ORION ENTRY MONITOR

Kelly M. Smith*

NASA is scheduled to launch the Orion spacecraft atop the Space Launch System on Exploration Mission 1 in late 2018. When Orion returns from its lunar sortie, it will encounter Earth's atmosphere with speeds in excess of 11 kilometers per second, and Orion will attempt its first precision-guided skip entry. A suite of flight software algorithms collectively called the Entry Monitor has been developed in order to enhance crew situational awareness and enable high levels of onboard autonomy. The Entry Monitor determines the vehicle capability footprint in real-time, provides manual piloting cues, evaluates landing target feasibility, predicts the ballistic instantaneous impact point, and provides intelligent recommendations for alternative landing sites if the primary landing site is not achievable. The primary engineering challenges of the Entry Monitor is in the algorithmic implementation in making a highly reliable, efficient set of algorithms suitable for onboard applications.

INTRODUCTION

An uncrewed Orion Multi-Purpose Crew Vehicle (MPCV) will be launched aboard the Space Launch System (SLS) in 2018 (as of this writing) to orbit the Moon in a distant retrograde orbit (DRO) for a test flight known as Exploration Mission 1 (EM-1). After returning from the Moon, Orion will enter the Earth's atmosphere with speed in excess of 11 kilometers per second, and then attempt its first precision-guided skip entry. Orion will target a water splashdown off the west coast of California near San Diego for a expeditious recovery and return to port.

For EM-1, Orion has requirements that drive development of onboard capabilities to evaluate the health of the current vehicle trajectory, to aid in entry downmode decision-making, and to provide recommendations for alternative landing locations. This collection of capabilities is housed beneath the moniker Entry Monitor.

The Entry Monitor will produce output for consumption by the crew via crew displays (Reference 1) and by flight controllers through telemetry. The resulting information significantly enhances onboard autonomy by providing information that aids in time-critical decision-making during entry. In the initial plunge into the atmosphere in the first entry, the vehicle rapidly depletes its ranging capability through drag. In less than a minute, the vehicle can lose thousands of miles of ranging capability. It is critical that GN&C failures are identified early and that corrective action is applied quickly in order to avoid losing the crew and vehicle to excessive deceleration loads, heating, structural failure, or catastrophic atmospheric skip-out.

The primary functions of the Entry Monitor are:

* Aerospace Engineer, GN&C Autonomous Flight Systems, NASA Johnson Space Center, 2101 NASA Parkway, Houston, TX, 77058

1. **Constant Bank Trajectory Prediction**: Calculates trajectory and intantaneous impact point produced by flying current bank angle held constant until pararchute deployment
2. **Ballistic IIP Prediction**: Calculates trajectory and instantaneous impact point (IIP) produced by initiating a ballistic entry from the current state.
3. **Capability Footprint Calculation**: Defines feasible zone where landing is possible through atmospheric maneuvering.
4. **Guidance Feasibility Indicator**: Determines feasibility of current guidance landing target with respect to current manevuering capability.
5. **Intelligent Landing Site Recommender**: Provides a set of feasible, prioritized landing target recommendations in the event that current landing target is no longer feasible.

The results of each function answers a specific question:

Entry Monitor Function	Question Answered
Constant Bank Trajectory Prediction	Where is the primary guidance system taking me?
Ballistic IIP Prediction	If I switched to a ballistic entry now, where would I land?
Capability Footprint Calculation	How far away can I divert?
Guidance Feasibility Indicator	Can I still reach my primary landing location?
Intelligent Landing Site Recommender	If I can't fly to my primary landing site, where are good landing locations?

FUNCTIONALITY

Numerical Integration & Equations of Motion

Numerical integration of the equations of motion for a non-thrusting entry vehicle is at the heart of the functionality for Constant Bank Trajectory Prediction, Ballistic IIP Prediction, and Capability Footprint Calculation. However, numerical integration is relatively computationally expensive compared to historically available onboard computing resources; this is a major reason why analytical methods have been preferred for operational vehicles. However, these analytical methods are generally approximations with simplifying assumptions (shallow flight path angles, remaining near a reference trajectory, etc) which restrict their extensibility and flexibility without re-tuning. However, the general differential equations of motion governing entry flight do not have a closed form solution. For flexibility across missions and flight phases, a numerical integrator was selected as the mechanism for predicting trajectories under the influence of lift, drag, and gravity.

The Entry Monitor employs a variable-step integrator, the Runge-Kutta 45 (RK45)2, with built-in adaptive step-size control. This design accelerated propagation speed by multiple orders of magnitude without loss of accuracy, compared to a fixed-step RK4 integrator. The RK45 method evaluates the equations of motion with six integration sub-steps in order to develop a state estimate X_4 with error $O(n^4)$ and another state estimate X_5 with error $O(n^5)$. The algorithm adjusts the step-size based on the differences between these two state estimates relative to the error tolerances specified by the user. For smooth, slowly changing dynamics, both state estimates tend to agree with one another, and then the step-size is increased. For rapidly changing dynamics, the state estimates may disagree, and the step-size is reduced until sufficient agreement can be achieved. This mechanism enables the vehicle to gingerly take small steps through flight regimes with rapidly changing dynamics, and it is able to sprint through flight regimes with slowly changing dynamics (exo-atmospheric flight).

The equations of motion are defined in inertial space about a rotating planet. A simple gravity model with the J_2 spherical harmonic is used, because higher order effects become sufficiently small as to be neglected over the relatively short entry flight durations. Lift and drag forces are computed assuming a constant reference area S_{ref}, constant lift-to-drag ratio L/D, constant aerodynamic drag coefficient (C_D), and an altitude-indexed density table $\rho(h)$. The fixed aerodynamic parameters are currently treated as constants primarily due to software simplicity and reduced execution time. So far, the additional aerodynamic modeling fidelity has not yet been shown to be required for this application.

In addition to the translational equations of motion, the bank angle and bank angle rate are also modelled using simplified rotational dynamics. Modeling the bank angle and bank angle rate was required to improve the accuracy with which ballistic entry impact points could be predicted. The current ballistic entry baseline is to perform a spin-up maneuver until a fixed bank angle rate is achieved. Within the Entry Monitor, this spin-up and constant spin rate dynamics are modeled within the equations of motion using a simple phase-plane deadband controller on bank angle error and bank angle rate error. This additional modeling fidelity markedly improved the ballistic impact point prediction accuracy.

The trajectory prediction engine begins at some initial condition specified by the user and continues until a termination condition is satisfied. The primary termination conditions include altitude, speed, propagation arc duration, integration steps used, and catastrophic skip-out. The output of the trajectory prediction engine is a time history of a small set of trajectory parameters and a final terminal state with additional metrics such as maximum altitude after entry interface and peak deceleration load. The design intent is to provide a common trajectory prediction engine that is utilized to predict a wide variety of entry trajectories.

Constant Bank Angle Trajectory Prediction

To enable the crew onboard Orion to monitor the performance of the primary entry guidance algorithm, PredGuid3, the current bank angle command is held constant from the current state until termination conditions are satisfied, typically at drogue parachute deployment altitude. The instantaneous impact point (IIP) will be displayed on a crew display as a marker on a map. This information will enable the crew to oversee the entry guidance system's performance as it flies toward the target. If the IIP is not converging on the target over time, then the crew can become alerted to a potential guidance failure and take corrective action.

Ballistic Instantaneous Impact Point Prediction

An emergency entry mode onboard Orion is the ballistic entry capability. In this mode, Orion will initiate a clockwise bank angle spin at +15°/s. The intent is to direct the lift in all directions equally in time such that the integrated effects of lift become negligible. Thus, the vehicle would behave like a ballistic body only subject to gravity and aerodynamic drag.

As documented in Reference 4, initiating a ballistic entry for lunar returns partway through entry can lead to catastrophic skip-out or overflight of the landing site, resulting in a landmass impact. Entry Monitor predicts the current instantaneous impact point (IIP) for a ballistic entry initiated from the current state. As such, Entry Monitor can report out whether or not a ballistic entry initiated from the current state leads to a successful water landing or not. This information would be useful to a crew in deciding the appropriate time for initiating a survivable ballistic entry.

The predicted ballistic IIP will be displayed to the crew on a crew display as a marker on a map. Additionally, the peak deceleration load would also be displayed as supplemental information.

Capability Footprint Calculation

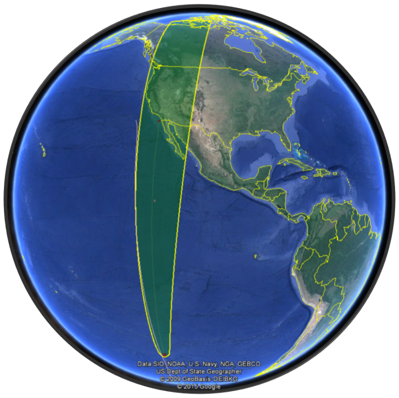
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
US Dept of State Geographer
© 2009 GeoBasis-DE/BKG
© 2015 Google

Figure 1. Notional Orion capability footprint at Entry Interface

To inform the crew about the vehicle's current atmospheric maneuvering capability, the vehicle's approximate capability footprint is computed by the Entry Monitor. The vehicle capability footprint describes the area on the surface of the planet that is physically reachable through atmospheric maneuvering. To compute the capability footprint, several trajectories are propagated with various constant bank angle profiles. The terminal states of all the trajectories are connected in a "connect-the-dots" manner. The resulting set of points defines the bounds of the capability footprint. Any locations outside the bounds of this capability footprint approximation are considered infeasible.

While various methods for rapidly computing the vehicle capability footprint have been documented in the literature, these methods 5 6 require the vehicle to fly near a quasi-equilibrium glide condition and/or at shallow flight path angles. These methods tend to be more applicable towards mid to high L/D vehicles rather than low L/D vehicles like Orion, and they do not readily extend to the problem of skip entry.

A simple footprint algorithm would simply iterate over a range of pre-defined constant bank angle profiles. To better approximate the footprint, either a dense sampling of bank angles between $\in [\pi \quad 0]$ or a judicious selection of bank angles is required. After numerically propagating each trajectory with its bank profile to termination, this simple scheme is sufficient to define the footprint for vehicle states for which skip-entry is not possible.

However, such a simple method does nothing to limit the deceleration loads experienced during entry. To mitigate this, the load relief algorithm from Ref7 was used inline to the trajectory prediction. This algorithm translates peak load constraints into an altitude rate error which can be used to modify the bank angle command to observe a trajectory inequality constraint. When the load relief algorithm is properly tuned for the vehicle, all trajectories observe the peak deceleration limits with only minimal overshoot. This has the effect of marginally lengthening the minimum range trajectory.

When skip-entry is possible (at sufficiently high speeds or high L/D), the vehicle capability footprint lengthens dramatically. It is desired to constrain the capability footprint such that the maximum range solutions do not climb above 600,000 feet altitude. This constraint is applied to honor a constraint within the entry guidance algorithm's numerical search algorithm. The Orion entry guidance algorithm, PredGuid, will reject candidate bank angles whose trajectories climb above 600,000 feet altitude as ”escape” trajectories. For this reason, the Entry Monitor constrains the peak altitude after entry interface (EI) in order to maintain compatibility with PredGuid.

In order to constrain the footprint to trajectories whose post-EI apogee exceeds 600,000 feet, the Entry Monitor performs univariate root-finding to find a constant bank angle profile whose trajectory achieves this apogee limit within some tolerance. This root-finding must be performed for both the left and right sides of the capability footprint to account for differing drag profiles caused by differences in atmospheric-relative velocity based on azimuth differences.

The vehicle apogee post-EI becomes extremely sensitive to bank angle in the region of interest, resembling an exponential function. As a result, local slope-based root-finding methods such as Newton's method tend to not perform well because the convergence is dependent on an excellent initial guess for this type of problem. Large regions of zero or near zero slope exist in the solution space, and Newton's method (and other gradient-based methods) tends to suffer when slope information is of poor quality.

The root-solving algorithm can accommodate a variable number of *a priori* existing data points to form the next guess. These three *a priori* points are generated in the first stage of the algorithm which attempts to provide slope information near the expected solution. These points are generated to “prime” the search algorithm which uses quadratic interpolation to converge on solution. If there only two *a priori* points available, then a bisection (bracketed) or secant search (unbracketed) method is used to produce the next guess. If there is one valid point, then the next guess is slightly offset from the known prior point to establish slope information for the subsequent iteration. This total concept is somewhat similar to the Brent-Dekker method 8, though it appears to enjoy superior convergence rates in initial testing. Throughout the search process, all terminal states are recorded so that non-skip-out trajectories can be included later in defining the footprint vertices.

If convergence can be achieved in less than the allocated number of trajectory propagations, then the surplus (unused) trajectory propagations are re-allocated to the subsequent stage, the Backfill Algorithm.

Backfill Algorithm Following convergence, the footprint algorithm will determine how many “spare” trajectory propagations it has available after completing the initial search priming stage and performing the skip-out search stage. If there are any spare trajectory propagations available, then the algorithm will proceed into its “Backfill” logic. The footprint algorithm is constrained to only perform a fixed number of trajectory propagations due to computational throughput constraints for the onboard processor.

In the Backfill logic, the algorithm attempts to "plug" the largest gaps between adjacent footprint vertices. When flying more than 1/4 of the planet's circumference past the target point, the method for computing downrange relative to the landing site begins to decrease range flown. This decrease in downrange relative to the target for long-range trajectories breaks the monotonicity assumption required for univariate line search. To avoid this issue, time of flight is used as a proxy variable which does not suffer from wrap-around issues. By finding the largest time of flight difference between two adjacent trajectories, the largest gap in footprint definition is identified. The midpoint between the two corresponding bank angles is calculated, and it is used to propagate an entry trajectory. This cycle repeats until all "extra" propagations are consumed in better defining the footprint.

Without this Backfill algorithm, occasionally one side of the capability footprint would move significantly, and this would be rendered as a large transient glitch in the footprint visualization. This scenario could occur when few values of static bank angle scan captured and the apogee search converged in very few iterations. For a polar entry flying North, the long-range trajectory may fly over the North Pole and descend over Asia. The great circle arc connecting the long-range terminal point in Asia to the last non-skip terminal state would cross from southwestern hemisphere all the way to the northeastern hemisphere. This visualization artifact was caused by the sparse definition of one side of the footprint and amplified by the great-circle arc interpolation algorithm for visualization. The Backfill technique significantly decreases the likelihood of such a glitch because it attempts to insert intermediate points to minimize the size of the gaps between adjacent vertices.

Guidance Feasibility Indicator

The vehicle capability footprint boundaries define the edges of a polygon which envelope the feasible landing locations. To determine whether or not the current target pursued by entry guidance is feasible, all that is required is to check whether or not the guidance target point is within the capability footprint polygon. This type of algorithm is well-known as the "point-in-polygon" algorithm, and the Entry Monitor uses the ray-crossing algorithm for its computational efficiency and ease of implementation. If the target is within the footprint boundary, then the target is feasible. If the target is outside the capability footprint, then the target is considered to be not feasible. This information will be displayed to the crew via crew displays.

Intelligent Landing Site Recommendations

In the event that the current guidance target is not achievable, then the Entry Monitor produces recommendations for n suggested landing sites that lie within the capability footprint.

Various methods for producing target recommendations were considered early in the development of the Entry Monitor. An early, discarded method involved simply finding three points centered (laterally) within the footprint at various downrange distances. The problem with this approach is that the Entry Monitor could recommend three landing points on landmass rather than water. Given that the vehicle is structurally designed for water landings and not impact with terrain, recommendations for landmass landings is problematic.

Therefore, the Entry Monitor uses a two-stage approach for producing recommended landing

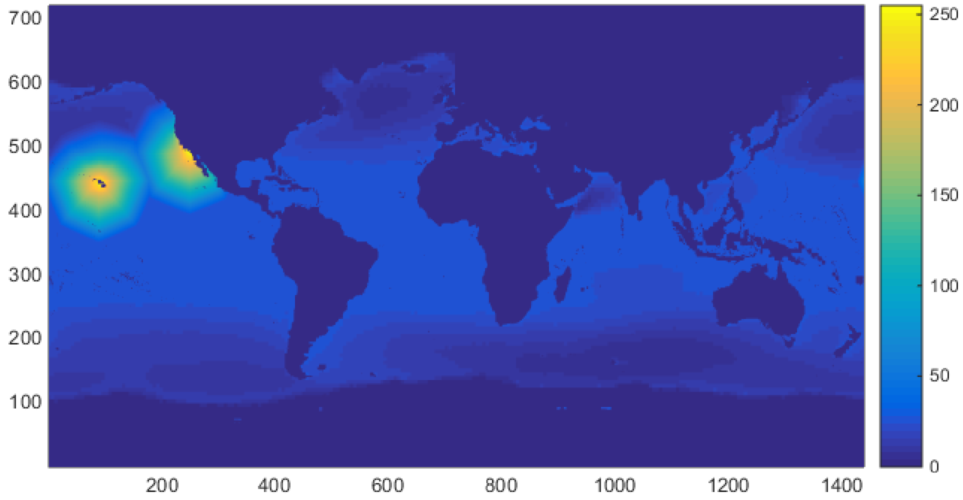

Figure 2. Native Resolution (1/4°per pixel) Notional Global Score Map, comprised of blend of sea conditions averages, proximity to potential recovery locations, and proximity to shipping lanes

sites. The first step is to check to see if any locations from a pre-screened list of landing locations are within the capability footprint. If there are n pre-screened landing locations that the vehicle is able to reach, then these are presented to the crew, and the recommendation task is done. However, in the scenario where less than n pre-screened landing locations are within its capability, then the Entry Monitor will query a global score map to find the best landing locations within its current capability to make up the remainder.

A global map that encoded suitability for Orion landings as an integer between $\in [0 \quad 255]$ was developed. The map allows for relative comparisons between potential landing site candidates. Landing suitability was determined as a function of probability of meeting sea condition constraints (based on C-ERA-40 sea condition climatological database), expected wait time for recovery (dependent on locations and types of recovery assets available), and proximity to global shipping lanes (crewed flights only). For uncrewed flights, the wait time for recovery was computed based on ship sailing distance from the nominal recovery location using Dijkstra's path-finding algorithm9. For crewed flights, the wait time for crew rescue was computed as the shortest travel time for aircraft to depart, fly to the spacecraft, and deploy parajumpers using the A* path planning algorithm10. Each of these individual datasets were mapped from their raw dimensional values (hours until recovery, probability of meeting sea condition constraints) into normalized, non-dimensional values between 0 and 1. Next, each normalized layer is convolved to produce a single map, which is then re-normalized to the $[0 \quad 255]$ domain. This output map is known as the global score map. It is pre-computed and stored onboard for reference by the Entry Monitor.

The raw resolution of the global score map is at 1/4°resolution, which corresponds to a 15×15 nautical mile grid at the Equator. For grid cells at increasing further distance from the Equator, the grid cell will encompass a smaller surface area. At 1/4°resolution, the global score map was represented as a $[720 \times 1440]$ matrix, containing over a million elements.

For storage in onboard flight software, such a large matrix can pose memory challenges. As a result, it was desired to reduce the amount of data storage. Consequently, the 1/4°raw resolution has been down-sampled to a 3°resolution by selecting the worst pixel over each 3° × 3° tile. Consequently, this map can be represented onboard as a much smaller $[60 \times 120]$ matrix, populated with 8-bit unsigned integers, only requiring a total of 7.2 kilobytes, a 99.3% reduction in memory requirements.

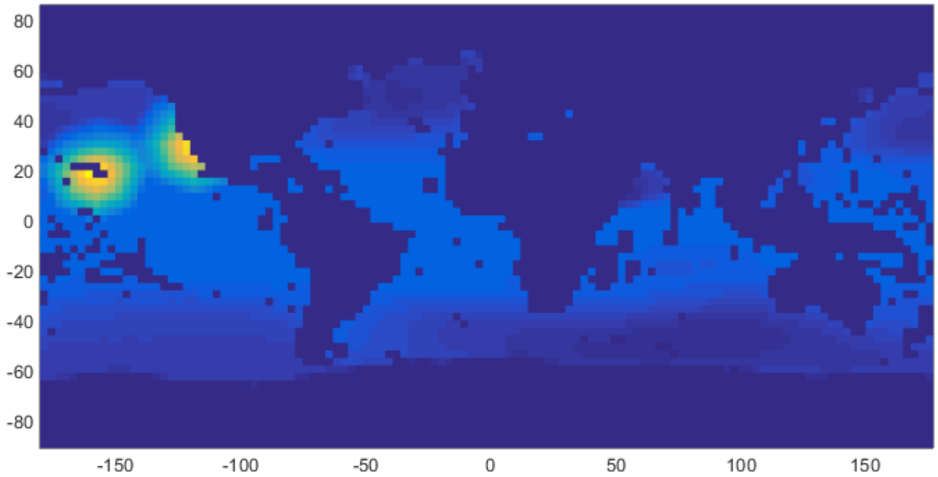

Figure 3. Coarse Resolution (3°per pixel) Notional Global Score Map

If the best pixels within the entire footprint were output to the crew as recommendations for landing sites, it would be a disappointing set of recommendations. Because the underlying score map datasets vary slowly spatially, it is unsurprising then that the best pixels in the score map would be located adjacent to one another. These adjacent recommended locations are, for all intents and purposes, a single landing location recommendation.

In order to enforce a diversity of options, the footprint is sliced into segments based on downrange flown (Figure 4). Within each footprint segment, a grid of latitude-longitude points is generated, and at each point, the value is estimated based on 2-dimensional linear interpolation (bilinear interpolation) of the neighboring pixel values defined in the compressed score map. The point with the highest score within each segment is then compared to all other segments' best pixels. The best pixels from the best n best footprint segments become the n landing site recommendations provided to the crew via crew display, listed in priority order.

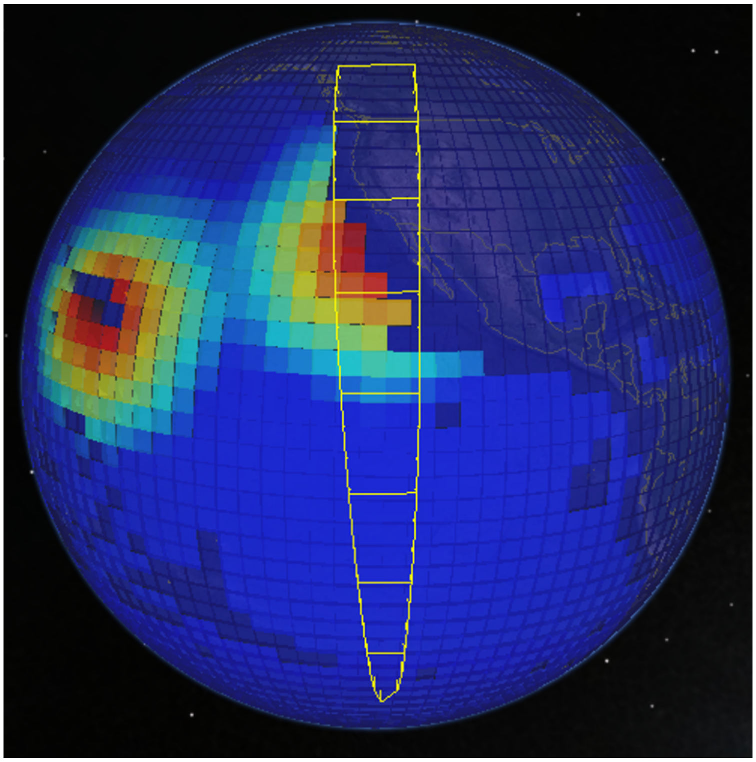

Figure 4. Footprint segments overlayed on score map

SUMMARY

The Entry Monitor has been developed for the Orion spacecraft to enable high levels of onboard autonomy. The information provided by the Entry Monitor enables the future crew to make real-time critical decisions during hypersonic entry when communication with the ground is impeded by ionized flow. The key challenges in the development stem from the design and implementation of robust, reliable, and efficient algorithms. As the saying goes, "In theory, there is no difference between theory and practice. But, in practice, there is." Designing the Entry Monitor algorithms for a skip-entry capable vehicle significantly complicates the problem as compared to a non-skip-capable vehicle.

REFERENCES

[1] D. Baird, M. Grant, B. Kadwa, E. Gillespie, D. Matthews, W. Penny, T. Zack, and B. Bihari, "Orion Entry Display Feeder and Interactions with the Entry Monitor System," *AIAA Guidance, Navigation, and Control Conference*, 2010.

[2] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2 ed., 1992.

[3] Z. R. Putnam, M. D. Neave, and G. H. Barton, "PredGuid Entry Guidance for Orion Return from Low Earth Orbit," *IEEE Paper 2010-1571*, March 2010.

[4] K. M. Smith and J. R. Rea, "Lunar Entry Downmode Options for Orion," *26th AAS / AIAA Space Flight Mechanics Meeting*, February 2016.

[5] A. Saraf, J. A. Leavitt, K. D. Mease, and M. Ferch, "Landing Footprint Computation for Entry Vehicles," *AIAA Guidance, Navigation, and Control Conference*, August 2004.

[6] P. Lu and S. Xue, "Rapid Generation of Accurate Entry Landing Footprints," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 3, 2010, pp. 756–767.

[7] P. Lu, “Entry Guidance: A Unified Method,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 3, 2014, pp. 713–728.

[8] R. P. Brent, *Chapter 4: An Algorithm with Gauranteed Convergence for Finding a Zero of a Function*. Englewood Cliffs, New Jersey: Prentice-Hall, 1973.

[9] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, ch. 3, p. 110. Pearson, 3 ed., 2009.

[10] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, ch. 3, p. 93. Pearson, 3 ed., 2009.