AN EVALUATION OF APOLLO POWERED DESCENT GUIDANCE

A Thesis

Presented to the

Faculty of

San Diego State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Aerospace Engineering

with a Concentration in

Guidance, Navigation, and Controls

---

by

Lloyd David Strohl III

Spring 2018

SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the

Thesis of Lloyd David Strohl III:

An Evaluation of Apollo

Powered Descent Guidance

Ping Lu, Chair
Department of Aerospace Engineering

Ahmad Bani Younes
Department of Aerospace Engineering

Peiman Naseradinmousavi
Department of Mechanical Engineering

Approval Date

ABSTRACT OF THE THESIS

An Evaluation of Apollo
Powered Descent Guidance
by
Lloyd David Strohl III
Master of Science in Aerospace Engineering with a Concentration in Guidance, Navigation, and Controls
San Diego State University, 2018

This is my abstract which describes my whole thesis.

Many extraterrestrial missions require a powered descent phase. Because this phase is late in the mission its fuel efficiency has an outsized effect on payload capacity. This thesis presents a strategy for optimizing fuel use using well tested guidance algorithms that reduces fuel consumption over conventional strategies by x%.

TABLE OF CONTENTS

LIST OF TABLES

PAGE

LIST OF FIGURES

GLOSSARY

ACKNOWLEDGMENTS

I would like to thank Dr. Lu for his serendipitous arrival at SDSU and consequent advice and instruction. With his mentoring I have been able to launch an enjoyable and fulfilling career in GN&C, something I could not have achieved otherwise.

CHAPTER 1
INTRODUCTION

1.1 LITERATURE REVIEW

1.2 CONTRIBUTION

1.3 ORGANIZATION

CHAPTER 2

PROBLEM STATEMENT

CHAPTER 3
METHODOLOGY

To investigate a guidance law with careful focus on the time-to-go approach, a law must be developed and implemented in a simulation framework. The Law's derivation is presented here in the context of optimal control, as is the time-to-go approach.

The simulation methodology is also described, including the numerical methods and the aerodynamic model.

3.1 GUIDANCE LAW

The guidance law under investigation is E-Guidance, first presented in Cherry 1964 [6]. E-Guidance was developed empirically by integrating the equations of motion and choosing basis functions for the thrust acceleration input $\boldsymbol{a}_T$ that provided the necessary degrees of freedom to satisfy the terminal constraints in Equations 3.5 and 3.6. With control over thrust acceleration (and therefore total acceleration), satisfying the initial conditions after integration of the equations of motion shows the need for two basis functions with vector coefficients. Cherry developed E-Guidance by considering first one guidance axis at a time, so the acceleration command took the form of Equation 3.1, where $p_1(t)$ and $p_2(t)$ are linearly independent functions of time.

$$\ddot{x} = c_1 p_1(t) + c_2 p_2(t) \tag{3.1}$$

Cherry sets $p_1(t) = 1$ and $p_2(t) = t$, mostly for simplicity while recognizing that it may be suboptimal, arriving at a form identical to the one presented below in Equation 3.13. The Cherry paper also proposes a time-to-go algorithm similar to that in Algorithm 1. It does not, however, do much to consider the optimality of this algorithm or, more critically, to deal with the problem of ignition timing, nor does most of the academic literature since.

Below E-Guidance as a control law will be derived using optimal control theory. An extension that also considers the final orientation will also be presented without claim about its optimality, but its performance will be compared with E-Guidance. The rationale behind choice of time-to-go algorithm and subsequent ignition timing will follow.

3.1.1 Equations of Motion

Derivation of an optimal powered descent guidance law begins with a formulation of the State Equation 3.2

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \tag{3.2}$$

The command input $\boldsymbol{u}$ for this problem is comprised of the thrust magnitude and direction. Aerodynamic effects are not considered for development of the law, though they will be simulated and investigated with regards to performance.

The state equations for the 3-dimensional powered descent guidance problem are as follows

$$\dot{\boldsymbol{r}} = \boldsymbol{V} \qquad \boldsymbol{r(t_0)} = \boldsymbol{r_0} \tag{3.3}$$

$$\dot{\boldsymbol{V}} = \boldsymbol{g(r)} + \boldsymbol{a_T} \qquad \boldsymbol{V(t_0)} = \boldsymbol{V_0} \tag{3.4}$$

with terminal constraints at a fixed final time t_f

$$\boldsymbol{r}(t_f) = \boldsymbol{r}_f^* \tag{3.5}$$

$$\boldsymbol{V}(t_f) = \boldsymbol{V}_f^* \tag{3.6}$$

where $\boldsymbol{a}_T$ is the thrust acceleration vector. $\boldsymbol{a}_T$ is limited such that

$$0 < a_{min} \leq ||\boldsymbol{a}_T|| \leq a_{max} \tag{3.7}$$

and $\boldsymbol{g}(\boldsymbol{r})$ is gravitational acceleration

$$\boldsymbol{g} = -\frac{\mu \boldsymbol{r}}{(\boldsymbol{r}^T \boldsymbol{r})^{(3/2)}} \tag{3.8}$$

Where μ is the standard gravitational parameter for the body in question. For Mars, $\mu \approx 4.282 * 10^{13}$

3.1.2 Performance Index

Fuel consumption is related to the thrust acceleration vector by engine parameters represented by some positive constant k

$$\dot{m} = -k||\boldsymbol{a}_T|| \tag{3.9}$$

A fuel optimal guidance law should therefore use the performance index

$$J = \int_{t_0}^{t_f} ||\boldsymbol{a}_T|| dt \tag{3.10}$$

Choosing to minimize the square of the total acceleration $\boldsymbol{a} = \boldsymbol{g} + \boldsymbol{a}_T$ gives a performance index

$$J = \frac{1}{2}\int_{t_0}^{t_f} (\boldsymbol{g} + \boldsymbol{a}_T)^T(\boldsymbol{g} + \boldsymbol{a}_T)dt \tag{3.11}$$

For a constant gravitational acceleration $\boldsymbol{g}$, this performance index attempts to minimize $||\boldsymbol{a}_T||^2$. It is not fuel optimal as in Equation 3.10, but it does provide a cost to large thrust accelerations and might be expected to give good fuel performance.

3.1.3 Guidance solution

Choosing the guidance command $\boldsymbol{u} = \boldsymbol{g} + \boldsymbol{a}_T$ and applying optimal control theory results in the following

$$H = \boldsymbol{p}_r^T\boldsymbol{V} + \boldsymbol{p}_V^T\boldsymbol{u} - \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u} \tag{3.12}$$

$$\begin{aligned}
\dot{\boldsymbol{p}}_r = -\frac{\partial H}{\partial \boldsymbol{r}} = 0 &\implies \boldsymbol{p}_r = -\boldsymbol{c}_2 \\
\dot{\boldsymbol{p}}_V = -\frac{\partial H}{\partial \boldsymbol{V}} = -\boldsymbol{p}_r &\implies \boldsymbol{p}_V = \boldsymbol{c}_1 + \boldsymbol{c}_2 t \\
\frac{\partial H}{\partial \boldsymbol{u}} = 0 &\implies \boldsymbol{u} = \boldsymbol{p}_V = \boldsymbol{c}_1 + \boldsymbol{c}_2 t
\end{aligned}$$

For convenience, let $\tau = t_f - t$

$$\boldsymbol{u} = \boldsymbol{k}_1 + \boldsymbol{k}_2\tau \tag{3.13}$$

where $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are constant vectors.

Integrating the equations of motion with $\dot{\boldsymbol{V}} = \boldsymbol{u}$ then gives

$$\int \dot{\boldsymbol{V}}(t)dt = \boldsymbol{k}_1(t - t_0) + \frac{1}{2}\boldsymbol{k}_2(t - t_0)^2 + \boldsymbol{V}(t_0) \tag{3.14}$$

$$\int \dot{\boldsymbol{r}}(t)dt = \frac{1}{2}\boldsymbol{k}_1(t - t_0)^2 + \frac{1}{6}\boldsymbol{k}_2(t - t_0)^3 + \boldsymbol{V}(t_0)(t - t_0) + \boldsymbol{r}(t_0) \tag{3.15}$$

Setting $t = t_f$ and letting $t_{go} = t_f - t_0$ satisfies the terminal constraints from Equations 3.5 and 3.6, resulting in 6 linear equations in 6 unknowns

$$\boldsymbol{k}_1 t_{go} + \frac{1}{2}\boldsymbol{k}_2 t_{go}^2 = \boldsymbol{V}_f^* - \boldsymbol{V}_0 \tag{3.16}$$

$$\frac{1}{2}\boldsymbol{k}_1 t_{go}^2 + \frac{1}{6}\boldsymbol{k}_2 t_{go}^3 = \boldsymbol{r}_f^* - \boldsymbol{r}_0 - \boldsymbol{V}_0 t_{go} \tag{3.17}$$

These equations can be separated into sets of two per vector component. Define an inertial guidance frame $\boldsymbol{e} = (\hat{x}, \hat{y}, \hat{z})^T$ such that guidance vector $\boldsymbol{u}$ is composed of components in $\boldsymbol{e}$, $\boldsymbol{u} = (u_x, u_y, u_z)^T$. For the equations in $\hat{x}$ we have

$$\begin{bmatrix} t_{go} & \frac{1}{2}t_{go}^2 \\ \frac{1}{2}t_{go}^2 & \frac{1}{6}t_{go}^3 \end{bmatrix} \begin{pmatrix} k_{1_x} \\ k_{2_x} \end{pmatrix} = \begin{pmatrix} V_{f_x}^* - V_{0_x} \\ r_{f_x}^* - (r_{0_x} + V_{0_x} t_{go}) \end{pmatrix} \tag{3.18}$$

Solving the two-equation system is accomplished by inverting the A matrix, leading to a coefficient matrix E

$$E = \begin{bmatrix} -2/t_{go} & 6/t_{go}^2 \\ 6/t_{go}^2 & -12/t_{go}^3 \end{bmatrix} \tag{3.19}$$

The coefficients in $\hat{x}$ are then

$$\begin{pmatrix} k_{1_x} \\ k_{2_x} \end{pmatrix} = E \begin{pmatrix} V_{f_x}^* - V_{0_x} \\ r_{f_x}^* - (r_{0_x} + V_{0_x} t_{go}) \end{pmatrix} \tag{3.20}$$

It can be shown that the equations in $\hat{y}$ and $\hat{z}$ take the same form. This $2x2$ E matrix is the origin of the name *E-Guidance*, the guidance law used in the Apollo lunar landing missions.

Of some interest is the addition of a final attitude constraint. For a vehicle whose attitude it determined by the thrust acceleration vector, this constraint can be implemented as a final thrust acceleration constraint as in Equation 3.21.

$$\boldsymbol{a}_T(t_f) = \boldsymbol{a}_{T_f}^* \tag{3.21}$$

This vector constraint cannot be satisfied with only two basis functions for the command $\boldsymbol{u}$, so a third linearly independent function must be introduced such that $\boldsymbol{u} = \boldsymbol{c}_1 p_1(t) + \boldsymbol{c}_2 p_2(t) + \boldsymbol{c}_3 p_3(t)$. A tempting choice for the third basis function is $p_3(t) = t^2$ for simplicity, with the other two functions the same as E-Guidance.

After applying the substitution from Equation 3.13, this choice gives a command $\boldsymbol{u} = \boldsymbol{k}_1 + \boldsymbol{k}_2\tau + \boldsymbol{c}_3\tau^2$. This form for the command input is also easily integrable and leads to a similar linear system of 9 equations in $\boldsymbol{k_1}$, $\boldsymbol{k_2}$, and $\boldsymbol{k_3}$. Using the same guidance frame $\hat{e}$ defined for Equation 3.18 and considering one coordinate a time we get a system similar to E-Guidance in Equation 3.22

$$\begin{pmatrix} k_{1_x} \\ k_{2_x} \\ k_{3_x} \end{pmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 18/t_{go}^2 & -24/t_{go}^3 & -6/t_{go} \\ -24/t_{go}^3 & 36/t_{go}^4 & 6/t_{go}^2 \end{bmatrix} \begin{pmatrix} V_{f_x}^* - V_{0_x} \\ r_{f_x}^* - (r_{0_x} + V_{0_x} t_{go}) \\ g_x + a_{f_x}^* \end{pmatrix} \tag{3.22}$$

3.1.4 Time-to-go

The E-Guidance solution depends upon a reliable estimate of remaining time-to-go (t_{go}). The Apollo mission's guidance used an estimate that updated continuously using Newton's method, but it was intended to only operate until start of the terminal descent phase at which point guidance switched to a manual vertical descent operation. Updating the t_{go}

estimate continuously is attractive since it should be robust; if conditions have to change during the mission a closed-loop (continuously updating) solution will adjust and a new, realistic t_{go} will feed into the guidance solution. This quality was important to the Apollo Guidance solution because it relied upon pilot inputs to define the landing location visually, which meant allowing for landing site redesignations mid-mission. If t_{go} was not recomputed after site redesignation, the guidance law would command unrealizable thrust acceleration commands.

For the purposes of this study, live landing site redesignation was not considered. Without the possibility of landing site redesignation, an open-loop t_{go} solution lends the guidance law more stability in that the performance is less dependent upon specific assumptions and conditions imposed by the t_{go} algorithm. For instance, one closed-loop t_{go} algorithm is implemented in Algorithm 1.

Algorithm 1 Fixed-Point-Iteration t_{go}

procedure FPI(t_{go_1})
  $tol \leftarrow c$
  while $|t_{go_0} - t_{go_1}| \geq tol$ **do**
    $t_{go_0} \leftarrow t_{go_1}$
    $\Delta V \leftarrow \sqrt{(\boldsymbol{V} - \boldsymbol{V_0} + \boldsymbol{g} \cdot t_{go})^T (\boldsymbol{V} - \boldsymbol{V_0} + \boldsymbol{g} \cdot t_{go_0})}$
    $t_{go_1} \leftarrow \frac{m_0}{\dot{m}} \left(e^{\frac{-\Delta V}{v_{ex}}} - 1 \right)$ $\quad \triangleright\ \dot{m} < 0$
  end while
  return t_{go_1}
end procedure

Each guidance update uses the previous update's t_{go} minus clock time as its initial guess t_{go_0}, and the max iterations may be limited to some reasonable number.

This algorithm requires an assumption about a fixed mass flow rate $\dot{m}$ which is not guaranteed by the guidance law. Adjustment of this mass flow rate estimate is very particular to the initial conditions of the mission, resulting in a necessarily conservative t_{go} to account for initial condition dispersion.

One attractive open-loop option is the time to perform a gravity turn landing at maximum thrust. Equation 3.23 for a gravity turn time-to-go, $t_{go_{GT}}$, was presented first in Cherry 1964 [6]. After engine ignition and initiation of the Powered Descent Guidance, the updated time-to-go is computed as $t_{go_{GT}}$ minus elapsed clock time. The algorithm for time-to-go using a gravity turn maneuver is given in Equation 3.23, where a_{GT} is the thrust acceleration magnitude applied during the maneuver. This equation gives two roots for a_{GT}, a positive root and a negative root. The desired root is positive, making it a simple matter to find. This quantity will be important in section 3.1.5.

$$
\begin{aligned}
\gamma &= \frac{\pi}{2} - cos^{-1}\left(\frac{\boldsymbol{r}^T\boldsymbol{V}}{||\boldsymbol{r}||||\boldsymbol{V}||}\right) \\
g_m &= ||\boldsymbol{g}|| \\
r_m &= ||\boldsymbol{r}|| \\
V_m &= ||\boldsymbol{V}|| \\
a &= 1/g_m^2 \\
b &= \frac{sin(\gamma)V_m^2}{2(r_m - R_M)g_m^2} \\
c &= -\frac{V_m^2(1 + sin(\gamma)^2)}{4(r_m - R_M)g_m} + 1 \\
a_{GT} &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \\
t_{go_{GT}} &= \frac{V_m}{2}\left(\frac{1 + sin(\gamma)}{a_{GT} + g_m} + \frac{1 - sin(\gamma)}{a_{GT} - g_m}\right)
\end{aligned}
\tag{3.23}
$$

A gravity turn landing does not directly apply to the general powered descent guidance problem under investigation because it does not seek to satisfy the constraints given in Equations 3.5 and 3.6. However, if the magnitude of the terminal velocity target in Equation 3.6 is small, the required time to decelerate from an initial $\boldsymbol{V}_0$ under only the forces of thrust acceleration $\boldsymbol{a}_T$ and gravity $\boldsymbol{g}$ is, at minimum, the gravity turn solution $t_{go_{GT}}$. Since the terminal position $\boldsymbol{r}_f$ is specified, the vehicle necessarily needs more time than given by the gravity turn solution to satisfy it. Assuming a small trajectory error requiring a small diversion to landing site, a small constant factor $c_t \approxeq 1.2$ can be applied to the gravity turn time-to-go to allow for redirection. Because the fuel optimal guidance law would minimize the index given in Equation 3.10, reducing the time of flight should be expected to reduce propellant consumption. This suggests that driving the factor c_t as close to 1 as possible while still landing softly is desirable. The choice of $c_t = 1.2$ will prove to be sufficiently conservative to survive initial condition dispersion, rocket parameter dispersion, and navigation error.

3.1.5 Ignition Timing

The literature has also given very little attention to the topic of ignition timing, or the initiation of powered descent guidance (PDI). Mission planning involves phases and guidance laws like E-Guidance

Here we talk about the thrust acceleration and range criterion for ignition.

... where h is altitude and s_{GT} is the downrange distance traveled during a gravity turn maneuver.

$$s_{GT} = \frac{V_m^2}{2 * a_{GT})} * cos(\gamma) * \frac{V_m^2 + 2 * g_m * h}{V_n orm^2 + g_m * h} * \frac{R_M}{r_m} \tag{3.24}$$

3.2 SIMULATION

The powered descent simulation (PD Sim) is developed using standard programming techniques and numerical methods. The code is modular to facilitate design and reflect the functions of a mission computer. The simulation has three degrees of freedom (3-DoF) with derived orientation for force modeling. The simulation takes a set of initial conditions $\boldsymbol{r}_0$ and $\boldsymbol{V}_0$, a set of final conditions $\boldsymbol{r}_f$ and $\boldsymbol{V}_f$, dispersed rocker parameters, and nominal rocket parameters for the guidance module to use in its calculations of throttle setting.

At the core of the simulation is numerical time integration. The method employed is a 4th order Runge-Kutta (RK4). The RK4 function is called at each time step to progress the simulation forward.

The separate modules called during simulation are the guidance computer which contains the guidance law and computes a commanded thrust acceleration vector, the navigation module which generates a state estimate from simulated instrument measurements for use by the guidance computer, and the aerodynamic module which contains a model of the landing vehicle to compute forces and orientation for use in integration.

The simulation stops when either the vehicle passes through zero altitude (crashes) or the time-to-go is less than a single time integration step. When the latter condition is met, the time integration step is reduced to the remaining time-to-go and one more iteration is performed.

The full simulation code is wrapped in a Monte-Carlo script. The Monte-Carlo script calls the PD Sim with simulation settings, vehicle parameters, initial condition, and navigation uncertainty. This allows each individual run to have dispersed conditions without modification of the code.

Further detail for each part of the simulation follows.

3.2.1 Numerical Integration

The numerical time integration is done via a standard Runge-Kutta formulation, presented here as in Ferziger and Perić [8]. The state vector $\boldsymbol{\phi}$ is passed to the equations of motion $\boldsymbol{f}(\boldsymbol{\phi}, t)$ several times as in Equation 3.25.

$$\begin{aligned}
\boldsymbol{\phi}^{*}_{n+\frac{1}{2}} &= \boldsymbol{\phi}_n + \frac{\Delta t}{2}\boldsymbol{f}(t_n, \boldsymbol{\phi}_n) \\
\boldsymbol{\phi}^{**}_{n+\frac{1}{2}} &= \boldsymbol{\phi}^n + \frac{\Delta t}{2}\boldsymbol{f}(t_{n+\frac{1}{2}}, \boldsymbol{\phi}^{*}_{n+\frac{1}{2}}) \\
\boldsymbol{\phi}^{*}_{n+1} &= \boldsymbol{\phi}^n + \Delta t \boldsymbol{f}(t_{n+\frac{1}{2}}, \boldsymbol{\phi}^{**}_{n+\frac{1}{2}}) \\
\boldsymbol{\phi}_{n+1} &= \boldsymbol{\phi}^n + \frac{\Delta t}{6}[\boldsymbol{f}(t_n, \boldsymbol{\phi}_n) + 2\boldsymbol{f}(t_{n+\frac{1}{2}}, \boldsymbol{\phi}^{*}_{n+\frac{1}{2}}) \\
&\quad + 2\boldsymbol{f}(t_{n+\frac{1}{2}}, \boldsymbol{\phi}^{**}_{n+\frac{1}{2}}) + \boldsymbol{f}(t_{n+1}, \boldsymbol{\phi}^{*}_{n+1})]
\end{aligned} \tag{3.25}$$

The EOM function computes the derivative of the state vector $\boldsymbol{\phi}$. It is the same as the modeled equations of motion in Equations 3.3 and 3.4 but with an additional force term, $\boldsymbol{F}_{LD}$, to represent the aerodynamic forces of Lift and Drag. $\boldsymbol{\phi}$ is passed in as a $6x1$ column vector such that $\boldsymbol{\phi} = [\boldsymbol{r}^T, \boldsymbol{V}^T]^T$, and the derivatives are calculated as in Equations 3.26 and 3.27.

$$\dot{\boldsymbol{r}} = \boldsymbol{V} \tag{3.26}$$

$$\dot{\boldsymbol{V}} = \boldsymbol{g(r)} + \boldsymbol{a_T} + \boldsymbol{F}_{LD} \tag{3.27}$$

For simplicity, the vehicle's thrust acceleration response to a guidance command is modeled as zero-lag, i.e. perfect control. It assumes that the vehicle instantly and perfectly responds to the commanded $\boldsymbol{a}_T$, so the command from the guidance computer, after being put through a thrust magnitude limiter, is the same as the term in Equation 3.27.

3.2.2 Guidance Computer

The guidance computer takes as input the current state as provided by the navigation system, the terminal constraints (final position and velocity in the case of E-Guidance, with final acceleration as well in the case of advanced E-Guidance with attitude constraint), nominal rocket max thrust, current vehicle mass, time-to-go, and selection of guidance law.

The output is a throttle setting as a fraction of the nominal max thrust and a direction for the thrust acceleration $\boldsymbol{a}_T$. These values are unlimited at the level of the guidance computer, i.e. the throttle setting can be larger than 1 requiring thrust magnitude greater than T_{max}.

In this implementation, the throttle is limited after being passed out of the guidance computer to satisfy the real performance constraint represented in Equation 3.7. Since the guidance computer only receives nominal T_{max} upon which to base its throttle setting, the resultant command can be off due to engine performance dispersion.

The guidance command is updated periodically at a rate lower than the simulation rate. It must also stop updating when t_{go} becomes small due to its presence in the denominator for several entries in the E matrices of Equations 3.18 and 3.22. For the last half second the thrust acceleration is held constant. This has very little impact on landing precision.

The time-to-go computation takes place in the guidance module. The computation method is as described in 3.1.4. A gravity turn estimate is used to initialize the simulation, but it is reset upon ignition based on on the current conditions and then allowed to run open-loop until simulation stop.

3.2.3 Ignition trigger

3.2.4 Navigation Module

The navigation module is fairly simplistic. It has two functions.

The first function is to produce navigation error in the form of normally distributed noise. The function takes the actual vehicle state $\boldsymbol{r}$ and $\boldsymbol{V}$, adds noise with a distribution specified in the simulation settings of the Monte Carlo script, and creates $\boldsymbol{r}_{nav}$ and $\boldsymbol{V}_{nav}$ vectors. The noise distribution is chosen to reflect realistic navigation error as provided by an inertial navigation system per the trade study presented in Moesser 2010 [16]. Typical standard deviations are 1 m position error and 1/3 m/s velocity error.

The second function is to filter the noisy navigation measurements to produce a smoother estimate of vehicle state. It does this using a simple low-pass filter as in Equation 3.28, where α is a chosen filter constant (here $\alpha = 0.3$), $\boldsymbol{x}$ is the current nav output of state, $\boldsymbol{x}_{prev}$ is the previous filtered state estimate, and $\boldsymbol{x}_{est}$ is the current filtered state estimate.

$$\boldsymbol{x}_{est} = \alpha \boldsymbol{x}_{prev} + (1 - \alpha)\boldsymbol{x} \tag{3.28}$$

A Kalman filter would likely be used on a more sophisticated navigation system for a real aerospace vehicle, but given the simplicity of the navigation error and perfect knowledge of the noise distrubution, modeling a Kalman filter would likely provide too good an estimate effectively canceling the effects of simulated navigation error.

3.2.5 Aerodynamic Model

The Aerodynamic model is comprised of Mars atmosphere data from (insert atmosphere data reference), and the vehicle model was developed by Cerimele et. al [5]

3.2.5.1 Atmosphere Model

The atmosphere model is based on NASA's Mars-GRAM [12], an engineering-level atmospheric model of the Martian atmosphere. The model used for this simulation is an empirically developed equation to represent the Martian atmosphere data within the mission's envelope. Mars-GRAM 2010 is based on NASA Ames Mars General Circulation Model for altitudes under 80 km.

For this simulation, the Mars-GRAM data is used to calculate a density ρ and a speed of sound V_{sound}, from which Mach number is calculated as $|V|/V_{sound}$ and lift and drag are calculated as in Equations 3.30 and 3.29, where S is a reference area.

$$L = \frac{1}{2}\rho|V|^2 SC_L \tag{3.29}$$

$$D = \frac{1}{2}\rho|V|^2 SC_D \tag{3.30}$$

A particular feature of the powered descent phase is the change in drag calculation while the engine is ignited. When the guidance system is operating and the rocket is firing, the drag is modeled as half the value computed when the vehicle is unpowered, i.e. during glide. This is due to the rocket plume's aerodynamics. This change is accomplished in the simulation during the ignition timing phase by simply dividing the reference area by a factor of 2.

3.2.5.2 Lift and Drag

The vehicle aerodynamic model is based on the work of Cerimele et al [5]. The vehicle is the CobraMRV, a "rigid, enclosed, elongated lifting body shape that provides a higher lift-to-drag ratio (L/D) than a typical entry capsule..." [5]. It was designed as an atmospheric entry and powered descent and landing vehicle for manned Mars missions compatible with the current NASA Human Mars mission architectures. These missions require delivery of roughly 20 tonnes of cargo to the surface.

The work of Cerimele er al provides a model of the lift and drag coeffecients C_L and C_D based on vehicle Mach number and angle of attack. Vehicle angle of attack is determined directly from the thrust acceleration vector $\boldsymbol{a}_T$ and the vehicle velocity vector $\boldsymbol{V}$.

During the unpowered phase of approach when ignition timing is being optimized, the vehicle is held at a constant angle of attack of $55°$ to achieve maximum drag.

3.2.5.3 Orientation

The vehicle's body Euler angles are computed in a 3-2-1 sequence from the body axes. Since the simulation assumes perfect control, the body axes are computed as Equations 3.31, 3.32, 3.33. Pitch, yaw, and roll are then calculated as the angles between the body axes and the planet-fixed guidance frame $\hat{e}$. Angle of attack is similarly calculated using the body axes and the velocity vector.

$$\boldsymbol{body}_Y = \boldsymbol{a}_T / ||\boldsymbol{a}_T|| \tag{3.31}$$

$$\boldsymbol{body}_P = \frac{\boldsymbol{r} \times \boldsymbol{body}_Y}{||\boldsymbol{r} \times \boldsymbol{body}_Y||} \tag{3.32}$$

$$\boldsymbol{body}_R = \boldsymbol{body}_Y \times \boldsymbol{body}_R \tag{3.33}$$

This calculation ensures zero side-slip, that the vehicle will always be pointed in the direction it is traveling. Figure **??** shows a small changing yaw angle throughout flight since it is computed relative to a fixed guidance frame centered on the landing site, and the vehicle's initial condition is rotated around the curvature of the planet.

3.2.6 Monte Carlo

CHAPTER 4

RESULTS AND DISCUSSION

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

CHAPTER 6
REFERENCING

Below a list of references are provided in the acceptable format for Master's thesis submission. References are to be numbered and should appear either alphabetically or in the order of appearance in the text. (LaTeX does the former for the student.) For students using LaTeX these are obtained using the plain style with BibTeX. The Department of Mathematics and Statistics will accept either the plain style or the SIAM style. (For the SIAM style, get a copy of the SIAM.BST file from your graduate adviser or the Mathematical Sciences computer system.) There are references for journal `articles` [1], `books` and `booklets` [4, 22], `inbooks`, `incollections`, and `inproceedings` [3, 7, 20]. *Note that when you have more than one citation in a single bracket they must be in increasing numerical order!* Other sources may be `proceedings` [2], technical reports (`techreport`) [18], theses (`mastersthesis`, or `PhDthesis`) [9], or `unpublished` material [19]. This should provide a fairly comprehensive list for any material that the student may encounter. For additional assistance, see the graduate adviser in your area of concentration. LaTeX source codes are available for copying. ***Style note***

If you cite a website [17] and you can't find the year on the website, you should put "n.d." (not dated) at the end. (this is true for other reference also.) It must also has the word "accessed" and the month and year you access the website. You can change how things with no author(s) are sorted in the bibliography by supplying a `key` entry (see `thbib.bib`), *e.g.* this news release [21] will be sorted under "U," the leading letter of the publishing agency (as preferred by the thesis publisher).

This [10] is an example of a patent. ***Notice:*** how the `month` and `year` fields in `thbib.bib` have been abused to force the "correct" format.

BIBLIOGRAPHY

[1] T. Abraham. Mathematical study of γ-rings in a Hilbert space. *J. Math. Anal. Appl.*, 19:125–128, 1984.

[2] P. Axelrod and C. P. Snow, editors. *Proceedings of the Conference on Mathematical Population Genetics*, New York, 1982. Marcel Dekker.

[3] B. J. Bach. Homotopy theory for the delay differential equation $\dot{y}(t) = Ay(t - \tau)$. In F. Neuerfeldt, editor, *Concepts in Differential Equations*, volume 2, pages 807–876. Academic Press, Washington, D.C., 1987.

[4] G. T. Bankhead. *Modeling and Control in the Mathematical Sciences*, volume 69 of *Lecture Notes in Mathematics*. Springer, Berlin, 1975.

[5] C. Cerimele et al. A rigid mid lift-to-drag ratio approach to human mars entry, descent, and landing. *AIAA Paper*, pages 2017–1898, 2017.

[6] G. W. Cherry. A general, explicit, optimizing guidance law for rocket-propelled spaceflight. *AIAA Paper*, pages 64–638, 1964.

[7] R. DeWitt. *Abstract Functional Equations for Fluids*, chapter 9, pages 274–293. Addison-Wesley, New York, 1983.

[8] J. H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. Springer, Berlin, 2002.

[9] H. B. Finknoddle. Random processes for α-surds in a complex topology. Master's thesis, San Diego State University, San Diego, CA, 1990.

[10] Bill Gates. The Knife and Fork: Novel Eating Utensils for the Mass Consumption of Plant and Animal Based Food Items, U.S. Patent No. 0,000,000. January 19, 2038.

[11] S J. Citron, S E. Dunin, and H F. Meissinger. A terminal guidance technique for lunar landing. *AIAA Journal*, 2:503–509, 1964.

[12] Hilary L. Justh. Mars global reference atmospheric model. NASA, https://see.msfc.nasa.gov/model-Marsgram, accessed February 2018, 2010.

[13] A. R. Klumpp. Apollo lunar descent guidance. *Automatica*, 10:133–146, 1974.

[14] L. Lamport. *LaTeX: A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, 1986.

[15] J. M. Mahaffy, D. A. Jorgensen, and R. L. Vanderheyden. Oscillations in a model of repression with external control. *Quart. Appl. Math.*, 50:415–435, 1992.

[16] T. J. Moesser. Guidance and navigation linear covariance analysis for lunar powered descent. Master's thesis, Utah State University, Logan, UT, 2010.

[17] Misc Peeps. The truth about everything. Wikipedia, http://www.wikipedia.com/, accessed August 2012, n.d.

[18] I. T. Simon and A. M. McGeorge. Integration of large Bessel functions. Technical Report 284, San Diego State University, San Diego, CA, 1989.

[19] J. K. Slemrod. P-infinity norms in the physical sciences. Unpublished report, 1990.

[20] B. W. Stuart. Contour mappings over closed Ω-groups. In G. V. Avery, editor, *Complex Algebras*, volume $VOLUME_REQUIRED$, page $PAGES_REQUIRED$, Philadelphia, PA, 1987. SIAM.

[21] UNITED STATES ENVIRONMENTS PROTECTION AGENCY. *EPA Challenges Colleges to Recycle at Football Games / Agency encourages fans to Reduce, Reuse, Recycle.* News Release, Sept. 7, 2010. http://yosemite.epa.gov/opa/admpress.nsf/2010%20Press%20Releases?OpenView, accessed September 2010.

[22] Writing style for the mathematical sciences. AMS, Providence, RI, 1979.

[11] [6] [13] [5]

APPENDIX A

MORE INFORMATION ON EQUATIONS

MORE INFORMATION ON EQUATIONS

To demonstrate how an appendix should be inserted into the thesis we have provided two appendices. This first appendix illustrates some more advanced techniques to improve the appearance of your equations. Below is a system of partial differential equations from a model for cellular control by an external nutrient. The equations are complicated and LaTeX tends to allow them to run into each other. To prevent this we have used the `\vrule` command to separate them. Note this is an ordinary TeX command and is not in L. Lamport's book [14]. Furthermore, we have some complicated boundary conditions that we needed to align, so we used the array command, but to get the equations looking right we also needed the `\dfrac` command instead of the `\frac` command. The equations for our model are as follows:

$$\begin{aligned}
\dot{U}_1(t) &= \tilde{f}(W_1(t-T)) - U_1(t) + \gamma_1 U_2(R\sigma, t), \\
\dot{W}_1(t) &= -\hat{b}_3 W_1(t) + \gamma_3 W_2(R\sigma, t), \\
\frac{\partial U_2}{\partial t} &= D_1 \nabla^2 U_2 - U_2 - \tilde{f}(W_1(t-T)) - \gamma_1 U_2(R\sigma, t), \qquad \text{(A.1)} \\
\frac{\partial V_2}{\partial t} &= D_2 \nabla^2 V_2 - b_2 V_2 + c_0\big(U_2 + U_1(t)\big), \\
\frac{\partial W_2}{\partial t} &= D_3 \nabla^2 W_2 - b_3 W_2 + (\hat{b}_3 - b_3) W_1 - \gamma_3 W_2(R\sigma, t) \\
&\quad + k \left[\left[\left(\frac{D_3}{r^2} \right) \frac{d}{dr} \left(r^2 \frac{dh}{dr} \right) - b_3 h \right] V_2(R,t) - h \dot{V}_2(R,t) \right],
\end{aligned}$$

for $t > 0$ and $R\sigma < r < R$ and with the boundary conditions:

$$\begin{array}{rclcrcl}
\dfrac{\partial U_2(R\sigma,t)}{\partial r} & = & \beta_1 U_2(R\sigma,t), & \qquad & \dfrac{\partial U_2(R,t)}{\partial r} & = & 0, \\[2ex]
\dfrac{\partial V_2(R\sigma,t)}{\partial r} & = & 0, & & \dfrac{\partial V_2(R,t)}{\partial r} & = & 0, \\[2ex]
\dfrac{\partial W_2(R\sigma,t)}{\partial r} & = & \beta_3 W_2(R\sigma,t), & & \dfrac{\partial W_2(R,t)}{\partial r} & = & 0.
\end{array}$$

Notice that the system is numbered only once by (A.1) and that this is centered as best we can on one line. All other lines have the *nonumber* command.

A.1 THEOREMS

The appendix can also include technical theorems and lemmas which are call in the same manner as before. For example,

Theorem A.1. *The system of equations (A.1) can exhibit periodic solutions for certain parameter values.*

Proof. The argument uses Hopf bifurcation techniques and is very complicated. See Mahaffy *et al* [15]. □

APPENDIX B

LISTS AND QUOTATIONS

LISTS AND QUOTATIONS

The thesis will rarely use list environments, but they are valuable for résumés. For more information on creating a résumé you may want to see the author of this document (you also need to learn quite a bit about `\parbox` commands). To create a list you will want to use one of `itemize`, `enumerate`, or `description`. For example:

continuous A function f is **continuous** at x if and only if for every $\varepsilon > 0$ there exists a $\delta(x) > 0$ such that whenever $|y - x| < \delta$, $|f(y) - f(x)| < \varepsilon$.

uniformily continuous A function f is **uniformly continuous** if and only if for every $\varepsilon > 0$ there exists a $\delta > 0$ such that whenever $|y - x| < \delta$, $|f(y) - f(x)| < \varepsilon$ independent of x and y.

equicontinuous A family of functions f_n is **equicontinuous** at a point x if and only if for every $\varepsilon > 0$ there exists a $\delta > 0$ such that whenever $|y - x| < \delta$, $|f_n(y) - f_n(x)| < \varepsilon$ for all functions f_n.

LaTeX provides an environment for block quotations. To agree with the thesis manual follow the format below for a quotation exceeding four lines. From Lewis Carrol's *Hunting of the Snark* we hear the Bellman tell his crew:

> The Bellman himself they all praised to the skies–
> Such a carriage, such ease and such grace!
> Such solemnity, too! One could see he was wise,
> The moment one looked in his face!
>
> He had bought a large map representing the sea,
> Without the least vestige of land:
> And the crew were much pleased when they found it to be
> A map they could all understand.
>
> "What's the good of Mercator's, North Poles and Equators,
> Tropics, Zones, and Meridian Lines?"
> So the Bellman would cry: and the crew would reply,
> "They are merely conventional signs!"
>
> "Other maps are such shapes, with their islands and capes!
> But we've got our brave Captain to thank"
> (So the crew would protest) "that he's bought us the best–
> A perfect and absolute blank!"