

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, explicit, path-adaptive descent guidance algorithms must provide the stability and customizability required for a safe and efficient descent to the lunar surface, while also meeting program and vehicle constraints. Several descent algorithms have been flown and tested for single-stage landers through the Apollo and Altair programs, but thus far little analysis has been conducted involving the application of these algorithms to a two-stage descent vehicle. Due to payload mass and fairing constraints of the existing fleet of launch vehicles, multi-stage descent architectures are a unique option for achieving the greatest possible mass to lunar surface. This paper seeks to compare the performance of guidance configurations of 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 baseline to compare the guidance algorithms tested. Time-to-go computational methods and ignition logic routines that may be employed in a lunar environment are also discussed. Preliminary results are presented that show relative performance metrics for a range of different guidance algorithm configurations.

INTRODUCTION

As NASA continues development on the Artemis program and maintains support on other missions that entail cis-lunar and surface capabilities, engineers still face a number of complex problems that must be solved. Long-term lunar missions designed for sustainability will become increasingly more complex from a systems perspective. Flexibility in trajectory design is required in order to enable landings at multiple locations on the lunar surface. In light of these obstacles, the need for autonomous and robust lunar descent guidance is apparent. The guidance laws used on board Apollo and other lunar lander missions have been proven over the last few decades to be near fuel-optimal.¹ While work is in continual development for finding more fuel-optimal solutions without a large increase in complexity,² the simplicity and elegance of algorithms with flight heritage cannot be ignored. Yet throughout the development and application of powered descent algorithms completed in previous years, little work has been done on testing planetary descent guidance laws in a two-stage vehicle configuration.

In two-stage descent vehicle architectures, program and operational constraints typically require some predetermined nonzero coast phase during first stage separation, and also potentially restrictions on liquid propellant usage for the first stage. One benefit of the explicit guidance

* 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

algorithms that have been proven in flight for single-stage descent is their robustness to path dispersions. Therefore, a reasonable hypothesis that this paper seeks to validate is that a two-stage descent vehicle could use these heritage single-stage algorithms in some configuration, such that trajectory complexities (coast phases, throttle constraints, etc.) could be handled autonomously by the path-adaptive nature of the algorithms while retaining near fuel-optimality.

This paper focuses on the near-optimality of guidance algorithms tested in a two-stage vehicle configuration. The Background section is dedicated to heritage planetary guidance laws, and illustrates how these laws would be applied in a two-stage vehicle. This section also discusses potential methods for determining fuel-optimal burn times, as well as first stage ignition logic. The Performance section covers simulation results that employ these guidance configurations for two-stage vehicles, and compares these simulated trajectories to ideal, fuel-optimal descent trajectories. This section also proposes performance metrics to address how far these guidance algorithms may deviate from their respective ideal optimal trajectories. The Conclusion section summarizes the results of this study and discusses future work.

BACKGROUND

Heritage Descent Guidance Laws

The generalized purpose 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 constraints.³ A simple solution that bounds the final position $\vec{r}_f$ and velocity $\vec{v}_f$ can be obtained by assuming that the acceleration on each Cartesian axis takes on a linear profile:⁴

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

One can solve for the coefficient vectors $\vec{c}_0$ and $\vec{c}_1$ analytically with a-priori knowledge of the current state $\vec{r}_0$ and $\vec{v}_0$, the target final state $\vec{r}_f$ and $\vec{v}_f$, and total burn time t_{go} (the computation of which is discussed further in the next subsection). If t in Equation (1) is the elapsed 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. the value of t resets to 0 every minor guidance cycle), 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:

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

An in-depth derivation of this specific guidance law can be found in the referenced literature.⁵

The actual acceleration in this example must also take into account gravitational effects that are not included in the derivation of Equation (3). Therefore, the *commanded* acceleration becomes:

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

Equation (4) can also be derived by taking a uniform gravity field into consideration from the start. In fact, the solution to this problem is the solution to the minimum-control-effort optimal control problem:

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

Equation (5) is 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 not the solution to the minimum-fuel problem, which takes form with a different performance index. However, Equation (4) provides a near-optimal solution in fuel usage,² and has the added benefit of simplicity.

A linear acceleration profile as in Equation (4) does not allow for a constraint on the final acceleration; in other words, the final pointing direction of the vehicle is unconstrained. Apollo guidance engineers viewed this as a setback,¹ because final attitude constraints were advantageous for crewed missions with visibility requirements at lower altitudes. To compensate for this constraint, an amended acceleration profile was introduced that was quadratic in each Cartesian axis over time:

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

By solving this equation for the coefficient vectors, plugging them back into 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 will force the vehicle to be completely vertically aligned with the local 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 (hereafter referred to as ‘‘Apollo guidance’’ in this paper) does not have a direct optimal control theory analog. It has been shown that Apollo guidance underperforms slightly in payload mass carried to the surface compared to the linear guidance law, but Apollo guidance benefits from the extra final state constraint.¹

In a recent paper,⁶ it was shown that a tunable guidance law could be introduced that takes on the following form:

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

Note that in Equation (10), gravity compensation is already taken into account. 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 law 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, 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 a Newton-Raphson scheme or other similar root-finding method.

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 number 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 scalar coefficients that depend on position, velocity, gravity, and a term Γ , which is a gain that controls 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 in a uniform gravity field. It has been shown that this solution is also stable for and compatible for use with Apollo guidance. Typically, after t_{go} is once computed using the quartic equation above, it is not calculated again in this fashion and is simply stepped down through time every guidance cycle. This is computationally efficient since the calculation for t_{go} involves a costly fourth-order root-finding problem.

Other solutions for estimating a fuel-optimal time-to-go are documented 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 problem.² Another solution is to make t_{go} 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 determined by trial and error through these simulations.¹ Although this method is computationally simple (due to the lack of selection of real vs. imaginary roots as in the quartic equation above, and the lack of iterating as in UPG), stability of this t_{go} value for dispersed starting conditions must be thoroughly vetted before flight.

Ignition Logic

After the time-to-go 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 converges to a specified value. The calculation for

determining the time to ignition, t_{ig} , at a fixed desired downrange distance at ignition is analytical and does not require propagation or iteration. Given a current state $[\vec{r}_0, \vec{v}_0]$, 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.

It has been shown in literature that the initialization point for the powered descent burn can be adjusted based on the UPG solution for a fuel-optimal “soft” descent.⁶ 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, which compensates for events such as delayed burns and divert maneuvering for hazard avoidance.

PERFORMANCE

Vehicle Simulation & Optimization

To test these heritage guidance methodologies in a two-stage vehicle configuration, a test vehicle was constructed for use in both the simulation and optimization portions of this study. This is a notional spacecraft that served as a baseline for this analysis, and is similar to the assumptions used in previous design trades for two-stage lunar landers. Table 1 displays the vehicle configuration and other system parameters.

Table 1. Test Vehicle Configuration

Parameter	Value
Initial Orbit: Apolune Altitude	100.0 km
Initial Orbit: Perilune Altitude	15.0 km
Initial Orbit: True Anomaly	-20.0 deg
Stage 1: Propellant Mass	8,000 kg
Stage 1: Inert Mass	10,000 kg
Stage 1: Fuel/Oxidizer	LH ₂ /LOX
Stage 1: Maximum Thrust	100,000 N
Stage 1: I _{sp}	450.0 sec
Stage 1: Throttle Range	20% min, 90% max
Stage 1: Minimum Burn Time	50.0 sec
Coast Time	Variable
Stage 2: Propellant Mass	8,000 kg
Stage 2: Inert Mass	10,000 kg
Stage 2: Fuel/Oxidizer	LH ₂ /LOX
Stage 2: Maximum Thrust	100,000 N
Stage 2: I _{sp}	450.0 sec
Stage 2: Throttle Range	20% min, 90% max
Stage 3 (Crew Capsule): Mass	5,000 kg
Target: Altitude	200.0 m
Target: Vertical Velocity	-15.0 m/s

It should be noted that while the first and second stages may seem identical, this is not quite the case. While they do share similar engine characteristics, the mass distributions of the two stages are slightly different. The second (upper) stage of the vehicle is assumed to contain more inert mass than the first (lower) stage in the form of landing legs and an extra set of docking interfaces (since

this second stage must be docked with both the first stage below and the crew capsule above). The lower stage compensates for that with a larger propellant tank: to arrive at a lunar orbit with a relatively low apolune, a significant expenditure of fuel from the first stage is required to carry the mass of the entire spacecraft from a preceding higher orbit. The propellant mass values in Table 1 are assumed to be at the start of powered descent (after any necessary orbit lowering maneuvers), which is where the scope and relevance of this analysis commences.

Both the simulation testbed and the optimization suite were in-house tools built for this study. The simulation testbed assumed perfect navigation and perfect controllability of the vehicle, and contained multi-stage logic, thrust-limiting capability, and multi-engine support. The simulation testbed was written in MATLAB and utilized the Aerospace Toolbox for certain functions. Furthermore, the simulation assumed a spherical lunar gravity model and therefore did not take into account higher-order gravity and perturbation effects. The optimizer used for the following comparisons is a multi-phase Hermite-Simpson collocation optimizer written in Python, named Ananke. There is still future work to be done in comparing both the optimizer and simulation testbed used for this study with other in-house MSFC tools that have passed verification and validation (V&V) processes. However, it was deemed appropriate to develop tools for this specific study that more appropriately match the problem scope.

Simulation Scenario and Optimality Comparisons

Both the optimizer and the simulation were configured using vehicle configuration data from Table 1. A fixed first stage ignition downrange distance of 545 km was set for this example (this was arbitrarily chosen to be near the optimum downrange ignition distance in order to maximize mass-to-surface for this particular vehicle configuration). For this first study, the linear acceleration law prescribed in Equation (4) was implemented for the first stage burn. Just before commencing the first stage burn, the root-finding problem described in Equation (11) was used to calculate time-to-go of the entire two-stage trajectory, using the landing site guidance targets as the final target state.

After the first stage depleted to a predetermined propellant mass threshold, the staging event was initiated, followed by a fixed coast time of 50 seconds. The clocking value was estimated to provide enough time for a staging, separation, and jettison event. In reality, this parameter will be driven by the specific separation mechanism of the vehicle, as well as the sizes of relevant components in the vehicle architecture to ensure adequate clearance before the second burn.

After completion of the coast phase, the time-to-go analytical root finding equation was once again solved to calculate the second stage burn time (using the same guidance landing targets). Then, for the second stage burn, the tunable Apollo guidance law presented in Equation (10) was used, testing gain values through the acceptable range of $k_r = [6.0, 12.0]$.

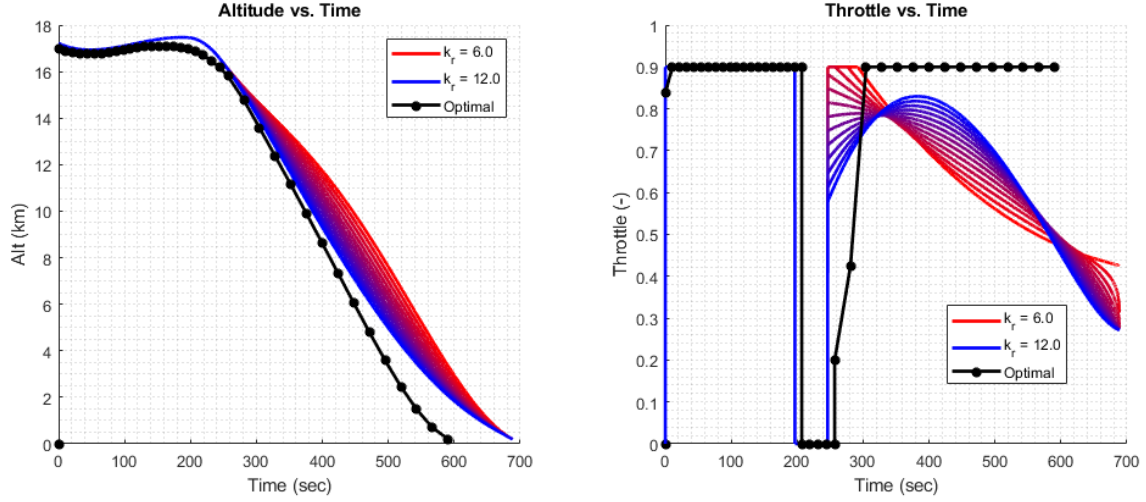


Figure 1. 50-second Coast Time Minimum-Control Trajectory Comparisons

As seen in Figure 1, the simulated trajectory running linear acceleration guidance for the first stage lines up well with the optimized minimum-fuel trajectory profile. It should be noted that the stage separation in both the optimizer and simulation is triggered when the propellant mass of the first stage has depleted to 4,000 kg, which is half of the total first stage propellant mass. Triggering separation before complete first stage depletion was another conscious decision to ensure that the first stage has enough propellant to fly back to a higher parking orbit for refueling and reusability. Operationally, this mass threshold would be dependent on program and vehicle architecture, so the event trigger value of 4,000 kg used in this paper was arbitrary.

When the linear acceleration law provided in Equation (4) was strictly implemented without modifications, some altitude lofting was observed during the first stage burn. This was due to the fact that the linear acceleration law does not take into account planetary curvature over the duration of the burn. To compensate for this, the acceleration target command in Equation (4) was modified in the following manner:

$$\vec{a}_c = \vec{a}_T - \kappa \vec{g} \quad \kappa \rightarrow [0.0, 1.0] \quad (12)$$

By modifying how the algorithm compensated for local gravity, the altitude lofting could be artificially removed from the trajectory profile. For the vehicle architecture described in Table 1, a value of 0.8 for κ was observed to adjust the first stage altitude profile such that it aligned well with the fuel-optimal trajectory.

From Figure 1, it is clear that the fuel-optimal trajectory optimized in Ananke takes on a “bang-bang” throttle control nature, which has been described as fuel-optimal in previous literature.² The reason that two collocation nodes do not appear directly on the lower or upper throttle boundaries is because of collocation point fidelity (however, if more points are used, these collocation points would eventually shift to either the lower or upper bounds). The total time of flight of the fuel-optimal trajectory is shorter than the simulated trajectory because of the “suicide burn” nature of the second stage burn in the fuel-optimal profile. This leads to a higher mass-to-surface (by approximately 480kg) compared to the simulated vehicle performance. This mass delta was expected due to the nature of the guidance laws not being perfectly fuel optimal, as described in previous sections. Figure 1 also shows that there is only a very miniscule mass difference between the lower gain limit and upper gain limit for tunable Apollo guidance (with a mass difference of only 40kg observed in this example).

In summary, selecting a linear acceleration law for first stage guidance and a tunable Apollo guidance law for second stage guidance is an acceptable algorithm suite for this specific vehicle configuration. To obtain a more thorough understanding of the effect of downrange starting distance – as well as the effect of the gain in the tunable Apollo guidance law – a parametric study was run with several different simulated cases, whose results are shown in the figure below.

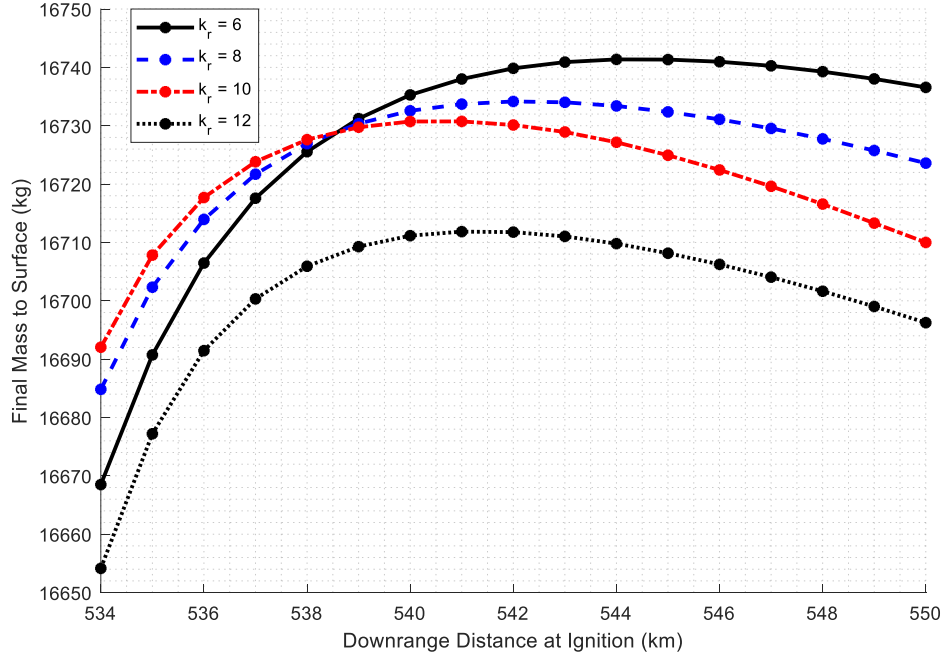


Figure 2. 50-Second Coast Phase, Variable Downrange Ignition & Apollo Gains

As seen in Figure 2, for each tunable Apollo gain tested, there is clearly some optimal downrange distance that corresponds to a peak value in mass-to-surface for the second stage of the vehicle. Narrowing in on this window was done by trial and error through simulation, but a more pragmatic approach could be taken through optimization to determine globally optimal downrange ignition distances. There are two general trends that can be observed: 1) the most optimal mass-to-surface will decrease as the tunable Apollo gain increases, and 2) the most optimal downrange distance for first stage ignition will generally (and very slightly) decrease as the tunable Apollo gain increases, with the notable exception of the case where $k_r = 12.0$. However, the difference from optimal mass-to-surface for all tested tunable gains is no larger than about 40 kg, which should be well within any designed mass margin for a vehicle of this size. Therefore, the benefits of the more constrained final attitudes provided by the quadratic acceleration profile (i.e. $k_r = 12.0$) is marginal at best.

It is then safe to say that, in this scenario and vehicle configuration, using the guidance architecture described above is an acceptable solution for safely guiding the two-stage vehicle to the surface of a planet. In this particular scenario, the coast phase was somewhat short, clocking in at only 50 seconds. Intuitively, it is expected that increases in coast time would only increase any deviation between the optimized trajectory and the simulated trajectories, since the guidance laws in this study generally do not compensate for this coast phase. So, holding the initial state and the guidance target constant, the coast time between vehicle stages was varied and the resulting mass-to-surface observed. A tunable gain of $k_r = 6.0$ was used for the second stage in all cases. The results of this trade are shown in Figure 3 below.

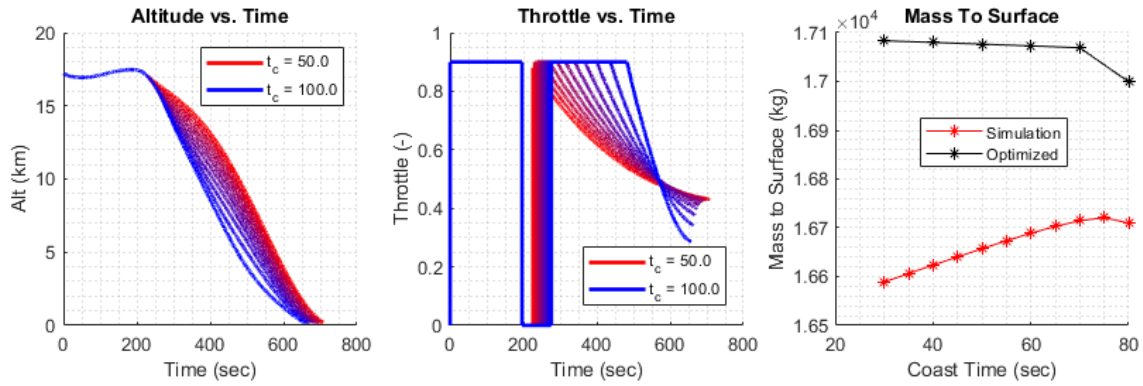


Figure 3. Coast Time Variation Plot

It is clear that longer coast times lead to slowly converging final masses between the simulated vehicle trajectory and the optimized trajectory for that coast time. This is because, as coast time increases to substantially higher values, the second stage guidance is forced to saturate throttle for a longer period of time to hit the specified target state. In terms of optimality, this is more fuel-optimal than a less extreme, more gradual throttle profile. Therefore, the simulation benefited from a total mass-to-surface increase of approximately 110kg over this defined trade space. It can also be seen in Figure 3 that the optimized mass-to-surface is less sensitive to coast time, but perceptibly decreasing. This is because the optimizer is not constrained to fly some predetermined guidance law, so any non-thrust phase will add to a deviation from the truly optimal bang-bang thrust profile. If the coast time was increased much further, the second stage guidance would eventually become unstable and would most likely lead to an improper solution, or even an altitude violation – in other words, the vehicle would careen into the surface before it could slow down enough. Thus, these coast time considerations must be thoroughly simulated, with a set of clear boundaries and thresholds based on the specific vehicle configuration and starting states.

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 presented in this paper, it is clear that using heritage, single-stage guidance algorithms on a two-stage vehicle configuration is a promising option for planetary landing guidance. The stability and feasibility of the algorithms used are heavily dependent on the specific spacecraft configuration. Engineers planning to use these algorithms in a two-stage fashion such as the one discussed in this paper should ensure that vehicle thresholds and tolerances are rooted in robust simulation analysis to ensure that their guided trajectories remain close to their corresponding nominal profiles.

While this study has completed the feasibility portion of using these guidance laws, more work is yet to be completed on different guidance trades and their optimality comparisons. A potential improvement to the work completed in this paper is to 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 naturally lead to the simpler convergence of a burn-coast-burn algorithm, depending on the problem space and initial conditions.

An important item to consider for future work is weighing the fuel importance of the first stage. Setting smarter mass staging thresholds is a crucial step to ensuring that future simulated trajectories align well with broader program requirements and objectives.

Future Vehicle Considerations

Several differences can be implemented into the vehicle architecture described in this paper in order to expand this study. For example, this study made use of variable throttling during the initial first stage burn, but this is not a guaranteed feature of a “braking phase” assist burn. Nominally, the objective of the assist burn is simply to bleed off as much energy as possible, allowing for a fuel-optimal second stage burn, which maximizes payload mass. A study that exercises the algorithms presented in this paper along with a first stage throttle constraint is warranted.

Marking the end of the first stage burn, and start of the separation and jettison event, is another critical point that requires supplemental analysis. In this study, a specific mass threshold initiates first stage engine cutoff. This value is driven by the fact that the amount of remaining propellant must be sufficient 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 parameter for this threshold. The separation event can also be triggered by a Δv threshold, which allows for more precise energy depletion constraints on the first stage burn. An altitude, or altitude rate, is yet another practical threshold parameter for terminating the burn. 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 margin for the guidance algorithm to correct for any errors accumulated during the 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. Mentioned in previous sections, 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 effect of coast time on payload mass to the surface is discussed in this paper, but the consequences of varying the fundamental system architecture makes for a compelling future study.

ACKNOWLEDGEMENTS

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

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