicinity of the origin (in this example the switch curves have been designed such that a state exiting the drift channel will re-intercept the switch curve with near-zero rate, thereby reducing overshoot). The trajectory then settles into a stable limit cycle whose characteristics are a function of the attitude deadband, the switch curve slope, the control power, and the actuator time delays.

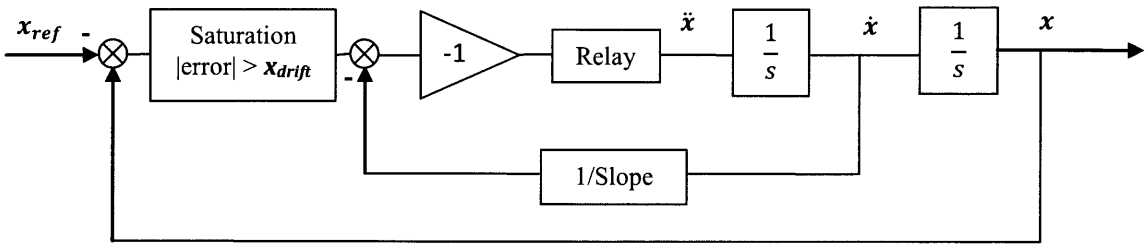


Figure 5-9: Linear feedback architecture for simplified controller

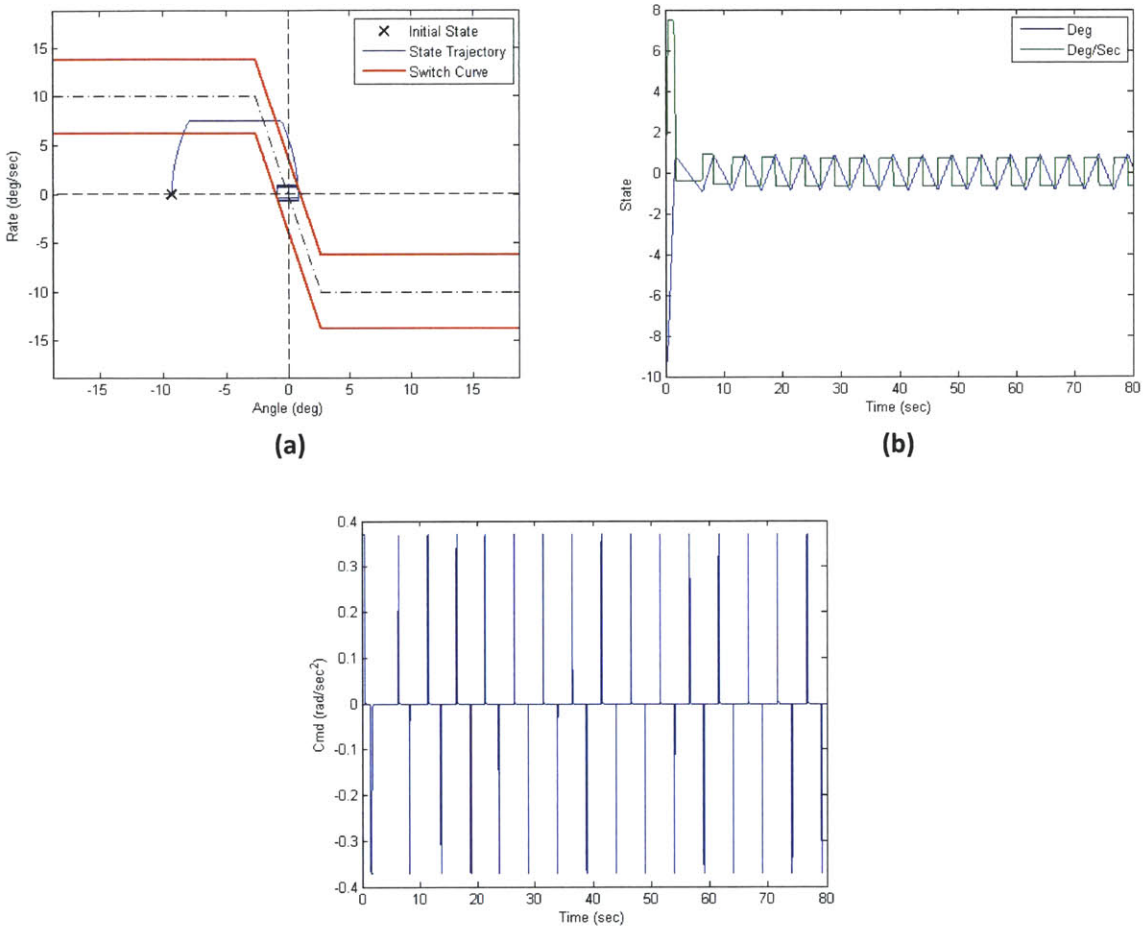


Figure 5-10: Typical step command plots; (a) phase plane, (b) states, (c) inputs

Phase plane architectures are particularly advantageous because they can tolerate a wide range of modeling errors without compromising stability (in the sense of Lyapunov). As shown in Figure 5-11, common sources of error such as reasonable increases in control power and feedback time delay tend to increase the rate amplitude of the limit cycle (and often frequency) without directly destabilizing the system. This robustness is an important characteristic of phase plane controllers and makes them well suited for applications where gradual stability and performance degradation are much preferred to rapid divergence. The orders of magnitude of the modeling errors shown in Figure 5-11 also indicate that the trajectory (and limit cycle) is more sensitive to time delays than control power modeling errors, meaning quantification of time delay on the actual system is very important in controller design. The change in limit cycle trajectory is also interesting to note. Increasing the control power actually

decreases the angle amplitude while increasing the rate amplitude, and increasing the time delay tends to increase both the angle and rate amplitudes. This means that increasing (or decreasing) control power should not decrease pointing accuracy, while increasing the time delay will have a significant impact on pointing accuracy and overall stability characteristics. This is particularly important for the roll and pitch DOF's because they will serve as the inner loops of the X and Y position controllers in our chosen control algorithm architecture.

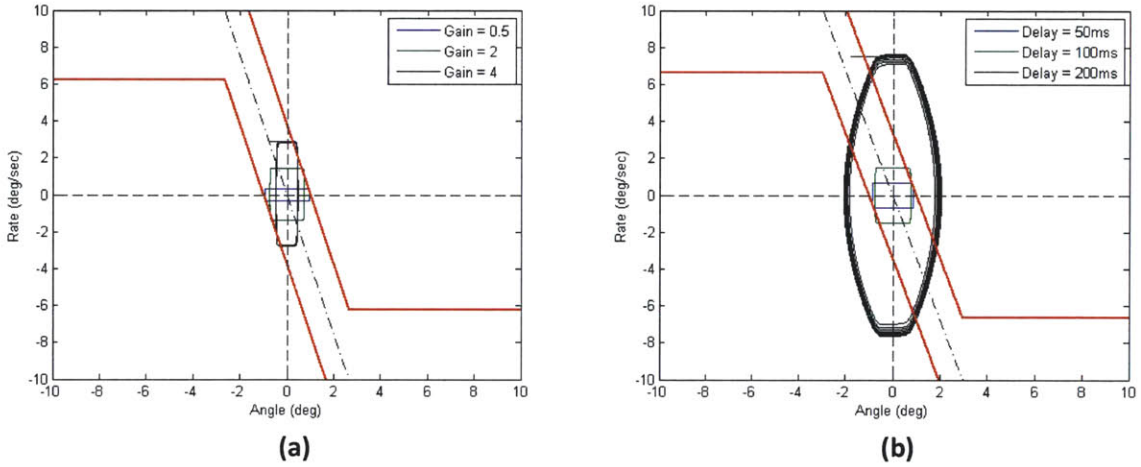


Figure 5-11: Effect of modeling errors; (a) control power, (b) time delay

Lastly, one of the challenges of phase plane controller design is stability and performance verification. The simulations indicate that under nominal conditions the state trajectory will always result in a stable limit cycle, but we would also like to prove this analytically. If the feedback architecture can be expressed in a single feedback loop, then describing functions can be used to predict limit cycle performance quite well. Unfortunately, our chosen feedback architecture contains two feedback loops and two nonlinearities, which make it difficult to apply the describing function technique. A simpler approach is to assume that a stable limit cycle will exist, and that the trajectory will be symmetric about the X and Y axes of the phase plane plot. The assumption is motivated by the minimum (symmetric) nonzero impulse limitation, which means that intuitively the system can never reach the reference angle state with zero rate and therefore must continually oscillate about the desired reference angle in a stable limit cycle. Now for a given switch curve, control power, and time delay we can solve the boundary value problem under the constraints that the trajectory must be symmetric about each individual axis to predict the approximate amplitudes and frequency of the limit cycle. For instances in which the minimum on/off time of the actuator is the dominant driver of the limit cycle, one can use

Equation 5-7 to calculate the maximum and minimum rate, Equation 5-8 to calculate the maximum and minimum angle, and Equation 5-9 to calculate the estimated limit cycle frequency in Hz. In this case, $\boldsymbol{x}$ refers to the angle, $\dot{\boldsymbol{x}}$ refers to the angular rate, $\ddot{\boldsymbol{x}}$ refers to the nominal angular acceleration induced by the actuators, $\boldsymbol{\sigma}$ refers to the minimum on/off time, and Δ refers to the slope of the switch curve. Note that this method is most valid when the overall time delay is much less than the minimum on/off time because it assumes that the limit cycle trajectory is entirely contained by the sloped switch curves.

$$\dot{x}_{max} = 0.5\ddot{x}\sigma \tag{5-7}$$

$$x_{max} = -0.5\ddot{x}\sigma^2 + \dot{x}_{max}(\sigma + 1/\Delta) + x_{deadband} \tag{5-8}$$

$$Hz = 1/(2\sigma + 4x_{max}/\dot{x}_{max}) \tag{5-9}$$

In instances where the overall time delay (denoted here by t_{delay}) is greater than the minimum on/off time and is assumed to be the dominant driver of the limit cycle characteristics, one can use Equation 5-10 to calculate the maximum and minimum angle and Equation 5-11 to calculate the estimated limit cycle frequency. This approach assumes that the corners of the limit cycle trajectory pass beyond the switch curve boundaries due to the time lag between commanded and actual angular acceleration. In this case the author has chosen to calculate $\dot{x}_{max}$ using an iterative search method to find the maximum and minimum rate based on the boundary condition requirements that $(x_0, \dot{x_0}) = (x_f, -\dot{x_f})$ during periods of angular acceleration.

$$x_{max} = -0.5\ddot{x}(t_{delay}/2)^2 + \dot{x}_{max}(3t_{delay}/2 + 1/\Delta) + x_{deadband} \tag{5-10}$$

$$Hz = 1/(2t_{delay} + 4x_{max}/\dot{x}_{max}) \tag{5-11}$$

The resulting solutions for an example controller (deadband of 5.0deg, drift rate of 25.0deg/sec, and slope of -1.43) are compared with results from a nonlinear simulation in Figure 5-12. Notice that the analytical solutions become less accurate for systems with small deadbands, large time delays, and large control powers, particularly when the trajectory passes all the way through the deadzone before ceasing to accelerate (as is the case in Figure 5-11 for 200ms delay). This general approach can be used to design a phase plane controller to produce a specific limit cycle, and these plots can be used to predict limit cycle stability and performance limits for a given phase plane controller.

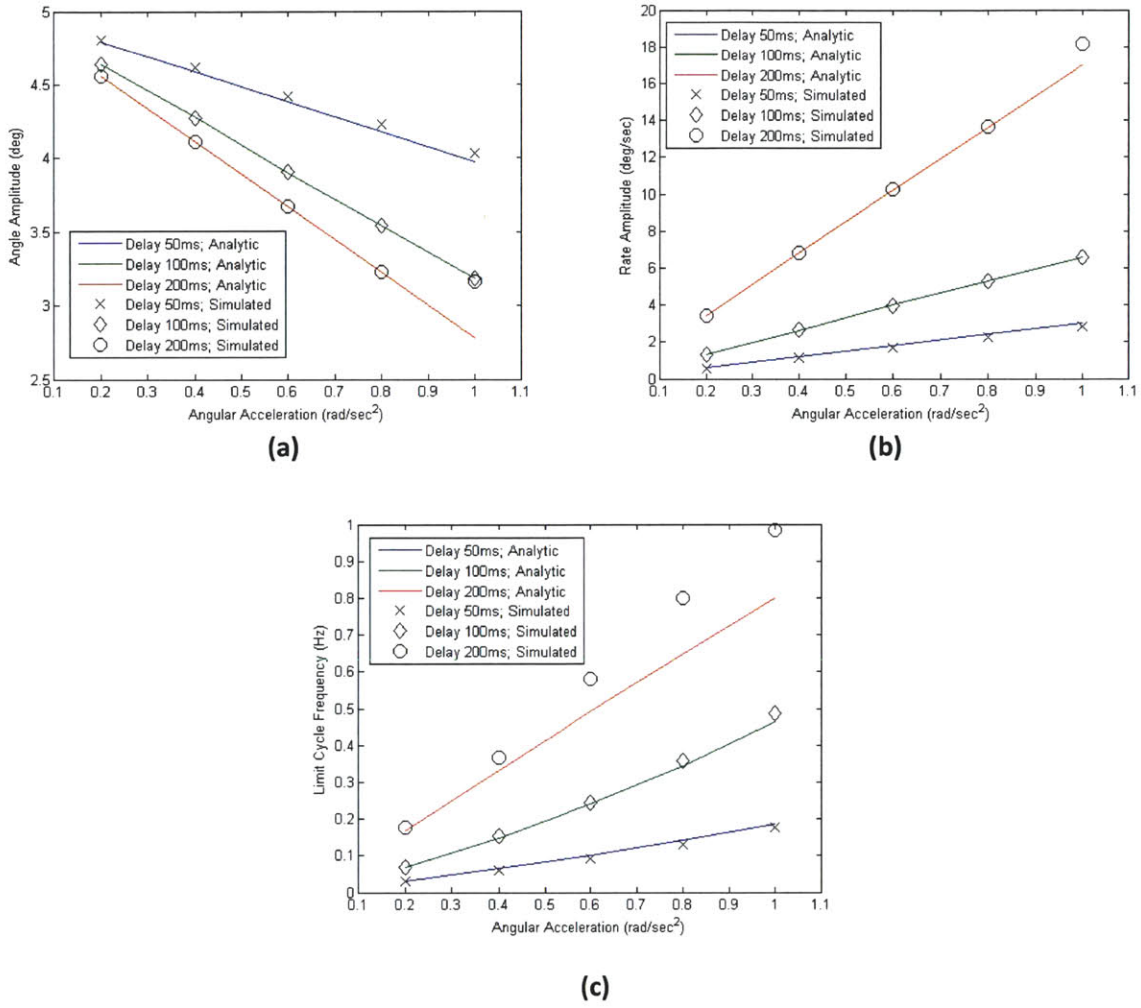


Figure 5-12: Limit cycle solutions; (a) angle amplitude, (b) rate amplitude, (c) frequency

Yaw

The yaw controller is the simplest to design because the chosen flight profile does not require any prespecified yaw maneuvers, and the selected actuator architecture does not incur any major bias moments about the yaw axis. The yaw control architecture can therefore be designed with a lower control power, drift rate, and deadband; ultimately, this results in a low frequency and amplitude limit cycle under steady state conditions. The chosen controller has a deadband of 2.0deg, a drift rate of 10.0deg/sec, and a slope of -3.27. The controller is designed such that a state exiting the center of the drift channel with an extra time delay of 30ms (margin, to reduce overshoot) will intersect the switch curve with near-zero rate. The step response plots (with an additional time delay of 30ms) are shown in Figure 5-13.

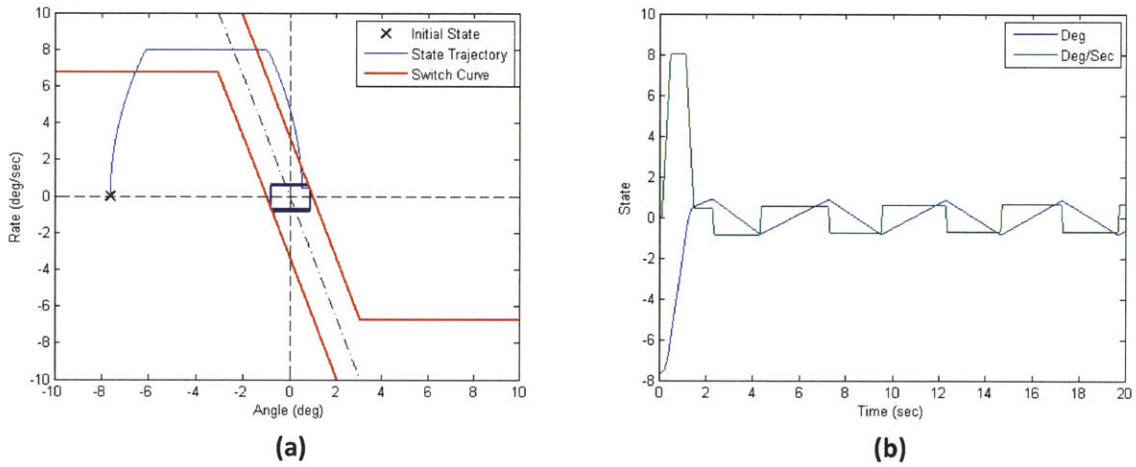
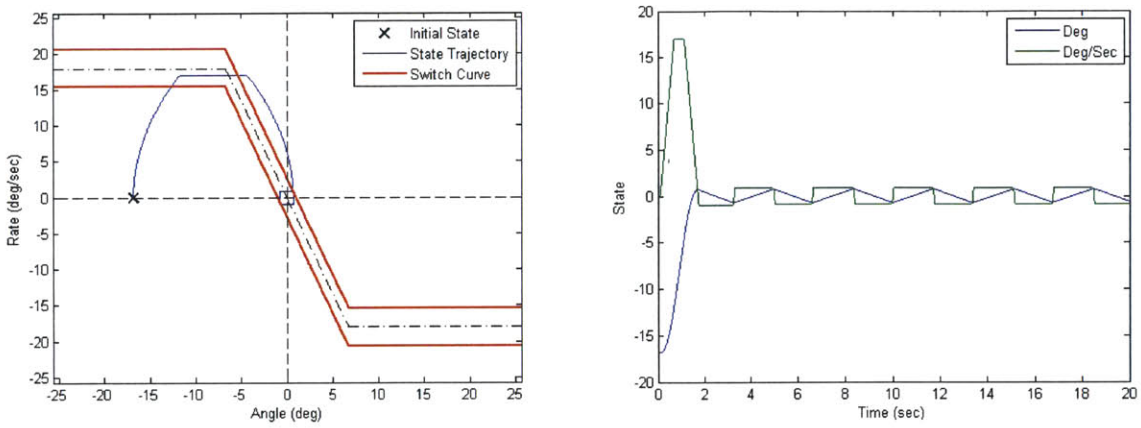


Figure 5-13: Yaw step response; (a) phase plane, (b) states vs. time

Roll and Pitch

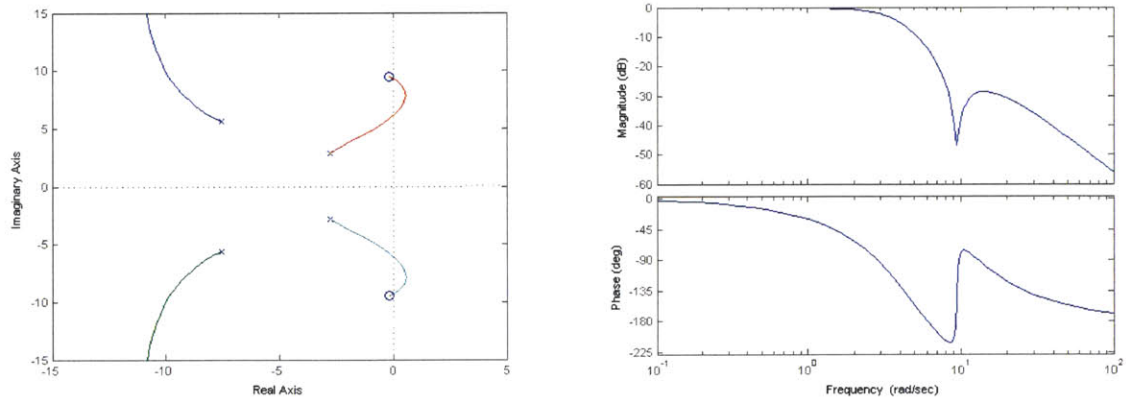
The roll and pitch controllers, which are considered to be the same for simplicity, pose a more challenging design problem. They must be capable of considerable disturbance rejection while still keeping limit cycle frequency and amplitudes low. The chosen controller has a deadband of 2.0deg, a drift rate of 18.0deg/sec, and a slope of -2.68; it is designed with the same time delay margin and trajectory criterion of the yaw controller. The predicted limit cycle frequency is approximately 0.25Hz, which is six times lower than the predicted propellant slosh frequency (1.5Hz). The nominal step response plots with 30ms time delay are shown in Figure 5-14. Note that the higher drift rate and control power decreases rise time and increases system bandwidth.



(a) (b)

Figure 5-14: Roll and pitch step response; (a) phase plane, (b) states vs. time

Notice that in the nominal control case we have chosen to only use one thruster per signed DOF. This means that a small asymmetric force will be imparted on the vehicle CM each time a roll or pitch ACS thruster is fired, and this value should be fed forward to the altitude controller. This choice also leaves the other five thrusters per DOF to serve as bias moment compensators, and they will be commanded by a separate bias moment estimator and controller. This function simply calculates the difference between the commanded and measured angular acceleration, passes the signal through a low pass filter and wide notch filter (centered at the lower range of the expected propellant slosh frequency), and then commands up to five actuators to impart a moment such that the resultant net bias moment is less than or equal to half of the moment capability of the single actuator (per signed DOF) used by the original phase plane controller. The prefilter, shown in Figure 5-15, includes the low pass and notch filter.



(a) (b)

Figure 5-15: Bias compensator signal prefilter; (a) Root Locus, (b) Bode

The resultant response to a worst-case 5.0cm CM offset at 100% throttle (86.0N*m) is shown in Figure 5-16. The state, which begins at the origin of the phase plane, quickly diverges to 19deg/sec before the bias compensator function commands the additional thrusters to fire and reverse the rate of the vehicle. Notice that even when the state is inside the deadzone a negative net bias moment (from the ACS thrusters) still accelerates the state. This means that the trajectory will eventually settle onto the left side of the switch curve and 'chatter' back and forth at a much higher frequency (albeit at a lower angle amplitude) than the intended limit cycle design case. This is an unavoidable consequence of using bang-bang actuators to counter bias moments, and the new limit cycle properties are highly dependent on bias moment magnitude. Propellant slosh excitation and excessive propellant usage will therefore be continuing concerns during the actual mission scenario. Also note that recovery time is greatly influenced by the time it takes for the bias compensator function to recognize that a bias moment exists. If the designer has confidence in the plant model, it may be very beneficial to increase the natural frequency of the low pass filter poles and reduce the width of the notch filter. It should also be understood that this type of disturbance is very unlikely in an actual lander at the beginning of the terminal descent flight stage. Instead, it is much more likely that the bias compensator would have encountered and identified the bias moment much earlier in the trajectory when the main descent engine was first used (for example, during the deorbit and braking burns), and the vehicle would have much more time to recover before beginning the terminal descent. The plots in Figure 5-16 should be viewed accordingly. Lastly, one should also realize that the bias compensation method discussed here may or may not use propellant without imparting ΔV (which wastes propellant), depending on how the actuators are arranged.

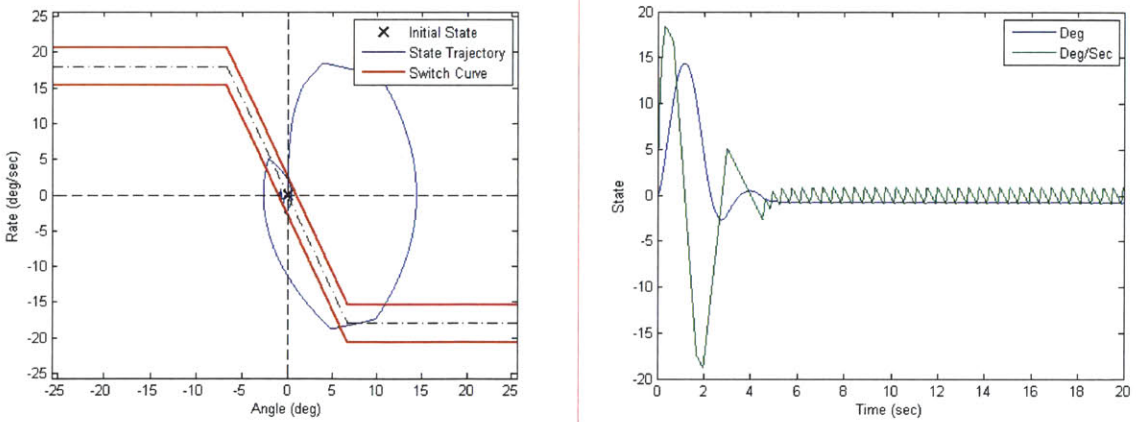


Figure 5-16: Roll and pitch response to 86.0N*m bias moment; (a) phase plane, (b) states vs. time

Position Controllers

The position controllers are responsible for comparing information from the desired guidance trajectory and the vehicle state to command the main throttleable engine (to control altitude) as well as the pointing direction of the main thrust vector (to control X and Y position).

Altitude

Under nominal conditions the altitude system does not contain complicated dynamics and will be discussed first. The plant model of interest is simply a double integrator combined with the first order lag and time delays associated with the throttleable engine. As the state vector of interest only contains three elements we will assume that they are all available to the control algorithm (without a complex estimator design), so we will use a LQR control algorithm architecture, which was discussed in Chapter 4. This problem formulation ensures stability (from a mathematical sense) and allows us to weight the importance of the state variables against the control effort. In this case, we are most concerned with tracking errors (position and velocity) from the desired trajectory and control signal saturations, and LQR will allow us to tailor these performance variables directly.

We begin by forming the state space matrices augmented with an integral term on the position error to null steady state error (where $\boldsymbol{Z}$ is the altitude and $\boldsymbol{F}$ is force), and then we weight the states and control effort according to preference. Note that a time delay of 0.05sec is included in the gain calculations.

$$\begin{bmatrix} \dot{Z} \\ \ddot{Z} \\ \dot{F} \\ \dot{Z}_I \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/\tau & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Z \\ \dot{Z} \\ F \\ Z_I \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/\tau \\ 0 \end{bmatrix} u \qquad \textbf{5-12}$$

As deviations from desired position and velocity are most important, we weight these terms highest. The resultant full state feedback gains are shown in Equation 5-13 along with the closed loop system eigenvalues. Note that $\boldsymbol{Z}$ is still the measured altitude and $\boldsymbol{Z_I}$ is the integral of the altitude error.

$$\boldsymbol{K} = [3.05 \quad 3.13 \quad 0.54 \quad -0.47] \qquad \textbf{5-13}$$

$$\boldsymbol{eig}[\boldsymbol{A} - \boldsymbol{BK}] = \begin{bmatrix} -4.94 \\ -1.19 + 1.06i \\ -1.19 - 1.06i \\ -0.19 \end{bmatrix} \qquad \textbf{5-14}$$

Time domain tracking performance for the closed loop system is shown below for initial condition errors of 1.0m and a range of velocities which are considered to be expected 'worst case' conditions for our system. Initial condition response and overall disturbance rejection is acceptable, but desired control efforts indicate that saturation (at 10% or 100% throttle) is likely under some initial conditions. The nonlinear simulation must therefore be used to help assess the impact of this occurrence. An anti-windup logic filter will also be added to the integral term to reduce the impact of control signal saturation on stability. The input sensitivity function (disturbance acceleration to y) shows sufficiently low gains at low and high frequencies, and the peak magnitude is well below 0dB; this supports the disturbance rejection capability seen in the linear simulation.

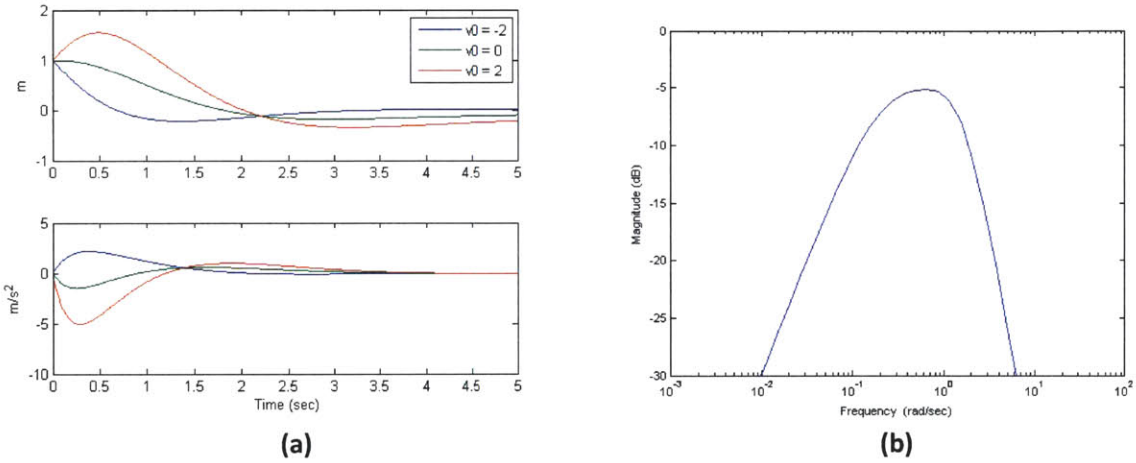


Figure 5-17: Altitude controller performance; (a) time domain tracking, (b) input sensitivity function

One should also note that the control signal in the nonlinear simulation is different from the control signal used to generate the plots in Figure 5-17. This is because the actual control signal is driven by precomputed reference altitude and descent rates, and is therefore calculated using Equation 5-15.

$$\boldsymbol{u = -K(1:3)\left[\begin{bmatrix} Z \\ \dot{Z} \\ F \end{bmatrix} - \begin{bmatrix} Z_{ref} \\ \dot{Z}_{ref} \\ 0 \end{bmatrix}\right] - K(4)\int (Z - Z_{ref})dt} \qquad \textbf{5-15}$$

This is a multi-input system, and cannot be analyzed using a SISO transfer function. Instead, we will rely on the assumption that the bandwidth of the closed loop system is reasonably above the bandwidth of the reference input signals such that the system tracks the reference signal well. The nonlinear simulation supports this.

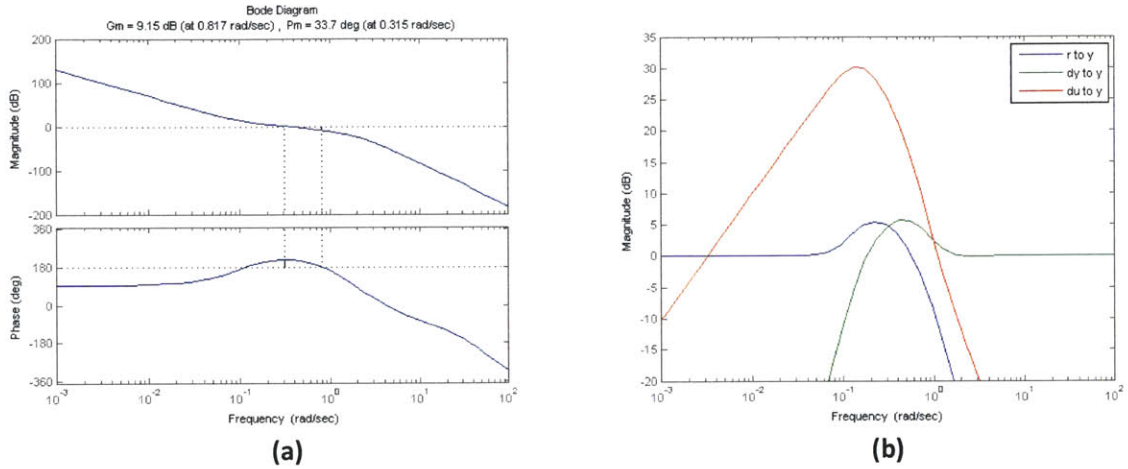
Lateral

Lateral position controllers (X and Y) are the most challenging to design because they must command the inner loop phase plane controllers to achieve a desired thrust pointing vector. As the phase plane controllers cannot be well approximated by simple linear equations, it is difficult to design a corresponding high bandwidth outer loop controller using standard LTI assumptions. One approach is to approximate the inner loop as a double integrator system and design the outer loop system bandwidth to be below (~1.0 decade) the estimated inner loop bandwidth. Keeping the bandwidths sufficiently far apart ensures that the outer loop dynamics will not significantly interfere with the inner loop dynamics, allowing the two systems to be designed separately without compromising stability. As this has been shown to be effective on previous space vehicles, we will also utilize this method. However, this decision to approximate the complex inner loop dynamics as a simple double integrator also means that the nonlinear simulation will now play a large role in controller design and verification, particularly as we try to increase open loop bandwidth for robustness to disturbances.

We begin by designing a compensator capable of stabilizing the plant while providing adequate bias moment and velocity rejection, a timely step response, and acceptable stability margins. The criteria for these characteristics are loosely derived from the magnitude of disturbances the spacecraft is likely to encounter and the 15.0sec terminal descent trajectory time limitation. The compensator is therefore designed so that the closed loop transfer function exhibits steady state conditions at 15.0sec in response to a lateral 1.0m step, a 1.0m/sec velocity, and a 5.0cm CM offset which meet the criteria established for a safe landing in Table 1-3 without pushing the open loop crossover frequency past 0.4rad/sec (above which Monte Carlo simulations show an unacceptably high probability of safe landing criteria violation and/or instability). The resultant compensator and frequency domain plots are shown in Equation 5-16 and Figure 5-18. Notice that the compensator (which is the transfer function from lateral position error to commanded lateral acceleration) includes an integrator to null steady state error, a pair of complex zeros for mid-frequency gain and phase increase, and two pairs of poles to reduce sensitivity to noise and unmodeled dynamics at high frequency. The open loop Bode plot, including a Pade approximation for the 50ms time delay, is shown in Figure 5-18a. The sensitivity (dy to y) and complementary sensitivity (r to y) transfer functions both peak below 5dB, and the input sensitivity transfer function (du to y) is reduced at both low and high frequencies (Figure 5-18b). One area of concern is that both the sensitivity and the input sensitivity transfer function magnitudes are greater

than zero at 1.0rad/sec, which is close to the unperturbed steady state limit cycle frequency of the inner loop phase plane system. This will be discussed more with the time domain performance plots.

$$Gc(s) = 0.003175\left(\frac{100s^2 + 14s + 1}{s(0.1089s^2 + 0.66s + 1)(0.4489s^2 + 0.94s + 1)}\right) \qquad \text{5-16}$$

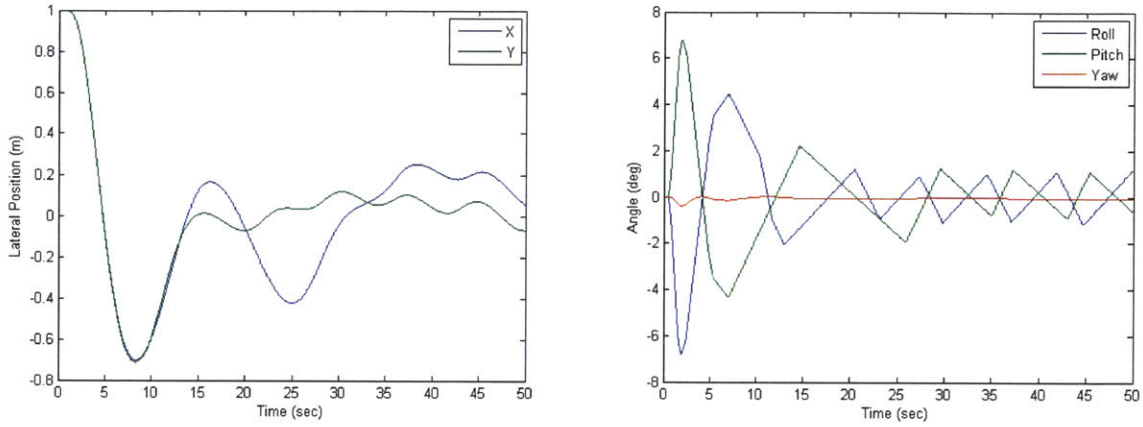


(a) (b)

Figure 5-18: Lateral position transfer functions; (a) open loop Bode, (b) sensitivity magnitudes

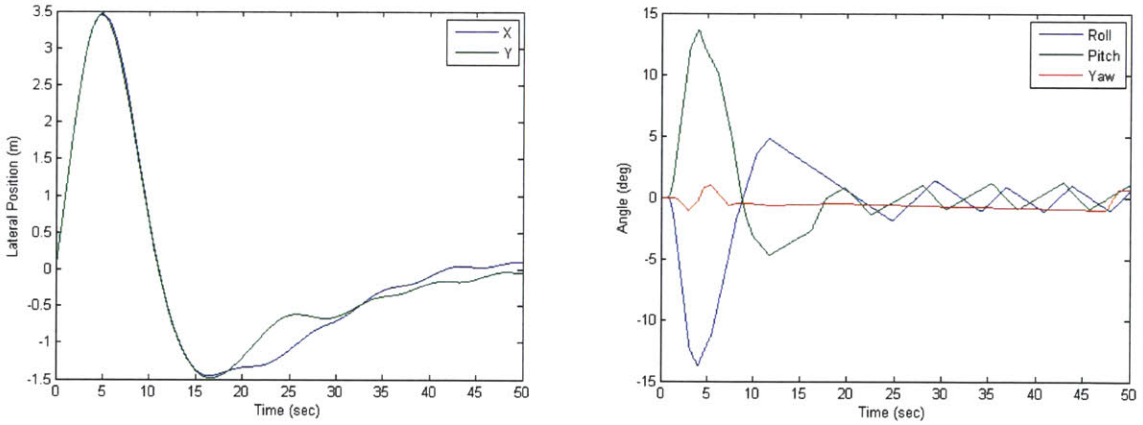
Nonlinear time domain response plots are shown for a step command (Figure 5-19), an impulse (Figure 5-20), and a bias moment (Figure 5-21). In all cases the attitude states have returned to steady state within 15.0sec and the position states are within the acceptable bounds. Also note that the disturbance in the position state that we expect from the attitude limit cycle is only present when the position is near a steady state condition, meaning that when the system is actively responding to significant external perturbations the limit cycle is nonexistent. This makes sense because under said conditions the state is pulled away from the deadband region of the phase plane and limit cycling cannot exist until the state returns to the vicinity of the origin. For this reason it is acceptable to have sensitivity and input sensitivity transfer function magnitudes greater than one near 1.0rad/sec; the 'input' disturbance caused by the limit cycle is only present for an unperturbed system. If the system remains unperturbed at steady state, simulation indicates that variations in position between ±0.3m are experienced due to the limit cycle disturbance. This condition can be seen in the final 25sec of Figure 5-19a where the X position begins to experience 0.4m amplitude oscillations as a result of the pitch limit cycle. If the final time of the simulation is increased, the Y position begins to exhibit similar characteristics. Also note that under bias moment conditions the limit cycle changes significantly. The bias causes the state to hug one

side of the switch curve, resulting in much smaller amplitude angle oscillations, as can be clearly seen in Figure 5-21b. The change in limit cycle characteristics due to bias moments are difficult to predict and will remain a concern for potential propellant slosh mode excitation.



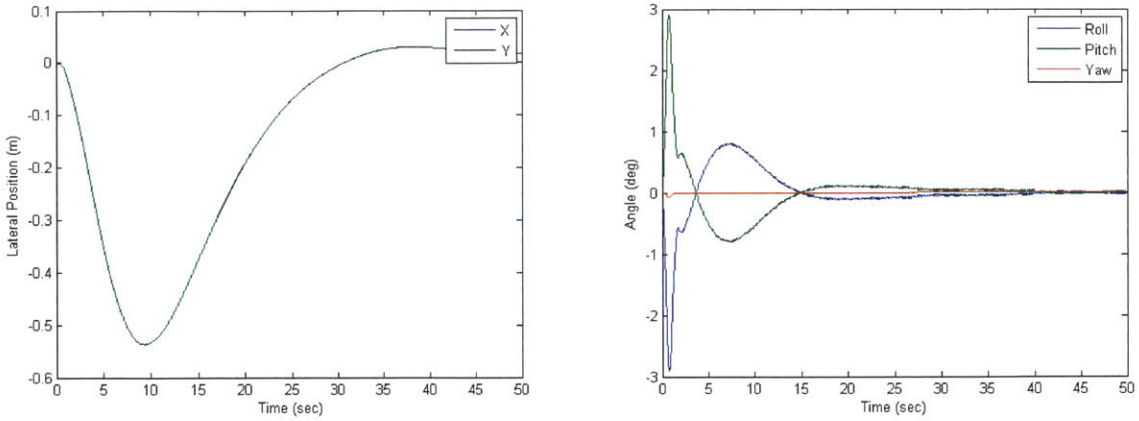
(a) (b)

Figure 5-19: Lateral 1.0m step response; (a) position, (b) angles



(a) (b)

Figure 5-20: Lateral 1.0m/sec impulse response; (a) position, (b) angles



(a) (b)

Figure 5-21: Lateral response to 5.0cm CM offset in both axes at T/W = 1; (a) position, (b) angles

Lastly, note that although the aforementioned approach produces acceptable closed loop performance in this case study, it should be recognized that higher bandwidth (and potentially better disturbance rejection etc.) may be feasible if we are confident in the limit cycle frequency. As the attitude limit cycle adds oscillatory characteristics to the lateral dynamics, the primary concern with increasing outer loop bandwidth is excitation of the limit cycle dynamics. This can be addressed via the introduction of a notch filter at the limit cycle frequency or by taking other means to minimize the input sensitivity function (du to y) at said frequency. This approach was not pursued further in this thesis because of uncertainty in limit cycle characteristics due to bias moment effects.

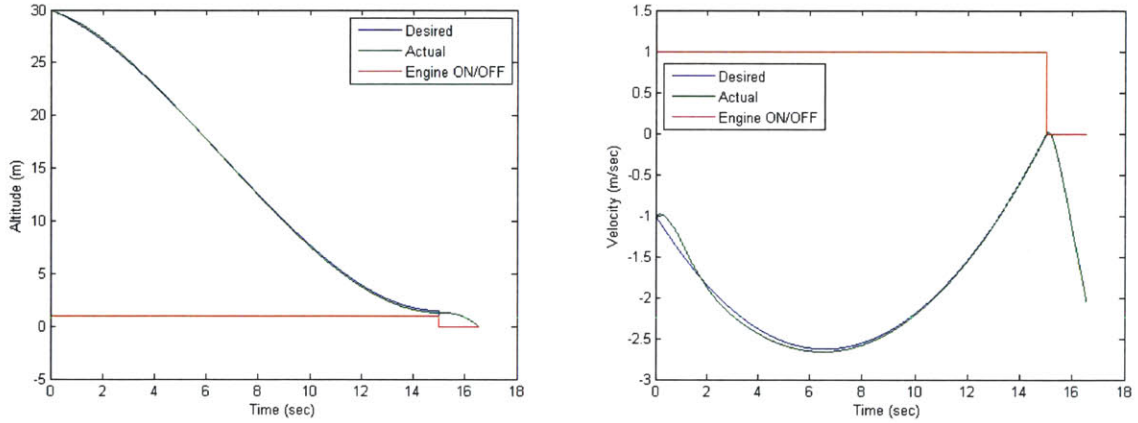
5.2.3 Performance and Verification

The nonlinear simulation discussed in the beginning of this chapter will be used to verify the collective performance of the control system architecture in all six DOF's. The simulation will analyze performance for nominal cases as well as off-nominal cases in order to characterize system robustness and alert the designer to sensitivities. The simulation does include actuator dynamics, 6DOF rigid body spacecraft dynamics, propellant slosh dynamics, bias moments, time delays, and parameter variation. It does not explicitly include sensor noise or lunar environment-based disturbances.

Nominal Performance

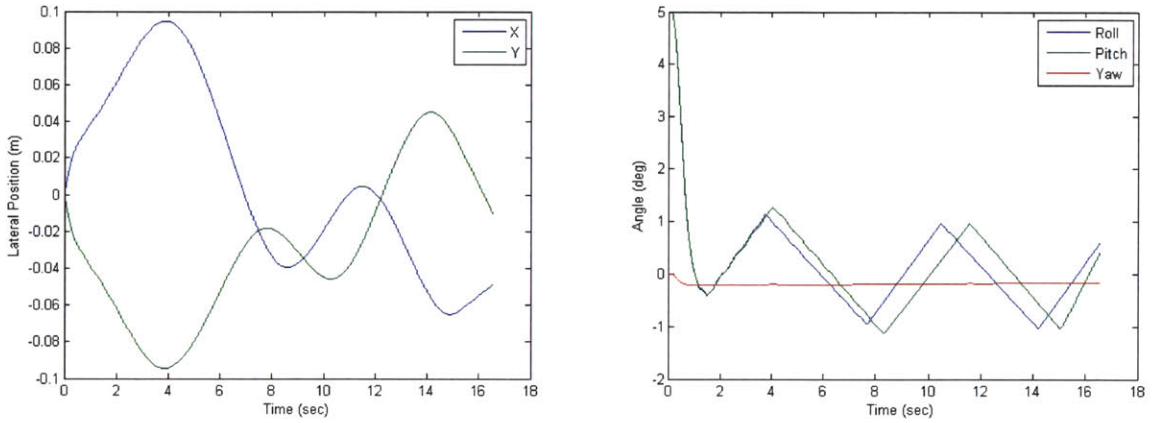
Under nominal conditions the lander begins terminal descent at 30.0m altitude, a 1.0m/sec descent velocity, and with its thrust vector pointed nearly perpendicular to the lunar surface (5.0deg pitch and roll). The following performance plots show that the lander tracks the fixed 'guidance' descent trajectory

well, commands engine shutdown after 15.0sec, and descends to the lunar surface with all state conditions within the desired margins. None of the control commands come close to their saturation limits, and the overall system remains stable in the sense of Lyapunov. Also notice that the attitude limit cycle induces small oscillations in the positions and that propellant slosh excitation produces small oscillations in the attitude. However, all oscillations are within acceptable limits and are not increasing in magnitude.



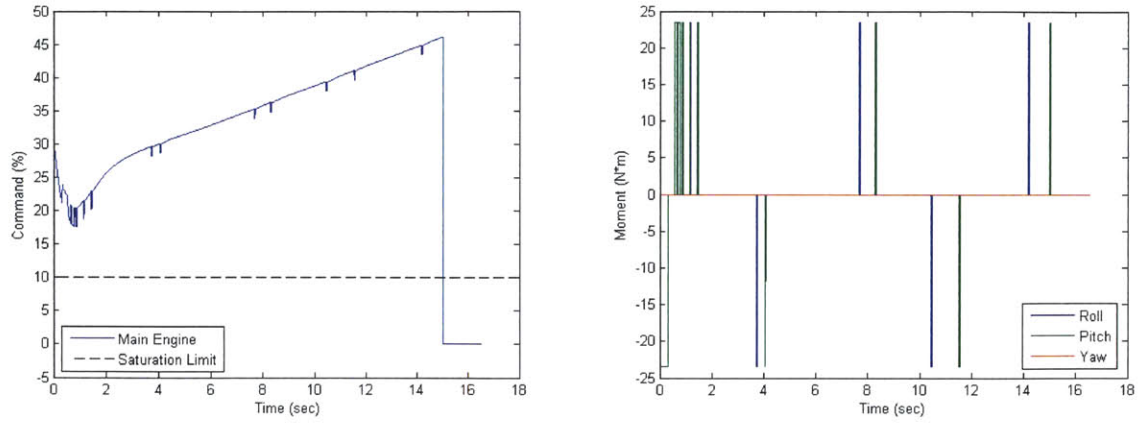
(a) (b)

Figure 5-22: Nominal descent states; (a) altitude, (b) vertical velocity



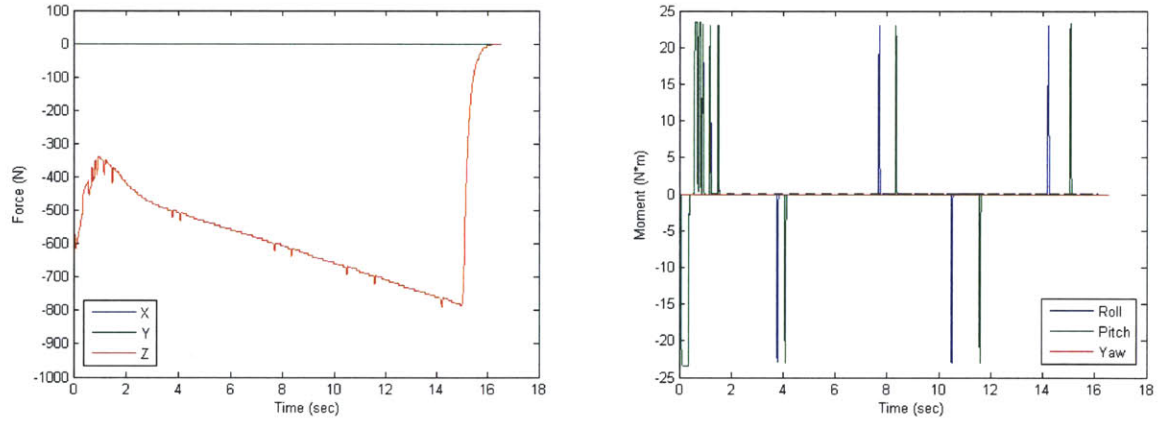
(a) (b)

Figure 5-23: Nominal descent states; (a) lateral positions, (b) angles



(a) (b)

Figure 5-24: Nominal descent control; (a) throttle, (b) desired moments



(a) (b)

Figure 5-25: Nominal descent actual body frame control; (a) forces, (b) moments

Variation of Single Parameters

Determining system stability and performance limits to disturbances and other modeling errors provides a good indication of the overall robustness. Single parameter variations will be used in this section to identify individual sensitivities, and Monte Carlo simulations will be used in the next section to assess overall robustness. Important single parameter results are shown in Table 5-3; the values listed are the maximum allowable variations before one or more of the landing conditions in Table 1-3 are violated. As expected, the results indicate that the system is very insensitive to actuator gain factors and initial attitude (which is a testament to the robustness of the phase plane controllers), and that the system is relatively insensitive to initial vertical state errors (which suggests that the throttleable engine and LQR

controller are performing well). However, the results also indicate that the system is relatively sensitive to time delays and initial velocity errors: time delays alter the limit cycle characteristics without destabilizing the outer position controller (which is concurrent with the results of the early phase plane controller analysis); the initial velocity errors are within the region of attraction of the system, but they do tend to induce a large position error which is not corrected within the 15.0sec terminal descent trajectory landing window. While this may seem like a concern (as initial lateral velocity errors are probable in an actual landing), the reader is reminded that for these case studies the landing trajectory is intentionally fixed, allowing us to focus solely on the design and analysis of the control systems. The actual landing system would greatly benefit from real-time trajectory generation at the beginning of the terminal descent phase which directly uses the current vehicle states as the initial conditions of the boundary value problem, thereby reducing initial state errors to approximately zero. The initial state limits shown in Table 5-3 are primarily for comparison to the other case studies following the same fixed trajectory and should be viewed accordingly. Lastly, and perhaps most importantly, none of the individual parameters shown in Table 5-3 actually induce divergent state trajectories, meaning our system exhibits graceful performance degradation from modeling errors rather than rapid divergence.

Table 5-3: Max single parameter variation limits before landing constraint violation (Case 1)

Parameter	Case 1	Comment
ACS Thrust Scale Factor	9	Applied to all thrusters
Main Engine Thrust Scale Factor	0.5, 2.6	Min, max
Initial Roll or Pitch Error	65deg	On one axis
Initial Lateral Position Error	15m	On one axis
Initial Vertical Position Error	35m, 8m	Height above, below 30m
Initial Lateral Velocity Error	3m/sec	On one axis
Initial Vertical Velocity Error	5m/sec, 6m/sec	Velocity above, below -1m/sec
Overall Time Delay	120ms	Applied to all control feedback states
X or Y CM offset	10cm	On one axis

Monte Carlo Simulations

Lastly, we can assess stochastic performance and stability for random variations of multiple parameters using Monte Carlo simulations. The parameter bounds were chosen to be well within the single parameter variation limits and were intended to reflect reasonable uncertainty in the overall system. If the designer has confidence in the chosen variation bounds, the Monte Carlo simulation tool can also be used to quantify confidence in mission success. In our case, the simulations will help to identify sensitivities in coupled parameters which may not be apparent from previous analysis methods, as well as for robustness comparison purposes to the other case studies.

Table 5-4: Monte Carlo simulation parameter variation bounds

Parameter	Minimum	Maximum
Lander Mass	300kg	400kg
Moment of Inertia Scale Factor	0.8	1.2
Initial Position Error [x y z]	N/A	± [1 1 1]m
Initial Velocity Error [x y z]	N/A	± [1 1 1]m/sec
Initial Attitude Error [x y z]	N/A	±[5 5 5]deg
Initial Rate Error [x y z]	N/A	±[5 5 5]deg/sec
Propellant Slosh Mass	20kg	60kg
Propellant Slosh Frequency	0.5Hz	3Hz
Overall Time Delay	N/A	50ms
CM offset (applied to both axes)	N/A	±5.0cm

Representative simulation results are shown in the figures below. The lunar position and desired guidance trajectory are shown in Figure 5-26, final position and velocity histograms are shown in Figure 5-27, and final angular position and rate are shown in Figure 5-28. Of the 100 runs simulated, only one was found to violate the safe landing state conditions at touchdown. The offending state was the lateral velocity, which was induced by the combination of a very low system mass (306kg), a low moment of inertia (0.89 scale factor), a CM offset of 3.2cm, and initial lateral position and velocity errors near 1.0m and 1.0m/sec. Further simulation of these conditions showed that stability was retained and a steady state was reached after 30sec. However, the Monte Carlo simulations also showed sporadic excitation of the propellant slosh modes in cases where the bias moment induced an attitude limit cycle near the frequency (or at a multiple) of the slosh natural frequency. In cases where damping was reduced to

0.005, the probability of excessive angular rate at touchdown was nearly 5%. It is also interesting to note that even in cases of slosh excitation and angular rate limit violation at touchdown, the outer position states did not show signs of instability, indicating a graceful degradation of general system performance even in the worst case conditions.

Ultimately, the Monte Carlo simulations support the conclusions of the previous analysis that the overall control system architecture is robust to a wide variety of parameter variations but does suffer from stability concerns regarding interaction between attitude limit cycle frequency (which is difficult to predict) and propellant slosh modes if damping is less than 1%.

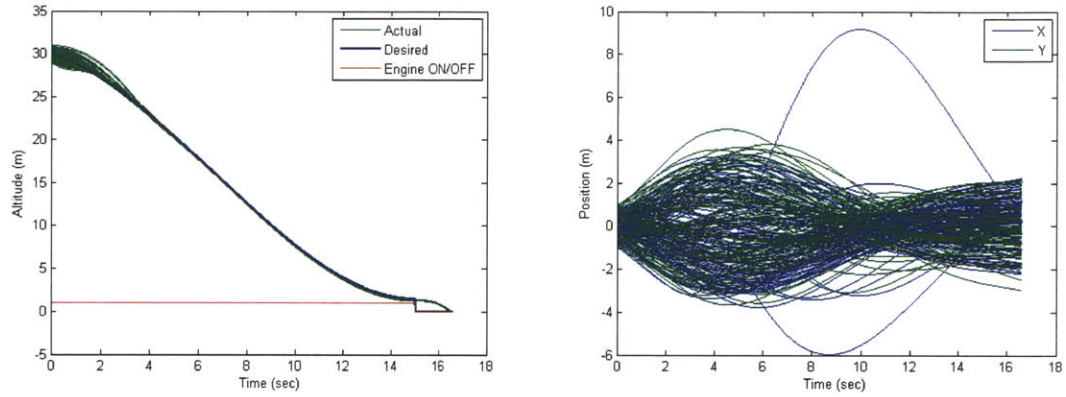


Figure 5-26: Monte Carlo position plots; (a) altitude, (b) lateral

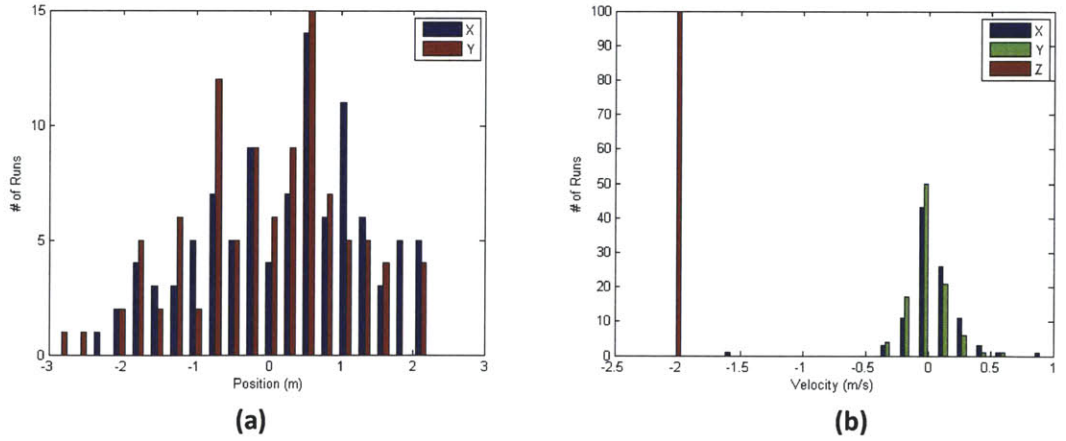


Figure 5-27: Monte Carlo final state histogram plots; (a) lateral positions, (b) velocities

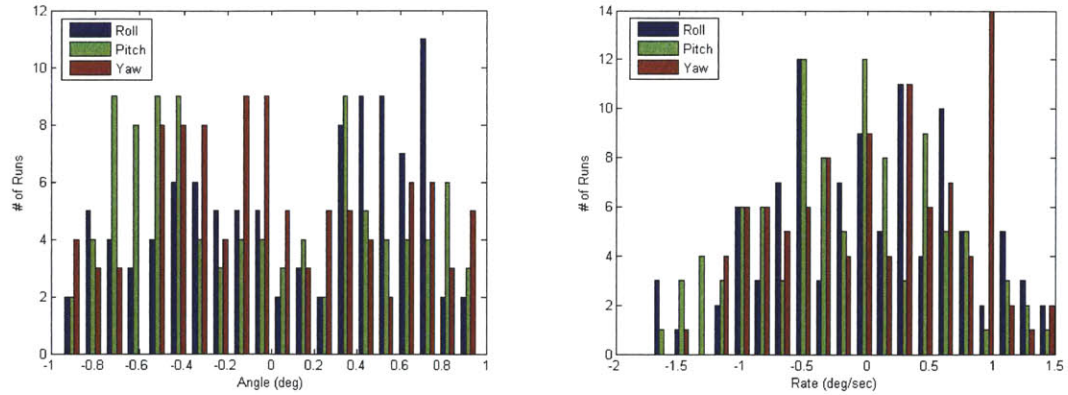


Figure 5-28: Monte Carlo final state histogram plots; (a) angles, (b) rates

5.2.4 Case 1 Summary

In this case study we attempted to separate the design of the high linear impulse actuation system and the low angular impulse actuation system by using a single fixed throttleable main engine to create the main thrust vector and multiple smaller fixed nonthrottleable thrusters to change the direction of the main thrust vector. The actuator and control algorithm architectures were designed to meet the overall performance criteria presented in Chapter 1 and were analyzed using analytical- and simulation-based techniques. A brief qualitative summary of the main conclusions drawn from the actuator architecture, the control algorithm architecture, and the comprehensive control system architecture are given below.

Actuator Architecture

The individual actuators performed well under nominal conditions. The throttleable main engine provided adequate throttle range and continuously variable controllability which enabled excellent altitude control and trajectory tracking, and the ACS thrusters enabled robust and high bandwidth attitude control. However, the advantages of the chosen actuator architecture were less evident in cases where the main thrust vector was misaligned with the vehicle CM. The explicit design requirement for the ACS to counter bias torques incurred by this misalignment ultimately became a driving factor in the selection of thruster number, location, and thrust magnitude.

Control Algorithm Architecture

The control algorithm architecture utilized a variety of feedback schemes and design methodologies which were directly influenced by the different actuator properties: LQR was used to generate a FSFB

linear feedback algorithm to control altitude using the continuously throttleable main engine; phase plane logic was used to control attitude using the signed binary properties of the ACS thrusters; and lead/lag networks were used to control lateral position by commanding the phase plane logic reference states. Analysis showed that the descent controller was remarkably robust to initial condition errors and plant parameter variations. The phase plane controllers where simple, tolerant of a wide variety of errors, and exhibited graceful performance degradation. However, they were not particularly well suited to handle bias moments. The lateral position controllers were most challenging to design because of the nonlinear inner loop characteristics of the phase plane systems. Consequently, controller bandwidth was kept low, which decreased disturbance rejection performance.

Control System Architecture

The control system architecture discussed in Case 1 was able to meet nominal stability and control performance requirements and demonstrate reasonable robustness to parameter variation and initial condition errors. However, the choice to use the ACS to counter bias moments induced by the main throttleable engine resulted in the potential for conditions of inefficient propellant consumption (used to counter the bias moment without imparting linear acceleration) and slosh mode excitation.

5.3 Case 2: Single Gimbaled Throttleable Engine for Altitude and Roll/Pitch Control; Multiple Fixed Nonthrottleable Thrusters for Yaw Control

The second case study will explore the concept of intentional coupling between the roll/pitch control system and the altitude (ΔV) control system. This coupling has the potential to enable both continuously variable rotational control and thereby produce more precise attitude control than expected with bang-bang actuators without compromising altitude control performance. Yaw control will still be accomplished with low-thrust fixed bang-bang actuators, which were shown to perform well in the first case study. The baseline actuator architecture is shown in Figure 5-29.

This type of actuator architecture design has been used successfully on all of the Apollo landing vehicles. The primary advantage is the ability to precisely and efficiently control vehicle attitude while countering bias moments incurred by CM offsets from the main thrust vector (a task at which the previous actuator architecture did not excel). As the gimbal mechanism enables near-continuous pointing of the thrust vector with little energy penalty for changing (or holding) gimbal angles, the control algorithm designer has potentially fewer nonlinear actuator-specific dynamics for which to account, and therefore has more freedom in the design process. However, as this design does require the single gimbaled throttleable engine to control multiple degrees of freedom simultaneously (roll, pitch, and all translational DOF's), inertial coupling and actuator saturation will be sources of concern.

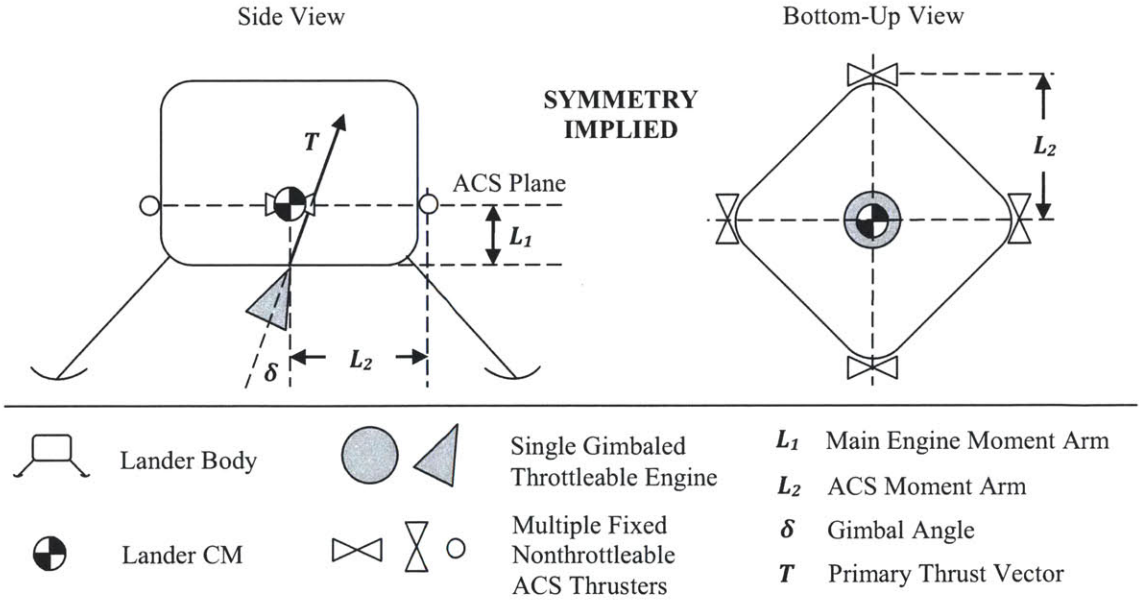


Figure 5-29: Case 2 baseline actuator architecture

5.3.1 Designing the Actuator Architecture

The actuator architecture refers to the specific characteristics and placements of the actuators presented in Figure 5-29. This includes the fixed nonthrottleable thrusters used for yaw control (referred to here as the ACS) as well as the main throttleable engine and its accompanying gimbal dynamics.

Main Engine Parameters

The single main engine (and its gimbal) is responsible for altitude, roll/pitch, and translational control. This means that the thrust and moment range must be selected to fulfill all associated controllability requirements while allowing adequate additional range to compensate for off-nominal cases. A reasonable approach is to select nominal thrust and throttle limits for altitude control capability, and then select the gimbal limits and moment arm for roll/pitch control capability. This approach was applied here, and the final parameters were adjusted based on iterative analysis and simulation results.

As the descent trajectory remains the same for all three case studies, the nominal thrust and throttle range for the main engine are selected to be identical to that of the main engine presented in the first case study; the specific characteristics can be found in Table 5-1. As this actuator performed adequately in the first case study, we expect it to perform adequately here, too.

The gimbal range and associated moment arm were selected to meet the desired control authority range of 0.02 to 1.0rad/sec^2 while being able to counter a 5.0cm offset of the main thrust vector from the vehicle CM. This case study will use a nominal moment arm of 0.5m and gimbal rotation limits of ±15.0deg. If thrust is equal to lunar weight, then the actuator produces approximately 5.0N*m of torque per gimbal degree and is capable of producing more than 1.6rad/sec^2 of angular acceleration. The additional gimbal rotation availability is intended to counter CM offsets, which can be computed as the inverse tangent of the ratio of the CM offset distance to the moment arm length (regardless of thrust level), meaning that a 5.0cm CM offset requires a nominal gimbal angle of 5.7deg. Also, notice that if hardware specifications required that the gimbal rotation limits were smaller, the nominal moment arm must increase proportionally. This would result in a taller lander vehicle with a higher CM and tighter attitude landing tolerances (both of which are undesirable). The full gimbal characteristics can be found in Table 5-5. The simulation model includes rotation quantization, second order dynamics, an associated time delay, and torque saturation limits.

Lastly, it should also be noted that the moment arm (L_1) changes with the vehicle CM as a function of propellant depletion. Although this change is not appreciable during the ~15sec terminal descent phase (typically <1cm vertically) and will not be discussed in detail here, it will be significant during the full deorbit/braking/descent portion of the mission and should not be ignored by the designer.

Table 5-5: Main engine gimbal characteristics

Characteristic	Value
Rotation Limits (δ)	±15.0deg (Roll and Pitch)
Rotation Quantization	0.10deg
Natural Frequency (ω_n)	6.0rad/sec
Damping Constant (ζ)	0.70
Time Delay (T_d)	0.05sec
Torque Limits (M_g)	20.0N*m
Moment Arm (L_1)	0.50m

ACS Parameters

The 'ACS' system for this case study refers only to the fixed nonthrottleable thrusters responsible for yaw control. As in the first case study, we wish to decrease actuator thrust and increase actuator moment arm (L_2) while still allowing for adequate control authority bandwidth and disturbance rejection capability. We will base the design on the yaw ACS architecture from the first case study, and will use eight thrusters located on a plane which nominally intersects the lander CM which allows for quantized throttleability and redundant rotational control without coupled translational acceleration. However, as the roll/pitch actuator architecture is less adept at handling high frequency disturbances (due to additional gimbal dynamics not present in the first case study), the thrust of each actuator will be reduced to 2.25N (~0.15rad/sec^2 when four thrusters fire) in order to limit undesirable rotation induced by inertial coupling during ACS firings. The remainder of the parameters are identical to those found in Table 5-2.

5.3.2 Designing the Control Algorithm Architecture

The control algorithm architecture is responsible for commanding the actuator architecture in order to control the vehicle's states: the fixed nonthrottleable ACS thrusters are commanded to control yaw; and the single throttleable gimbaled main engine is commanded to control roll/pitch, altitude, and lateral

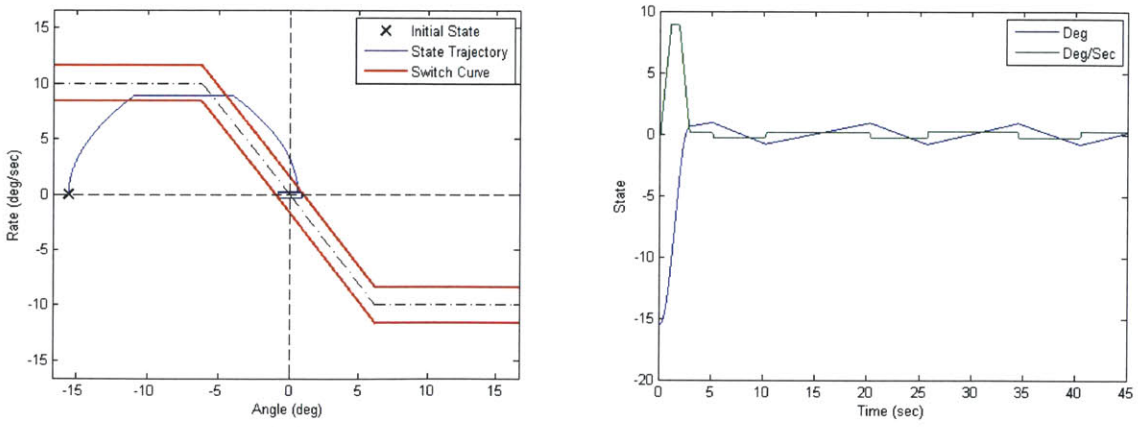
translation. Unlike in the first case study, the actuator responsible for roll and pitch can be well approximated by linear dynamics, and this gives the control algorithm designer much more freedom in the design and analysis process. However, the gimbal does add additional non-minimum phase translation dynamics, which can make the algorithm architecture design process more complicated and will ultimately limit closed loop system bandwidth. As the attitude dynamics form a natural 'inner loop,' to the translational dynamics, they will be discussed first.

Attitude Controllers

The attitude controllers are responsible for calculating the differences between the actual and desired vehicle angles (and associated substates) and determining the desired control effort to be produced by the actuator architecture. The yaw control algorithm will command the ACS thrusters, and the roll/pitch algorithm will command the gimbal angles. Symmetry is assumed in roll and pitch.

Yaw

The yaw DOF is the simplest because it is not subject to large disturbances or complex dynamics. In fact, the dynamics differ very little from those discussed in the first case study, so a similar design process will be applied here. The plant is approximated as a double integrator with a small first order lag and time delay, and the phase plane controller is designed with a deadband of 1.0deg, a drift rate of 10.0deg, a slope of -1.61, and an estimated limit cycle of 0.07Hz with an angle amplitude of 0.86deg and a rate amplitude of 0.25deg/sec. The control power is intentionally kept low to reduce the effects of cross-coupled rotation. An example step response (with additional delay of 30ms) is shown in Figure 5-30.



(a) (b)

Figure 5-30: Yaw step response; (a) phase plane, (b) states vs. time

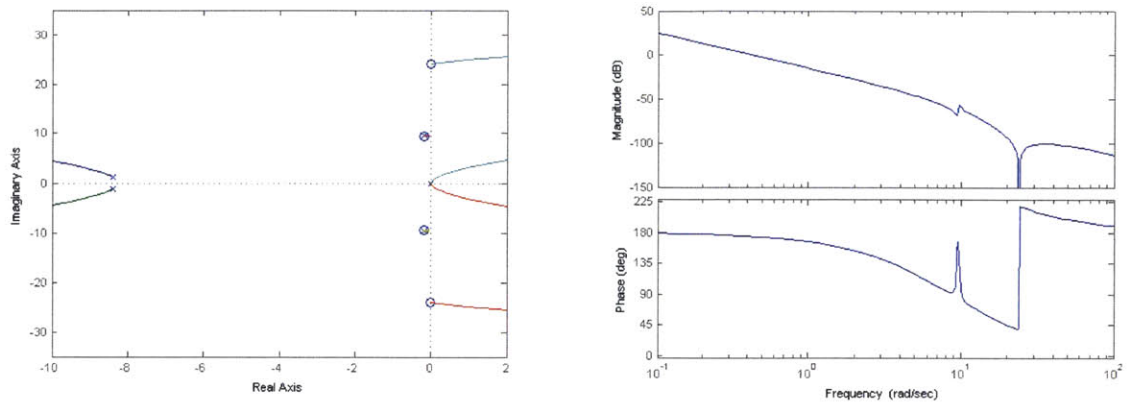
Roll and Pitch

Although the gimbal actuator provides a nearly continuous linear control signal, it also contributes additional dynamics to the roll and pitch system. The state vector of interest for each DOF now consists of three variables and their derivatives. The state space equations are shown in Equation 5-17. They include the basic double integrator, the two gimbal poles, the slosh pole-zero pair, and the 'tail-wags-dog' (TWD) gimbal zeros. Notice that the propellant slosh dynamics are assumed to only be functions of attitude and gimbal dynamics.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{x}_s \\ \ddot{x}_s \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-h_s(k_s h_s + m_s g)}{I} & 0 & \frac{-h_s k_s}{I} & \frac{-h_s \zeta_g}{I} & \frac{\omega_n^2 - TL_1}{I} & \frac{2\zeta\omega_n}{I} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -g - \frac{k_s h_s}{m_s} & 0 & \frac{-k_s}{m_s} & \frac{-\zeta_g}{m_s} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{-\omega_n^2}{I_g} & \frac{-2\zeta\omega_n}{I_g} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x_s \\ \dot{x}_s \\ \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ -1/I \\ 0 \\ 0 \\ 0 \\ 1/I_g \end{bmatrix} M_g \qquad \textbf{5-17}$$

In this case, attitude is denoted by θ, propellant slosh location is denoted by x_s, and gimbal angle is denoted by δ. For clarification, h_s is the propellant slosh mass height above the vehicle CM (0.3m), k_s is the propellant slosh mass equivalent spring constant (2660, resulting in a natural frequency of ~1.5Hz), m_s is the mass of the propellant slosh (30kg), g is the vertical acceleration (nominally lunar surface gravity, 1.635m/sec^2), I is the vehicle's principle moment of inertia (45kg*m^2), ζ_g is the slosh damping factor (0.015), ω_n is the gimbal natural frequency (6.0rad/sec), ζ is the gimbal damping factor (0.7), T is nominal vehicle thrust (572N), L_1 is the gimbal moment arm (0.5m), I_g is the gimbal moment of inertia (0.5m*kg^2), and M_g is the moment of the gimbal actuator.

The nominal plant dynamics are shown in Figure 5-31. The Root Locus shows that the resultant system has six poles and four zeros. The two poles on the far left are from the gimbal dynamics, the two poles at the origin are the basic double integrator attitude system, the oscillatory pole-zero pair is due to slosh dynamics, and the imaginary zeros are the result of the TWD effect of the gimbal-vehicle interaction. As expected, the Bode plot shows an infinite DC gain (with accompanying -180deg phase lag at zero frequency), a small resonant peak at the slosh frequency, and a sharp notch at the TWD frequency.



(a) (b)

Figure 5-31: Roll/Pitch plant characteristics; (a) Root Locus, (b) Bode

As the system is well characterized by linear dynamics, the control algorithm designer has a great deal of freedom. In this case, although the system order is relatively large, some of the dynamics do not heavily influence closed loop performance. For example, the TWD zeros are at a high enough frequency that they will only become dominant factors if system bandwidth is very large, and the oscillatory slosh dynamics can also be easily dealt with by making sure the associated resonant peak is well below the 0dB limit for the range of expected slosh frequencies and damping ratios (gain stabilization). The remaining dynamics can now be characterized by the double integrator and gimbal poles, which can be well controlled by a simple lead controller. The resultant controller (which is the transfer function from attitude error to gimbal torque) is shown in Equation 5-18. Note that the DC gain must be scaled based on engine thrust (nominal gain for vehicle $\boldsymbol{T/W}$=1).

$$\boldsymbol{Gc_{roll,pitch}(s) = 6.0\left(\frac{2s+1}{0.1s+1}\right)} \qquad \textbf{5-18}$$

The controller is designed to give the system a fast rise time (less than one second) while maintaining adequate robustness to parameter variation (which is important, because the effective control signal will be a function of other variables which are likely to change, such as thrust and moment arm). The initial design was selected by placing the zero and pole to increase the closed loop damped natural frequency of the open loop double integrator poles while providing adequate gain and phase margin near the open loop crossover frequency at 2.0rad/sec; the design was later refined through simulation. The controller and complete open loop attitude system Bode plots are shown in Figure 5-32. Notice that the slosh peak is well below the 0dB line, and the TWD zeros are well above the system's bandwidth. A Pade approximation is included to account for the 50ms time delay.

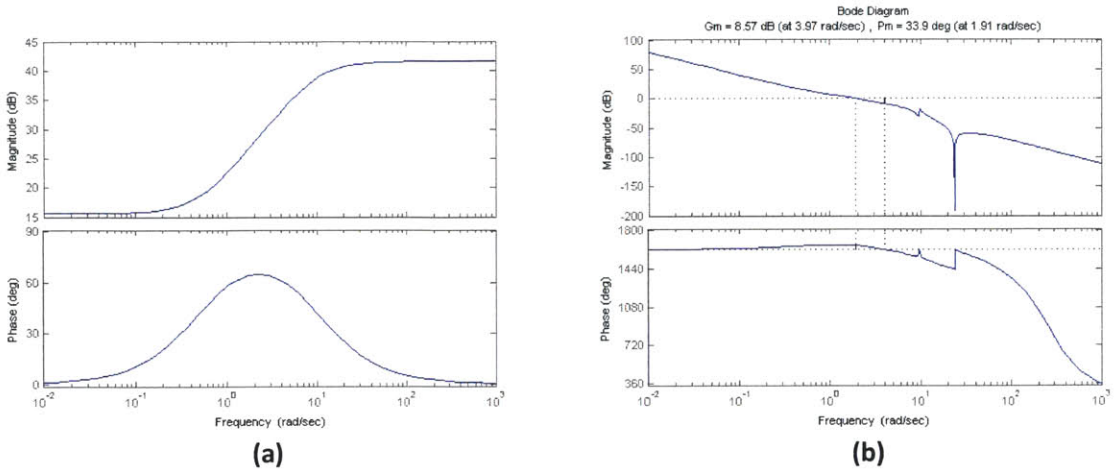
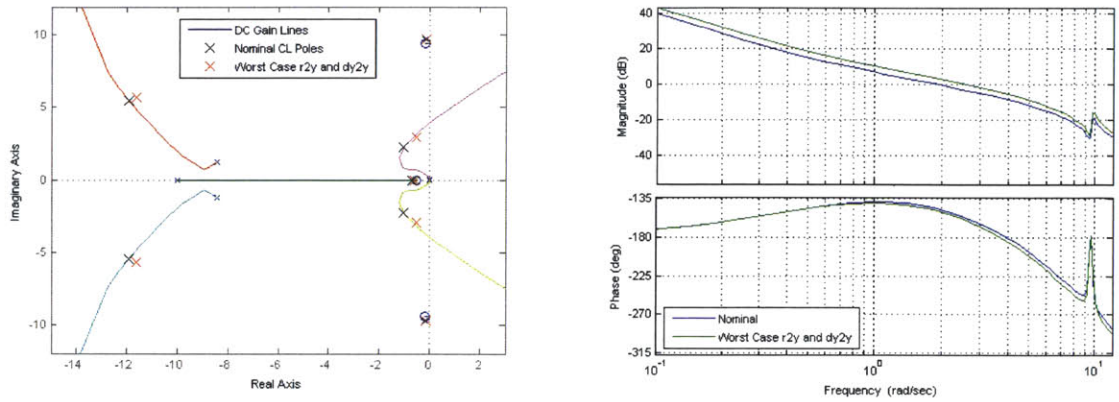


Figure 5-32: Roll/Pitch Bode plots; (a) controller, (b) full open loop system

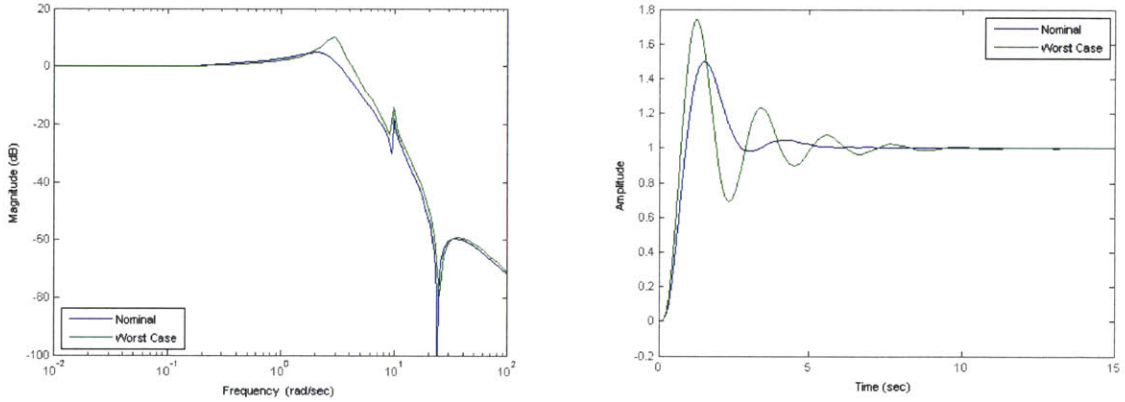
The Robust Control Toolbox in MATLAB was also used to verify robustness to parameter variation by determining the system conditions which produce the highest closed loop and sensitivity transfer function peak magnitudes for a ±10% variation in gimbal natural frequency, vehicle moment of inertia, and gimbal moment arm. Analytic results shows that the system is stable under all variations and can tolerate an additional gain change of -26% to +34% and an additional phase change of ±17.0deg. The worst case condition occurs for the maximum allowable gimbal moment arm and the minimum allowable moment of inertia and gimbal natural frequency. This combination essentially increases the DC gain of the system and drives the open loop double integrator poles back towards the imaginary axis. The analysis also indicates that the system is relatively sensitive to changes in the natural frequency of the gimbal poles, which alters the nominal open loop dynamics. Lastly, further analysis also shows that reasonable variations in slosh mass, natural frequency, or slosh height did not destabilize the system as all resonant open loop peaks were effectively gain stabilized. The nominal and worst case closed loop poles along with associated open loop Bode plots are shown in Figure 5-33; the TWD zeros are not shown within the plot window of the Root Locus as they were not a dominant influence on the overall dynamics.



(a) (b)

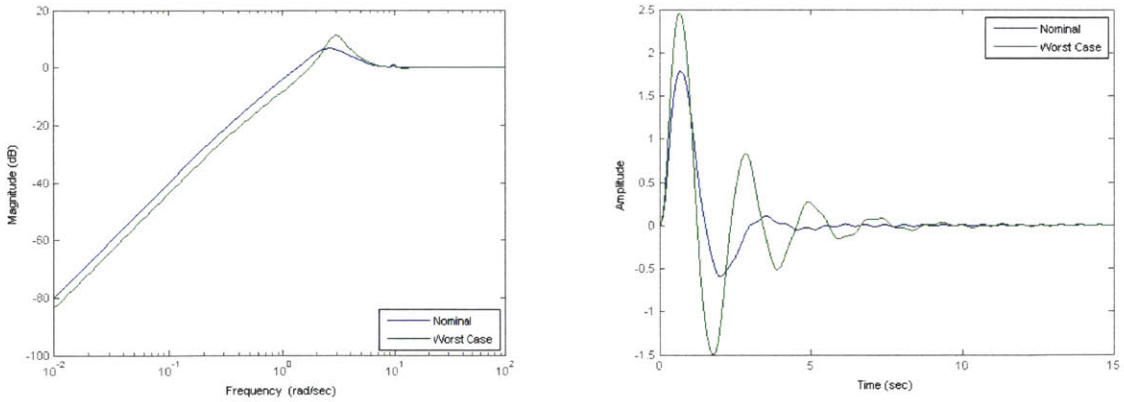
Figure 5-33: Roll/Pitch system with parameter variation; (a) Root Locus, (b) open loop Bode

The closed loop (r to y) and sensitivity (dy to y) transfer functions along with associated impulse and step responses can be seen in Figure 5-34 and Figure 5-35. The nominal closed loop system peaks at 5.0dB near 2.0rad/sec, and the worst case closed loop system peaks at 10.0dB near 2.8rad/sec. The nominal sensitivity function peaks at 6.5dB near 2.8rad/sec, and the worst case sensitivity function peaks at 11.5dB near 3.0rad/sec. Simulations indicate that these are acceptable ranges. Sources of concern are the large overshoots exhibited by both the closed loop and sensitivity functions under worst case conditions, as well as the small peak above 0dB in the sensitivity transfer function at the propellant slosh frequency (which could be excited by an outer-loop lateral position controller).



(a) (b)

Figure 5-34: Roll/Pitch closed loop performance; (a) frequency response, (b) step response



(a) (b)

Figure 5-35: Roll/Pitch closed loop sensitivity; (a) frequency response, (b) impulse response

Position Controllers

The position controllers are responsible for computing the errors between desired and actual position and commanding the necessary actuators to reduce the resultant error. Position control will be accomplished indirectly via vehicle (and thrust vector) tilt, while altitude will be controlled directly via the main engine throttle.

Altitude

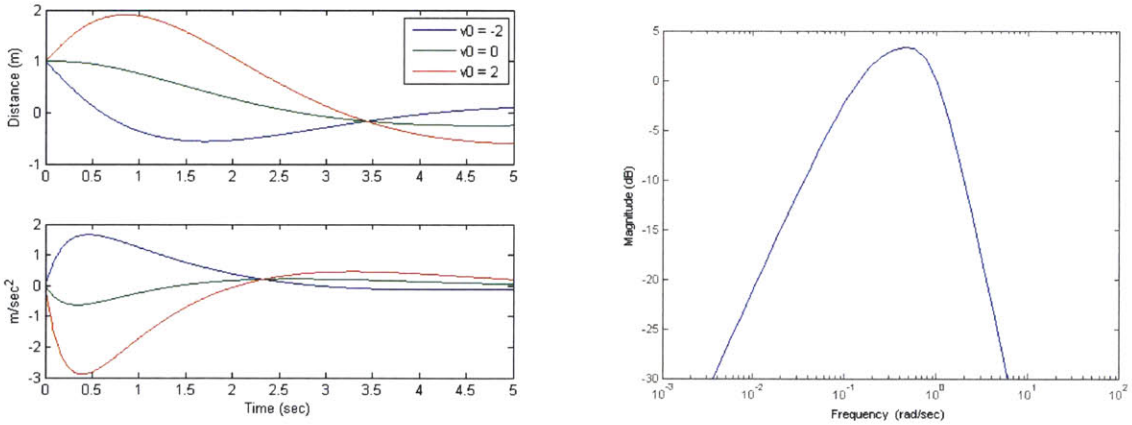
The altitude control problem is very similar to that presented in the first case study. The trajectory and primary actuator are identical, and the only difference is the requirement that the single main engine also control roll/pitch (and lateral translation). In this case we will treat the variation in gimbal angles as a disturbance to the vertical channel and use the same FSFB LQR approach that was shown to work well in the first case study. Once again, the plant model reduces to a double integrator combined with a first order lag and associated time delay identical to Equation 5-12. Although we would like to keep controller bandwidth high (as it was in the first case study) for disturbance rejection purposes, we must lower the bandwidth to reduce the chances of an adverse interaction between multiple control signals applied to the same actuator. In other words, the single gimbaled main engine must respond directly to the commands from the altitude and roll/pitch controllers as well as indirectly to the commands from the lateral position controllers. Changes in gimbal angle will present themselves as disturbances to the

altitude dynamics, and changes in throttle setting will present themselves as disturbances to the attitude/lateral dynamics. As long as the control signals are small, the associated disturbances will also be small. However, if we try to push controller bandwidth too high (in either/both the attitude or/and the altitude channels), the associated disturbances will be large. Simulation indicates that this may result in system instability, particularly if nonlinearities begin to dominate the dynamics (e.g. saturation of control signal, rotation due to inertial coupling). The initial altitude controller has therefore been chosen in a similar manner as in the first case study but with additional penalty on control effort. The gains of the final controller and resultant closed loop poles are shown in Equations 5-19 and 5-20. Notice that the two real poles are largely unchanged while the natural frequency of the complex poles has been reduced.

$$\boldsymbol{K = [1.04 \quad 1.64 \quad 0.29 \quad -0.14]} \tag{5-19}$$

$$\boldsymbol{eig[A - BK] = \begin{bmatrix} -4.99 \\ -0.63 + 0.62i \\ -0.63 - 0.62i \\ -0.19 \end{bmatrix}} \tag{5-20}$$

Time and frequency domain disturbance rejection performance is shown in Figure 5-36. Notice that the shapes of all curves are similar to those in Figure 5-17 but exhibit characteristics of decreased bandwidth (e.g. slower recovery time, higher input sensitivity magnitude). The input sensitivity function shows acceptably low gains at low and high frequencies, but the peak at mid frequencies (near the crossover frequency of the roll/pitch system) is above 0dB. Simulation has shown this to be a reasonable compromise necessary to decrease overall chances of instability.



(a) (b)

Figure 5-36: Altitude controller performance; (a) time domain tracking, (b) input sensitivity function

Lateral

As in the first case study, the lateral position controllers are the most difficult to design. Although we do have confidence in the linearization of the vehicle and actuator dynamics (which was a concern in the first case study), the dynamics themselves are complex enough to warrant careful attention. Furthermore, actuation in the lateral channels must be accomplished indirectly by changing vehicle attitude. As this must be done by use of the gimbal, the translation dynamics now also include a non-minimum phase component (in the M_g to x transfer function), which makes achieving robust, high bandwidth performance challenging. The full system dynamics (attitude controller omitted, slosh mass assumed to not affect composite CM) are shown in Equation 5-21. All symbols are the same as in Equation 5-17 with the addition of lateral position and vehicle mass, denoted by x and m, respectively.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{x}_s \\ \ddot{x}_s \\ \dot{\delta} \\ \ddot{\delta} \\ \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{-h_s(k_s h_s + m_s g)}{I} & 0 & \frac{-h_s k_s}{I} & \frac{-h_s \zeta_g}{I} & \frac{\omega_n^2 - TL_1}{I} & \frac{2\zeta\omega_n}{I} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -g - \frac{k_s h_s}{m_s} & 0 & \frac{-k_s}{m_s} & \frac{-\zeta_g}{m_s} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-\omega_n^2}{I_g} & \frac{-2\zeta\omega_n}{I_g} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{-T}{m} & 0 & 0 & 0 & -\frac{T}{m} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x_s \\ \dot{x}_s \\ \delta \\ \dot{\delta} \\ x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ 1/I \\ 0 \\ 0 \\ 0 \\ 1/I_g \\ 0 \\ 0 \end{bmatrix} M_g \qquad \text{5-21}$$

One approach to control algorithm design is to recognize that the inner loop attitude system is of relatively high order and the lateral position system may benefit from a FSFB controller. However, we showed through the design of the roll and pitch controllers that some of the higher order dynamics were not dominant, and it may be reasonable to approximate the inner loop system as a SISO transfer function as long as necessary robustness verification measures are taken. This will allow us to design a classical error-based feedback controller, which has been shown to work well on Earth-based lander prototypes and in lunar lander studies [57] [26] [42]. We will begin by approximating the closed loop dynamics with the feedback architecture shown in Figure 5-37. Notice that the TWD and slosh dynamics have been removed, and that lateral acceleration is a function of both gimbal angle and vehicle attitude.

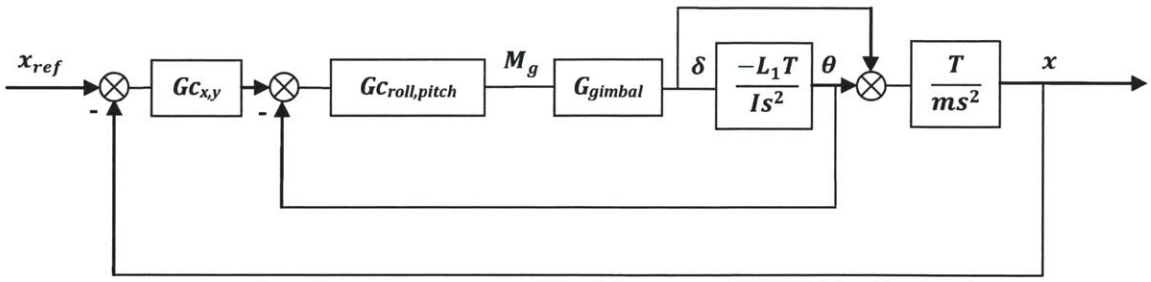
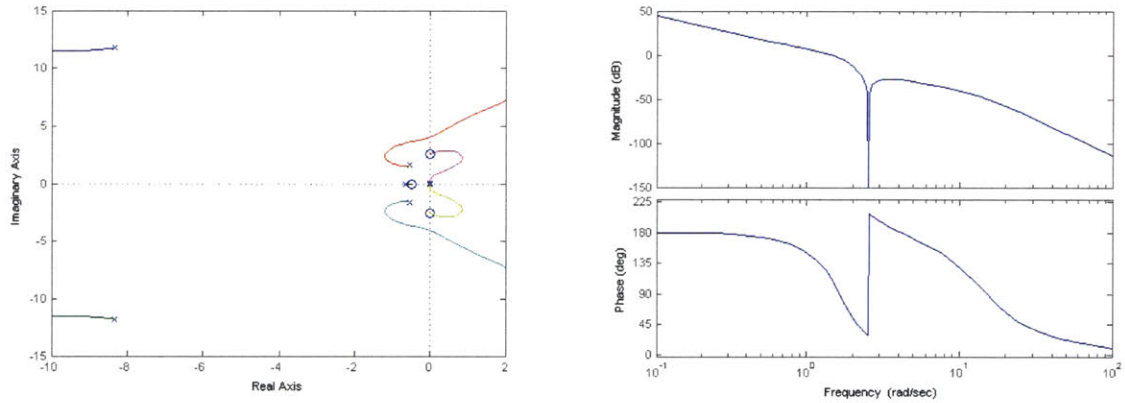


Figure 5-37: Lateral position feedback architecture

The nominal open loop lateral system (which includes the closed loop inner attitude system) now contains two sets of oscillatory poles, a double integrator, a real pole, a real zero, and two purely imaginary zeros. The associated Root Locus and Bode plots are shown in Figure 5-38. Notice that the three zeros tend to trap the nearby poles, which makes it very difficult to push open loop system bandwidth above 1.0rad/sec.



(a) **(b)**

Figure 5-38: Lateral plant characteristics; (a) Root Locus, (b) Bode

The initial controller was designed to include an integrator to null steady state error and a second order lead compensator for gain and phase increase at mid-to-high frequencies. The zeros were selected to give maximum phase margin near 1.0rad/sec (which was the target crossover frequency), which allowed the controller magnitude to be above 0dB at 0.01 rad/sec (to provide timely disturbance rejection to CM offsets and lateral velocity errors). Although the author found that reasonable stability, performance, and robustness were achievable for the single DOF nonlinear system with a wide range of controller parameters, interaction with the other DOF's was much more difficult to account for. The initial

controller was therefore heavily tuned in accordance with results from the 6DOF nonlinear simulations in order to reduce the instances of instability due to these nonlinear interactions. The controller output (commanded attitude) was also limited to ±60.0deg, and an anti-windup function was added to the integrator. The final controller (which is the transfer function from position error to commanded attitude) is shown in Equation 5-22, and the associated Bode plots are shown in Figure 5-39. Notice that the final crossover frequency is near 0.5rad/sec. Although the lower bandwidth compromises step response and disturbance rejection, it reduces undesirable interaction with the other DOF's and ultimately results in more consistent favorable terminal landing conditions.

$$Gc_{x,y}(s) = 0.0152\left(\frac{16s^2 + 5.7s + 1}{s(0.01s^2 + 0.14s + 1)}\right) \qquad 5\text{-}22$$

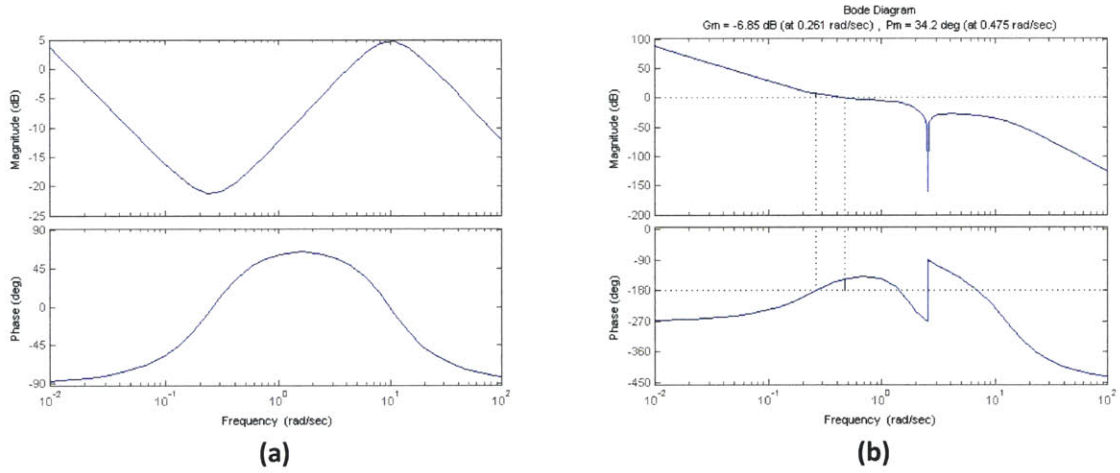
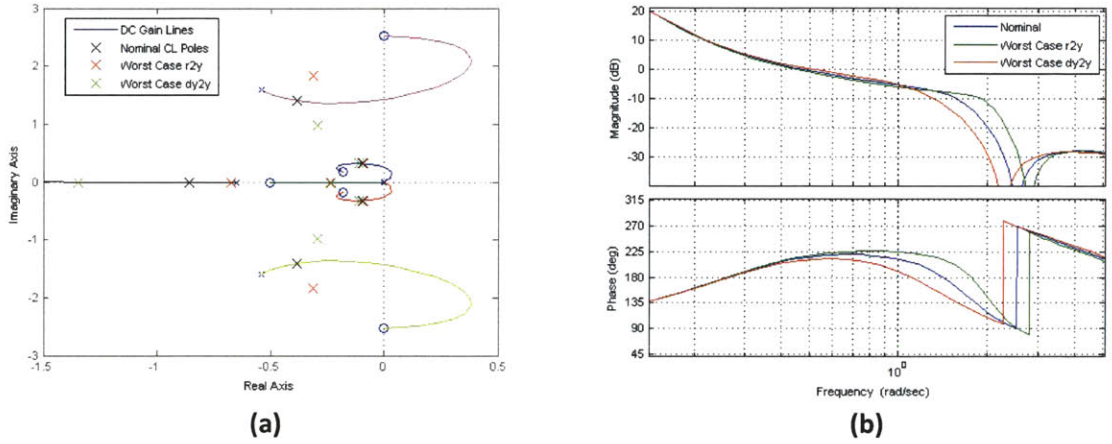


Figure 5-39: Lateral Bode plots; (a) controller, (b) full open loop system

Robustness tests with a ±10% variation in vehicle moment of inertia, nominal thrust, and gimbal natural frequency also indicate that the linear system is stable under all variations and can tolerate an additional gain change of -39% to +63% and an additional phase change of ±27.1deg. The worst case closed loop (r to y) condition occurs for the minimum moment of inertia, the maximum vehicle mass, and the minimum gimbal natural frequency. This increases the inner loop gain while decreasing the outer loop gain; the net result is the two pairs of lower frequency oscillatory poles slide slightly closer to the imaginary axis. The worst case sensitivity (dy to y) condition occurs for maximum moment of inertia, minimum system mass, and maximum gimbal natural frequency (the opposite of the r to y conditions). This causes the mid-frequency poles to decrease in both natural and damped frequency; the net result is

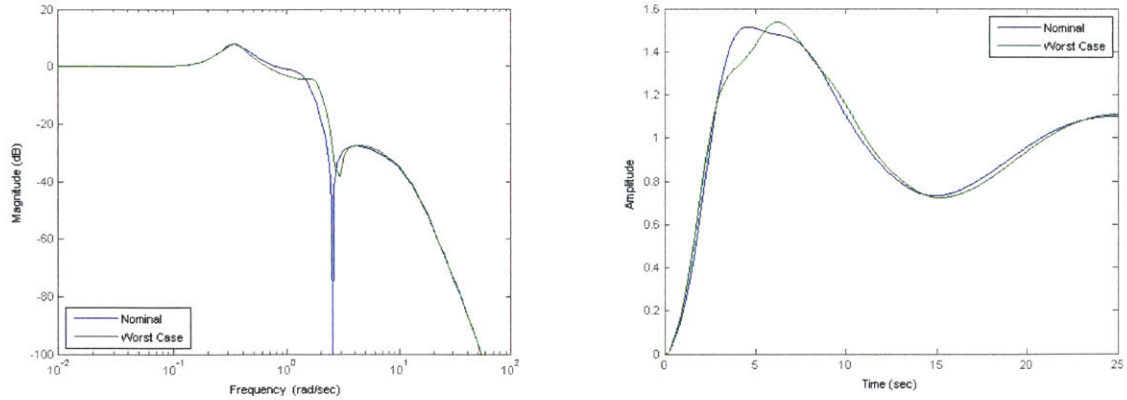
a slightly more sluggish disturbance rejection. The nominal and worst-case pole locations are shown in Figure 5-40 (higher order dynamics not shown) along with the associated open loop Bode plots. Notice that the parameter variations change the location of the notch in the magnitude plots, but that the crossover frequency (which remains largely unchanged) is low enough that gain and phase margin are not adversely affected.



(a) (b)

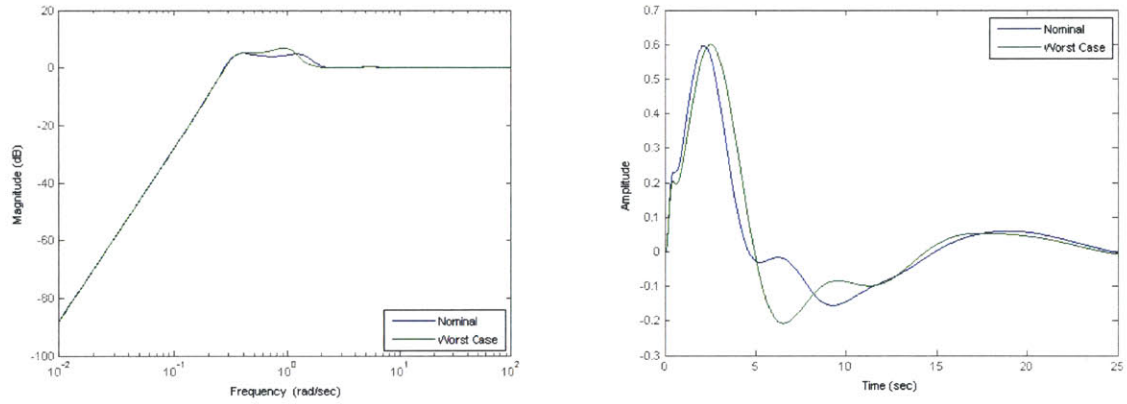
Figure 5-40: Lateral system with parameter variation; (a) Root Locus, (b) open loop Bode

The closed loop (r to y) and sensitivity (dy to y) transfer functions along with associated impulse and step responses can be seen in Figure 5-41 and Figure 5-42. Both the nominal closed loop system and the worst-case closed loop system peak at 7.8dB near 0.3rad/sec. The nominal sensitivity function peaks at 5.3dB near 0.4rad/sec, and the worst-case sensitivity function peaks at 6.8dB near 1.0rad/sec. The peak magnitudes are similar to those found in the inner loop analysis, and the associated frequencies are lower; simulation indicates that these are acceptable.



(a) (b)

Figure 5-41: Lateral closed loop performance; (a) frequency response, (b) step response



(a) (b)

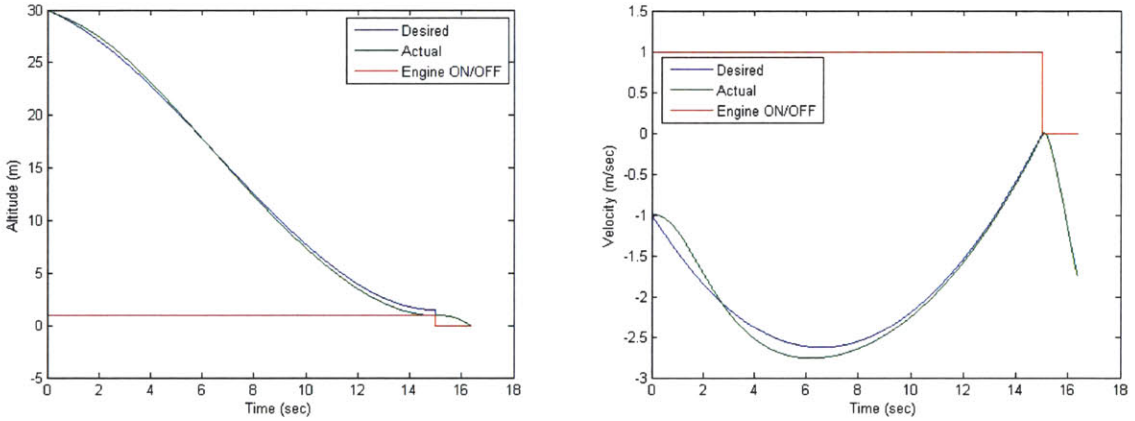
Figure 5-42: Lateral closed loop sensitivity; (a) frequency response, (b) impulse response

5.3.3 Performance and Verification

The same nonlinear simulation environment used in the first case study is used to verify the collective performance of the control system architecture in all six DOF's; the simulation has been modified to include the additional actuator dynamics. As previously mentioned, it was used heavily in the design process of the lateral position controllers. It will be used here to verify the overall performance and robustness of the complete vehicle.

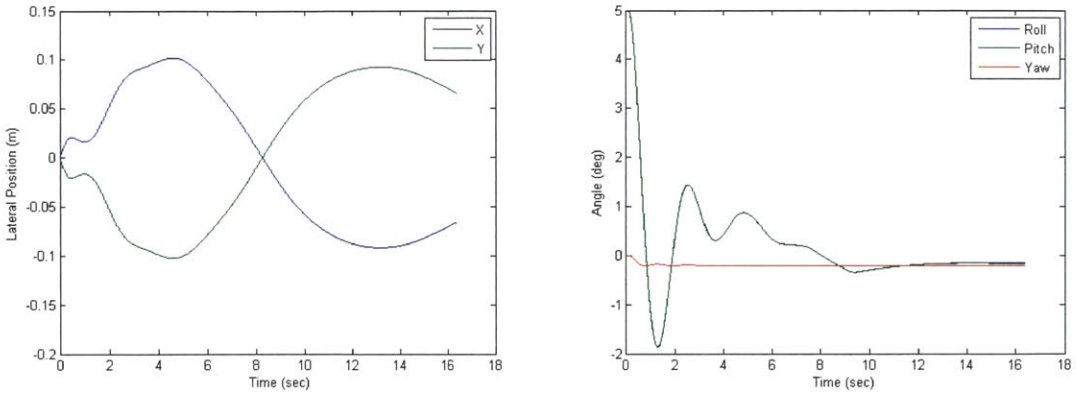
Nominal Performance

Under nominal conditions the lander begins terminal descent at 30.0m altitude, a 1.0m/sec descent velocity, and with its thrust vector pointed nearly perpendicular to the lunar surface (5.0deg pitch and roll). The following performance plots show that the lander tracks the fixed 'guidance' descent trajectory well, commands engine shutdown after 15.0sec, and descends to the lunar surface with all state conditions within the desired margins. The vertical position error at engine cut-off is now slightly larger than in the first case study due to lower controller tracking bandwidth, but attitude is generally less oscillatory and gimbal commands cease entirely after ten seconds. The gimbal quantization indicates that a low frequency limit cycle would exist for longer duration descent trajectories, although it does not play a significant role in this scenario. Lastly, notice that the actual roll/pitch moment applied to the vehicle is proportional to the gimbal angle; this implies that there is negligible coupled disturbance between the altitude and attitude/position channels, which is a desirable condition for reasonable stability and performance.



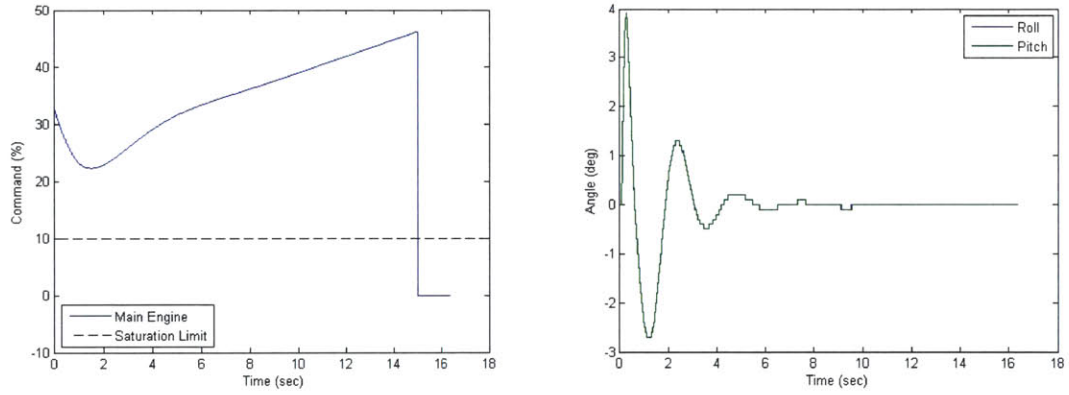
(a) (b)

Figure 5-43: Nominal descent states; (a) altitude, (b) vertical velocity



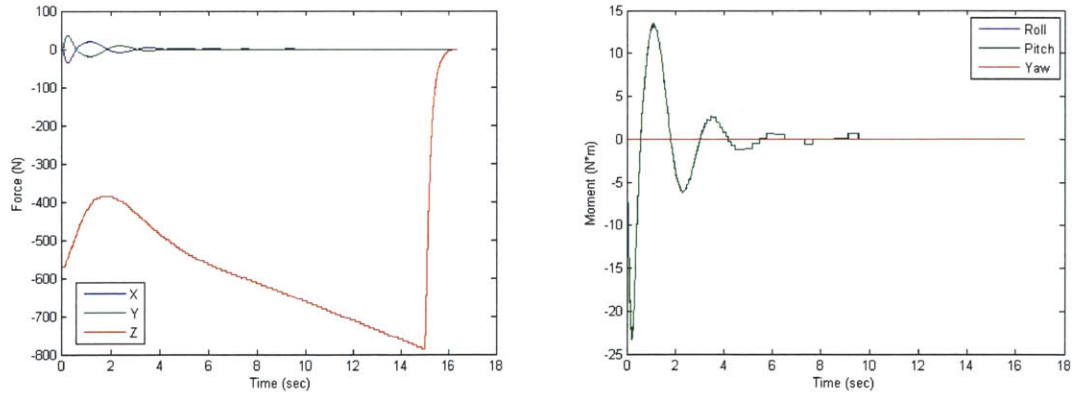
(a) (b)

Figure 5-44: Nominal descent states; (a) lateral positions, (b) angles



(a) (b)

Figure 5-45: Nominal descent control; (a) throttle, (b) gimbal angles



(a) (b)

Figure 5-46: Nominal descent actual body frame control; (a) forces, (b) moments

Variation of Single Parameters

The simulation is also used to determine limits of individual parameter variation for which safe landing conditions can still be achieved. This type of analysis is most useful in revealing stability and performance sensitivities to changes in single factors and is also useful for comparison between case studies. The results for this case study, along with the results from the first case study, are shown in Table 5-6. The variation limits are similar between the two case studies, and this is to be expected because the two control system architectures were designed to meet the same performance and robustness goals. The most notable difference, which is not apparent from the table of variations, is the qualitative behavior of the vehicle as some of the parameter variation limits are approached: worst case conditions in the first case study often resulted in larger limit cycles which were still stable in the sense of Lyapunov; worst case conditions in the second case study sometimes resulted in divergent oscillatory behavior, which is indicative of right hand plane (RHP) poles. This is partially due to the inability of the linear control laws to handle the nonlinearities of the actual system.

Table 5-6: Max single parameter variation limits before landing constraint violation (Cases 1 and 2)

Parameter	Case 1	Case 2	Comment
ACS Thrust Scale Factor	9	20	Applied to all thrusters
Main Engine Thrust Scale Factor	0.5, 2.6	0.5, 2.6	Min, max
Initial Roll or Pitch Error	65deg	90deg	On one axis
Initial Lateral Position Error	15m	15m	On one axis
Initial Vertical Position Error	35m, 8m	30,16m	Height above, below 30m
Initial Lateral Velocity Error	3m/sec	3m/sec	On one axis
Initial Vertical Velocity Error	5m/sec, 6m/sec	5m/sec, 8m/sec	Velocity above, below -1m/sec
Overall Time Delay	120ms	160ms	Applied to all control feedback states
X or Y CM offset	10cm	5.5cm	On one axis

Monte Carlo Simulations

Finally, Monte Carlo simulations were conducted to verify the robustness of the complete 6DOF nonlinear system to the same types of stochastic parameter and initial condition variation performed on the first case study. These limits of these variations are shown in Table 5-4. Representative results from 100 runs are shown in Figure 5-47 (position plots), Figure 5-48 (position and velocity histograms), and Figure 5-49 (attitude and rate histograms). Ultimately, the results were comparable to those of the first case study. Only one run was found to violate the terminal landing conditions set in Table 1-3, and this was due to gimbal angle and throttle saturation, which was induced by a 4cm CM offset on both axes, a 42ms time delay, a low vehicle mass, and large initial velocity and position errors.

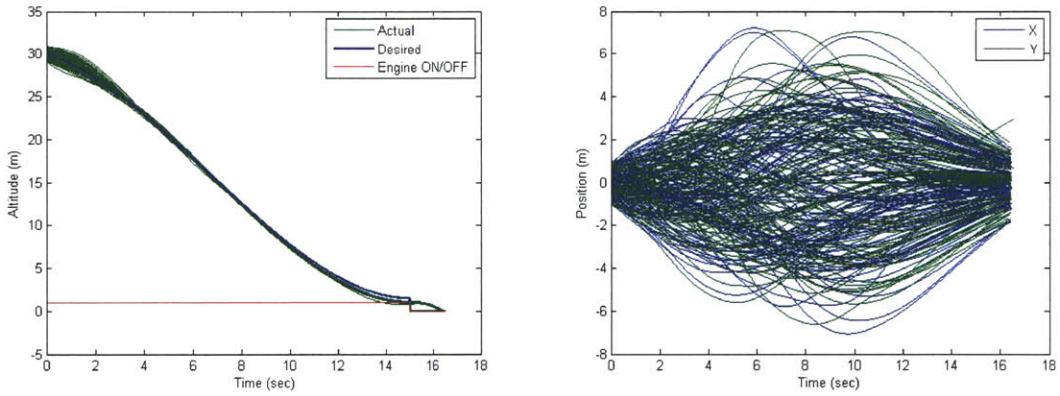


Figure 5-47: Monte Carlo position plots; (a) altitude, (b) lateral

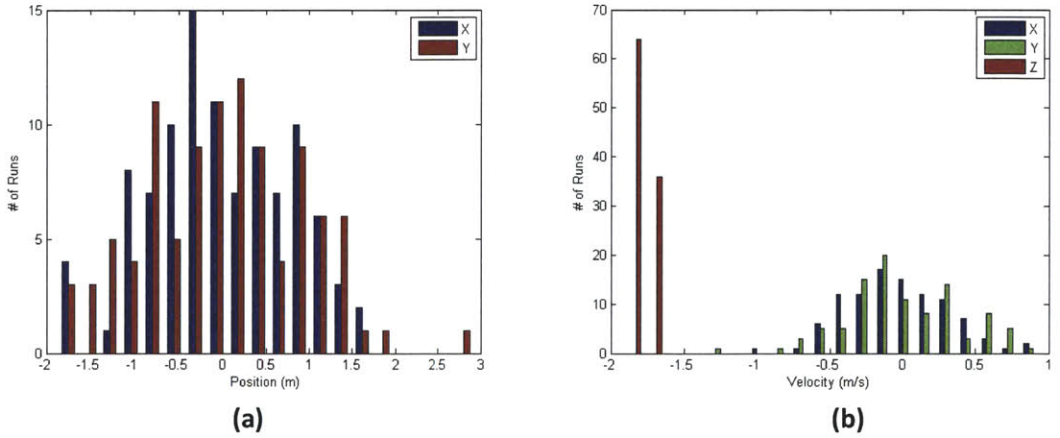


Figure 5-48: Monte Carlo final state histogram plots; (a) lateral positions, (b) velocities

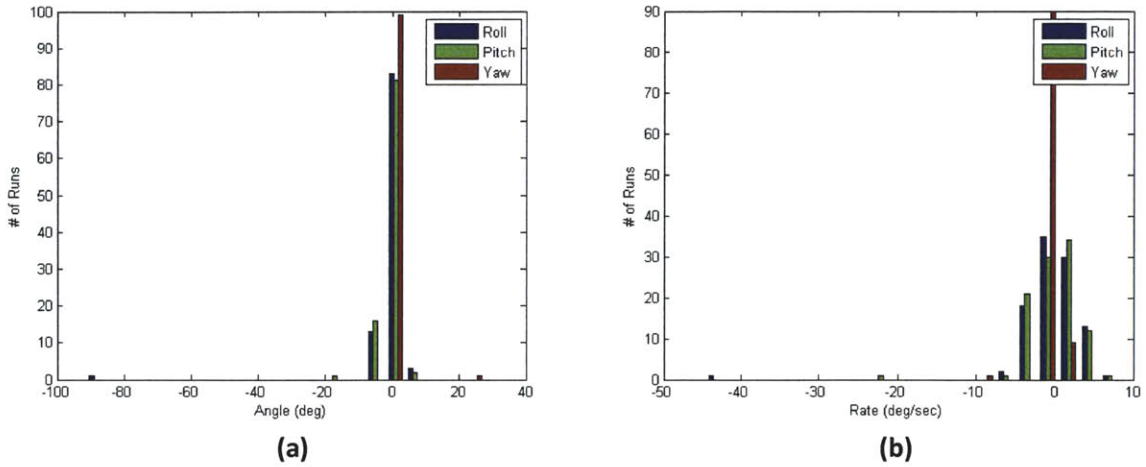


Figure 5-49: Monte Carlo final state histogram plots; (a) angles, (b) rates

5.3.4 Case 2 Summary

In this case study we attempted to couple the design of the high linear impulse actuation system with the roll/pitch actuation system by using the single throttleable gimbaled main engine to exert force along the Z body axis and to exert moments about the body X and Y axes. Fixed nonthrottleable ACS thrusters were used to control yaw. The actuator and control algorithm architectures were designed to meet the same overall performance criteria that were presented in Chapter 1 and used in the first case study.

Actuator Architecture

The actuators performed well under nominal conditions. The fixed nonthrottleable ACS thrusters were able to overcome reasonable disturbances and provide adequate control power to maintain the desired yaw angle, and the throttleable gimbaled main engine was able to precisely and efficiently counter CM offsets while still providing adequate control power for the altitude and roll/pitch systems. However, using the same actuator to control multiple degrees of freedom simultaneously meant that gimbal angle saturation and control signal coupling (e.g. increasing throttle also increases moment) had to be specifically addressed and their effects thoroughly evaluated.

Control Algorithm Architecture

The control algorithm architecture used a variety of algorithms and verification techniques. A phase plane and an LQR controller were adapted from the first case study to control yaw and altitude,

respectively, while roll/pitch and lateral position were controlled with classical error feedback algorithms. The classical controllers were generated using linear performance and robustness evaluation techniques and then further refined using the nonlinear simulation. Although the gimbal mechanism offered the potential for near-continuous and linear roll/pitch control authority, it introduced additional dynamics which ultimately limited the ability to respond to high frequency commands. Nonlinear coupling between different degrees of freedom also became a driving factor in the algorithm design and verification process, and required the designer to rely heavily on simulation.

Control System Architecture

The control system architecture discussed in Case 2 was able to meet similar general stability, performance, and robustness metrics as in Case 1. It was able to efficiently and precisely counter CM offsets while still providing adequate control to all other degrees of freedom. However, the choice to use a single actuator to control multiple degrees of freedom meant that control signal coupling and resultant dynamics were a persistent concern.

5.4 Case 3: Multiple Nonthrottleable Thrusters for Altitude and Attitude Control (Three Fixed, One Gimbaled)

The third case study explores the possibility of further control system coupling by using the same set of nonthrottleable actuators to control altitude (ΔV) and attitude. By combining the attitude and altitude actuation systems, this architecture offers the potential for fewer vehicle systems by alleviating the need for separate actuators to control separate degrees of freedom. In this case, all four thrusters will be used to control the primary thrust vector magnitude and location (thereby controlling altitude, roll, and pitch), and one thruster will also be gimbaled to provide small moments about the Z axis (thereby controlling yaw). The baseline actuator architecture is shown in Figure 5-50.

This type of actuator layout is very similar to that which was used on the Surveyor landing vehicles, as it requires all actuators to operate simultaneously to create and control a single primary thrust vector. The advantage of this type of architecture is its potential for mass/cost reduction (fewer separate actuator systems) and redundancy (if five or more thrusters used). The use of only nonthrottleable actuators also means throttleable engines (which have been used on all known successful lunar landers) are not required, and this may be one of main reasons similar architectures are under consideration by the ESA lunar lander design team [28] and at least one of the GLXP teams [40]. However, as the use of nonthrottleable actuators to simultaneously control multiple DOF's requires the hardware to operate in a pulsed actuation scheme, control signal saturation and digital signal effects will be sources of concern.

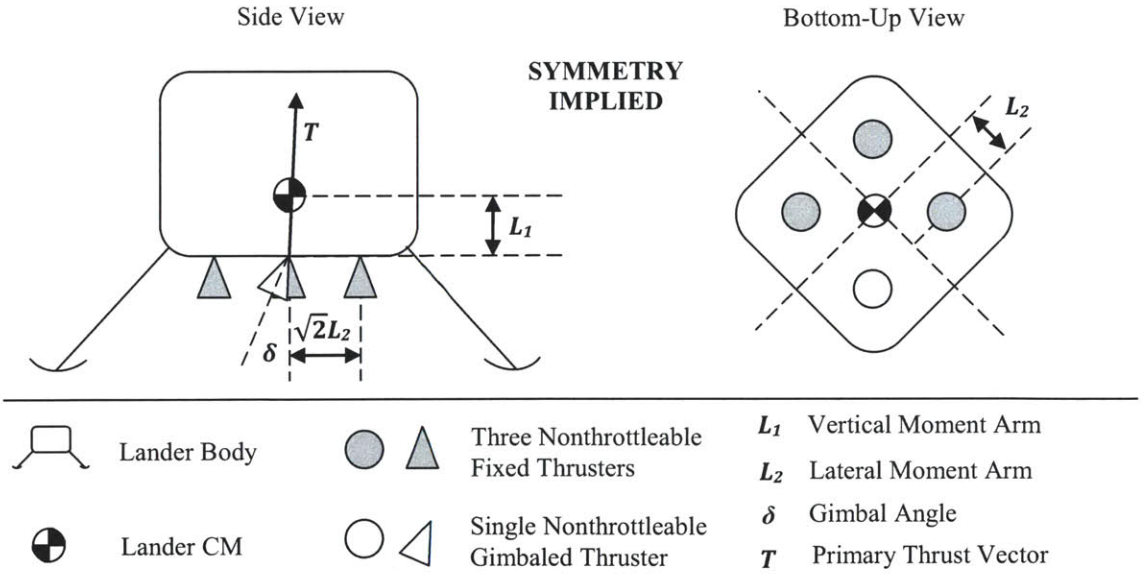


Figure 5-50: Case 3 baseline actuator architecture

5.4.1 Designing the Actuator Architecture

The actuator architecture is responsible for executing the desired closed loop control commands and must be designed to meet the general control authority guidelines established at the beginning of this chapter. In the previous case studies we were able to largely decouple the actuator architecture parameters from the actual control algorithm design/scheme, but in this case we must be careful to account for the desired feedback architecture and its implications for actuator performance.

Thruster Parameters

The chosen architecture requires that four thrusters control all degrees of freedom simultaneously. This means that each actuator must possess the capability to operate within the confines of the pulse scheme commanded by the control algorithm architecture (which will be discussed in Section 5.4.2), and it also means that the composite set of thrusters must be capable of providing the control authority necessary to control all vehicle states with a 5.0cm CM offset. Initial actuator architecture values were selected from the relationships derived in Equation 3-1 and then refined through analysis and simulation. The final values are shown in Table 5-7.

Notice that the fixed thrust value results in a net thrust (2x the lander's lunar surface weight) greater than one would expect to desire during the nominal descent trajectory used for these case studies. This value was chosen so that the thrusters could nominally operate in a 50% 'throttle' pulsed scheme (on and off for equal time) to provide enough impulse to offset lunar gravity; the remainder of the 'throttle' could then be used to create differential thrust and angular control moments. Although this should work well under nominal conditions, it does mean that the maximum and minimum 'throttle' in the vertical channel must be limited (which may decrease descent trajectory efficiency) to allow for adequate portions of the command period to be allotted for the execution of angular control commands. Simulation shows that for a given continuous 'throttle' range of 20%-80%, then 25%-75% vertical channel 'throttle' limits allow for adequate angular control under most conditions while still providing adequate disturbance rejection and tracking of the vertical trajectory. The resultant vertical control powers are therefore $\pm 0.82 m/sec^2$ (± 0.5 lunar g's), and the angular control powers are at least $0.13 rad/sec^2$ (for ±5% throttle) and at most $0.76 rad/sec^2$ (for ±30% throttle). Lastly, notice that delays associated with the thrusters are larger than those of the previous ACS systems. This is natural consequence of having actuators with higher thrust. These penalties should not be ignored, as they will play a large role in the design of the feedback architecture and will ultimately limit control rate.

Table 5-7: Nonthrottleable thruster characteristics

Characteristics	Value
Thrust ($T/4$)	286.0N
Moment Arm (L_2)	0.50m
Time Constant (τ)	0.02sec
Minimum On/Off Time (σ)	0.04sec
Time Delay (T_d)	0.03sec
Pulse Quantization	0.001sec

Gimbal Parameters

The gimbal is responsible for angling the nozzle of the fourth thruster to create a yaw control moment. The chosen parameters are shown in Table 5-8; the maximum control power during hovering flight is 0.44rad/sec^2 and the minimum control power during hovering flight is 0.003rad/sec^2. In this case we have chosen to model the gimbal as a first order differential equation (see Equation 4-5) with a small associated time delay, rotation saturation limits, and rotation quantization. Notice that if the gimbal rotation axis is not aligned with the vehicle CM (nonzero L_1), then a nonzero gimbal angle will also incur a smaller undesired moment about the roll and pitch axes. This moment could be fed-forward to the roll and pitch controllers, but simulations show that it is sufficient to simply treat the roll/pitch moments as disturbance signals and allow the roll/pitch/yaw controllers to operate independently.

Table 5-8: Gimbal characteristics

Characteristic	Value
Rotation Limits (δ)	±15.0deg
Rotation Quantization	0.10deg
Time Constant (τ)	0.10sec
Time Delay (T_d)	0.01sec
Primary Moment Arm ($\sqrt{2}L_2$)	0.71m
Secondary Moment Arm (L_1)	0.10m

5.4.2 Designing the Control System Architecture

The control system architecture is most difficult to design for this case study because it is responsible for commanding four nonthrottleable thrusters and one gimbal to work together to simultaneously control all six degrees of freedom. As the actuators are unable to change their individual thrust levels, each actuator must use short pulses to vary their impulse in order to impart the desired linear and angular acceleration on a 'real-time' basis. In this case study we will continue to use different controllers for each DOF: an altitude controller to generate the desired net vertical impulse; roll/pitch controllers to generate the desired roll and pitch impulses; and a yaw controller to generate the desired yaw impulse. The desired vertical impulse will be used to set the nominal pulse length of all four thrusters, the desired roll and pitch impulses will be used to add/subtract pulse lengths to respective pairs of thrusters, and the desired yaw impulse will be used to orient the gimbal. Although pulsed control schemes were briefly discussed in the first case study, they will be discussed here in more detail.

Pulsed control schemes require that actuators turn on and off quickly to produce a desired total impulse. They are most common in systems with fixed nonthrottleable actuators and ACS systems where a linear control signal is desired but cannot be directly realized on hardware by changing thrust levels. Instead, pulsed schemes command the actuator to turn on and off over a specified command period such that the resultant impulse is equal to the impulse which would have occurred had the actuator's thrust level been changed but been allowed to remain on over the entire time period. This principle is demonstrated in Figure 5-51. Note that additional actuator dynamics (see Figure 5-7) have been omitted for clarity but will be included in all simulation and analysis.

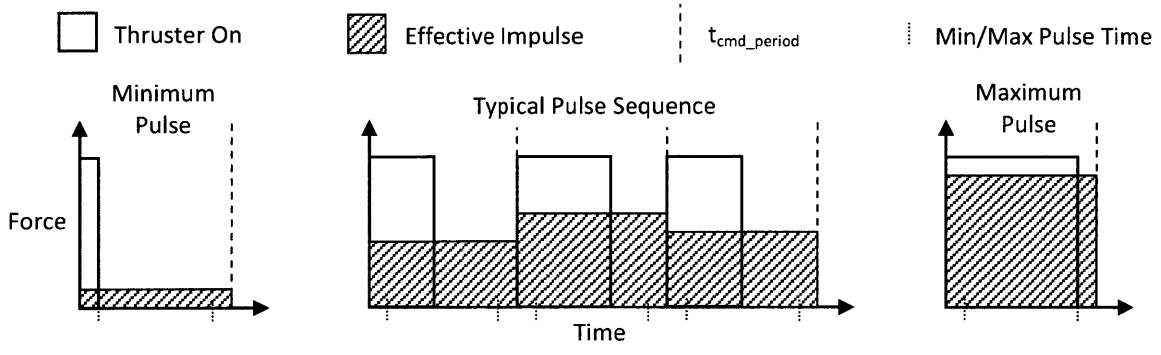


Figure 5-51: Typical pulse profiles within continuously variable control range

As seen on the left and right sides of Figure 5-51, this type of control scheme will always have a minimum and maximum pulse which correspond to the lower and upper 'throttle' limits of the continuously variable control signal. They are functions of the actuator's minimum on time and the chosen control command period (these are the same 'throttle' limitations discussed in the Thruster Parameters section). Although we can command the thrusters to be completely off or completely on (the true minimum and maximum pulses within a given command period), we are relying on the thrusters to work together to produce a continuously variable impulse. If one or more thrusters are commanded into these saturation regions, precise impulse control will be compromised and collective control of all DOF's will be penalized. We must therefore carefully design both the actuator architecture and the control algorithm architecture to ensure that chances of this occurring are sufficiently reduced. This is typically accomplished by picking a large command period and low-impulse descent trajectory (which will compromise propellant use efficiency) to ensure that control signal saturation does not occur. Unfortunately, the pulsed control scheme introduces an effective time delay (see Equation 4-10), which is roughly proportional to the length of the control command period. The command period must therefore be chosen carefully to balance these penalties: it must be long enough to provide adequate impulse variability; it must be short enough as to not add excessive time delay; and it must also be achievable by the actuators (taking into account combustion dynamics, valve times, etc.). For this case study we will use a 5Hz control scheme with a fixed 200ms command period, which simulation and analysis has shown to perform well.

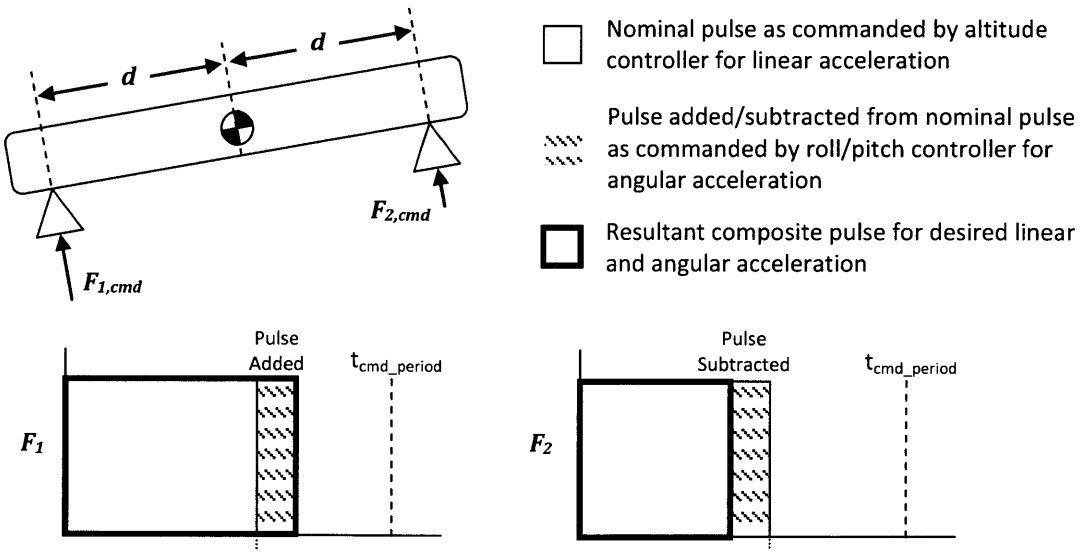


Figure 5-52: Example pulses for closed loop control

A typically actuation sequence for a generic vehicle is shown in Figure 5-52. The vehicle has two fixed nonthrottleable thrusters equidistant from the CM, which are responsible for controlling altitude and attitude. The vertical channel controller commands a vertical impulse, which is divided and sent to both actuators (the plain pulses). In addition to this, the rotational channel controller commands a clockwise rotational impulse, which is also divided and then added to F_1's pulse and subtracted from F_2's pulse (the hash-marked pulses). The composite pulses (bolded) show that the net resultant impulse on the vehicle is equal to that which was commanded by the vertical and rotational controllers. This is the same general implementation scheme we will use in this case study.

Two additional important factors which will affect vehicle performance are the implementation scheme and the state sampling method. In other words, the point in time at which the state is measured relative to the beginning of a given period will affect the accuracy of the error measurement, and the relative location of the actuator pulses within a given period will affect the state trajectory within the command period. If these two factors are not accounted for in the design and analysis process, stability and performance can be affected. For a clearer example, consider the scenario of a generic vehicle with a CM offset under 'steady state' conditions (hovering flight, no net attitude error, no net position error) presented in Figure 5-53. The control algorithms command a net vertical force equal to gravity and a moment to counter the CM offset. Under the basic implementation scheme, both thrusters turn on at the beginning of the command period and then turn off after their respective desired impulses have been exerted. Notice that the impulse centroids from the two thrusters are offset by a finite time. The associated approximate angle and rate plots (boundary conditions were chosen such that the integral of both states are zero over the command period) show a positive angular acceleration while both thrusters are on, a negative angular acceleration when the second thruster turns off, and zero angular acceleration when both thrusters are off. Although the net change over the command period in angle and rate is zero, the boundary conditions show that both angle and rate are nonzero at the beginning and end of the command period. If we choose to sample the state at the end of the time period we reduce the time delay between measurement and control command, but the nonzero state values will cause the controller to (incorrectly) believe there is an error which must be reduced. If we choose to sample the state as the average over the command period our controller (correctly) believes there is no state error, but we introduce an additional time delay of approximately half of the command period.

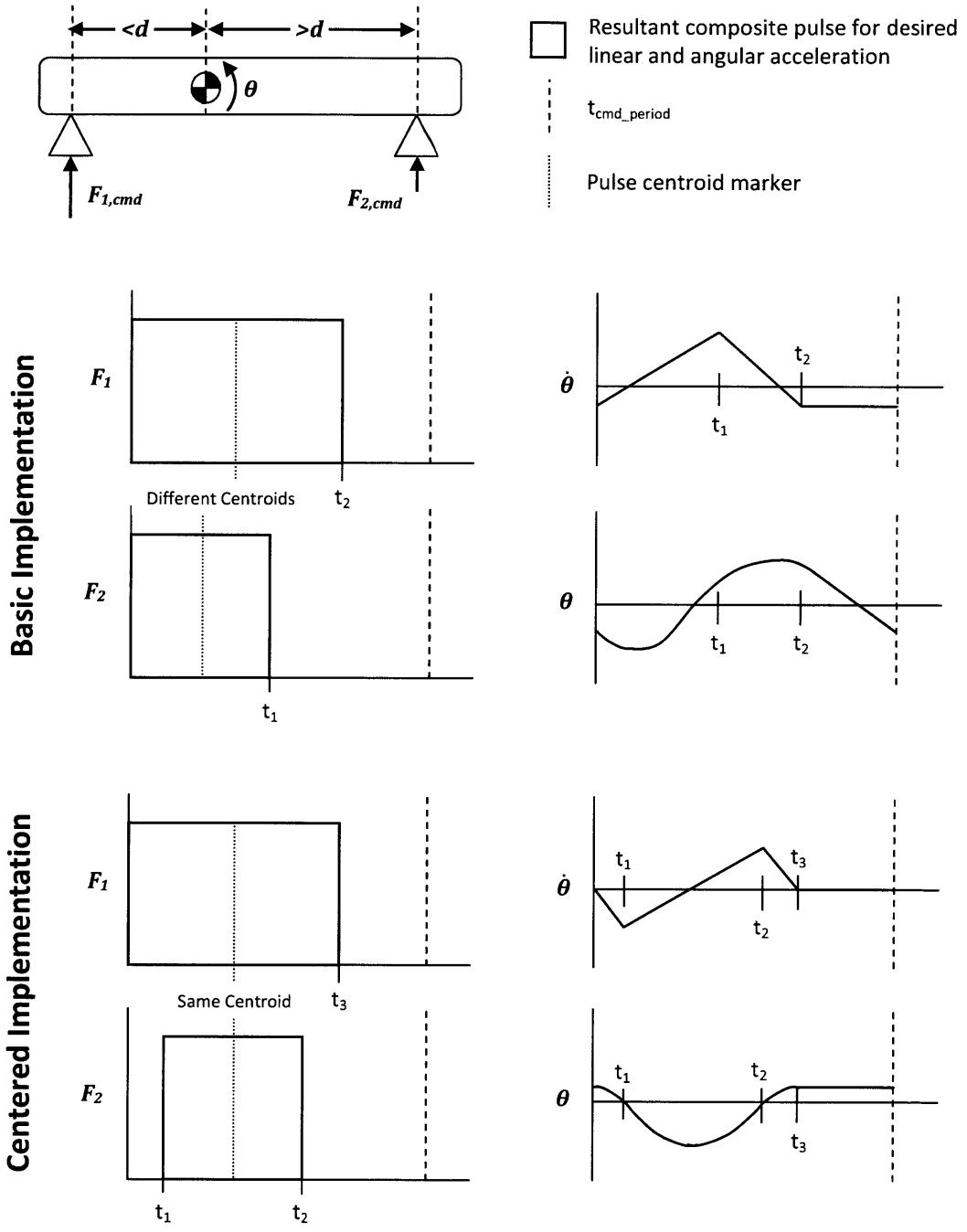


Figure 5-53: Example of how pulse scheme implementation affects state response

Now consider the second centered implementation scheme shown in Figure 5-53. The thrusters remain on for the same amount of time as in the basic implementation, but the second pulse has been shifted so that the two pulse centroids are now aligned. The approximate state responses show a negative angular acceleration before the second pulse begins, a positive angular acceleration after the second pulse begins, a negative angular acceleration after the second pulse ends, and zero angular acceleration after the first pulse ends. Notice that the state profiles are different from the basic implementation case and that once again, where/how we choose to sample the states will affect the accuracy of the error measurement and the induced time delay. In this case study, simulation showed that the most reliable and acceptable performance (compared to left and right pulse edge alignment schemes) was obtained with the centered pulse implementation and state sampling at the final time step of the command period.

A final factor which must be carefully considered by the control system architect is propellant slosh mode excitation as a direct result of the pulsed control scheme. As any pulsed control scheme subjects the vehicle to repeated high angular accelerations for short periods of time, it is a natural (and unfortunate) consequence that some propellant slosh will occur even under 'steady state' conditions due to limit cycling, bias moments, and stochastic disturbances. In our case, we defined the propellant slosh frequency to be between 0.5Hz and 3.0Hz. In previous case studies we were able to effectively gain stabilize these frequencies under most conditions by keeping the crossover frequencies of our open loop systems well below the aforementioned modes. But in this case, simply reducing the crossover frequency of the linear system is not sufficient because the slosh mode will almost always be disturbed by a signal at or near the frequency of the pulsed control scheme (for example, see Figure 5-53). If the slosh mode is found to be unacceptably excited, we are left with three primary options: change the frequency of the command period; try to modify the 'disturbance' signal to the slosh mass; or inform the lander design team that more physical damping is required. In this case, changing the command period by ±1Hz did not sufficiently reduce slosh excitation (below 4Hz compromised disturbance rejection by adding too much time delay, and above 6Hz increased chances of saturation by compromising actuator 'throttle' range). However, it was found that the centered pulse scheme implementation effectively modified the slosh mass disturbance signal and significantly reduced the chances of excessive excitation. Unfortunately, as we will show in the Performance and Verification section, sporadic excitation will remain a concern, and simulation shows that the vehicle would greatly benefit from doubling the nominal damping ratio from a ζ of 0.015 to 0.03.

A Brief Note on Controller Design and Implementation

In the two previous case studies, all controllers were designed and analyzed in the continuous time domain and then verified in the near-continuous time nonlinear simulation. In this case, one might argue that the control signal is inherently digital, and the controllers should therefore be designed and analyzed in the discrete time domain. While this is a viable option, the author has found that nearly identical results can be achieved if controllers are augmented with a pure time delay and designed and analyzed in continuous time [40] [68]. This approach has the advantage of allowing us to use the same analysis tools as in previous case studies, which makes comparison easier. All initial controller design and analysis presented in the following sections will therefore be in the continuous time domain. The continuous controllers will also be implemented in the nonlinear simulation (discrete step size of 0.001sec) and will be sampled discretely at the final simulation time step of each 5Hz control period by the Thruster/Engine Selection Function shown in Figure 5-5.

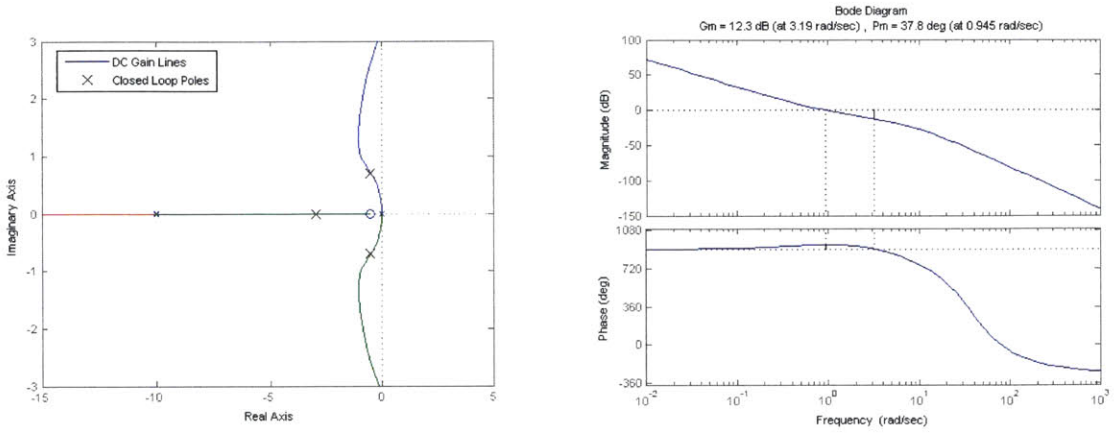
Attitude Controllers

The attitude controllers are responsible for computing errors in rotational states and commanding the four thrusters and single gimbal to impart the necessary impulse to remove these errors. As discussed in the previous section, the controllers will command a desired moment which will be transformed into an associated pulse time (or gimbal angle). All thrusters will work in unison to create angular impulses to reduce the chances of single thruster saturation, meaning that a roll/pitch control signal will increase the pulse length of two adjacent thrusters and decrease the pulse length of the two opposite thrusters. As the yaw controller uses a gimbal rather than differential impulse, it will be discussed first.

Yaw

The yaw controller will assume that its associated thruster is operating at 50% throttle and that the plant it must control is simply a double integrator multiplied by the first order gimbal system and a pure time delay of 0.25sec. We have already shown that this type of system can be well controlled by a lead controller, so we nominally select the same gimbal controller used in the second case study (which was shown to be robust to disturbances and parameter variation) and modify it to achieve a rise time of approximately 1.0sec. The resultant controller (which is the transfer function from yaw error to gimbal rate) is shown in Equation 5-23 along with the associated Root Locus and Bode plots in Figure 5-54.

$$Gc_{yaw}(s) = 0.25\left(\frac{2s+1}{0.1s+1}\right) \qquad \text{5-23}$$



(a) (b)

Figure 5-54: Yaw system; (a) Root Locus, (b) Bode

Roll and Pitch

The roll and pitch controllers (which are identical because the vehicle is assumed to be symmetric) are responsible for determining the portion of the thrusters' pulse widths which must be added or subtracted to produce a net desired angular impulse. The plant of interest contains the basic double integrator, second order slosh dynamics, first order combustion dynamics, and a pure time delay of approximately 0.25sec. The system (with time delay omitted) is shown in Equation 5-24, where $\boldsymbol{X}$ represents the state vector of the attitude system and $\boldsymbol{M}$ represents the actual applied moment due to differential thrust.

$$\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{x_s} \\ \ddot{x_s} \\ \dot{M} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{-h_s(k_s h_s + m_s g)}{I} & 0 & \frac{-h_s k_s}{I} & \frac{-h_s \zeta_g}{I} & \frac{1}{I} \\ 0 & 0 & 0 & 1 & 0 \\ -g - \frac{k_s h_s}{m_s} & 0 & \frac{-k_s}{m_s} & \frac{-\zeta_g}{m_s} & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{\tau} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x_s \\ \dot{x_s} \\ M \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1/\tau \end{bmatrix} u \qquad \text{5-24}$$

As an alternative to the classical error feedback compensators used in the previous case studies, we would like to explore a FSFB architecture in the form of an LQR algorithm. However, we anticipate that

full state knowledge availability will be an issue (slosh dynamics, for example, are hard to measure), so we will also include an estimator modeled after the controller in the form of an LQE algorithm. When combined, the LQR and LQE form a very powerful SISO LQG algorithm, which will be referred to here as a dynamic output feedback (DOFB) compensator. The resultant feedback architecture is shown in Figure 5-55, where $\boldsymbol{K}$ is a vector containing the controller feedback gains and $\boldsymbol{L}$ is a vector containing the estimator feedback gains.

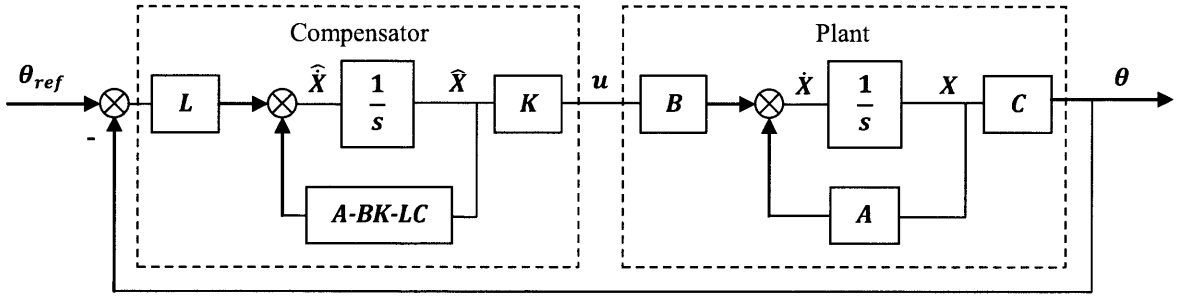


Figure 5-55: Roll and pitch feedback architecture

This feedback architecture is particularly advantageous for three reasons. Firstly, it guarantees closed loop stability of the linear system, even when the controller and compensator are designed separately. In fact, the closed loop system poles are simply the combined controller and estimator poles. Secondly, the design process is well suited for handling the tradeoffs associated with control effort and performance, regardless of the order of the system. This is very important because we are particularly concerned about control signal saturation. Thirdly, as we can express the compensator as a SISO system, we can also form the open loop system transfer function and apply traditional error feedback analysis techniques to determine stability and robustness margins.

The controller was designed to place the most importance on attitude and attitude rate deviations without exceeding reasonable control effort limits, and the estimator was designed to expect the most process noise on the attitude rate and slosh mass velocity states. The resultant feedback gain vectors are shown in Equations 5-25 and 5-26. As expected, $\boldsymbol{K}$ contains the highest gains on the attitude and attitude rate states, and $\boldsymbol{L}$ contains the highest gains on the attitude rate and slosh mass velocity states.

$$\boldsymbol{K} = [\mathbf{25.73} \quad \mathbf{48.65} \quad \mathbf{-5.22} \quad \mathbf{-9.73} \quad \mathbf{0.02}] \qquad \textbf{5-25}$$

$$L = \begin{bmatrix} 6.26 \\ 19.59 \\ 4.83 \\ 47.94 \\ 0.00 \end{bmatrix} \qquad \text{5-26}$$

The compensator (which is the transfer function from roll/pitch error to $\boldsymbol{u}$) is shown in Equation 5-27. It was generated using MATLAB solvers for an equivalent discrete control system to account for time delays. Notice that it is of higher order than the classical compensators we designed in the previous case studies, but its main components are very similar. It contains a single zero near the origin, two mid frequency poles to reduce sensitivity at high frequencies, a second order pole-zero notch filter near the slosh dynamics, and a high frequency pole-zero pair.

$$\boldsymbol{Gc_{roll,pitch}(s) = 622.14\left(\frac{(s+50)(s+0.4135)(s^2+0.3307s+94.58)}{(s+50.01)(s^2+5.702s+15.08)(s^2+1.954s+97.83)}\right)} \qquad \text{5-27}$$

The Root Locus and open loop Bode plots in Figure 5-56 show that the slosh poles are well contained, all poles are sufficiently far away from the imaginary axis, and the system has acceptable gain and phase margin characteristics.

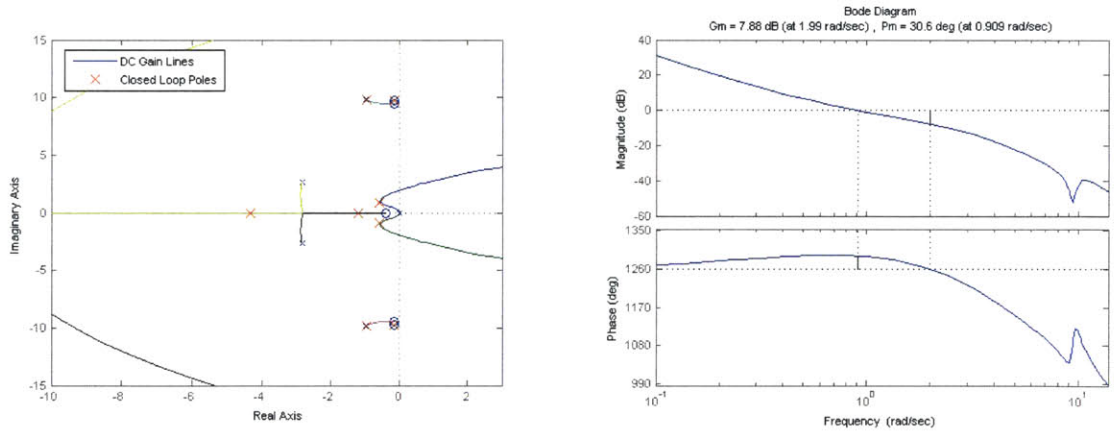
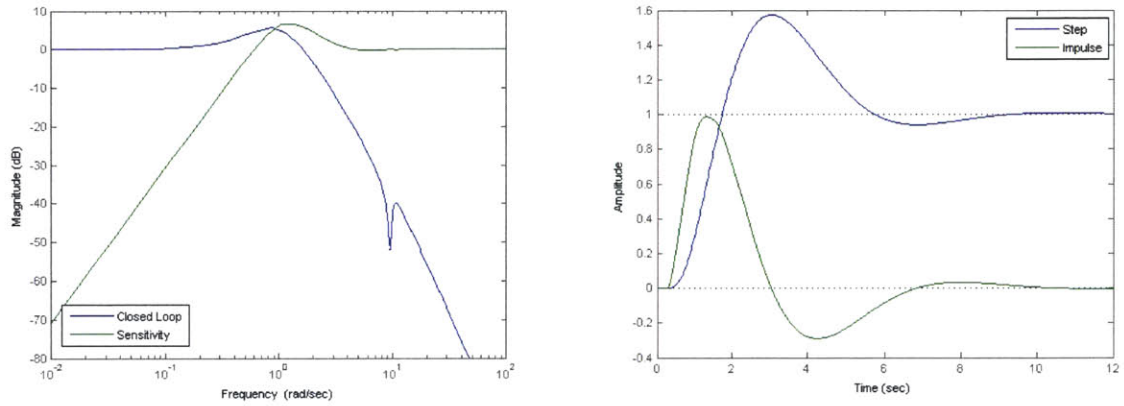


Figure 5-56: Roll/Pitch system; (a) Root Locus, (b) Bode

The performance and sensitivity plots, shown in Figure 5-57, also show acceptable closed loop characteristics. The closed loop transfer function (r to y) magnitude peaks at 5.5dB near 0.81rad/sec and the sensitivity transfer function (dy to y) peaks at 6.6dB near 1.41rad/sec. The step response shows a rise time of 1.5sec, a peak overshoot of 55% near 3.5sec, and near-steady state conditions after 8.0sec.

The impulse response shows an initial overshoot of 100% near 1.5sec, a secondary overshoot of 30% near 4.2sec, and near-full recovery after 8.0sec. Further nonlinear simulation and analysis shows that these are reasonable performance and sensitivity characteristics for our vehicle.



(a) (b)

Figure 5-57: Roll/Pitch performance and sensitivity; (a) frequency domain, (b) time domain

Position Controllers

As in all previous cases, the position controllers are responsible for computing errors in the vertical and lateral position states and commanding necessary linear accelerations. The altitude controller will be responsible for generating the nominal pulse sent to all four thrusters each control time period, and the lateral controllers will send reference attitude commands to the inner loop roll/pitch controllers.

Altitude

Although we observed earlier in this case study that a pulsed control scheme will place stricter limits on the upper and lower 'throttle' setting of the vertical actuators (which will have a large impact on the full deorbit and descent trajectory) than required in previous cases, this does not significantly affect our performance during the short ~15sec terminal descent because the nominal acceleration range specified by our trajectory does not require a large throttle range. In fact, simulation and analysis shows that the exact altitude controller used in the second case study also performed well in this case study (see Equation 5-19.

Lateral

In the two previous case studies we observed that design and verification of the lateral position controllers was most challenging; this case is no exception. Once again, we will use the attitude system to form the inner loop for the lateral position controller as shown in Figure 5-58, where $\boldsymbol{Gp_{roll,pitch}}$ is the plant transfer function of the attitude system. The controller will be responsible for supplying the reference roll/pitch to the attitude system as a function of lateral position state error. The lateral position plant equations now contain the attitude closed loop system combined with a lateral double integrator and associated time delay. Interestingly, this means that our state vector now contains the original attitude states ($\boldsymbol{X}$), the estimated attitude states ($\widehat{\boldsymbol{X}}$), lateral position ($\boldsymbol{x}$), and lateral velocity ($\dot{\boldsymbol{x}}$). The lateral plant state vector ($\boldsymbol{X_{x,y}}$) is shown in Equation 5-28.

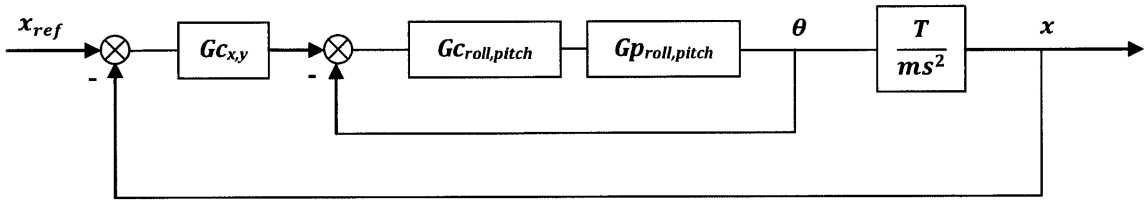
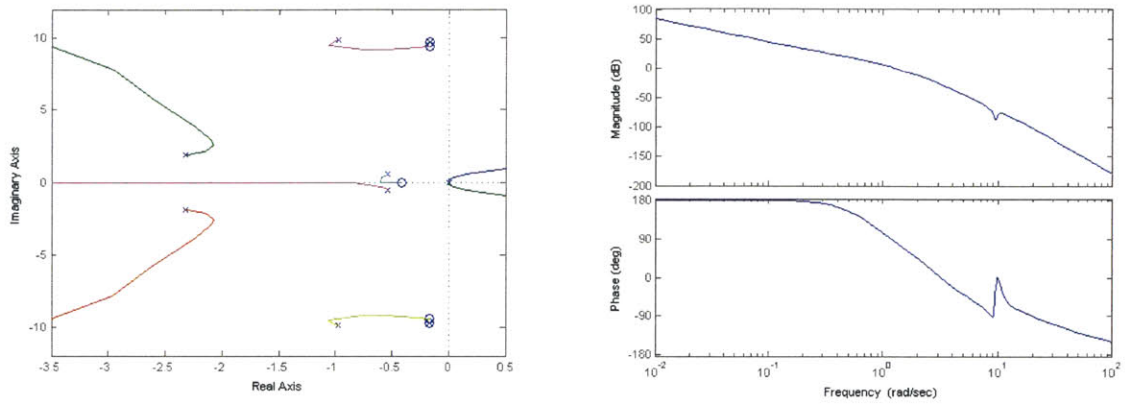


Figure 5-58: Lateral position feedback architecture

$$X_{x,y} = \begin{bmatrix} X \\ \widehat{X} \\ x \\ \dot{x} \end{bmatrix} \qquad \textbf{5-28}$$

As this plant is more complex than the attitude plant, its Root Locus (higher order dynamics not shown) and Bode plots are displayed in Figure 5-59. As expected, the system exhibits infinitely large magnitude and -180deg phase lag at low frequencies due to the lateral plant double integrator as well as a small notch at the slosh frequency.



(a) (b)

Figure 5-59: Lateral plant characteristics; (a) Root Locus, (b) Bode

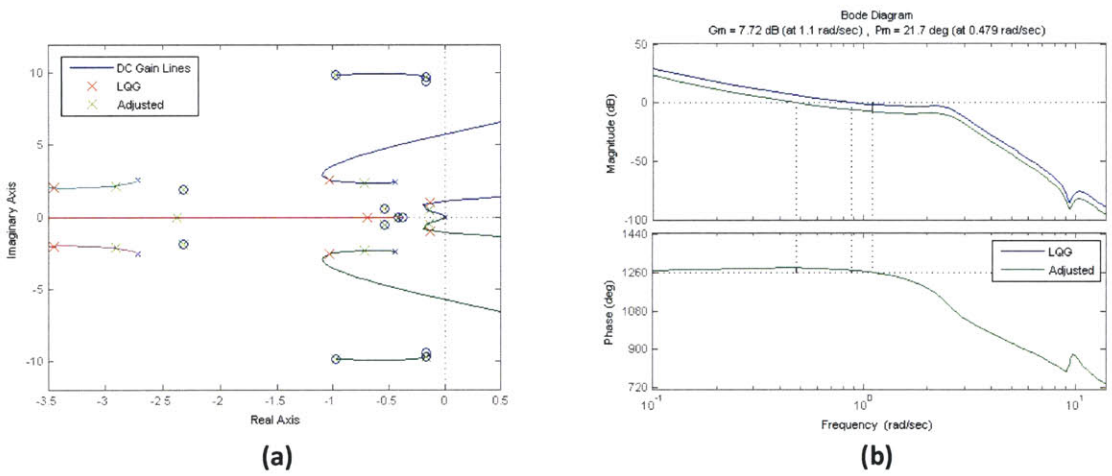
We now design a DOFB compensator. In this case, we will intentionally omit a position integrator from the compensator because the LQG approach has difficulty achieving reasonable gain and phase margins due to the added phase lag of the integrator and the digital control scheme. Instead, we use the same separate bias moment estimator and compensator function which we developed in the first case study. This will allow the lateral DOFB compensator to focus on driving the lateral velocity to zero within the terminal descent time frame. The compensator is therefore designed with a large penalty on lateral position/velocity deviation and process noise on the lateral velocity state. The resulting gains, which were refined through simulation and analysis, are shown in Equations 5-29 and 5-30. The scale factor of 0.5 shown in front of the estimator gains was included after the original LQR/LQE design process to reduce the crossover frequency of the open loop system and provide adequate gain and phase margin.

$$\boldsymbol{K} = [\mathbf{2.67} \quad \mathbf{1.99} \quad \mathbf{-0.53} \quad \mathbf{-0.39} \quad \mathbf{0.00} \quad \mathbf{-0.61} \quad \mathbf{0.55} \quad \mathbf{0.12} \quad \mathbf{-0.12} \quad \mathbf{0.00} \quad \mathbf{1.25} \quad \mathbf{2.43}] \qquad \textbf{5-29}$$

$$\boldsymbol{L} = \mathbf{0.5}\begin{bmatrix} \mathbf{0.02} \\ \mathbf{0.01} \\ \mathbf{-0.01} \\ \mathbf{-0.01} \\ \mathbf{-0.75} \\ \mathbf{-0.01} \\ \mathbf{-0.00} \\ \mathbf{0.00} \\ \mathbf{0.01} \\ \mathbf{0.75} \\ \mathbf{2.52} \\ \mathbf{3.18} \end{bmatrix} \qquad \textbf{5-30}$$

The full compensator is shown in Equation 5-31. Now the compensator is of very high order and less intuitive to understand, but we can recognize the basic elements of several lead compensators and high frequency poles. The Root Locus and Bode plots in Figure 5-60 show reasonable pole locations and acceptable gain and phase margins, but they also show multiple pole-zero cancellations (which are not uncommon in DOFB compensators and are typically acceptable if the poles are in the LHP) and a 'shelf' near -9.0dB at mid frequencies. Ultimately, simulation shows that both of these characteristics are acceptable. Note that 'LQG' in the legend refers to the original DOFB compensator generated with LQR/LQE and 'Adjusted' refers to the scale factor (0.5) placed on $\boldsymbol{L}$ to reduce the open loop crossover frequency of the complete system. The compensator shown in Equation 5-31 contains this scale factor.

$$\boldsymbol{Gc_{x,y}(s) = 5.47\left(\frac{(s+0.3735)(s^2+1.067s+0.5734)(s^2+4.648s+9.058)}{(s+3.597)(s+0.4152)(s^2+0.8819s+6.007)(s^2+5.439s+14.06)}\right)} \qquad \textbf{5-31}$$



(a) (b)

Figure 5-60: Lateral position system; (a) Root Locus, (b) Bode

Lastly, we can examine the closed loop and sensitivity time and frequency domain performance to support the conclusion that the complete 6DOF nonlinear system will be robust to disturbances and parameter variations. Figure 5-61 shows that the closed loop transfer function (r to y) has a peak of 8.5dB near 0.50rad/sec and a step response with a settling time near 25.0sec. Figure 5-62 shows that the sensitivity transfer function (dy to y) has a peak of 8.6dB near 0.55rad/sec and an impulse response settling time of approximately 20.0sec. The slow step response and impulse rejection were intentionally included in the design in an effort to reduce roll/pitch angles and rates at touchdown. Once again,

nonlinear simulation was used to confirm this. The controller output (the commanded reference attitude) was limited to ±60.0deg.

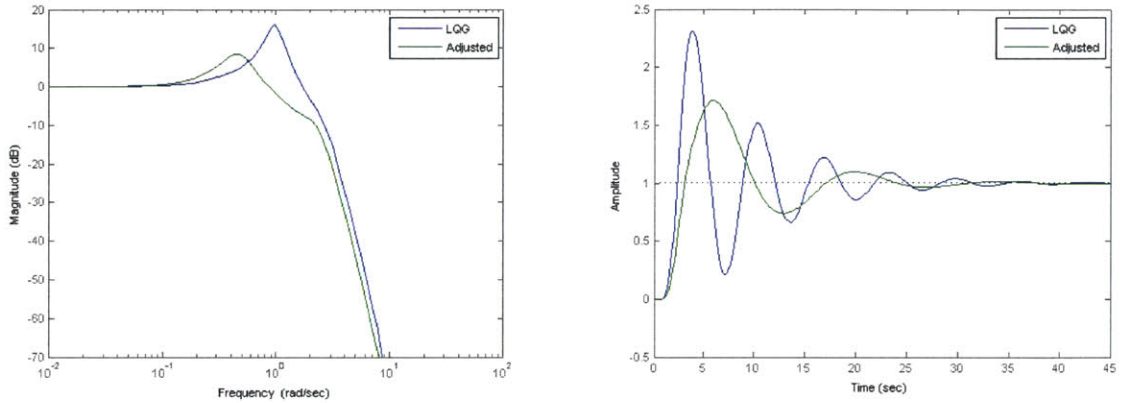


Figure 5-61: Lateral closed loop performance; (a) frequency response, (b) step response

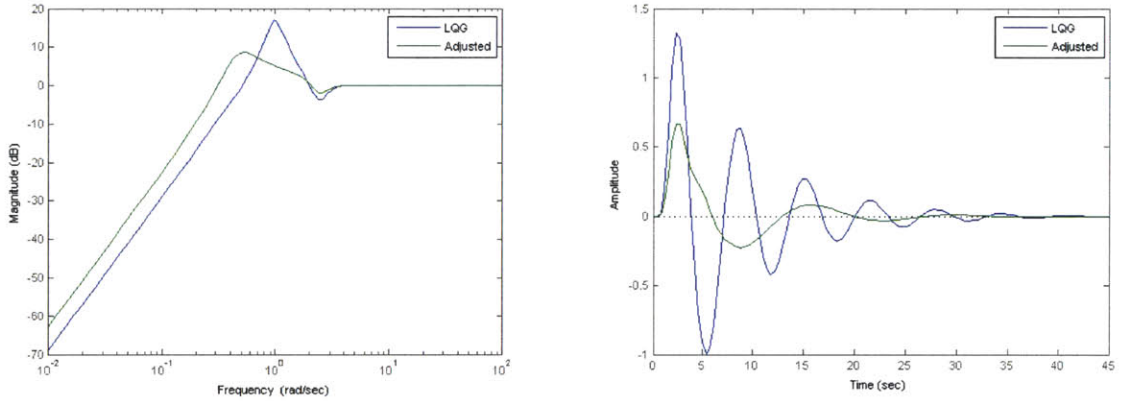


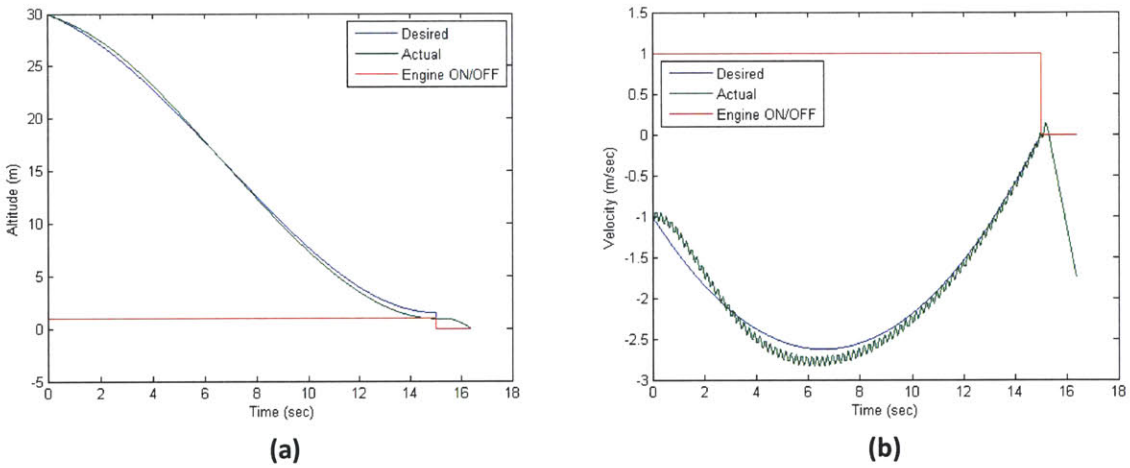
Figure 5-62: Lateral closed loop sensitivity; (a) frequency response, (b) impulse response

5.4.3 Performance and Verification

The final step of the process is to use the same nonlinear simulation environment to verify the collective performance of the control system architecture in all six DOF's. Results will be discussed for the nominal case, for variation of single parameters, and for Monte Carlo runs.

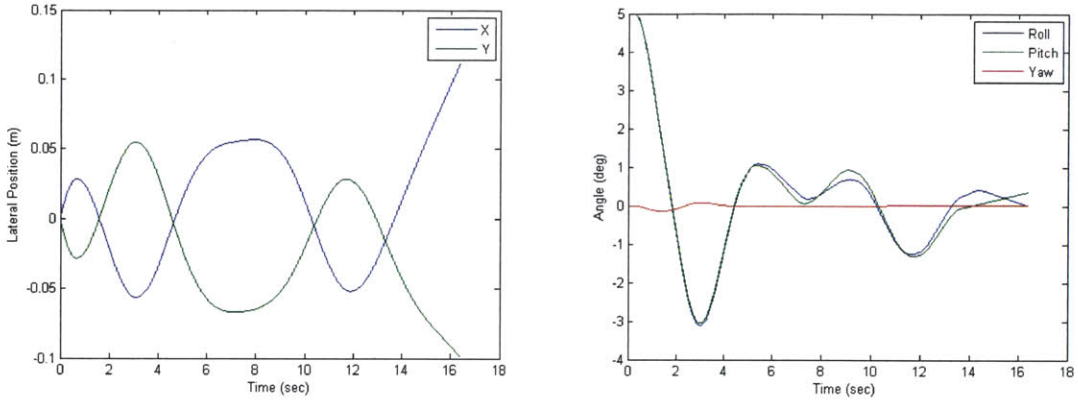
Nominal Performance

Under nominal conditions the lander, which begins at 30.0m altitude with a 1.0m/sec descent velocity and 5.0deg roll and pitch, tracks the altitude trajectory (Figure 5-63a) nearly identically to the second case study. However, the difference in the actuation system is clearly seen in the high frequency oscillations of the vertical velocity (Figure 5-63b) which result from repeated pulsed firings. The attitude and lateral position (Figure 5-64) show acceptably small oscillations. Note that the apparent divergence of the lateral position towards the end of the descent is misleading, as it is the result of engine/control system cutoff after 15.0sec. The thruster pulse widths and gimbal angle (Figure 5-65) are well within the saturation limits, and neither show signs of controller-induced growing oscillations. Lastly, the force and moment plots (Figure 5-66) clearly show the pulsed actuation scheme. The apparent variance in force/moment level is a direct result of the first order combustion dynamics; all thrusters exert a constant thrust of 286.0N at steady state.



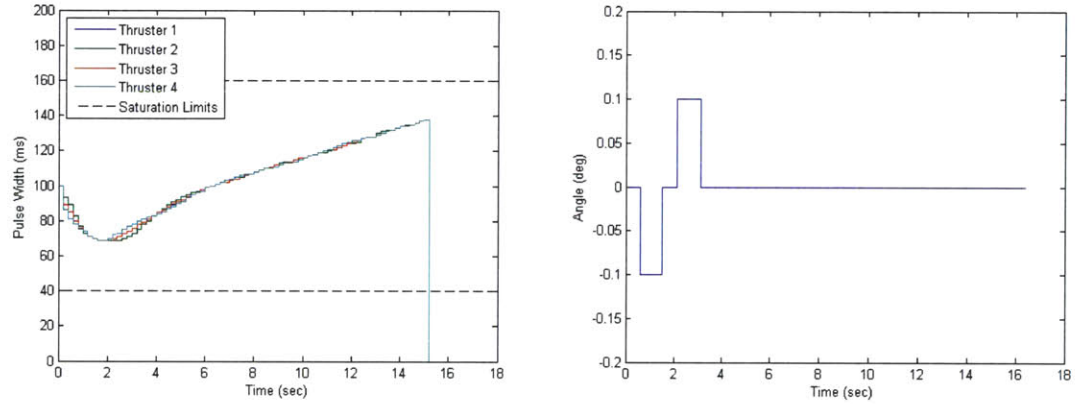
(a) **(b)**

Figure 5-63: Nominal descent states; (a) altitude, (b) vertical velocity



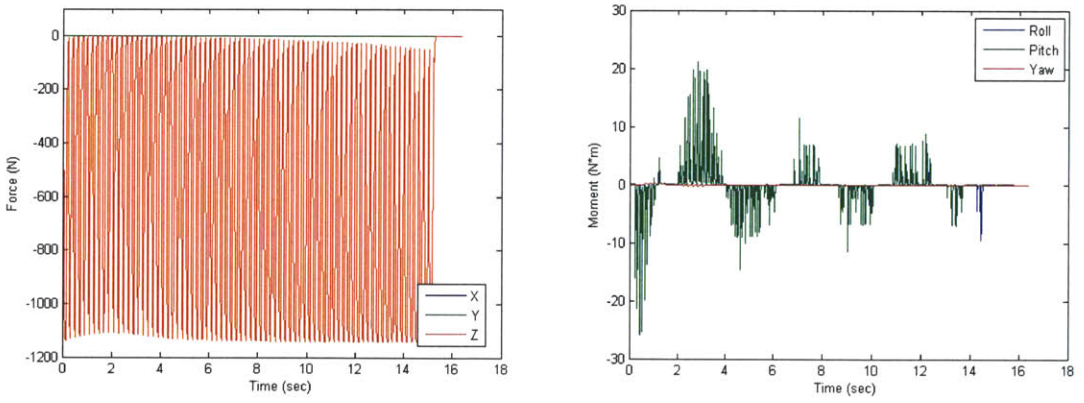
(a) (b)

Figure 5-64: Nominal descent states; (a) lateral positions, (b) angles



(a) (b)

Figure 5-65: Nominal descent states; (a) pulse widths, (b) gimbal angle



(a) (b)

Figure 5-66: Nominal descent actual body frame control; (a) forces, (b) moments

Variation of Single Parameters

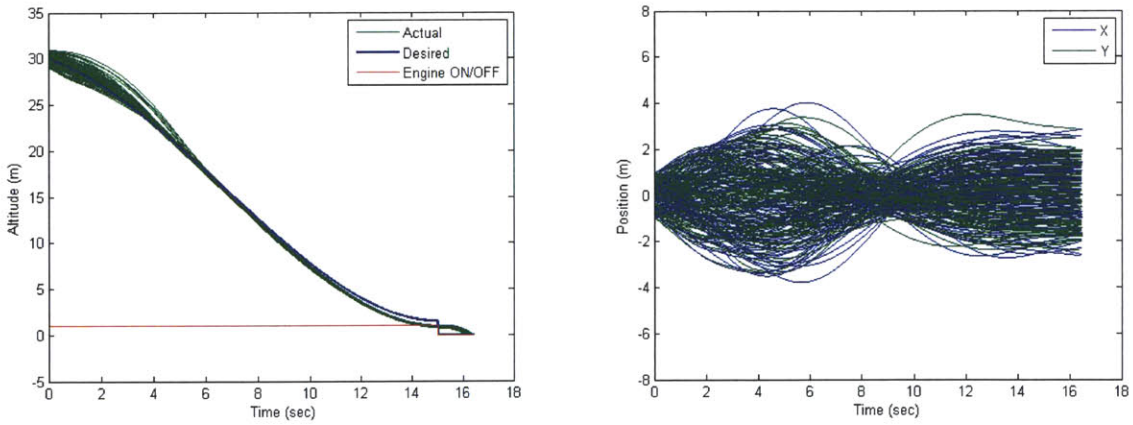
The results for this case study, along with the results from the first and second case studies, are shown in Table 5-9. Notice that while the third case is very insensitive to time delays (we intentionally designed it this way in order to handle delays incurred by discrete control), it is also much more sensitive than the first two case studies to many other parameter variations and initial condition errors. In almost all instances the landing constraint violation was linked to saturation of the control signal, meaning that multiple thruster pulse widths were commanded beyond 40ms or 160ms. Control signal saturation typically resulted in an inability to reduce vertical position error followed by attitude and position divergence due to an inability to exert adequate angular impulse. This means that, unlike in the first case study where graceful performance and stability degradation was the norm, the loss of control of a single DOF sometimes cascaded into the loss of control of multiple DOF's. It should be emphasized that these failure characteristics are rooted in the decision to use a single set of nonthrottleable fixed actuators to control multiple degrees of freedom.

Table 5-9: Max single parameter variation limits before landing constraint violation (all Cases)

Parameter	Case 1	Case 2	Case 3	Comment
ACS Thrust Scale Factor	9	20	N/A	Applied to all thrusters
Main Engine Thrust Scale Factor	0.5, 2.6	0.5, 2.6	0.8, 1.6	Min, max
Initial Roll or Pitch Error	65deg	90deg	90deg	On one axis
Initial Lateral Position Error	15m	15m	7m	On one axis
Initial Vertical Position Error	35m, 8m	30,16m	15m, 10m	Height above, below 30m
Initial Lateral Velocity Error	3m/sec	3m/sec	3m/sec	On one axis
Initial Vertical Velocity Error	5m/sec, 6m/sec	5m/sec, 8m/sec	2m/sec, 6m/sec	Velocity above, below -1m/sec
Overall Time Delay	120ms	160ms	230ms	Applied to all control feedback states
X or Y CM offset	10cm	5.5cm	11cm	On one axis

Monte Carlo Simulations

Finally, Monte Carlo simulations were conducted to verify the robustness of the complete 6DOF nonlinear system to the same stochastic parameter and initial condition variations performed on the first and second case studies. These variation limits are shown in Table 5-4. Representative results from 100 runs are shown in Figure 5-67 (position plots), Figure 5-68 (position and velocity histograms), and Figure 5-69 (attitude and rate histograms). Once again, the overall performance is similar to that seen in the first two case studies. Notice that the initial overshoot of the altitude trajectory is greater than in the previous case studies because of the vertical channel control signal saturation limits. Final position and velocity remain well within the desired limits, and angles and rates are generally acceptable. Four runs were found to violate the terminal landing conditions set in Table 1-3 due to slightly high angle and angular rate states. Further simulation of all four runs showed that the offending states were the result of excited slosh modes (amplitude doubling approximately every 15sec), which would have fully destabilized the system after approximately 120sec. All four cases also shared CM offsets greater than 2cm, and moment of inertia scale factors and slosh mass frequencies at the upper or lower limits of their allowable ranges.



(a) (b)

Figure 5-67: Monte Carlo position plots; (a) altitude, (b) lateral

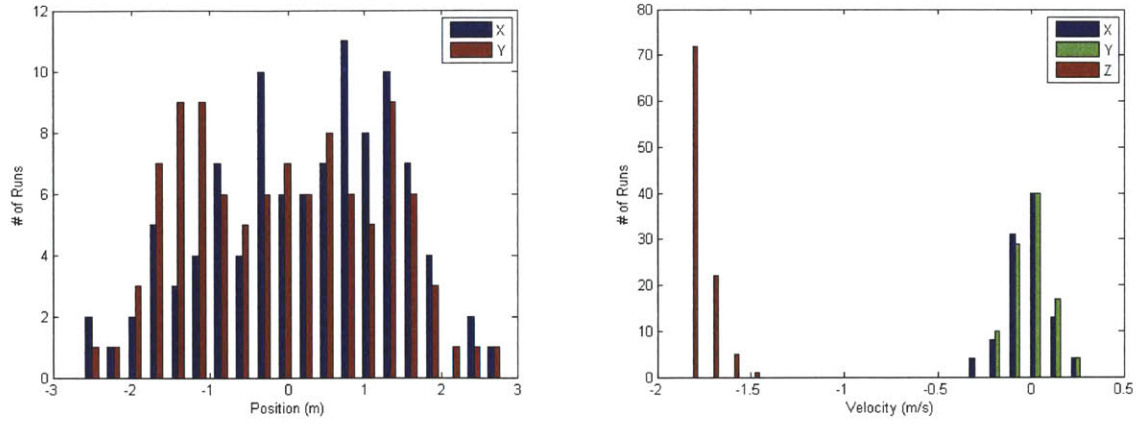


Figure 5-68: Monte Carlo final state histogram plots; (a) lateral positions, (b) velocities

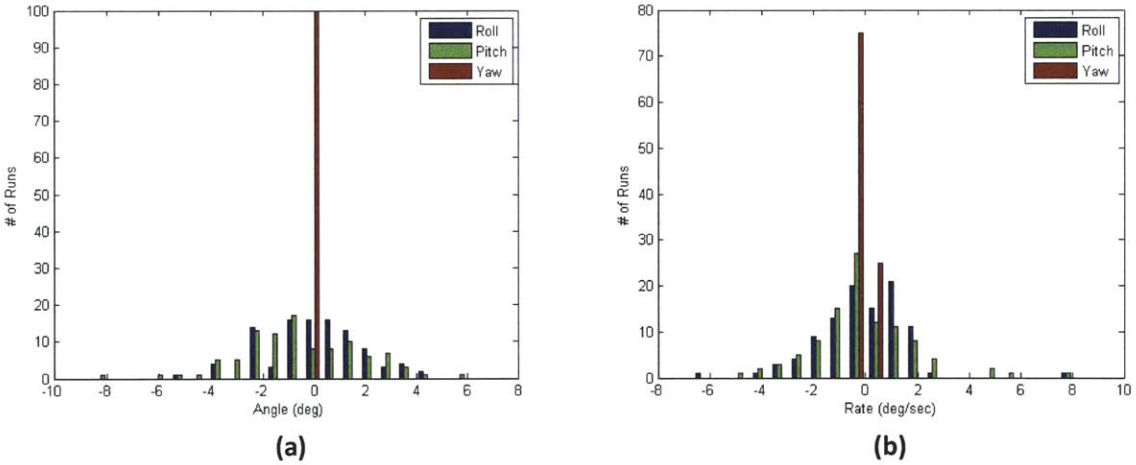


Figure 5-69: Monte Carlo final state histogram plots; (a) angles, (b) rates

5.4.4 Case 3 Summary

In this case study we explored the concept of complete actuation system coupling by using four nonthrottleable actuators to control roll, pitch, altitude, and lateral translation. One of the thrusters was placed on a gimbal to control yaw. The actuator and control algorithm architectures were designed to meet the same overall performance criteria that were presented in Chapter 1.

Actuator Architecture

The actuator architecture performed well under nominal conditions. The four thrusters were able to provide enough control power and resolution to execute most commanded control sequences and

counter reasonable disturbance torques. The gimbal also worked well for controlling yaw, and its coupling effects with the other DOF's was found to be minimal. However, using the same set of nonthrottleable actuators to control multiple DOF's simultaneously meant that control signal saturation was a persistent concern, as the resultant loss of controllability was rarely contained to a single channel.

Control Algorithm Architecture

The control algorithm architecture used DOFB compensators to control roll/pitch and lateral translation, a classical controller for yaw, and the same FSFB controller for altitude as was used in the second case study. Actuation was accomplished with a centered-pulse 5Hz control scheme with upper and lower saturation limits and state sampling at the final time step of each command period. Linear analysis techniques were used in the design process and refined through nonlinear simulation. In general, the control algorithm architecture worked well, but performance was sensitive to the implementation method (e.g. control rate, sensing method, pulse implementation scheme) and slosh mass excitation was difficult to account for.

Control System Architecture

The control system architecture discussed in Case 3 was able to meet slightly lower general stability, performance, and robustness metrics than in Case 1 and Case 2. It performed acceptably under nominal conditions, and it was able to accommodate time delays and bias moments well. However, the pulsed control scheme, which was necessitated by the decision to use a single set of nonthrottleable actuators to simultaneously control multiple degrees of freedom, caused control signal saturation and slosh mode excitation to be continuing sources of concern.

5.5 General Case Studies Summary

This chapter presented and discussed three specific case studies, which demonstrated the importance of understanding the relationship between actuator and control algorithm architecture development. The first case study examined the potential advantages and disadvantages of attempting to decouple the high linear impulse actuation system with the low angular impulse actuation system by using a single fixed-axis throttleable engine to control vertical acceleration and multiple fixed-axis nonthrottleable ACS thrusters to control angular acceleration. Reasonable stability, performance, and robustness margins were demonstrated, but bias moments induced by CM offsets from the main engine thrust vector and subsequent propellant slosh mode excitation made it difficult to fully decouple the design and analysis of the two actuation/control systems. The second case study explored the tradeoffs of partial coupling between the linear and angular impulse systems by using a single gimbaled throttleable engine for linear acceleration and roll/pitch angular acceleration and a separate set of fixed nonthrottleable ACS thrusters for yaw angular acceleration. Analysis and simulation showed that stability, performance, and robustness were comparable to the first case study, but gimbal dynamics and the effects of using a single actuator to simultaneously control multiple degrees of freedom made design and analysis challenging. Lastly, the third case study investigated the compromises associated with complete coupling between actuation systems by using four nonthrottleable thrusters (three fixed, one gimbaled) to simultaneously control all degrees of freedom. Results showed that stability and performance was comparable to the previous case studies but that robustness to parameter variation and initial condition errors was reduced due to the inherent penalties associated with the pulsed control scheme (which was necessitated by the actuator architecture). Consequently, design and analysis for this case study was most challenging.

In summary, all three of these case studies demonstrated that reasonable stability, performance, and robustness margins could be achieved for a wide variety of actuator and algorithm architectures if given adequate control authority range and resolution. However, the true purpose of this chapter was to use the aforementioned examples to illustrate the importance of the control system architecture paradigm by showing how the selection of actuator type/placement can affect algorithm design (and vice versa), and ultimately affect overall vehicle stability, performance, and robustness. In lieu of these results, it is recommended that lunar lander design teams and engineers apply the paradigm to actuator design/selection/placement and algorithm design/analysis in order to gain an understanding of the comprehensive effects of their decisions.

6 Conclusion

While a great deal of published work can be found on the development and implementation of both lunar lander actuators and control algorithms, most work addresses them as separate subjects which can be studied without explicit knowledge of the other. The few documents that do address their interaction tend to be for specific concept/flight vehicles. This thesis has attempted to augment the knowledge base by approaching actuator and algorithm development in a generalized and unified fashion by highlighting both the subtle and the obvious interactions between actuator types and placements, how this affects control algorithm design, and how these combined factors affect the overall performance of a lunar lander. Chapter 2 provided a review of past and planned lunar lander missions; Chapter 3 discussed different types of actuators and actuator placements (actuator architectures); Chapter 4 presented select feedback structures and control algorithms (control algorithm architectures); and Chapter 5 used three case studies to illustrate the importance of understanding and applying the control system architecture paradigm.

Results indicate that reasonable stability, control, and robustness can be achieved for a wide variety of actuator and control algorithm architectures as long as the actuators are able to supply adequate control authority and the control algorithms are designed to account for a variety of parameter variations and disturbances. However, the case studies repeatedly showed that the limitations of the chosen actuator architecture can play a significant role in the selection of the feedback architecture, the design of the control algorithms, and even impact overall vehicle performance. It is therefore recommended that both designers of the actuator architecture and designers of the control algorithm architecture make a concerted effort to understand the larger impact of their decisions in order to improve the chances of overall mission success.

Works Cited

[1] Andrew Ball, *Planetary Landers and Entry Probes.*: Cambridge University Press, 2007.

[2] Eric Mueller, Karl Bilimoria, and Chad Frost, "Effects of Control Power and Inceptor Sensitivity on Lunar Lander Handling Qualities," in *AIAA Space 2009*, Pasadena, CA, 2009, pp. 1-21.

[3] NASA. (2011, August) NSSDC Master Catalog: Surveyor. [Online]. http://nssdc.gsfc.nasa.gov/nmc/spacecraftDisplay.do?id=1966-045A

[4] NASA. (2012, February) Surveyor 1 Transit. [Online]. http://commons.wikimedia.org/wiki/File:Transit-Surveyor_1.png

[5] Sam W. Thurman, "Surveyor Spacecraft Automatic Landing System," in *AAS Guidance and Control Conference*, Breckenridge, 2004.

[6] C.E. McDermotto, R.R Breshears, and J. McCafferty, "Dynamic Performance of Surveyor Throttleable Engine when Propellants Contain Dissolved Helium," *Journal of Spacecraft*, vol. 4, no. 5, 1967.

[7] M. Edmund Ellion, H. DiChristina, A.R. Maffei, and A. Briglio Jr., "Development of the Surveyor Vernier Propulsion System," *Journal of Spacecraft*, vol. 4, no. 3, 1967.

[8] Gilbert Carman and Moises Montez, "Apollo 12 (Mission H-1) Spacecraft Analysis Volume IV: Descent and Ascent Dispersion Analyses Part 1 - Lunar Descent," NASA, MSC, Internal Note No 69-FM-275 1969.

[9] Thomas J. Kelly, "Manned Lunar Lander Design: The Project Apollo Lunar Module (LM)," in *AIAA Space Programs and Technologies Conference*, Huntsville, 1992.

[10] NASA. (2012, May) NASA Photos. [Online]. http://science.ksc.nasa.gov/mirrors/images/html/as10.htm

[11] NASA. (2012, February) NSSDC Master Catalog: Apollo 11 Lunar Module / EASEP. [Online]. http://nssdc.gsfc.nasa.gov/nmc/spacecraftDisplay.do?id=1969-059C

[12] Floyd Bennett, "Apollo Lunar Descent and Ascent Trajectories," in *AIAA 8th Aerospace Sciences Meeting*, New York, 1970, pp. 1-33.

[13] George W. Cherry, "Memo: Controlling the Attitude and Attitude Rate of the LEM with the Gimballed Descent Propulsion System," Cambridge, 4-66, 1966.

[14] David G. Hoag, "Apollo Navigation, Guidance, and Control Systems: A Progress Report," MIT Instrumentation Laboratory, 1969.

[15] Frederick Martin and Richard Battin, "Computer-Controlled Steering of the Apollo Spacecraft," *Journal of Spacecraft and Rockets*, vol. 5, no. 4, pp. 400-407, April 1968.

[16] William Widnall, "Lunar Module Digital Autopilot," *Journal of Spacecraft and Rockets*, vol. 8, no. 1, pp. 56-62, January 1971.

[17] David Hoag, "The History of Apollo Onboard Guidance, Navigation, and Control," *Journal of Guidance, Control, and Dynamics*, vol. 6, no. 1, pp. 4-13, January-February 1983.

[18] NASA: Anne M. Platoff. (2012, February) A Summary of Lunar Exploration Missions. [Online]. http://ares.jsc.nasa.gov/HumanExplore/Exploration/EXLibrary/docs/facts/LSumm1.htm

[19] Encyclopedia Astronautica. (2012, February) Luna E-6. [Online]. http://www.astronautix.com/craft/lunae6.htm

[20] Zarya. (2012, February) Soviet, Russian and International Space Flight: Luna - Exploring the Moon:

The Mission of Luna 16. [Online]. http://www.zarya.info/Diaries/Luna/Luna16.php

[21] NASA. (2012, February) NSSDC Master Catalog: Luna 16. [Online]. http://nssdc.gsfc.nasa.gov/nmc/masterCatalog.do?sc=1970-072A

[22] Zarya. (2012, February) Soviet, Russian and International Space Flight: Luna - Exploring the Moon: The Mission of Luna 21. [Online]. http://www.zarya.info/Diaries/Luna/Luna21.php

[23] C. D. Brown, *Spacecraft Propulsion*. Washington D.C.: American Institute of Aeronautics and Astronautics, 1996.

[24] Allan Y. Lee et al., "Preliminary Design of the Guidance, Navigation, and Control System for the Altair Lunar Lander," in *AIAA Guidance, Navigation, and Control Conference*, Toronto, 2010.

[25] Larry Kos, Tara Polsgrove, Ronald Sostaric, Ellen Braden, and Jacob Sullivan, "Altair Descent and Ascent Reference Trajectory Design and Initial Dispersion Analyses," in *AIAA Guidance, Navigation, and Control Conference*, Ontario, Canada, 2010, pp. 1-11.

[26] Michael C. Johnson, "A Parameterized Approach to the Design of Lunar Lander Attitude Controllers," in *AIAA Guidance, Navigation, and Control Conference*, Keystone, 2006.

[27] Berengere Houdou, James Carpenter, and ESA MoonNEXT Team. (2012, February) The MoonNEXT Mission. [Online]. http://www.lpi.usra.edu/meetings/leagilewg2008/presentations/oct30am/Carpenter4037.pdf

[28] Richard Fisackerly et al., "The ESA Lunar Lander Mission," in *AIAA Space 2011*, Long Beach, CA, 2011, pp. 1-8.

[29] Encyclopedia Astronautica. (2012, February) Chang'e. [Online]. http://www.astronautix.com/craft/change.htm

[30] Wikipedia. (2012, February) Chinese Lunar Exploration Program. [Online]. http://en.wikipedia.org/wiki/Chinese_Lunar_Exploration_Program

[31] Chandrayaan-2. (2012, February) What is Chandrayaan-2. [Online]. http://www.chandrayaan-i.com/index.php/chandrayaan-2.html

[32] Chandrayaan-2. (2012, Febraury) Chandrayaan-2 may miss scheduled 2013 take off date. [Online]. http://www.chandrayaan-i.com/index.php/chandrayaan-2/chandrayaan-2-updates/141-chandrayaan-2-may-miss-the-scheduled-2013-take-off-date.html

[33] Chandrayaan-2. (2012, February) Chandrayaan-2 is dealing with weight issues. [Online]. http://www.chandrayaan-i.com/index.php/chandrayaan-2/chandrayaan-2-updates/140-chandrayaan-2-is-dealing-with-weight-issues.html

[34] Wikipedia. (2012, February) Lunar Lander Challenge. [Online]. http://en.wikipedia.org/wiki/Lunar_Lander_Challenge

[35] X Prize Foundation. (2012, February) 2009 Northrop Grumman Lunar Lander X CHALLENGE. [Online]. http://space.xprize.org/lunar-lander-challenge

[36] Sonja Alexander. (2012, February) NASA and X Prize Announce Winners of Lunar Lander Challenge. [Online]. http://www.nasa.gov/home/hqnews/2009/nov/HQ_09-258-Lunar_Lander.html

[37] Armadillo Aerospace. (2012, February) News Archive. [Online]. http://www.armadilloaerospace.com/n.x/Armadillo/Home/News?news_id=368

[38] X PRIZE Foundation. Google Lunar X Prize. [Online]. http://www.googlelunarxprize.org/lunar/about-the-prize

[39] Wikipedia. (2012, February) Google Lunar X Prize. [Online].

http://en.wikipedia.org/wiki/Google_Lunar_X_Prize

[40] Joseph M. Morrow, Sarah L. Nothnagel, Phillip M. Cunio, and Jeffrey A. Hoffman, "Verification and Validation of a Cold Gas Propulsion System for the TALARIS Hopper Testbed," in *AIAA Space 2011*, Long Beach, CA, 2011.

[41] Sarah L. Nothnagel, "Development of a Cold Gas Propulsion System for the TALARIS Hopper," Massachusetts Institute of Technology, Cambridge, MA, S.M. Thesis 2011.

[42] Akil J. Middleton, "Modeling and Vehicle Performance Analysis of Earth and Lunar Hoppers," Massachusetts Institute of Technology, Cambridge, MA, S.M. Thesis 2010.

[43] Christopher J. Han, "Development of Modular Real-Time Software for the TALARIS Lunar Hopper Testbed," Massachusetts Institute of Technology, Cambridge, MA, S.M. Thesis 2011.

[44] Doug Messier. (2012, February) Parabolic Arc: GLXP News: Moon Express Lunar Lander Takes First Flight. [Online]. http://www.parabolicarc.com/2011/06/30/glxp-news-moon-express-lunar-lander-takes-first-flight/

[45] Dominick Reuter. (2012, February) Reuter Photo. [Online]. http://reuterphoto.photoshelter.com/gallery-image/G0000IjUR.hu8TqY/I0000VtQgwhz3c9E/152

[46] George P. Sutton and Oscar Biblarz, *Rocket Propulsion Elements.*: John Wiley & Sons, 2001.

[47] George P. Sutton, *History of Liquid Propellant Rocket Engines.*: American Institute of Aeronautics and Astronautics, 2006.

[48] Claude Joyner II, Daniel Levack, Russel Rhodes, and John Robinson, "Propulsion System Choices and Their Implications," in *AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, Nashville, TN, 2010, pp. 1-28.

[49] Matthew Casiano, James Hulka, and Vigor Yang, "Liquid-Propellant Rocket Engine Throttling: A Comprehensive Review," in *AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Denver, CO, 2009, pp. 1-39.

[50] G. A. Dressler, "Summary of Deep Throttling Rocket Engines with Emphasis on Apollo LMDE," in *AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Sacramento, CA, 2008, pp. 1-19.

[51] Erin Betts and Robert Frederick Jr., "A Historical Systems Study of Liquid Rocket Engine Throttling Capabilities," in *AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Nashville, TN, 2010, pp. 1-18.

[52] K. Hodge, T. Crofoot, and S. Nelson, "Gelled Propellants for Tactical Missile Applications," in *AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Los Angeles, CA, 1999, pp. 1-8.

[53] J.A. Hardgrove and H.C. Jr. Krieg, "High Performance Throttling and Pulsing Rocket Engine," TRW Space & Technology Group, Redondo Beach, CA, 1984.

[54] M. Larin, "Standardization of Rocket Engine Pulse Time Parameters," in *AIAA Thermophysics Conference*, Anaheim, CA, 2001, pp. 1-6.

[55] Gilbert S. Stubbs, Alexander Penchuk, and Robert W. Schlundt, "Digital Autopilots for Thrust Vector Control of the Apollo CSM and CSM/LM Vehicles," in *AIAA Guidance, Control, and Flight Mechanics Conference*, Princeton, 1969.

[56] C. Donovan Hyatt, Joseph R. Riccio, and Landon Moore, "Common Lunar Lander Vehicle Propulsion System Conceptual Design," in *AIAA/SAE/ASME/ASEE 29th Joint Propulsion Conference and Exhibit*, Houston, 1993.

[57] Masten. (2012, February) Masten Space Systems. [Online]. http://masten-space.com/

[58] MSFC, "System Performance Analysis Databook for the Warm Gas Test Article," NASA, Huntsville, RLTB-0xx Version 1.5a, 2011.

[59] Jovan Boskovic, Joseph Jackson, Raman Mehra, and Nhan Nguyen, "Multiple-Model Adaptive Fault-Tolerant Control of a Planetary Lander," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 6, pp. 1-15, November-December 2009.

[60] P.S. Stafford and R.N. Ingoldby, "Guidance and Control System Design for a Six Engine Martian Soft Lander," 1968.

[61] Bong Wie, *Space Vehicle Dynamics and Control.*: AIAA, 2008.

[62] Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini, *Feedback Control of Dynamic Systems*, 2nd ed. USA: Addison-Wesley Publishing Company, 1991.

[63] Jean-Jacques E. Slotine and Weiping Li, *Applied Nonlinear Control*, 1st ed. Engelwood Cliffs, New Jersey, USA: Prentice-Hall, 1991.

[64] Michael Athans and Peter L. Falb, *Optimal Control*, 1st ed. USA: McGraw-Hill Book Company, 1966.

[65] Robert F. Stengel, *Optimal Control and Estimation*, 1st ed. New York, USA: Dover Publications, 1994.

[66] Sigurd Skogestad and Ian Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 1st ed. West Sussex, England: John Wiley & Sons, 1996.

[67] A. Gelb and W. E. Vander Velde, *Multiple-Input Describing Functions and Nonlinear System Design*. New York: McGraw-Hill, 1968.

[68] Jonathan P. How and Emilio Frazzoli. (2010, Fall) MIT OCW: Feedback Control Systems. [Online]. http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-30-feedback-control-systems-fall-2010/

[69] Leonard Lublin, Simon Grocott, and Michael Athans, "H2 (LQG) and Hinf Control," in *The Control Handbook.*: CRC Press, INC, 1996.

[70] H. Norman Abramson, "The Dynamic Behavior of Liquids in Moving Containers With Applications to Space Vehicle Technology," NASA and Southwest Research Institute, Washington, D.C., NASA SP-106, 1966.

[71] Mahmut Reyhanoglu, "Ch 24: Modeling and Control of Space Vehicles with Fuel Slosh Dynamics," in *Advances in Spacecraft Technologies.*: InTech, 2011.

[72] Allan Lee, Alan Strahan, Rebekah Tanimoto, and Arturo Casillas, "Preliminary Characterization of the Altair Lunar Lander Slosh Dynamics and Some Implications for the Thrust Vector Control Design," in *AIAA Guidance, Navigation, and Control Conference*, Ontario, Canada, 2010, pp. 1-34.

[73] P.D. Hattis, "Qualitative Differences Between Shuttle On-Orbit and Transition Control," *Journal of Guidance, Control, and Dynamics*, vol. 7, no. 1, 1984.

[74] A.N. Penchuk, P.D Hattis, and E.T. Kubiak, "A Frequency Domain Stability Analysis of a Phase Plane Control System," *Journal of Guidance, Control, and Dynamics*, vol. 8, no. 1, 1985.

[75] G.E. Smith et al., "Application of LEM Technology to NASA Lunar Landing Research Program," 1963.

[76] Fermin Noel Garcia, "A Nonlinear Control Algorithm for Fuel Optimal Attitude Control Using Reactions Jets," Cambridge, 1998.

[77] Janice E. Voss and Philip D. Hattis, "Control Issues During Shuttle Resupply of Liquids On-Orbit," in *AIAA Guidance and Control Conference*, Seattle, 1984.

[78] Kenneth J. Cox and Philip D. Hattis, "Some Shuttle Orbit Flight Control Design Lessons: Direction for the Space Station," in *SAE Aerospace Control and Guidance Systems Committee*, Boise, 1987.

[79] Jiann-Woei Jang et al., "Ares I Flight Control System Design," in *AIAA/AAS Astrodynamic Specialist Conference*, Toronto, 2010.

[80] Sam W. Thurman and Henryk Flashner, "Robust Digital Autopilot Design for Spacecraft Equipped with Pulse-Operated Thrusters," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 5, 1996.

[81] S. Niwa, Zhou Suzuki, Nakai Kawasaki, and S.&K. Higashino, "Guidance and Control for Lunar Lander System," in *International Symposium on Space Technology and Science*, Kogashima, Japan, 1992.

[82] Jiann-Woei Jang, Michael Plummer, Nazareth Bedrossian, Charles Hall, and Pol Spanoss, "Absolute Stability Analysis of a Phase Plane Controlled Spacecraft," in *AIAA/AAS Spaceflight Mechanics Conference*, San Diego, 2009.