AAS 18-136

POWERED EXPLICIT GUIDANCE MODIFICATIONS & ENHANCEMENTS FOR SPACE LAUNCH SYSTEM BLOCK-1 AND BLOCK-1B VEHICLES

Paul Von der Porten,[*] Naeem Ahmad,[†] Matt Hawkins,[‡]
and Thomas Fill[§]

NASA is currently building the Space Launch System (SLS) Block-1 launch vehicle for the Exploration Mission 1 (EM-1) test flight. NASA is also currently designing the next evolution of SLS, the Block-1B. The Block-1 and Block-1B vehicles will use the Powered Explicit Guidance (PEG) algorithm (of Space Shuttle heritage) for closed loop guidance. To accommodate vehicle capabilities and design for future evolutions of SLS, modifications were made to PEG for Block-1 to handle multi-phase burns, provide PEG updated propulsion information, and react to a core stage engine out. In addition, due to the relatively low thrust-to-weight ratio of the Exploration Upper Stage (EUS) and EUS carrying out Lunar Vicinity and Earth Escape missions, certain enhancements to the Block-1 PEG algorithm are needed to perform Block-1B missions to account for long burn arcs and target translunar and hyperbolic orbits. This paper describes the design and implementation of modifications to the Block-1 PEG algorithm as compared to Space Shuttle. Furthermore, this paper illustrates challenges posed by the Block-1B vehicle and the required PEG enhancements. These improvements make PEG capable for use on the SLS Block-1B vehicle as part of the Guidance, Navigation, and Control (GN&C) System.

INTRODUCTION

NASA is currently building the Space Launch System (SLS) Block-1 launch vehicle for the Exploration Mission 1 (EM-1) test flight. The Block-1 vehicle consists of a Core Stage (CS) powered by four RS-25 engines, two Solid Rocket Boosters, an upper stage known as the interim cryogenic propulsion stage (ICPS), and an Orion payload. The CS will place the ICPS and its Orion payload into a 40.7 km by 1805.7 km (22 Nmi by 975 Nmi) elliptical Earth orbit. Afterwards, the ICPS will be responsible for a perigee raise maneuver and a translunar-injection (TLI) burn to place Orion in its desired initial orbit, followed by a burn to place ICPS on a lunar swingby into a heliocentric disposal orbit. (Reference 1)

[*] Aerospace Engineer, EV42/Guidance, Navigation, and Mission Analysis Branch, NASA Marshall Space Flight Center, Huntsville, AL 35812.

[†] Aerospace Engineer, EV42/Guidance, Navigation, and Mission Analysis Branch, NASA Marshall Space Flight Center, Huntsville, AL 35812.

[‡] PhD, Guidance Engineer, Jacobs Space Exploration Group, 1500 Perimeter Parkway, Huntsville, AL 35806.

[§] Principal Member Technical Staff, Strategic & Space Guidance and Control Group, Charles Stark Draper Laboratory, 555 Technology Square, MS 70, Cambridge, MA 02139.

NASA is also currently designing the next evolution of SLS, the Block-1B. This evolution of the launch vehicle replaces the ICPS with the Exploration Upper Stage (EUS) powered by four RL-10 engines. Planned missions for the Block-1B require EUS to perform ascent completion burns to place the EUS/payload stack into a circular Low Earth Orbit (LEO), apogee raise burns to insert the stack into a highly elliptical orbit, TLI burns (for Lunar Vicinity missions), and Earth Departure Burns (EDBs) (for Earth Escape missions). (References 1-4)

The Marshall Space Flight Center is responsible for the guidance, navigation, and control (GN&C) algorithms, including closed loop guidance, for the Block 1's CS and Block-1B's CS and EUS. In both cases, the closed loop guidance algorithm of choice is a modified version of the Space Shuttle's Powered Explicit Guidance (PEG) (Reference 5). PEG is a predictor-corrector scheme that solves the two point boundary value problem (current state to target state) and produces a steering law for the vehicle's attitude to follow in order to minimize propellant consumption. The SLS PEG algorithm is divided into an outer loop and an inner loop (Reference 5). The inner loop contains the calculations that are commonly found in PEG development papers (Reference 6) with the outer loop serving as a wrapper to the mainline PEG algorithm.

Some other notable distinctions between the Space Shuttle and SLS versions of PEG include 1) Shuttle's PEG was called at 0.5 Hz, while SLS calls PEG at 1 Hz, 2) Shuttle's PEG was checked for convergence after just one pass for each call, while SLS allows up to 10 iterations of the inner loop for each call before PEG is declared unconverged, 3) the mass-flowrate-to-initial-mass time constant (τ) for each PEG phase was computed in PEG's time-to-go computation algorithm for Shuttle, while these are calculated in the outer loop wrapper for SLS, and 4) PEG had 3 algorithmically driven constant thrust and constant acceleration phases for Shuttle, while SLS allows for a data driven approach to program the number of phases flown out by PEG.

This paper covers several enhancements to the PEG inner loop algorithm and its outer loop wrapper needed for both the Block-1 and Block-1B evolutions of SLS. First, the Block-1 PEG modifications since Shuttle are discussed. This is followed by a brief discussion concerning two different flight technique approaches used to target the ascent leg of the Block-1B. This discussion is used to present a challenging guidance problem studied early in the design of Block-1B which required some bullet proofing of the PEG algorithm. Finally, the Block-1-to-Block-1B PEG enhancements are discussed. The Block-1 and Block-1B discussions focus on the explanations and/or developments of new or augmented algorithms.

BLOCK-1 MODIFICATIONS TO PEG SINCE SHUTTLE

In many ways, Block-1's PEG inner loop is a straight adaption of Shuttle's PEG with the ascent desired velocity mode since both vehicle's ascent burn arcs are on the order of ~0.227 rad (~13°). Therefore, all of the Block-1 modifications with respect to the Shuttle version of PEG are in the way PEG is driven via the outer loop wrapper. This section features an overview of the parameterized/data driven approach to programming PEG's constant thrust/acceleration phases, a discussion on how PEG performs some additional lofting to fulfill a Launch Abort System (LAS) jettison requirement, details on the Block-1 EO PEG logic, and finally, a description of SLS's Thrust Factor algorithm that provides PEG with updated propulsion information.

Multi-phase PEG and PEG Phase Manager

The SLS implementation of PEG allows for multiple phases. Phases can be constant thrust or constant acceleration, with either a fixed reference duration or a fixed amount of propellant. New phases are distinguished by a change in propulsion parameters, a new stage, or a mass jettison. For SLS, phases are typically constant thrust. However, just like for Shuttle, a placeholder constant acceleration phase, known as the g-limit phase, is included after the last loaded phase to account for acceleration limiting. There is a maximum acceleration allowed for the vehicle, which is typically reached at the end of flight when the vehicle is the lightest.

PEG's predictor algorithm requires the burn time (t_b) and the mass-flowrate-to-initial-mass time constant (τ) for the current phase and all future phases. PEG's outer loop uses a phase manager to handle this information for multiple phases (with exception of the t_b for the phase prior to the g-limit phase which is still handled in the inner loop's Time-to-Go function). The time remaining for a fixed-time phase is simply the pre-loaded burn time. The amount of time since the start of the burn is deducted for the current phase, with a protection to avoid using a negative burn time if the real burn takes longer than the pre-loaded value. For fixed propellant phases, the mass flow rate is used to determine the length of the burn. Each phase's τ is computed using the ratio of the engines' mass flow rate and the initial mass for the phase. SLS's mass estimator provides the current phase's initial mass, which is then decremented for subsequent phases using each phase's t_b and any projected mass jettisons.

Lofting parameter for Launch Abort System (LAS) Jettison

The Orion payload includes a LAS on top of the stack. After booster separation, the LAS is jettisoned based on a combination of the time since service module (SM) panel jettison, dynamic pressure and calculated freefall time for the SM. Should an SM abort be needed, it is imperative that freefall time is above a required amount of time. LAS jettison represents a significant change in mass; however, PEG is not able to directly account for the complex LAS jettison criteria to predict jettison. Comparison of PEG's performance in a closed loop simulation with a fully optimized trajectory shows that the vehicle performance could be improved by lofting more prior to LAS jettison. Adjusting the amount of lofting allows for optimization of the tradeoff between jettisoning the LAS earlier but incurring larger gravity losses on the one hand, and jettisoning the LAS later but incurring smaller gravity losses on the other.

To induce lofting, the target radius is artificially increased prior to LAS jettison with an altitude bias. Additionally, CS flight is divided into two constant thrust phases. The first phase is a fixed time phase, using an approximate LAS jettison time for the reference time and the mass of the LAS and SM panels (the SM panels have a small enough mass to tolerate mass estimate errors associated with this approach) for the reference mass jettison. After LAS jettison, the altitude bias is removed and the desired radius is used for the radius target. PEG then enters a fixed propellant phase until either the acceleration limit is reached or the insertion targets are reached.

The reference time for the pre-jettison phase is set to the time between SRB separation and LAS jettison from the fully optimized trajectory. The altitude bias for a vehicle configuration is found by varying the bias over a range of values and simply selecting the value that maximizes the orbital insertion mass in the 6-degrees-of-freedom simulation.

Engine-Out Logic

The SLS vehicle needs to be able to respond to failure of a single core engine (i.e. loss of thrust) while ensuring crew safety. The general strategy to handle an engine-out is specific to the vehicle configuration, orbital geometry, and other considerations. A full treatment of engine-out logic is out of the scope of this paper. In general, PEG's response to an engine-out consists of getting updated information about the engines, and selecting a new target if the current targets are no longer feasible.

In the event of an engine-out, PEG will reset a priori engine data (see the next section) to reflect the loss of the engine and continue flying as though the vehicle has one fewer engine. The desire is to complete the mission if viable by inserting into the original target orbit. When this is not possible, alternate targets are used. Alternate mission targets are developed in a 3-DOF simulator for different flight regimes. Finally, 6-DOF Monte Carlo simulations are used to determine times when the vehicle can or cannot reach a given target orbit. For Block 1, PEG uses the sensed inertial velocity to determine thresholds to target different orbits. For example, for an engine out after a certain inertial velocity, the original targets are maintained, while before that threshold, new targets are chosen.

For engine outs during Boost Stage, before PEG takes over, additional analysis is used to favorably change boost stage steering. The goal is to allow boost steering to handle some of the trajectory change, leaving PEG less to do to reach the alternate targets.

The strategy for engine outs for the Block-1B vehicle is still in development. Ultimately PEG will still decide between maintaining nominal targets or loading modified targets.

Thrust Factor

Similar to Shuttle's FT_FACTOR algorithm, Thrust Factor provides PEG with better propulsion information by scaling phase τ's and t_b's for PEG's predictor to correspond to the estimated thrust level. Thrust Factor also provides robustness by implicitly protecting for unknown engine failures such as stuck throttle.

Thrust Factor is a ratio of estimated thrust to predicted thrust. An estimate of thrust comes from the sensed acceleration and estimated mass. The predicted thrust is an input into PEG's outer loop. The raw computation of Thrust Factor is shown in Equation (1)

$$TF_{raw} = \left(\frac{\tilde{m}a_{sens}}{T_p}\right)\left(\frac{\eta_{max}}{\eta}\right) \tag{1}$$

In Equation (1), $\tilde{m}$ is the estimated mass, a_{sens} is the magnitude of acceleration from navigation, T_p is the predicted thrust, η_{max} is the maximum throttle level and η is the current throttle level. In a case of vehicle throttling during flight, the raw estimate will remain constant because the lower sensed thrust is the expected thrust. Due to noise in the computed raw signal, a low pass filter is applied prior to using Thrust Factor to update parameters for PEG. The low pass filter with gain, K_{TF}, and the previous filtered thrust factor signal (TF_{prev}) is provided in Equation (2).

$$TF = K_{TF}TF_{raw} + (1 - K_{TF})TF_{prev} \tag{2}$$

The acceleration signal is computed using states from SLS's navigation which are measured at the Inertial Navigation System (INS) and not the vehicle's center of gravity (c.g.). Due to ignoring the moment arm from the INS to the c.g., the signal lacks compensation for the centripetal acceleration term, which can be significant when body rates are appreciable. Therefore, TF is held constant during the initial period of the CS burn when body rates are relatively high to track PEG's initial solution. In addition, for a known engine-out failure, Thrust Factor utilizes an engine status flag to snap the filtered signal to initial estimate (i.e. 0.75 for a single engine-out).

BLOCK-1B ONE-TARGET VS TWO-TARGET ASCENT GUIDANCE PROBLEMS

The Block-1B ascent profile starts very similar to the Block-1 vehicle. A boost stage trajectory powered by two Solid Rocket Boosters and the CS is carried out from launch to booster separation. This is followed by the completion of the CS burn. Shortly after CS separation, the EUS performs a burn to insert the remaining stack into a circular LEO.

There are essentially two flight technique options in flying out the ascent profile. The first approach, dubbed the “one target” approach, has the CS burn to completion and uses PEG to fly out the entire burn arc from booster separation to the LEO insertion (including accounting for the coast phases from CS shutdown to EUS ignition). The other method, known as the “two target” approach, calls for flight performance reserve to be held on the CS and has PEG perform separate targeted burns for the CS and EUS (i.e. after completion of the CS targeted burn, PEG is reset to fly out the EUS ascent burn). Both approaches were studied by the SLS Guidance and Trajectories Teams, with the two-target approach ultimately baselined in the concept of operations since that approach greatly reduced dispersed trajectory variations at the end of core stage flight, yielding lower required EUS flight performance reserve through the last burn prior to payload separation. Further discussion on this topic is out of scope for this paper, other than to introduce the one-target approach as a challenging guidance problem.

From a PEG perspective, the long burn arc, ~0.873 rad (~50°), of the one-target method presents several convergence issues when applying the Block-1 and Shuttle versions of the algorithm. In order to resolve these issues for the “no-failure” one-target case, several improvements are needed, including scaling the velocity-to-be-gained in the PEG Corrector and limiting the tangent of the thrust angle in PEG’s Linear Tangent Guidance (LTG) steering law. With the additional challenge of handling a single engine out failure (either on CS or EUS), use of Shuttle’s elevation limit (not needed for Block-1) and protection of a sign reversal in the components of the thrust turning rate vector are also needed. Though the one-target approach was abandoned, these enhancements are kept for Block-1B in order to bullet-proof the PEG algorithm. In fact, the elevation limit is required in order to close engine-out two-target scenarios. The aforementioned enhancements are discussed in the next section.

BLOCK-1-TO-BLOCK-1B ENHANCEMENTS

As mentioned earlier, for Block-1B the MSFC-developed SLS GN&C does not just perform the CS portion of the vehicle’s ascent leg, but all upper stage burns (via the EUS) as well. Due to the thrust-to-weight of the EUS, Block-1B experiences much longer burn arcs than Block-1’s CS: ~0.611 rad (~35°) for EUS ascent burn and ~0.785 rad (~45°) for some long in-space burns. The Block-1B engine-out failure cases can yield even longer EUS ascent burn arcs, upwards of ~0.785 rad (~45°). These burn arcs, as well as the arcs observed with the one-target method described in the previous section require PEG to fly well outside of the assumptions built into the original ascent Block-1 PEG algorithm. Furthermore, the need for EUS to perform TLI and hyperbolic in-space burns requires additional desired velocity routines for PEG’s corrector and additional plane control options. This section contains the enhancements needed since the Block-1 version of PEG in order to bullet-proof the code against one-target method stressing guidance problem, allow PEG to perform Block-1B engine-out ascent, perform any potentially desired plane control option, and carry out planned in-space burns.

Safeguards for Constructing Turn Rate Vector

The geometry of the PEG steering law (Reference 6) is illustrated in Figure 1. The solution strategy in the PEG algorithm places $\hat{\boldsymbol{\lambda}}_C$ vector along the unit velocity-to-be-gained ($\boldsymbol{v}_{go}$) from thrust. So $\hat{\boldsymbol{\lambda}}_C$ and burn time satisfy the terminal velocity constraint. All that remains is the solution for the turn-rate vector, $\dot{\boldsymbol{\lambda}}_C$, that satisfies the terminal radius constraint.

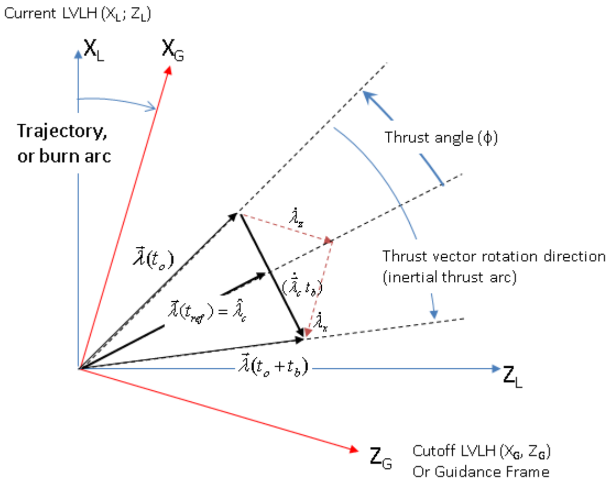
Current LVLH (X_L; Z_L)
X_L
X_G
Trajectory, or burn arc
Thrust angle (ϕ)
$\dot{\lambda}_z$
$\vec{\lambda}(t_o)$
Thrust vector rotation direction (inertial thrust arc)
$(\dot{\vec{\lambda}}_c t_b)$
$\vec{\lambda}(t_{ref}) = \hat{\lambda}_c$
$\dot{\lambda}_x$
$\vec{\lambda}(t_o + t_b)$
Z_L
Z_G
Cutoff LVLH (X_G, Z_G) Or Guidance Frame

Figure 1. PEG LTG Ascent Geometry

The orthogonality between $\hat{\boldsymbol{\lambda}}_C$ and $\dot{\boldsymbol{\lambda}}_C$ is a deliberate assumption to aid in the analytic solution of the desired turning rate required to achieve the insertion position target. It is known to be near optimal in most cases. But for the correct solution to this ascent problem, where there is no downrange constraint, $\dot{\boldsymbol{\lambda}}_C$ would have no downrange component (Reference 7). Consequently, at extremely low thrust-to-weight levels, two basic assumptions start to break down, resulting in sub-optimal profiles. At one extreme, the lower thrust to weight levels require $\hat{\boldsymbol{\lambda}}_C$ to have a larger radial component, relative to the Cutoff LVLH frame, to counter gravitational losses over longer burn times. This results in larger thrust vector turn-rate magnitudes as the turn-rate's impact on insertion altitude is diminished by the more radial $\hat{\boldsymbol{\lambda}}_C$ basis vector. At the other extreme, the longer burn times result in a larger central angle traversed in getting to cutoff. This larger central angle effectively produces more curvature which compromises the flat earth assumption, producing a more radial thrust profile at the same time. Attempts to address these issues theoretically introduces greater complexity, compromising the simplistic design which makes the PEG algorithm appealing. Furthermore, these issues generally arise in only stressing off-nominal scenarios such as early engine failures. When these issues are mitigated by certain safeguards until either the acceleration level increases or until the remaining trajectory arc better approximates a flat earth, the PEG algorithm has shown resiliency in flying out of these difficulties to a satisfactory cutoff state, and in many cases with only marginal losses relative to a fuel-optimal trajectory.

What follows is a description of some of these safeguards developed to address these stress situations. The basic premise is to place limits on the construction of the turn-rate vector in the LTG law to reduce the thrust angle. In most cases, similar but less advanced strategies were used to address these situations in the Space Shuttle Program.

Limiting Tangent of Thrust Angle. Generally, the optimality of the PEG LTG construct, which constrains $\hat{\boldsymbol{\lambda}}_C$ and $\dot{\boldsymbol{\lambda}}_C$ to be orthogonal, degrades when the thrust angle becomes large. A useful initial strategy is to limit the thrust angle, to effectively relax the altitude constraint and reduce the turn-rate. The thrust angle is defined by Equation (3).

$$\tan(\phi) = \kappa\dot{\lambda} \tag{3}$$

Here, κ is the expansion time in burn-elapsed time and defines the reference time t_{ref} in Figure (1) with $t_{ref} = t_0 + \kappa$. Thus, the $\dot{\lambda}$ constraint for a limit on thrust angle is given by Equation (4).

$$\dot{\lambda}_{\phi_L} = \frac{\tan(\phi_L)}{\kappa} \tag{4}$$

Applying this constraint results in an earlier predicted cutoff time that drifts upward while the initial thrust angle is held fixed. Once the turn-rate comes off this limit, the predicted cutoff time stabilizes to a good static value.

Elevation-limit. As either $\hat{\boldsymbol{\lambda}}_C$ becomes more radial, or the burn arc becomes greater, eventually there is a risk that the initial thrust vector could have a component that is retrograde. Engineers analyzing Space Shuttle early two-engine out abort capability during ascent experienced just such situations with the introduction of improved thrust prediction in PEG (Reference 8). Near the start of that two-engine out boundary, the improved thrust prediction drove the thrust direction retrograde relative to the radial defined by the predicted cutoff position vector. The geometry is illustrated in Figure 2. This situation also existed on some of the early engine out scenarios of the Space Shuttle Return-to-Launch-Site abort with insufficient trajectory lofting during the fuel-dissipation phase, resulting in the PEG algorithm producing initial fly-back thrust attitudes that were downrange away from, rather than pointing back towards the launch site.

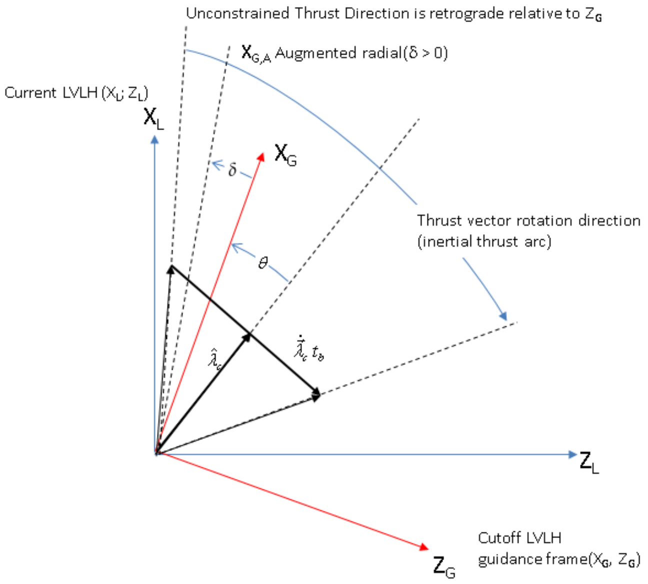
Unconstrained Thrust Direction is retrograde relative to Z_G
$X_{G,A}$ Augmented radial($\delta > 0$)
Current LVLH (X_L; Z_L)
X_L
X_G
δ
θ
Thrust vector rotation direction
(inertial thrust arc)
$\hat{\lambda}_c$
$\dot{\lambda}_c t_b$
Z_L
Z_G
Cutoff LVLH
guidance frame(X_G, Z_G)

Figure 2. Elevation Limit Geometry

The Elevation-limit constraint was conceived to inhibit issuing of retrograde steering commands and significantly improve flight performance in these low thrust-to-weight scenarios where the orthogonality between $\hat{\boldsymbol{\lambda}}_C$ and $\dot{\boldsymbol{\lambda}}_C$ was known to be sub-optimal. The solution to the constraint is obtained geometrically by computing the thrust vector turning rate magnitude ($\dot{\lambda}$) for the current direction of $\hat{\boldsymbol{\lambda}}_C$ which limits the current thrust direction in the linear tangent steering law to the radial defined by the predicted cutoff position vector. The elevation-limit constraint effectively scales down the magnitude of $\dot{\boldsymbol{\lambda}}_C$ to restrain the initial thrust direction from having a retrograde component. It is important to note that this constraint is applied after the construction of $\dot{\boldsymbol{\lambda}}_C$. So it makes no change in the vector directions of the basis vectors $\dot{\boldsymbol{\lambda}}_C$ or $\hat{\boldsymbol{\lambda}}_C$ – only in the magnitude of $\dot{\boldsymbol{\lambda}}_C$.

For SLS, the elevation limit was not required for the Block 1 vehicle due to ascent being executed entirely with the CS over a relatively small burn arc. For Block-1B, the presence of the EUS exposed guidance to more severe low thrust to weight conditions and retrograde commanded attitudes during engine-out failures. Additionally the elevation limit constraint has been augmented to enable specifying a declination angle, δ, measured positive up-range from the guidance cutoff radial, to provide compensation for finite burn arc effects, as shown in Figure 2. The unconstrained thrust angle in the LTG law is given by Equation (3). Figure 2 shows it exceeding both the elevation limits imposed by either X_G or $X_{G,A}$. The constrained thrust angle relative to the augmented radial is given by Equation (5).

$$\tan(\theta + \delta) = \kappa\dot{\lambda}_{EL} \tag{5}$$

Equation (5) defines $\dot{\lambda}_{EL}$ as the turn rate limit magnitude imposed by the elevation limit. Its solution is evaluated in Equation (6).

$$\dot{\lambda}_{EL} = [\tan(\theta + \delta)]/\kappa = \left[\sqrt{\frac{1}{\cos^2(\theta+\delta)}} - 1\right]/\kappa \tag{6}$$

Sign Reversal of Thrust Turning Rate Vector. Figure 1 illustrates the typical LTG geometry during ascent. $\hat{\boldsymbol{\lambda}}_C$ has a low elevation angle, and the turn-rate vector, $\dot{\boldsymbol{\lambda}}_C$ has a negative vertical (x) component and a positive downrange (z) component in the guidance (cutoff) frame. The vertical turn-rate component satisfies the altitude constraint at cutoff and the downrange component is solved to satisfy the orthogonality between $\hat{\boldsymbol{\lambda}}_C$ and $\dot{\boldsymbol{\lambda}}_C$. The orthogonality is satisfied by applying the first order expression for the position to be gained from thrust (Reference 6).

$$\boldsymbol{R}_{go} = S\hat{\boldsymbol{\lambda}}_C + (Q - S\kappa)\dot{\boldsymbol{\lambda}}_C \tag{7}$$

With the components R_{gox} and R_{goy} computed to satisfy the radial and out-of-plane positions at cutoff, respectively, dotting Equation (7) with $\hat{\boldsymbol{\lambda}}_C$ and enforcing orthogonality between $\hat{\boldsymbol{\lambda}}_C$ & $\dot{\boldsymbol{\lambda}}_C$ yields Equation (8).

$$R_{goz} = \left[S - \left(R_{gox}\lambda_{cx} + R_{goy}\lambda_{cy}\right)\right]/\lambda_{cz} \tag{8}$$

With R_{goz} determined, the desired turn rate is given by Equation (9).

$$\dot{\boldsymbol{\lambda}}_C = \left(\boldsymbol{R}_{go} - S\hat{\boldsymbol{\lambda}}_C\right)/(Q - S\kappa) \tag{9}$$

Generally, if the pitch of $\hat{\boldsymbol{\lambda}}_C$ increases to where it goes retrograde relative to $\boldsymbol{X}_G$ (or the burn arc increases to where $\hat{\boldsymbol{\lambda}}_C$ becomes retrograde relative to $\boldsymbol{X}_G$), the construction of $\dot{\boldsymbol{\lambda}}_C$ to satisfy orthogonality results in a reversal of the $\dot{\boldsymbol{\lambda}}_C$. This results from Equations (7) and (8) above - with everything being relatively constant from one-cycle to the next - if $\lambda_{c,z}$ changes from a positive to negative component. In that case, R_{goz} changes sign in Equation (8), resulting in $\dot{\lambda}_{c,z}$ changing sign in Equation (9). This undesirable result is shown going from left-to-right in Figure 3.

The reversal is avoided by defining an augmented guidance frame in the turning rate computations that represents the guidance frame rotated by the amount the pitch of $\hat{\boldsymbol{\lambda}}_C$ relative to $\boldsymbol{Z}_G$ exceeds some specified maximum angle θ_{max} . The augmented frame is defined as expressed in Equation (10).

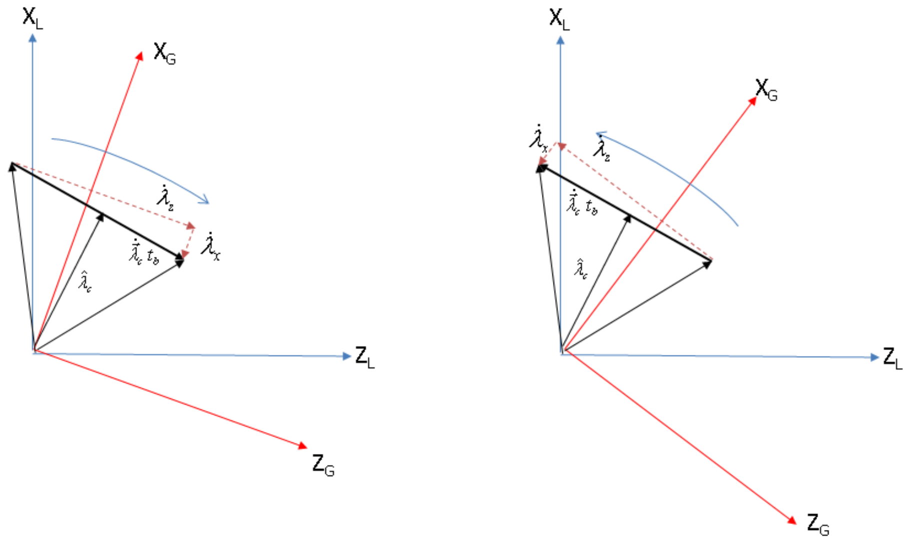

Figure 3. Turn-Rate Reversal Geometry

$$\theta_{aug} = \cos^{-1}\left(\hat{\boldsymbol{\lambda}}_C \cdot \hat{\boldsymbol{Z}}_G\right) \tag{10a}$$

$$\delta = \begin{cases} 0, \; \hat{\boldsymbol{\lambda}}_C \cdot \hat{\boldsymbol{Z}}_G \geq \cos(\theta_{max}) \\ \theta_{aug} - \theta_{max}, \; otherwise \end{cases} \tag{10b}$$

$$\begin{bmatrix} \hat{\boldsymbol{X}}_T \\ \hat{\boldsymbol{Z}}_T \end{bmatrix} = \begin{bmatrix} \cos(\delta) & -\sin(\delta) \\ \sin(\delta) & \cos(\delta) \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{X}}_G \\ \hat{\boldsymbol{Z}}_G \end{bmatrix} \tag{10c}$$

The augmented frame is used in the construction of $\dot{\boldsymbol{\lambda}}_C$.

Scaling Identity Jacobian in PEG Corrector

The PEG algorithm uses a predictor/corrector to converge the solution with the velocity-to-be-gained vector ($\boldsymbol{v}_{go}$) as the iteration variable. The correction process corrects $\boldsymbol{v}_{go}$ to null the miss between the predicted cutoff velocity vector ($\boldsymbol{v}_p$) and the desired velocity at cutoff ($\boldsymbol{v}_d$). The correction of $\boldsymbol{v}_{go}$ implicitly equates to an identity relationship with the change in the predicted cutoff velocity using the following set of equations:

$$\boldsymbol{v}_{miss} = \boldsymbol{v}_p - \boldsymbol{v}_d \tag{11a}$$

$$\Delta\boldsymbol{v}_{go} = -\boldsymbol{v}_{miss} \tag{11b}$$

$$\boldsymbol{v}_{go_n} = \boldsymbol{v}_{go_{n-1}} + \Delta\boldsymbol{v}_{go} \tag{11c}$$

Thus the Jacobian is assumed to be

$$\mathbf{J} = \frac{\delta\boldsymbol{v}_{miss}}{\delta\boldsymbol{v}_{go}} = \mathbf{I} \tag{12}$$

For long-arc burns, the identity matrix assumption breaks down, resulting in corrections that oscillate and over-correct from one iteration to the next. Applying a contraction factor, $0 < \alpha < 1$, to the usual PEG ascent correction reduces the magnitude of the $\boldsymbol{v}_{go}$ corrections on each iteration, eliminating the over-corrections and thus accelerating convergence. A factor of $\alpha = 0.5$ was empirically found to provide reliable and quick convergence compared to the usual correction logic. Therefore, rather than attempting to derive analytic or numerical expressions for the Jacobian, a simple approximation, that provides adequate performance, is given by Equation (13).

$$\frac{\delta v_{miss,x}}{\delta v_{go,x}} \approx \frac{\delta v_{miss,y}}{\delta v_{go,y}} \approx \frac{\delta v_{miss,z}}{\delta v_{go,z}} \approx 0.5 \tag{13}$$

This equates to the approximation in Equation (14).

$$\mathbf{J} = \ 0.5\mathbf{I}, \text{ or } \Delta \boldsymbol{v}_{go} = -0.5\boldsymbol{v}_{miss} \tag{14}$$

The convergence results of the challenging one-target case, with and without Equation (14), are shown in Figure 4. The simple approximation is shown to provide good convergence while PEG flies out the entirety of the CS burn, post CS cutoff coast and separation, and EUS burn.

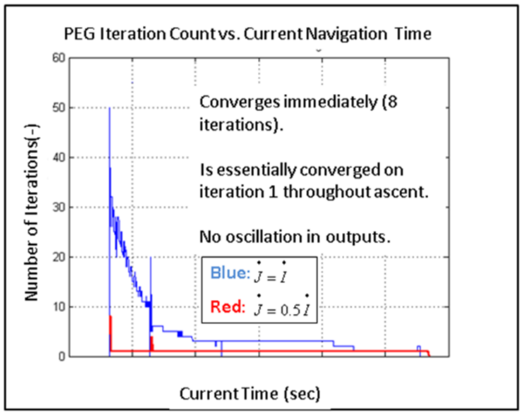

Figure 4. Convergence Performance of One-Target Case

Plane Constraint Strategy

A strategy to unify the plane constraint for both ascent and in-space burns is implemented in PEG for Block-1B. For each type of burn, one of four unique modes can be selected. These modes are PLANE_OFF, RV_NULL, V_NULL, and INTERCEPT.

In the PLANE_OFF mode, either the orbit plane is unconstrained during the entire burn, or the plane constraint is released late in the burn. In this mode the target plane is defined by PEG's predicted burn cutoff position ($\boldsymbol{r}_p$) and velocity ($\boldsymbol{v}_p$) vectors using Equation (15).

$$\hat{\boldsymbol{\imath}}_y = \vec{\boldsymbol{v}}_p \times \vec{\boldsymbol{r}}_p \tag{15}$$

This mode is being utilized in Block-1B's Earth Departure Burn (EDB) and Disposal RCS Correction Maneuver burn.

In the RV_NULL mode, both out-of-plane position and velocity at burn cutoff are controlled. The target plane is specified through an input target set comprising of the right ascension of the ascending node (Ω) and inclination (inc), and is computed using Equation (16).

$$\hat{\boldsymbol{\imath}}_y = [-\sin(\Omega)\sin(inc) \quad \cos(\Omega)\sin(inc) \quad -\sin(inc)]^T \tag{16}$$

Near completion of the burn, plane control is released due to the inability to control position as the time to burn cutoff trends to zero. At the point of release, any turn-rate to control the cutoff position is frozen using the linear tangent steering basis vectors to define a reference inertial turn-rate vector using Equation (17).

$$\boldsymbol{\omega} = \hat{\boldsymbol{\lambda}}_C \times \dot{\boldsymbol{\lambda}}_C \tag{17}$$

After release, the target plane is defined by Equation (15) and $\boldsymbol{\omega}$ is used in updating $\dot{\boldsymbol{\lambda}}_C$ for any changes in $\hat{\boldsymbol{\lambda}}_C$. This mode is utilized during Block-1B's CS and EUS ascent flight phase.

In the V_NULL mode, only the out-of-plane velocity at burn cutoff is controlled. The target cutoff plane is given by Equation (16). However, since there is no out-of-plane position constraint, the out-of-plane component of $\dot{\boldsymbol{\lambda}}_C$ is zero. Currently, the plane constraint for Block-1B's apogee raise (for EM-2) and TLI burns is configured as V_NULL.

Lastly, in the INTERCEPT mode, out of plane position at target intercept is controlled. In this mode, the desired orbit plane is defined by the predicted position vector at burn cutoff and the target intercept position vector. In this case, the unit normal vector $\hat{\boldsymbol{\imath}}_y$ is defined by Equation (18).

$$\hat{\boldsymbol{\imath}}_y = \boldsymbol{r}_T \times \boldsymbol{r}_p \tag{18}$$

The target intercept position vector, $\boldsymbol{r}_T$ is specified by input target parameters. The INTERCEPT option allows the stage freedom in plane control, while placing the payload on a path where it can take out undesirable plane error.

Desired Velocity Routine for Linear Terminal Velocity Constraint

Linear Terminal Velocity Constraint (LTVC) is a Space Shuttle-based routine used to determine the desired velocity at some point $\boldsymbol{r}_0$ in order to intercept some other point $\boldsymbol{r}_1$ later in the conic trajectory (Reference 9). A linear relationship exists between the radial and horizontal components of velocity at $\boldsymbol{r}_1$ as shown as defined by Equation (19)

$$v_{r1} = c_1 + c_2 v_{h1} \tag{19}$$

In the above equation, v_{r1} and v_{h1} are radial and horizontal components, respectively, of the velocity vector at intercept vector $\boldsymbol{r}_1$. c_1 and c_2 are coefficients that define flight path angle. Setting c_1 and c_2 to zero indicates that v_{r1} is zero and constrains the intercept vector to either apogee or perigee of the orbit. Hence, $\boldsymbol{r}_1$ is known and can be either apogee or perigee of the target orbit.

With v_{r1} set to zero, the LTVC routine is implemented to compute the radial and horizontal components of the desired velocity vector at $\boldsymbol{r}_0$. These components are then projected onto the plane $\hat{\boldsymbol{\imath}}_y$ as shown in Equation (20)

$$\boldsymbol{v}_d = v_{0r}\hat{\boldsymbol{\imath}}_x + v_{0h}\hat{\boldsymbol{\imath}}_z \tag{20}$$

Here, $\hat{\boldsymbol{\imath}}_x$ and $\hat{\boldsymbol{\imath}}_z$ are unit vectors that point along the radial and downrange directions, respectively, at burn cutoff. $\boldsymbol{v}_d$ is the desired velocity that is needed for PEG.

This desired velocity routine is used to target apogee raise and TLI burns.

Desired Velocity Routine for Hyperbolic Target

Earth Escape cargo missions such as Europa Clipper require SLS Guidance to target hyperbolic orbits for which specific energy (C_3) is greater than zero. For a given launch date, guidance targets hyperbolic excess velocity vector ($\boldsymbol{v}_e$) which is a function of C_3, declination and right ascension of launch asymptote.

PEG's corrector routine uses the above input and PEG's predicted position vector ($\boldsymbol{r}_p$) to form the required velocity using Equation (21) (Reference 10).

$$\boldsymbol{v} = \frac{1}{2} v_e \left\{ (D+1)\hat{\boldsymbol{v}}_e + (D-1)\hat{\boldsymbol{r}}_p \right\} \tag{21a}$$

$$D = \sqrt{1 + \frac{4\mu}{r_p v_e^2 (1 + \hat{\boldsymbol{r}}_p \cdot \hat{\boldsymbol{v}}_e)}} \tag{21b}$$

Equation (21) is then projected on the desired target plane to form desired velocity as shown in Equation (22)

$$\boldsymbol{v}_d = \boldsymbol{v} - \left(\boldsymbol{v} \cdot \hat{\boldsymbol{\imath}}_y\right)\hat{\boldsymbol{\imath}}_y \tag{22}$$

CONCLUSION

This paper discusses how the Space Shuttle PEG algorithm has been modified to accommodate the initial evolution of the SLS Vehicle, Block-1. Further discussion centers on the challenges to overcome and new capabilities required in PEG in order to carry out more demanding Block-1B missions. Finally, several enhancements to PEG going from Block-1 to Block-1B are covered. These improvements make PEG capable for use on the SLS Block-1B vehicle as part of the Guidance, Navigation, and Control (GN&C) System.

ACKNOWLEDGMENTS

The authors thank the other SLS Guidance Team members who have analyzed and made contributions to PEG during the design of SLS including Dr. Robin Pinson, Chris Hall, Jay Hauglie, Keith Jackson, and Jason Everett. Likewise, the authors thank Dr. Greg Dukeman who serves as a subject matter expert technical advisor to the SLS Guidance Team.

REFERENCES

[1] B. Donahue et al., "Space Launch System: Development Status." AIAA SPACE 2016, Long Beach, CA, 13-16 September 2016.

[2] W. H. Gerstenmaier, NASA Human Exploration and Operations, "Human Exploration & Operations Update on Mission Planning for NAC." NASA Advisory Council, 30 November 2016.

[3] B. Hill, S. Creech, "NASA's Space Launch System: A Revolutionary Capability for Science." NASA Advisory Council, July 2014.

[4]ESD 30000, "Space Launch System (SLS) Mission Planner's Guide.", Initial Baseline, NASA Marshall Space Flight Center, Huntsville, AL, 12 April 2017

[5] R. Pinson, P. Von der Porten, N. Ahmad, NASA Marshall Space Flight Center, S. Bocchino, Bevilacqua Research Corporation, C. Hall, Qualis Corporation, M. Hawkins, Jacobs Technology Inc., "Space Launch System Guidance Description for Exploration Mission 1." NASA Marshall Space Flight Center, Huntsville, AL, 3 August 2016.

[6] R. L. McHenry, T. J. Brand, A. D. Long, B. F. Cockrell, J. R. Thibodeau, "Space Shuttle Ascent Guidance, Navigation, and Control," *The Journal of the Astronautical Sciences*, Vol. 27, January-March 1979.

[7] T.J. Fill, "Introduction to Bi-Linear Tangent Steering For Shuttle Ascent and Aborts," EGB-89-108, Shuttle-89-022, C.S. Draper Laboratory, Cambridge, MA, 1 May 1989.

[8] T. J. Fill, *PEG Prediction Corrections*, Shuttle Memo 10E-80-11, The Charles Stark Draper Laboratory, Inc., Cambridge, MA, March 1980.

[9] S.W. Shepperd, "Linear Terminal Velocity Constraint Conic Required Velocity Routine Revisited", Shuttle Memo No. 10E-77-24, C.S. Draper Laboratory, Cambridge, MA, 27 April 1977.

[10] R. Battin, *An Introduction to the Mathematics and Methods of Astrodynamics*, American Institute of Aerodynamics and Astronautics, Inc., Reston, VA, 1999, p. 555.