

PATH-ADAPTIVE GUIDANCE ALGORITHM TRADES FOR A TWO-STAGE LUNAR DESCENT VEHICLE

Jason M. Everett^{*}, Anand R. Iyer[†]

For the next generation of NASA's missions, the necessity for path-adaptive guidance algorithms has become clear in order to provide the stability and customizability required for a safe and efficient descent to the lunar surface, while meeting specified program and vehicle constraints. Several descent algorithms have been tested and flown for single-stage landers through the Apollo and Altair programs, but thus far little analysis has been conducted involving the application of these algorithms for a two-stage descent vehicle. Due to the limited payload mass constraints of the existing fleet of launch vehicles, multi-stage descent architectures have become a viable course of action. This paper seeks to compare the performance of guidance configurations for a lunar lander system consisting of two stages, one of which separates partway through descent. Through development of this paper, an optimization suite has been written that is specifically designed for optimizing planetary non-atmospheric two-stage descent trajectories, and is used as a comparison baseline for the guidance algorithms tested. Time-to-go computational methods and ignition logic routines that may be employed in a lunar environment are also discussed. Further work is to be completed on trajectory design trades as well as the effects of modifying guidance targets in simulation based on trajectories that are optimized for different performance indices.

INTRODUCTION

As NASA continues development on the Artemis program and maintains support on other missions that require cis-lunar capabilities, engineers still face a number of obstacles that must be overcome. Longer-term lunar missions designed to be sustainable are inherently more complex. Maximizing payload mass to the Moon's South Pole demands higher efficiency in fuel usage. Flexibility in trajectory design is a requirement in order to enable landings to multiple locations on the surface as well as to facilitate the development of reusable spacecraft. In light of these obstacles, the need for autonomous and robust lunar descent guidance is apparent. The guidance laws employed on Apollo and other lunar lander missions have been proven over the last few decades to be near fuel-optimal¹. Work is continuing to be done on finding more fuel-optimal solutions without a large increase in complexity²; however, the simplicity and elegance of the algorithms with flight heritage cannot be ignored. Each algorithm that has been flown or tested in simulation has its respective advantages and disadvantages. Yet, throughout the development and application of powered descent algorithms completed in past studies, little work has been done on testing planetary descent guidance laws in a two-stage vehicle configuration.

^{*} Aerospace Engineer, EV42/Guidance, Navigation, and Mission Analysis Branch, NASA Marshall Space Flight Center, Huntsville, AL

[†] Aerospace Engineer, EV42/Guidance, Navigation, and Mission Analysis Branch, NASA Marshall Space Flight Center, Huntsville, AL

In two-stage vehicle architectures, program constraints and realistic operations typically require at least some predetermined nonzero coast phase, and commonly a restriction on propellant usage of the first stage of descent (if, for example, that first stage is to perform a maneuver after separation to return to a nominal parking orbit). One benefit of the explicit guidance algorithms that have been proven in flight for single-stage powered descent vehicles is their robustness to path dispersions. Therefore, a reasonable hypothesis that this paper seeks to validate is that a two-stage descent vehicle can use some form or configuration of these heritage single-stage algorithms, and that mission complexities (such as zero-thrust coast phases) will be handled autonomously by the path-adaptive nature of the algorithms, while still retaining near fuel-optimality.

This paper aims to focus on the near-optimality of the guidance algorithms tested in a two-stage vehicle configuration. The Background section is dedicated to both the heritage of previous planetary guidance laws and illustrates how these laws can be employed on a two stage vehicle. This section also examines viable methods for determining fuel-optimal burn times and burn starting conditions. The Performance section shows simulation results that implement these guidance configurations for two-stage vehicles, and compares these simulated trajectories to ideal optimal descent trajectories. This section also proposes performance metrics to address how far these guidance algorithms deviate from their respective ideal optimal trajectories. The Conclusion section summarizes the results of this paper and discusses future work for this study.

BACKGROUND

The following section outlines the derivations of guidance laws that have been used in historical powered descent scenarios (both in flight and in simulation). Notes on current simulation methods for approximating a near fuel-optimal burn time, as well as a proper starting ignition state, are also presented in this section.

Heritage Descent Steering Laws

The generalized problem of powered descent guidance can be constructed as the following statement: given a current state $\vec{r}_0$ and $\vec{v}_0$, subject to a specified set of equations of motion, the desired commanded acceleration $\vec{a}_T(t)$ is solved for, such that the vehicle is guided to some desired final state $\vec{r}_f$ and $\vec{v}_f$ over some period of time $t = t_{go}$. Several guidance laws have been developed over the past five decades that solve this problem with tradeoffs in fuel optimality, time minimization, and vehicle pointing constraints³. A simple solution that bounds the final position and velocity $\vec{v}_f$ and $\vec{r}_f$ can be obtained by assuming that the acceleration on each Cartesian axis takes on a linear profile⁴:

$$\vec{a}_T(t) = \vec{c}_0 + \vec{c}_1 t \quad (1)$$

One can solve for the coefficient vectors $\vec{c}_0$ and $\vec{c}_1$ analytically with the knowledge of the current state $\vec{r}_0$ and $\vec{v}_0$, and the target final state $\vec{r}_f$ and $\vec{v}_f$, and by knowing an a-priori total burn time t_{go} (whose computation is discussed more in the next section). If t in Equation (1) represents the time since the solution calculation of $\vec{c}_0$ and $\vec{c}_1$, and if the coefficient vectors are calculated each time cycle (i.e. $t = 0$), then the equation simplifies to:

$$\vec{a}_T(t) = \vec{c}_0 \quad (2)$$

Given the equations of motion for a uniform gravity field, and solving for $\vec{c}_0$ and plugging into Equation (2), the desired acceleration due to thrust at any given time is:

$$\vec{a}_T = \frac{6}{t_{go}^2}(\vec{r}_0 - \vec{r}_f - \vec{v}_0 t_{go}) + \frac{2}{t_{go}}(\vec{v}_0 - \vec{v}_f) \quad (3)$$

A more in-depth derivation of this specific guidance law can be found in previous literature⁵. The actual commanded acceleration in this example must also take into account gravitational effects that are not included in Equation (3). Therefore, the *commanded* acceleration becomes:

$$\vec{a}_c = \vec{a}_T - \vec{g} = \frac{6}{t_{go}^2}(\vec{r}_0 - \vec{r}_f - \vec{v}_0 t_{go}) + \frac{2}{t_{go}}(\vec{v}_0 - \vec{v}_f) - \vec{g} \quad (4)$$

The solution to this problem is synonymous with the solution to an optimality problem under the performance index of minimum control effort (acceleration)⁵:

$$J = \frac{1}{2} \int_{t_0}^{t_f} \vec{a}_c^T \vec{a}_c d\tau \quad (5)$$

Subject to the following system dynamics of a uniform gravity field:

$$\dot{\vec{r}} = \vec{v} \quad (6)$$

$$\dot{\vec{v}} = \vec{a}_c + \vec{g} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (7)$$

It is important to note that this is the solution to the minimum control problem, and not the solution to the minimum-fuel problem. However, Equation (4) provides for a near-optimal solution in fuel usage², which is attractive for its simplicity.

A linear acceleration profile as in Equation (4) above does not allow for a constraint on the final acceleration, which means that the final pointing direction of the vehicle was unconstrained. Apollo guidance engineers saw this as a setback¹, because final attitude constraints were advantageous for crewed missions with visibility requirements. To compensate for this, a new acceleration profile that is quadratic in each Cartesian axis over time was introduced:

$$\vec{a}_T = \vec{c}_0 + \vec{c}_1 t + \vec{c}_2 t^2 \quad (8)$$

By solving this equation for the coefficient vectors, plugging them back in to Equation (8), and assuming instantaneous calculation of the coefficients (i.e., $t = 0$), the following solution is obtained, where $\vec{a}_{T_f}$ is the desired final acceleration due to thrust:

$$\vec{a}_T = \frac{12}{t_{go}^2}(\vec{r}_0 - \vec{r}_f - \vec{v}_0 t_{go}) - \frac{6}{t_{go}}(\vec{v}_0 - \vec{v}_f) + \vec{a}_{T_f} \quad (9)$$

If $\vec{a}_{T_f} = -\kappa \vec{g}$, where $\kappa \geq 1.0$, this allows for a constraint to have the vehicle completely vertically aligned with the gravity vector at cutoff. Similar to Equation (4), the actual commanded acceleration must take gravity into consideration. Unlike the linear acceleration law, this quadratic acceleration law (referred to as ‘‘Apollo guidance’’ in this paper) does not have a direct optimal control theory analog. It has been shown that Apollo guidance slightly underperforms in payload mass carried to the surface compared to the linear guidance law, but Apollo guidance benefits from the extra final state constraint¹.

In a recent paper⁶, it was shown that a tunable guidance law could be introduced that takes on the following form:

$$\begin{aligned} \vec{a}_T(t) = \frac{2}{t_{go}} \left(1 - \frac{1}{3} k_r \right) [\vec{v}_f - \vec{v}(t)] + \frac{k_r}{t_{go}^2} [\vec{r}_f - \vec{r}(t) - \vec{v}(t)t_{go}] + \dots \\ \left(\frac{1}{6} k_r - 1 \right) \vec{a}_{T_f} + \left(\frac{1}{6} k_r - 2 \right) \vec{g} \end{aligned} \quad (10)$$

It is noted that in the equation above, the compensation for gravity is naturally handled. If the tunable trajectory gain k_r is set to 6, the equation will reduce to the linear acceleration law of Equation (4). When k_r is set to 12, it reduces to the Apollo guidance of Equation (9). Therefore, this trajectory shaping gain allows for the tradeoff of favorable end-state pointing conditions with a near fuel-optimal powered descent.

Other more complex algorithms have been studied that led to fuel-optimal solutions with roots in optimal control theory. One such guidance law, known as Universal Powered Guidance (UPG)², allows for the construction of several different performance indices, which allows, for example, a weighting of importance between final touchdown state and fuel used. The solution to the commanded thrust vector is found by numerical iteration on nonlinear algebraic equations, by using either a Newton-Raphson scheme or otherwise.

Time-to-Go Computation

The calculation of the time remaining in the powered descent burn, or time-to-go (written as t_{go} through this paper), can be handled in a wide variety of ways. One method is described in previous literature⁵, and is characterized by analytically solving the roots of the following quartic equation:

$$t_{go}^4 + at_{go}^2 + bt_{go} + c = 0 \quad (11)$$

In Equation (11), a , b , and c are all scalar coefficients that are dependent on position, velocity, gravity, and a term Γ , which is a gain to control how time-optimal or fuel-optimal the solution to the equation will be. This quartic equation is derived from the transversality condition of the Euler-Lagrange equations for the minimum-control performance index realized in a uniform gravity field. It has been shown that this solution is also stable for and compatible for use with Apollo guidance. Typically, once t_{go} is initially computed using the quartic equation above, it is not calculated again and is simply stepped down through time each guidance cycle. This is computationally efficient because the calculation for t_{go} involves a costly fourth-order root-finding problem.

Other solutions for estimating a fuel-optimal time-to-go have been shown in literature^{1,2,4}. It has been shown that the assumption of a constant-acceleration gravity turn problem is a good estimate for t_{go} . In UPG, t_{go} is provided through iteration of the problem². Another solution for handling t_{go} is to make it a loadable parameter to the vehicle. In this case, t_{go} is typically found by running simulations before flight, and an optimal t_{go} is found by trial and error through simulations¹. Although this method is computationally simple (due to the lack of selection of real vs. imaginary roots as in the quartic equation above, and no iterations as in UPG), stability of this t_{go} value for dispersed starting conditions must be thoroughly vetted in simulations before flight.

Ignition Logic

After the time-to-go has parameter has been calculated, the engine burn initiation time must also be determined. The ignition of powered descent in a lunar landing scenario has typically been handled by the on-board calculation of current downrange distance from a specified landing site. Ignition occurs when this downrange distance decreases to a specified value. The calculation for determining the time to ignition t_{ig} with a fixed desired downrange distance at ignition is analytical and does not require propagation or iteration. Given a current inertial position and velocity vector $\vec{r}_0$ and $\vec{v}_0$, a desired final state $\vec{r}_f$, and a desired initial downrange ignition distance θ_{dr} , the value of t_{ig} can be solved for analytically using Keplerian equations.

There have been cases shown in literature where the initialization point for the powered descent burn can be adjusted based on the UPG solution for a fuel-optimal “soft” descent⁶. This solution is continually monitored until the downrange distance required to complete the soft descent is equal to the current downrange distance. If UPG is implemented in this manner, then t_{go} can also be calculated by adjusting the time solution from the “soft” descent landing in UPG by some “relaxing” factor, to compensate for events such as delayed burns or divert maneuvering for hazard avoidance.

PERFORMANCE

Vehicle Simulation & Optimization

To test these guidance laws, an example vehicle is constructed in simulation space and the vehicle’s configuration as given will be held similar in both the simulation and optimization aspects of this study. This is a notional spacecraft designed to provide a baseline for this analysis, and is similar to the assumptions used in design trades for two-stage lunar landers. Table 1 displays the vehicle configuration, as well as other physical parameters of the system (e.g. the spacecraft’s initial state).

Table 1. Test Vehicle Configuration

Parameter	Value
Initial Orbit: Apoapse Altitude	100.0 km
Initial Orbit: Periapse Altitude	15.0 km
Initial Orbit: True Anomaly	-15.0 deg
Stage 1: Propellant Mass	8,000 kg
Stage 1: Inert Mass	2,000 kg
Stage 1: Maximum Thrust	60,000 N
Stage 1: I_{sp}	450.0 sec
Stage 1: Throttle Range	20% min, 80% max
Stage 1: Minimum Burn Time	50.0 sec
Fixed Coast Time	50.0 sec
Stage 2: Propellant Mass	8,000 kg
Stage 2: Inert Mass	2,000 kg
Stage 2: Maximum Thrust	60,000 N
Stage 2: I_{sp}	450.0 sec
Stage 2: Throttle Range	20% min, 80% max
Target: Altitude	200 m
Target: Vertical Velocity	-15 m/s

Both the simulation testbed and the optimization suite were in-house tools built up for this study. The simulation testbed assumes perfect navigation and perfect controllability of the vehicle, and contains multi-stage logic, thrust-limiting capability, and multi-engine support. The simulation testbed is written in MATLAB and utilizes the Aerospace Toolbox for certain functions. Furthermore, the simulation assumes a spherical lunar gravity model and therefore does not take into account higher-order gravity and perturbation effects.

The optimizer used for the following comparisons is Ananke, a multi-phase Hermite-Simpson collocation optimizer written in Python. While out of the scope of this study, the author plans to release more documentation in the future about the development and the derivation of the optimizer gradients in the near future. There is future work to be done in comparing both the optimizer and the simulation testbed used for this study with other V&V tools that are used in-house at MSFC. However, it was deemed appropriate to develop tools for this specific study that more appropriately match the problem scope.

Optimality Comparisons

First, a baseline trajectory is generated through Ananke, utilizing a minimum-control performance index. The minimum-control index is chosen because, as mentioned in sections before, common planetary landing guidance laws pertain to minimum-control rather than minimum-fuel. For this first case, the linear acceleration law prescribed in Equation (4) is employed for both the first and second stage burns. For both burns, the input guidance target for the guidance law is the final guidance target of the surface. The results are shown in Figure 1.

As seen in Figure 1, the simulated trajectory running linear acceleration guidance lines up well with the optimized trajectory. It should be noted that the stage separation in simulation is triggered when the propellant mass of the first stage is equal to that of the optimized trajectory at the ideal point of separation. The coast phase was estimated to provide enough time for a staging, separation, and jettison event. This is, by nature, an event of variable duration, and is driven by the separation mechanism as well as the ultimate sizes of the vehicle configuration in its entirety. Also, after coast phase, the second-stage time-to-go analytical root finding computation is skipped and the nominal time-to-go is provided directly from the optimized time of flight of the second leg of the trajectory.

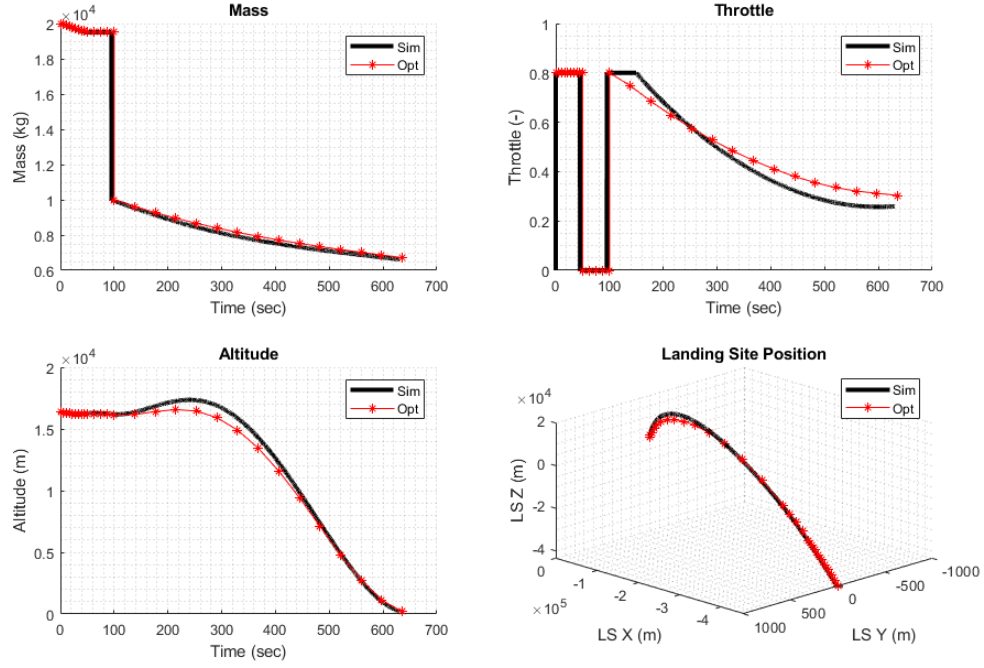


Figure 1. 50-second Coast Time Minimum-Control Trajectory Comparisons

From the results of the optimized trajectory it can be seen that the optimizer had chosen to make the first burn as short as possible given the time constraint, only utilizing a small portion of the available energy in the first stage. And thus, this is a clear side effect of optimizing on the total minimum-control over the entire course of the trajectory when designing around a two-stage vehicle architecture. Where the idea of employing minimum-control may be a decent analog for creating near fuel-optimal trajectories for a single-phase descent trajectory, it no longer applies as well to a multi-phase trajectory, unless properly designated to only apply in a specific phase of flight.

In the two-stage vehicle profile that is used in this study, it is intuitively clear that the performance index should only consider control/fuel in the second stage, allowing the control of the first stage to either float or to be constrained by some other factor. Figure 2 shows an example of a trajectory that was optimized for mass-to-surface, rather than minimum-control; still, the mass threshold and time-to-go triggers from this mass-optimal trajectory are used to prime the guided trajectory in the simulation, which uses minimum-control guidance. An important note on fuel-optimal trajectories is the natural bang-bang throttle tendency, where the vehicle will either tend towards full throttle or minimum throttle boundaries, with no smooth throttle profile in-between. A way to compensate for this effect is to generate optimized trajectories with a lower throttle limit on the vehicle, and then allow the simulated vehicle to advance to its full throttle. As seen, mass-to-surface improves using this methodology, but trajectory design becomes a greater factor rather than simple algorithm testing, which is out of the scope of this paper.

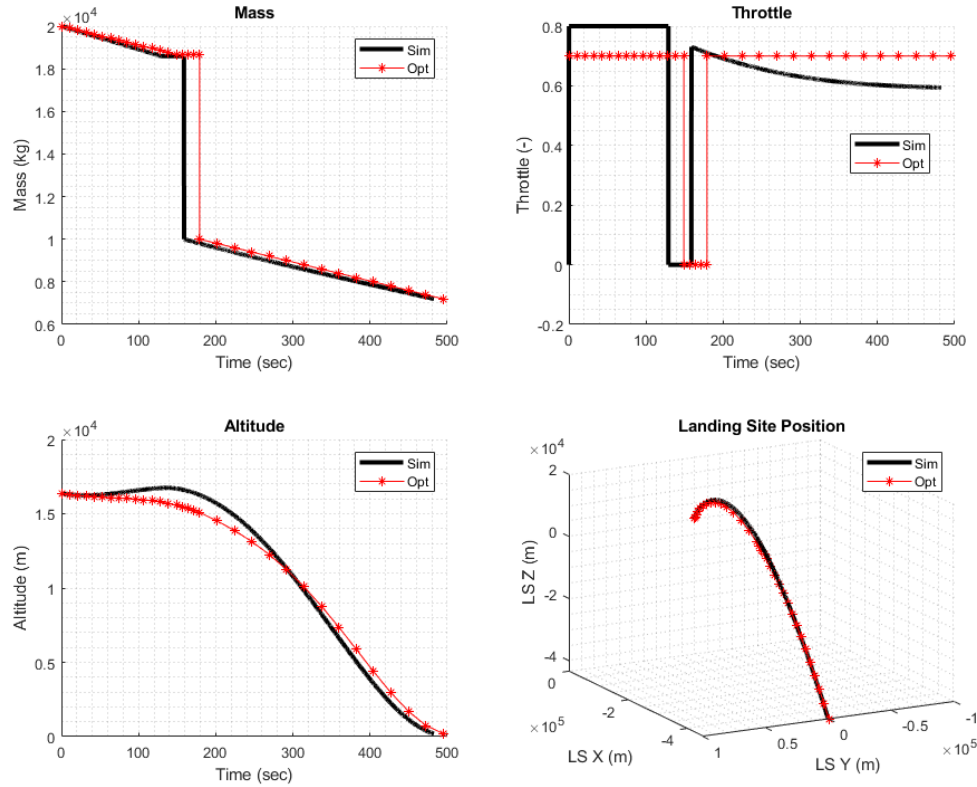


Figure 2. 30-second Coast Time Minimum-Fuel Trajectory Comparisons

Altitude lofting effects that are seen in Figure 1 and Figure 2 can be mitigated by a more careful selection of guidance targets and starting states, but for simplicity of comparison, the starting state is held constant throughout this analysis. Altitude lofting would also disappear if the trajectory was optimized on the premise of a minimum-fuel performance index. However, for this specific vehicle configuration and initial state, the optimizer had found a solution (and the linear acceleration guidance law commanded a solution) that included an altitude lofting effect in order to obtain the minimum-control solution to the trajectory.

It is then safe to say that, in this scenario and vehicle configuration, using the flight-proven planetary descent linear acceleration law for both phases of flight is an *acceptable* solution to this problem, but not necessarily near-optimal if the guidance and separation targets are not carefully selected. It should be noted that this specific scenario contains a somewhat short coast phase. It is expected that deviation between the optimized trajectory and the simulated trajectory would increase as coast time increases, since the guidance laws employed for this study do not compensate for the coast phase. So, holding the initial state and the guidance target constant, the coast time between vehicle stages is varied and the resulting mass to surface is checked. The results are shown in Figure 3 below.

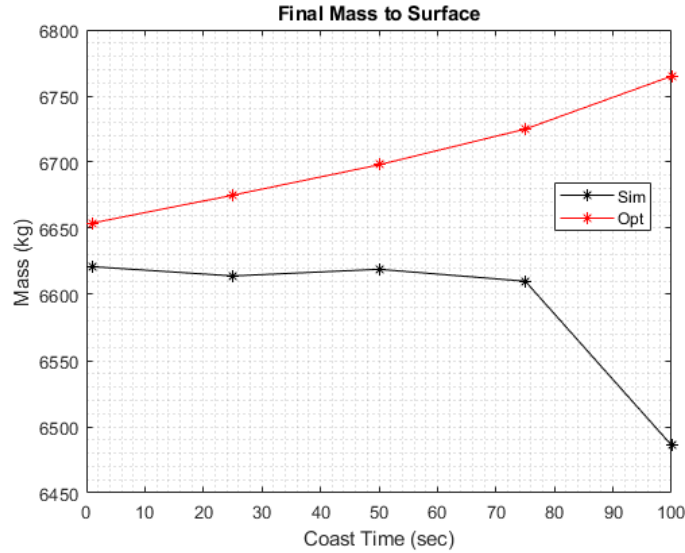


Figure 3. Coast Time Variation Plot

As is clear in Figure 3, longer coast times lead to significantly larger mass discrepancies between the simulated trajectory and the ideal optimal trajectory. Throttle saturation at the upper limit becomes prevalent for a longer period of time during the second stage burn as coast time increases. Intuitively, it would make sense that the guided scenario that matches most closely to a control-optimal trajectory is one with no coast that separates the two stages.

CONCLUSION

This paper summarizes a method for testing near-optimality of two-stage guidance configurations, and discusses tradeoffs between potential steering law candidates in different mission configurations. From the preliminary studies of this paper, it is clear that using heritage, single-stage guidance algorithms on a two-stage vehicle configuration is a promising possibility for planetary landing guidance. The stability and feasibility of the algorithms used are heavily dependent on the specific spacecraft configuration. Engineers planning to employ these algorithms in a two-stage fashion such as the one discussed in this paper should ensure that vehicle thresholds and tolerances are based on hard parameters (such as post-staging coast time) to ensure that their guided trajectories remain close to the nominal trajectories.

Whereas this study had completed the feasibility portion of using these guidance laws, more work is yet to be completed on different guidance trades and their optimality comparisons, and the authors plan to produce future works that include these more in-depth trades. A potential improvement to the work completed in this paper is to expand the minimum-control trajectories presented in this paper to a minimum-fuel exercise in similar format, and employ fuel-optimal guidance algorithms for comparison, such as Universal Powered Guidance. It should be noted that the “bang-bang” nature of a truly fuel-optimal descent trajectory may lead to the more realistic crafting of a burn-coast-burn trajectory, depending on the problem space and initial conditions.

An important item for future work is to consider weighting fuel and control optimality importance for each stage of the vehicle. If mass-to-surface is the overall objective, then it would seem logical for an optimizer to arrive at a trajectory solution that uses a significant portion of propellant from the first stage. A similar study should be conducted to produce data similar to that in Figure 3, but instead of baselining guidance targets and stage separation criteria off a minimum-

control optimized trajectory, the optimized trajectory should be generated with minimum fuel selected as the performance index for the second stage only. A more in-depth trajectory optimization study, specifically, is thus warranted.

Future Vehicle Considerations

A number of changes and differences can be implemented to the mission concept of operations, primarily for the first burn and staging event, for the purposes of future work and further analyses and trades. This study implemented variable throttling during the initial stage 1 burn, but this is not a necessity. Nominally, the objective of an assist burn is simply to bleed off as much energy as possible, which allows the second stage to conduct its burn more efficiently and increases payload mass to the surface. Thus, a simpler method would be to saturate the thrust at 100% throughout the course of the burn while running a guidance routine to control commanded acceleration. This would result in a change to the ignition timing and downrange distance in order to maintain optimality given this new constraint, but it could feasibly be a more energy-efficient route for the overall spacecraft trajectory.

Marking the end of the burn and start of the separation and jettison event is another critical point where alternate parameters can be implemented. In this study, a specific mass quantity initiates first stage engine cutoff. This value is driven by the fact that the amount of remaining propellant must be enough for further deorbit and surface disposal after the staging event (or for boosting back up to a higher orbit in pursuit of reusability objectives). However, mass is only one consideration for this changeover point. The separation event can also be triggered by Δv , which conveniently allows for an accurate assessment of energy usage on the first stage burn, as well as the remaining energy drop required for the second burn. An altitude, or altitude rate, is yet another practical checkpoint for terminating the burn. For most guidance schemes, this could necessitate liberal use of first stage vehicle propellant, but operations based on height above surface level translate to a more tightly controlled trajectory. Lofted trajectories might benefit from altitude rate triggers – as the change in altitude inflects past zero at its peak, staging and jettisoning the first stage would provide more than enough margin for the remainder of the descent to allow the guidance algorithm to correct for any errors accrued during the intermediate coast phase.

Coast time is also an important variable that can impact the effectiveness of the guidance modes both upstream and downstream of the event. This phase must obviously be clocked to provide enough time for the safe staging, separation, and jettison of the two stages before resuming the descent burn with the second stage. The actual time interval would be inherently variable, since everything from spacecraft dimensions to the physical separation modes contributes to this duration. The effect of coast time on payload mass to the surface has been analyzed and discussed in this paper, but the consequences of varying the fundamental system architecture makes for a compelling study.

ACKNOWLEDGEMENTS

The authors would like to thank the following engineers at NASA's Marshall Spaceflight Center for support of this work: Greg Dukeman (EV42), for his wealth of guidance knowledge and information; Evan Anzalone (EV42), for his continued support in guiding the authors in preparing this paper for publication; and Naeem Ahmad (EV42), for his.

REFERENCES

¹ Allan R. Klumpp, "Apollo Lunar-Descent Guidance", Charles Stark Draper Laboratory, 1971

² Ping Lu, "Propellant-Optimal Powered Descent Guidance", *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 4, 2018

³ Ronald R. Sostaric, “Powered Descent Guidance Methods for the Moon and Mars”, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2005

⁴ George W. Cherry, “A General, Explicit, Optimizing Guidance Law for Rocket-Propelled Spaceflight”, *AIAA/ION Astrodynamics Guidance and Control Conference*, August 1964

⁵ Christopher N. D’Souza, “An Optimal Guidance Law for Planetary Landing”, AIAA-97-3709, Charles Stark Draper Laboratory, 1997

⁶ Ping Lu, “Augmented Apollo Powered Descent Guidance”, *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 3, 2019