AAS 20-635

ANALYSIS AND APPLICATION OF NOVEL AND HERITAGE ACCELERATION LIMITING ALGORITHMS FOR SLS ON ASCENT

Jason Everett,[*] John Wall,[†] and Naeem Ahmad[‡]

National Aeronautics and Space Administration (NASA) is currently building the Space Launch System (SLS) Block-1 launch vehicle, to be used as the crewed heavy-lift vehicle for the Artemis series of missions. The SLS Block-1 guidance subsystem[1] utilizes a nonlinear algorithm derived from Shuttle heritage[2] for limiting the vehicle's induced maximum axial acceleration during ascent flight, known as g-limiting. Even though g-limiting has demonstrated stability and robustness through several design analysis cycles and the extensive Shuttle flight history, there are no available documents that demonstrate that this algorithm has been proven stable through conventional controls stability analysis. This paper highlights the nonlinear nature of g-limit, presents an alternative methodology to employ a linear version of this algorithm, performs assessment of selected linear gains using classical stability analysis and, conducts a comparison of both approaches through Six Degrees-of-Freedom (6-DOF) Monte Carlo (MC) simulations.

INTRODUCTION

The g-limiting algorithm is the means by which Space Launch System (SLS) limits its maximum acceleration primarily for the reason of preventing excessive structural loads and maintaining crew safety. Without g-limiting, SLS will experience higher stress loads than those for which the vehicle is designed, resulting in potential structural failure. Therefore, it is imperative that g-limit functionality is robust and any weakness of this algorithm is known *a priori*. The g-limiting algorithm in use for SLS gained its reputation from its Space Shuttle flight heritage.[2] Moreover, through several recent design analysis cycles, this heritage implementation of Shuttle's g-limiting algorithm has been thoroughly tested and has been vetted for SLS ascent for the Artemis I and Artemis II missions. The algorithm is also robust to certain failure scenarios, including the loss of a single core engine and loss of communication.

This heritage Shuttle g-limiting algorithm is a nonlinear time-varying system that contains a dynamic, multiplicative feedback law. Although this algorithm has been thoroughly

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

[†]Aerospace Engineer, EV41/Control System Design and Analysis Branch, Dynamic Concepts, Inc. (Jacobs ESSCA Group) Huntsville, AL 35806.

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

tested through high-fidelity SLS simulation environments, and has strong flight heritage, there are no available documents that demonstrate that this algorithm has been proven stable through conventional controls stability analysis. While the basis of flight heritage made a strong point in adopting this algorithm as a baseline for SLS, formal stability assessments were not part of its adoption criteria. Heritage aside, formal stability analysis of this nonlinear system, despite being a challenging controls problem, can provide an additional assurance for the robustness of the algorithm as adopted for SLS. Furthermore, stability analysis of the g-limiting feedback control ensures consistency with the treatment of other components of Guidance, Navigation, and Controls (GN&C) subsystems, such as the flight control system, for which stability analysis is an integral requirement.

Earlier attempts of converting the nonlinear model to a linear, time-invariant (LTI) system for stability analysis proved infeasible due to reasons which will be discussed in the upcoming sections. Efforts to linearize the inherently nonlinear heritage g-limiting algorithm led the SLS GN&C team to redefine physical interpretation of the components of the non-linear algorithm, which led to formulation of a fully LTI alternative controller formulation. In the next sections, the existing g-limiting and newly developed linear algorithms are compared and discussed in terms of stability and performance. To verify the design, a Six Degrees-of-Freedom (6-DOF) ascent simulation is used to evaluate performance including propellent usage and orbit insertion targets.

OPERATION OF THE HERITAGE G-LIMITING ALGORITHM

The g-limiting algorithm is a closed loop control system wherein throttle commands are issued in response to sensed acceleration feedback in order to maintain a desired target acceleration. Put plainly, the g-limiting algorithm solves the following problem: *given a desired acceleration, with access to sensed vehicle acceleration, what throttle command would allow the vehicle to achieve the target acceleration*?

The solution to this problem is a multiplicative feedback law that can be derived in a straight forward fashion when the appropriate assumptions are made. First, given a desired acceleration target a_{cmd}, it is clear that the desired engine thrust to be produced T_{cmd} is proportional to the desired acceleration by the current vehicle estimated mass m in the following fashion:

$$T_{cmd} = ma_{cmd} \tag{1}$$

In order to arrive at the desired throttle command, η_{cmd}, with a measurement of sensed acceleration, a_{sensed}, Equation 1 can be expanded into Equation 4, where T_{max} and $T_{current}$ are the maximum and current thrust levels:

$$\eta_{cmd} T_{max} = m a_{cmd} \tag{2}$$

$$\eta_{cmd} T_{max} = \frac{T_{current}}{a_{sensed}} a_{cmd} \tag{3}$$

$$\eta_{cmd} T_{max} = \frac{\eta_{current} T_{max}}{a_{sensed}} a_{cmd} \tag{4}$$

The SLS GN&C architecture does not have innate knowledge or feedback of the exact engine controller output throttle $\eta_{current}$. Therefore, an assumption is made that the current throttle of the engine is sufficiently equivalent to the previously commanded throttle of the engine, η_{prev}. This assumption is valid as long as the engine thrust has had ample time to respond to the previously commanded throttle. This fundamental assumption in the use of the heritage algorithm will be illustrated in the forthcoming sections. After some simplifications, Equation 4 becomes:

$$\eta_{cmd} = \eta_{prev} \frac{a_{cmd}}{a_{sensed}} \tag{5}$$

Equation 5 above represents the nonlinear control law, a *multiplicative* operation with two feedback loops: division by a_{sensed} and product with η_{prev}. The operation of this method assumes the achieved throttle has converged to the previous command and the resulting acceleration has been reached. Using this multiplicative approach, a small offset/overshoot of actual acceleration (when compared to commanded acceleration) becomes apparent. This offset is adjusted for by introducing a PI+I system upstream to this multiplicative feedback law, such that instead of using a fixed a_{cmd}, a new acceleration multiplier, a_{des}, is adjusted to remove any errors between the target and sensed acceleration. This PI+I system, while resembling a classical linear controller, serves only to adjust the set point of the main control law, inherently nonlinear with its multiplicative feedback. Further discussion of the PI+I system and the physical meaning of this a_{des} term, is discussed in the sections that follow.

STABILITY ANALYSIS OF THE HERITAGE G-LIMITING ALGORITHM

As is appropriate for the design and certification of a control system, and consistent with the other control loops employed during the operation of the SLS (eg. Flight Control System (FCS), Thrust Vector Control (TVC)), the authors sought to determine the stability properties of the heritage closed-loop g-limiting algorithm. This section presents an exploration into the stability properties and linearized treatment of the inherently nonlinear heritage algorithm and then proposes a fully linear control system alternative.

Figure 1 shows, in block diagram form, the nonlinear g-limiting algorithm as connected to a representation of the vehicle plant. This simple representation of the system was simulated in MATLAB to evaluate the response and stability characteristics of the acceleration limiting algorithms.

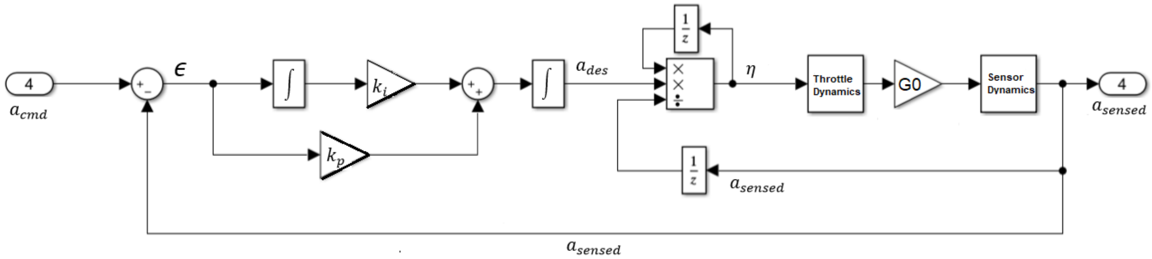

Figure 1: Nonlinear G-Limiting Algorithm Representation

The plant model represents the conversion of throttle command, η to sensed acceleration, a. The plant model includes the simple transfer function dynamics of the engine throttle response and acceleration sensor dynamics. The block, $G0$, represents the actual throttle to truth acceleration relationship. In this paper, the time domain simulation of $G0$ utilizes the relationships in Equations 6 through 9, yielding a partially linearized set of computations which, through simulation, were shown to closely match the full dynamical system using test-correlated tabular thrust, specific impulse I_{sp}, and mass flow rate tables.

$$\dot{m} = \frac{F_0}{I_{sp} g \eta_0} \eta \tag{6}$$

$$F = \frac{F_0}{\eta_0} \eta \tag{7}$$

$$m = m_0 - \dot{m}_0 \eta t \tag{8}$$

$$a = \frac{F}{m} \tag{9}$$

While the mass depletion rate $\dot{m}$ and thrust F shown in Equations 6 through 8 are readily linearized into the simple throttle gain relationships shown above, the resulting mass after $\dot{m}$ integration is then divided into thrust to produce acceleration (Equation 9, yielding a nonlinear relationship between acceleration and throttle.

While the acceleration, via mass flow rate, is indeed a function of the throttle control variable, η, the thrust relationship to throttle dominates as the primary means of acceleration control. As such, the mass integration step and the relatively low gain $\dot{m}$ relationship to throttle, allows the mass parameter to be treated as an input exogenous to the system, yielding a fully linear relationship between throttle and acceleration.

As such, and for stability analysis and interpretation, the resulting plant model relationship between throttle and acceleration can be treated as a time-varying gain, G_0, shown in Equation 10

$$a = G_0 \eta = \left(\frac{F_0}{m(t)} \right) \eta \tag{10}$$

This time-varying gain block of the g-limiting plant is analogous to the launch vehicle TVC control effectiveness, which is commonly addressed in the autopilot application with the use of gain scheduling. The exception to this analogy is that unlike the time-varying autopilot problem, the mass depletion effect additionally produces the very disturbance the g-limiting algorithm is seeking to reject.

Even though the time varying gain and the external disturbance are directly related, the assumption that they are decoupled from the control system throttle command enables classical control system analysis treatment as though the two effects of mass depletion are unrelated. Since mass depletion is the key driver of the g-limiting control problem at hand, one can also interpret the throttle de-coupled mass loss as an analogously increasing disturbance force input. An increasing force would produce the same effect disturbance-wise as a linearly decreasing mass. The analysis of such a ramp force disturbance to a dynamic system is well-known to the control system analysis discipline.

The final observation is that in typical operation the throttle feedback mechanism of g-limiting maintains control of the acceleration, keeping it at constant value. In this scenario, G_0 can be simplified further to a constant gain block, corresponding to the operating point acceleration, a_0, shown in Equation 11. This simplification removes the time-varying consideration of the plant enabling the interpretation of the acceleration problem as a control system whose main goal is to reject a ramp-style disturbance force.

$$a = G_0\eta = a_0\eta = \left(\frac{F}{m}\right)_0 \eta \tag{11}$$

LINEARIZATION OF THE HERITAGE NONLINEAR SYSTEM

While the plant dynamics can be sufficiently linearized for stability analysis of the acceleration control problem at hand, one quickly observes that the g-limiting control algorithm shown in Figure 1 contains both linear blocks (familiar to classical controls practitioners) as well as some nonlinear operations. At first look of the complete nonlinear model (both the PI+I and the multiplicative feedback law), the front end of the control system (the PI+I) looks fairly typical in that an error signal is computed based on the desired acceleration target and the sensed acceleration. The error signal is processed through a typical proportional+integral control scheme, followed by another integrator, a PI+I. The output of the final integrator, a_{des}, then feeds into a nonlinear operation. The operation of the nonlinear portion of the controller, as described earlier, computes an incremental scaling of the previously issued throttle command based on PI+I processed error signal and the previous acceleration.

In order to treat the nonlinear control system stability, the authors sought to interpret the nonlinear incremental scaling block as a linear operation so that a linear system response could be computed and analyzed using classical LTI control techniques. Figure 2 shows a block diagram representation of the fully linearized system, where the nonlinear operations have been replaced with a scaling block, SF. The scaling operation converts the PI+I output into the throttle command.

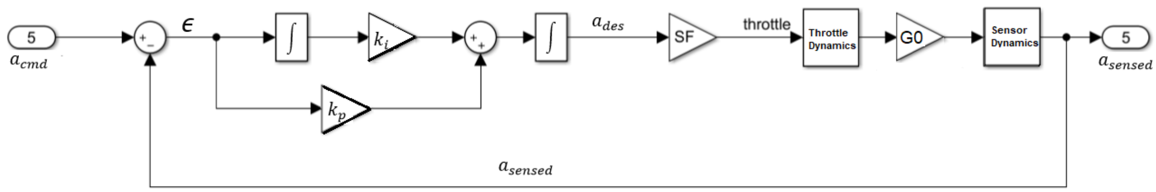

Figure 2: Linear G-Limiting Algorithm Representation

Following the operation of the nonlinear block shown in Figure 1, Equation 12 represents a linear version of the scaling block, wherein the throttle is computed by multiplying the processed acceleration error, a_{des} signal by a "throttle per accel" gain or more specifically, Equation 13 "throttle per thrust over mass".

$$\eta = \text{SF}a_{des} = \frac{\eta_0}{a_0}a_{des} \tag{12}$$

$$\eta = \text{SF}a_{des} = \frac{\eta_0}{\frac{thrust0}{mass0}}a_{des} \tag{13}$$

At this point it is advantageous to describe a subtle but important change of the representation of the PI+I system when comparing the nonlinear multiplicative feedback model as shown in Figure 1 to the strict PI+I linear feedback model shown in Figure 2. In the nonlinear control model, the transformation from an acceleration control signal error to a throttle is performed by the multiplicative feedback gain, and the PI+I is merely used to compensate for any small tracking deviations that may appear due to the nonlinear, time-variant nature of the heritage g-limiting algorithm. Put plainly, the signal that comes out of the PI+I component of the nonlinear system is still in units of acceleration, and the multiplicative feedback gain is the component of the model responsible for converting this acceleration signal into a desired throttle.

In stark contrast, when using the same linear PI+I system but with a scaling block instead of the the multiplicative feedback law, the PI+I system now takes on an entirely different meaning. Whereas in the nonlinear system the PI+I component simply handles a small nonlinear offset that appears, the PI+I component of the linear model is now entirely responsible for handling the conversion from the acceleration error signal to throttle response. In fact, it can be shown that if the scale factor SF is "absorbed" into the k_i and k_p gains, then the PI+I system in the linear model would then relate directly to converting a desired acceleration error signal into a commanded throttle value. In fact, the "PI" component in this case would convert an acceleration error signal into a throttle *rate*, and the extra "+I" component then integrates this throttle rate into a commanded throttle signal.

For this analysis, instead of completely absorbing this scale factor into the k_i and k_p gains of the PI+I system, it is left as a separate scale factor which exists downstream of the PI+I component. This allows for an extra tuning parameter that can be used for stability analysis, analogous to control allocation for vehicle attitude control. Leaving this scale factor separated also means that the PI+I system no longer directly converts the acceleration error signal to a throttle rate and throttle - instead, it converts the acceleration signal into an intermediary control signal, which is then converted into throttle using the SF gain. Stability analysis of this SF gain is analyzed in future sections.

To summarize, as the fully linear version of the g-limiting control system closely resembles the nonlinear operation by visual observation, the PI+I component of the model takes on radically different meaning. Therefore this linear system is not so much a *linearization* of the nonlinear version, but rather an *alternative* linear architecture to the inherently nonlinear multiplicative feedback heritage algorithm. As such, classical linear stability techniques cannot strictly be applied to the nonlinear system. To contrast the heritage nonlinear algorithm and the linear alternative, the next Section displays some time domain sensitivities under various conditions.

Time Domain Response Comparison, Simple Model

A time domain simulation using the simple model described in the previous sections was performed to contrast the two approaches to the g-limiting algorithm: the heritage nonlinear version with multiplicative feedback, and the fully linear version with the scaling alternative of Equation 13. Both simulations initialize the vehicle at the same g-limiting target acceleration and continue to deplete mass, requiring the g-limiting algorithm to reduce throttle to keep the acceleration at the target. The command quantization, present in the production system, has not shown any adverse effects in either algorithm approach but has been disabled in the simulations below to more clearly highlight the contrasting behaviors of the two system responses.

Figure 3 shows a comparison of the achieved acceleration, in red, and the a_{des} control command, in black where both signals are normalized with respect to the target acceleration. Immediately clear from the simulations is while both achieve the target acceleration, a_{des} is quite different between the two. The linear version shows a decreasing signal that follows the expected trend of the throttle response whereas the nonlinear system appears to settle to a fixed value. This is a clear visual representation of the effect described in the previous section, where the PI+I system clearly has different responsibilities across the two models. In the nonlinear model, the PI+I system merely acts as a "tracking improvement" offset that helps maintain a desired acceleration more accurately. Therefore, in the nonlinear model, a_{des} is still an acceleration signal. However, in the linearized model, a_{des} now takes on a different physical meaning in the control model: a_{des} now represents a more familiar linear control law control command which, when multiplied by a scale factor, is converted into a commanded throttle.

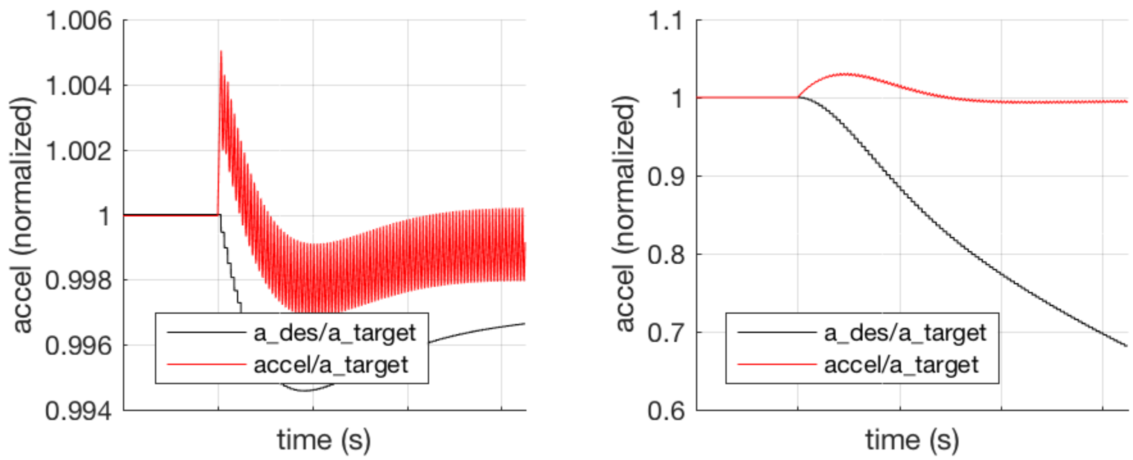

Figure 3: Time Response: Heritage (Left) vs. Linear Alt. (Right)

With these two systems and their representations clearly defined, it is advantageous to perform sensitivity trades on both algorithm update rate and engine throttle response. The nonlinear system, with its multiplicative feedback law using the previously commanded throttle, is sensitive to update rate, whereas the linear system is not sensitive to update rates - as expected. Whereas the baseline algorithm shown in Figure 3 updates every 0.7 seconds, Figure 4 shows a comparison of the nonlinear (left) and linear (right) systems run at 50Hz.

Note that a throttle rate command limit, normally avoided in normal operating conditions, is continually saturated in the nonlinear system when simulated at the higher execution rate and additionally shows divergent behavior. The faster update rate scenario represents a case in which the fundamental assumption of thrust to throttle command convergence has been violated. For such a high update rate, the throttle dynamics are too slow to respond and the current vehicle throttle is nearly or exactly equal to the previously commanded throttle, resulting in poor g-limiting response. The linearized system, no longer dependent on this commanded-to-actual throttle assumption, is free from this update rate sensitivity. In typical SLS operation, a 0.7 Hz update rate is selected for the heritage algorithm to ensure that sufficient time has passed for the actual physical engine throttle response to settle onto the previously commanded throttle.

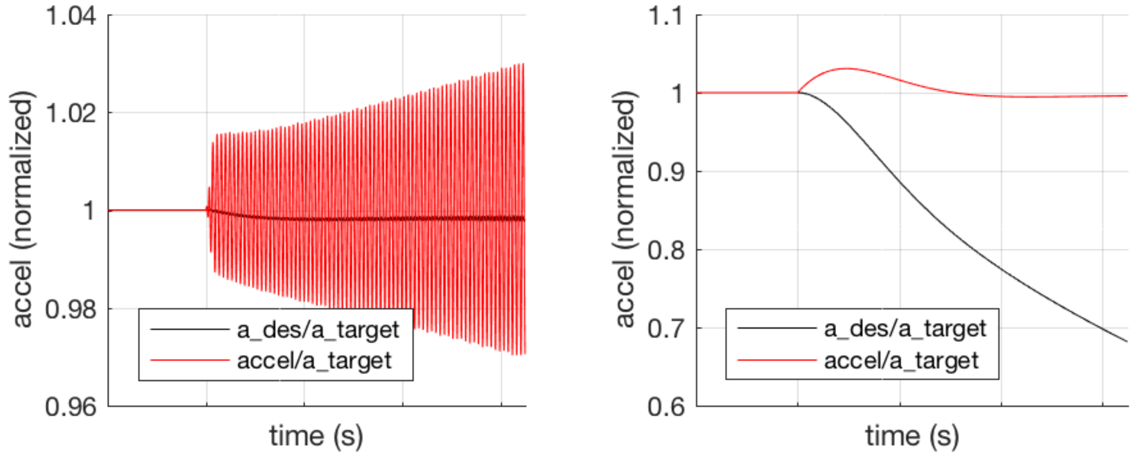

Figure 4: Time Response at 50Hz: Heritage (Left) vs. Linear Alt. (Right)

With the commanded-to-actual throttle assumption being a key requirement for application of the heritage algorithm, it is then intuitive that an increase in execution rate would lead to similar instabilities as a decreased engine throttle controller bandwidth. And, as expected, at the original 0.7 Hz update rate, the nonlinear system appears to be more sensitive to a reduced response of the throttle dynamics. Figure 5 shows a comparison where the nonlinear system (left) exhibits unstable oscillations in the presence of throttle dynamics reduced to approximately one tenth of the original bandwidth, whereas the linear system is nearly unaffected by the reduced responsiveness. This represents another example in which the throttle response time assumption for the use of the heritage algorithm has been violated. As the previously commanded throttle significantly deviates from the actual engine throttle, the stability of the nonlinear g-limiting algorithm is no longer maintained.

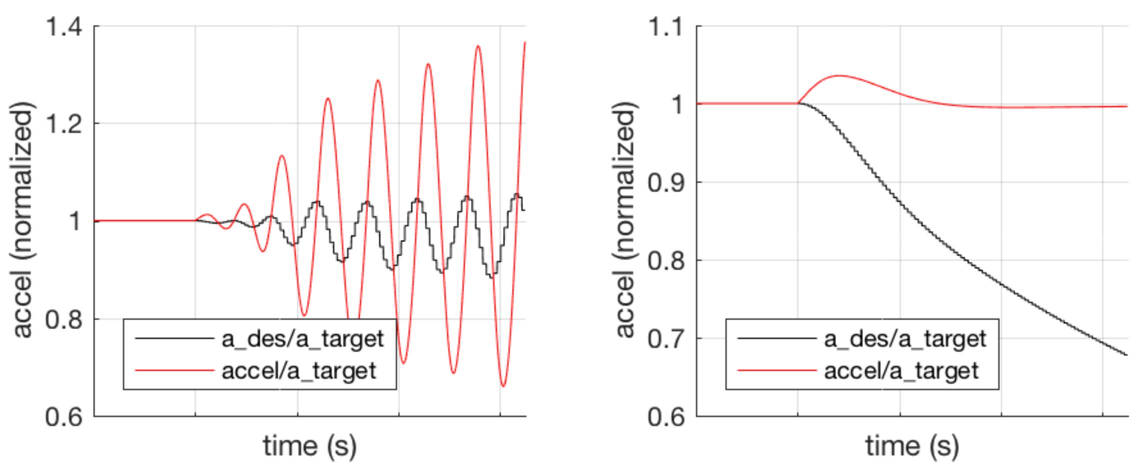

Figure 5: Time Response with 1/10 throttle bandwidth: Heritage (Left) vs. Linear Alt. (Right)

Described in the sections above, the responsibility of the PI+I component in the nonlinear system is to simply improve the tracking of the desired acceleration signal that gets fed into the multiplicative feedback law. In fact, the multiplicative feedback law itself (without the introduction of the upstream PI+I compensation component) is capable of maintaining a targeted acceleration with moderate accuracy. However, without the PI+I compensation, a small offset/overshoot is introduced into the system due to the several nonlinearities that exist in the derivation of the multiplicative feedback law. Figure 6 shows the nonlinear g-limiting algorithm without the PI+I error compensation (right), where $a_{des} = a_{target}$ in comparison to the baseline system (left). The inclusion of the PI+I system into the multiplicative feedback control law serves to produce an offset acceleration signal enabling more accurate convergence onto the true desired acceleration; thus successfully compensating for any offsets introduced by the feedback law or vehicle conditions.

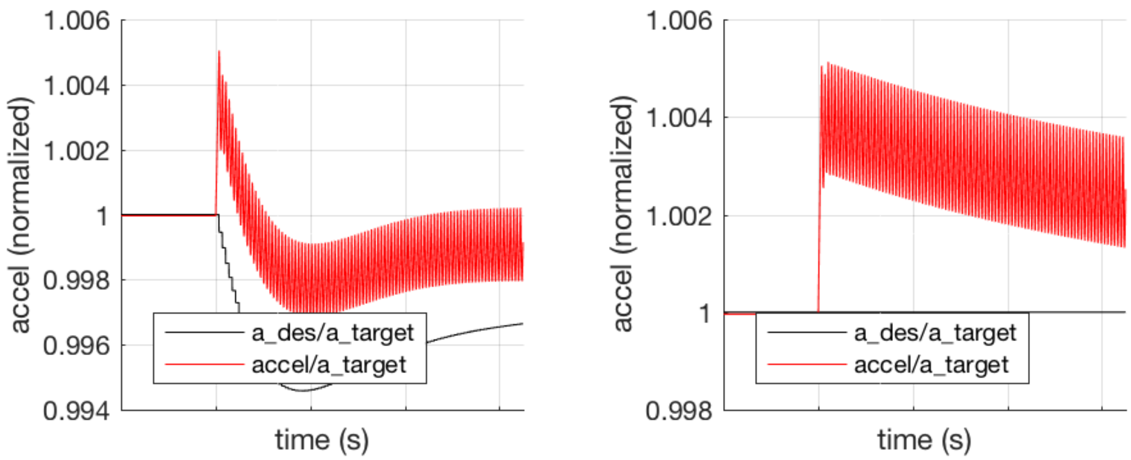

Figure 6: Time Response with 1/10 throttle bandwidth: Heritage (Left) vs. Linear Alt. (Right)

Stability Margin Assessment of Linearized G-Limiting Algorithm

The fully linear g-limiting algorithm, while seeking to achieve the same objectives as the heritage algorithm, carries with it the advantage that linear stability techniques can be readily applied. To assess the linear system stability in terms of classic gain and phase margins, the open loop response was computed for the plant and control in series, with the loop broken at the a_{des} command, shown in Figure 2. Equation 14 shows the expression for the open loop system response. The SF term is the glimit linear scaling computed based on the assumed throttle and acceleration whereas the thrust, F, and mass, m, are the current vehicle operating conditions. The proportional gain, k_p and integral gains k_i were not adjusted from the original nonlinear algorithm in the results of this section but are indeed selectable parameters. The throttle and sensor transfer functions, $G_{throttle}(s)$ and $G_{sensor}(s)$ are second order approximations of the low pass dynamics produced by throttle response and anti-aliasing filters, respectively.

$$OL(s) = \text{SF}\frac{F}{m}G_{throttle}(s)G_{sensor}(s)\frac{k_p + \frac{k_i}{s}}{s} \quad (14)$$

Figure 7 shows the Nichols chart for the open loop response of the system at the various flight times using the minimum (left) and maximum scaling alternatives (right). The minimum scaling alternative corresponds to the condition in which the glimit algorithm begins its operation: maximum acceleration and maximum throttle. The maximum scaling is based on minimum assumed commanded throttle and minimum achieved acceleration. While both conditions are beyond the anticipated operating limits of the glimit algorithm, they are analyzed to show an expected range of variation in loop gain resulting from a mis-match between controller scaling and actual conditions. For a fixed linear scaling of the control algorithm during operations, the variations shown equivalently represent the range of possible operating conditions for the vehicle throttle and acceleration. The ample margin space in the Nichols charts shown in Figure 7 indicates that the linear algorithm is very robust to system uncertainty including mis-matches between the asssumed linear scaling and the actual throttle and acceleration conditions.

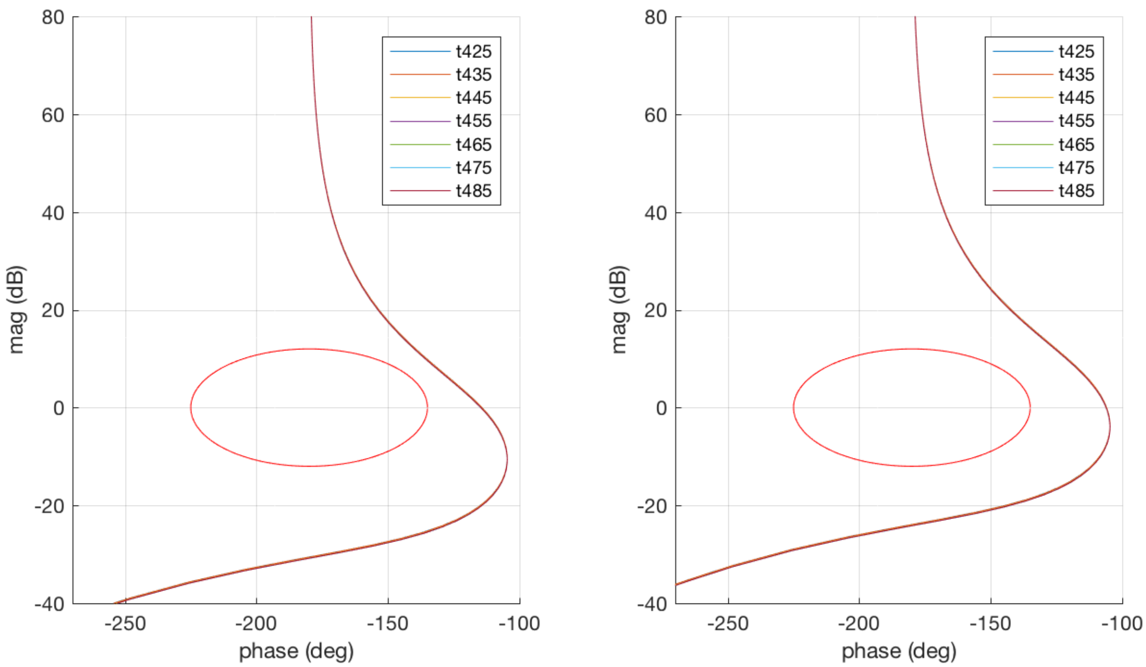

Figure 7: Nichols Response with Min Scaling (Left) and Max Scaling (Right)

Figure 8 shows the resulting Nichols plots corresponding to the times of operation as mass and throttle continually decrease as simulated and shown in Figure 3. The left plot shows the result with the baseline throttle response whereas the right plot shows the decreased stability margins produced by having one tenth the throttle response bandwidth. While the presence of throttle dynamics are what determine the maximum system gain bandwidth for the linear g-limiting algorithm, there is plenty of gain margin available even after the drastic decrease in the throttle response. This result corroborates the time domain simulation of the degraded throttle response scenario using the *linear* algorithm, Figure 5, and it also illustrates how the linear margins do not correspond to the inherently nonlinear heritage algorithm.

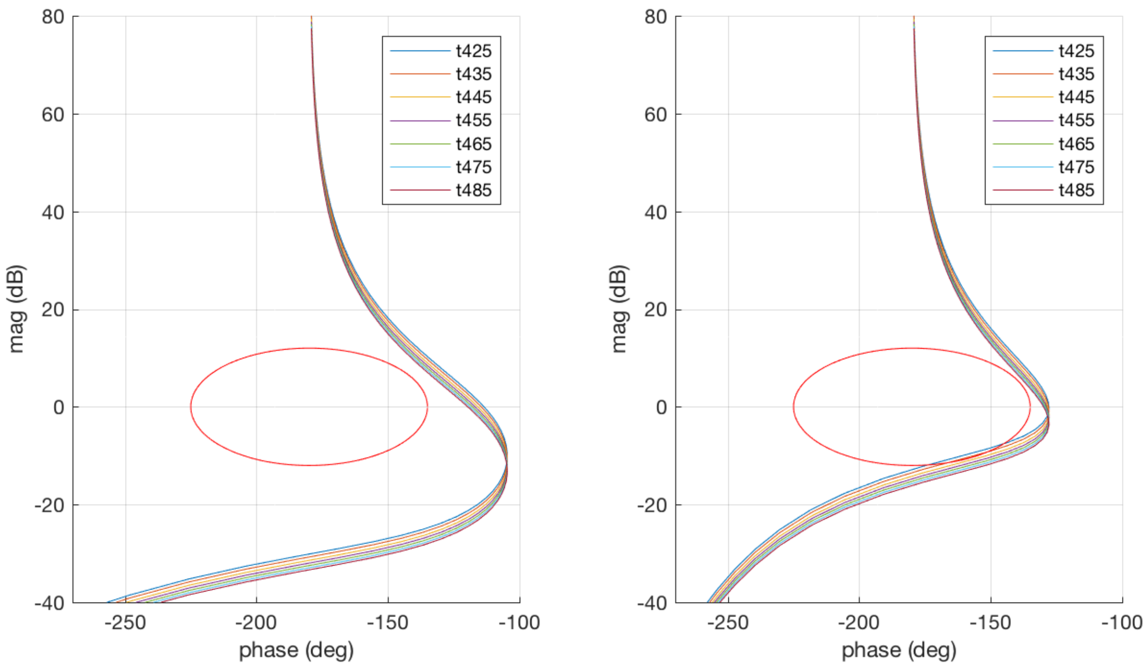

Figure 8: Nichols Response baseline (Left) and 1/10 throttle bandwidth (Right)

As shown in this section, the linear alternative to the baseline glimiting algorithm appears to exhibit generous stability margins across the range of accel and throttle conditions as well as under the extreme scenario of a very slow throttle response.

TUNING OF THE LINEARIZED ALGORITHM, SIMPLE MODEL

The previous sections analyzed a linear version of the glimiting algorithm using the same proportional and integral gains as were set in the heritage algorithm. To eliminate the overshoot with the linear system and utilize the available excess gain margin, a simple scaling of the gains was employed. Figure 9 shows the Nichols and time response of the system with 5 times the original k_p and k_i gains. Clearly the time response has much improved when the available gain margin is utilized.

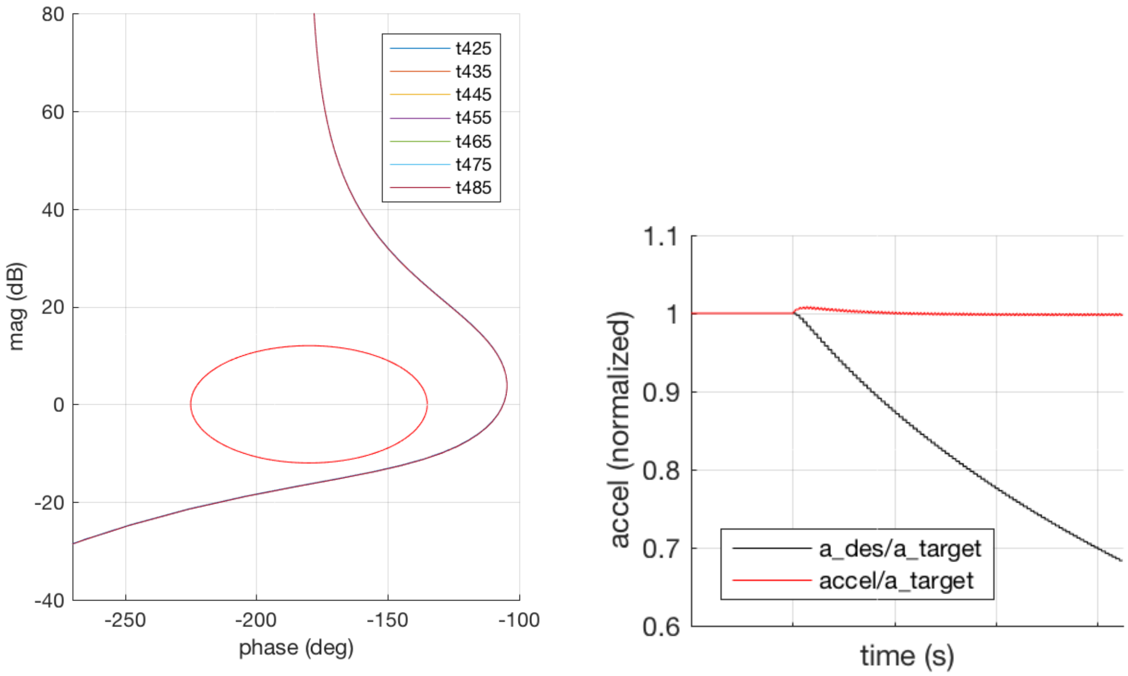

Figure 9: Nichols and Time Response of Linear Algorithm with 5x gains

As noted before, the time domain analysis comparing the nonlinear and linear version of the systems were produced without the effect of the throttle command quantization (simple rounding). Figure 10 compares the achieved acceleration responses of the nonlinear system and the linear system, where the linear system includes the 5x scaling on the gains. As evidenced by the similarity in the responses when quantization included, the effect masks the underlying differences highlighted before. The comparison of Figure 10 also illustrates that the well-tuned linear alternative g-limiting algorithm performs similar to the baseline approach.

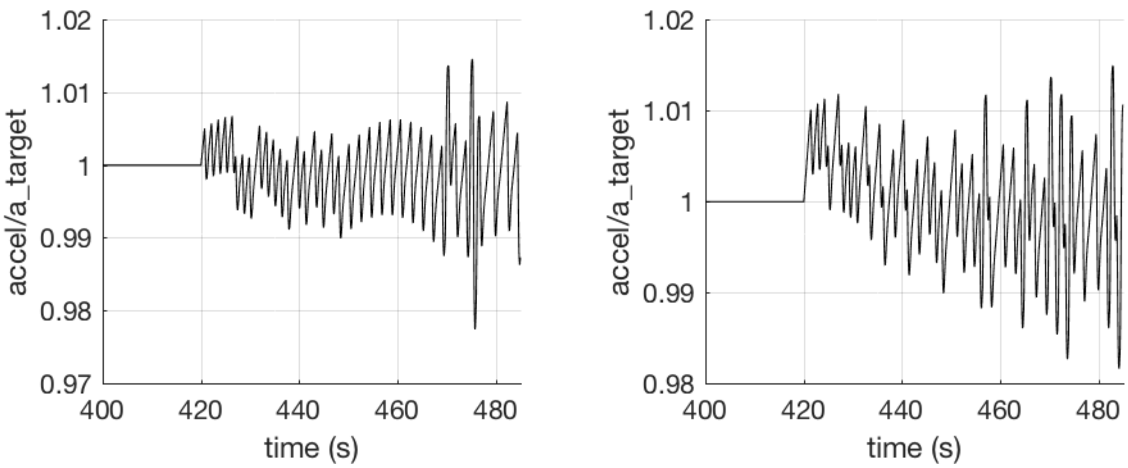

Figure 10: Nonlinear (left) and Linear (right) Accel Response

Figure 11 compares the resulting throttle responses for the nonlinear and linear systems, again showing very similar results.

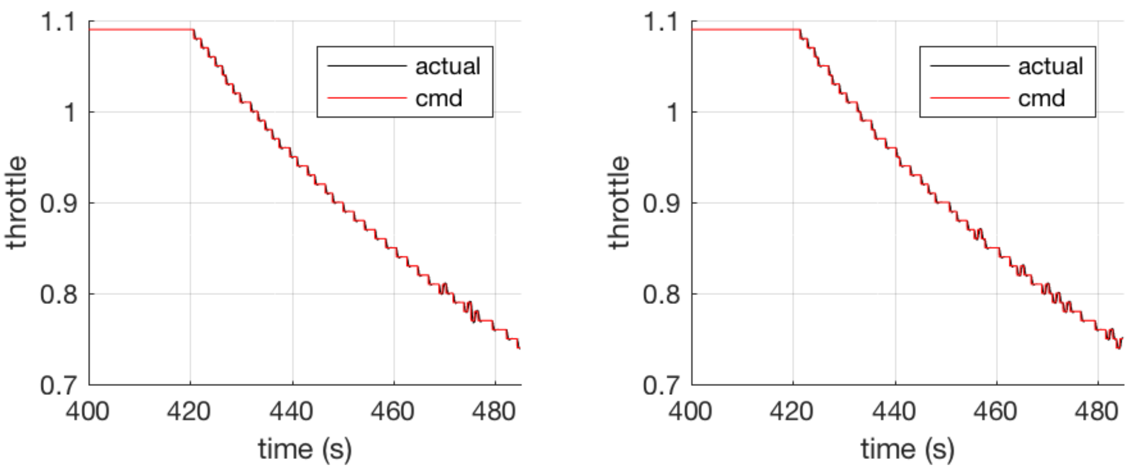

Figure 11: Nonlinear (left) and Linear (right) Throttle Response

6-DOF IMPLEMENTATION OF LINEARIZED G-LIMITING ALGORITHM

Preliminary 6-DOF analysis was performed using Marshall Space Flight Center (MSFC)'s flagship C++ simulation environment, Marshall's Aerospace Vehicle Representation in C (MAVERIC), in order to compare these two algorithms in a realistic flight environment. Figure 12 shows the results of 50-Monte Carlo sets which employ three unique variants: an untuned linearized g-limiting algorithm, a tuned linearized g-limiting algorithm, and the baselined heritage nonlinear algorithm.

As shown, the nominal responses of both the linearized and bibkubaer g-limiting algorithms in Figure 12 are similar to the profiles found in Figure 3. By visual inspection, it can be seen that the behavior of the linear g-limiting algorithm seems to take an expected linearized/smooth steadying response to targeting an acceleration, whereas the nonlinear algorithm clearly takes a more nonlinear response to targeting a specific acceleration. Preliminary results also showed that, when comparing mass-to-orbit performance of nominal Block-1B flight, the linearized g-limiting algorithm performed just as well as the nonlinear algorithm.

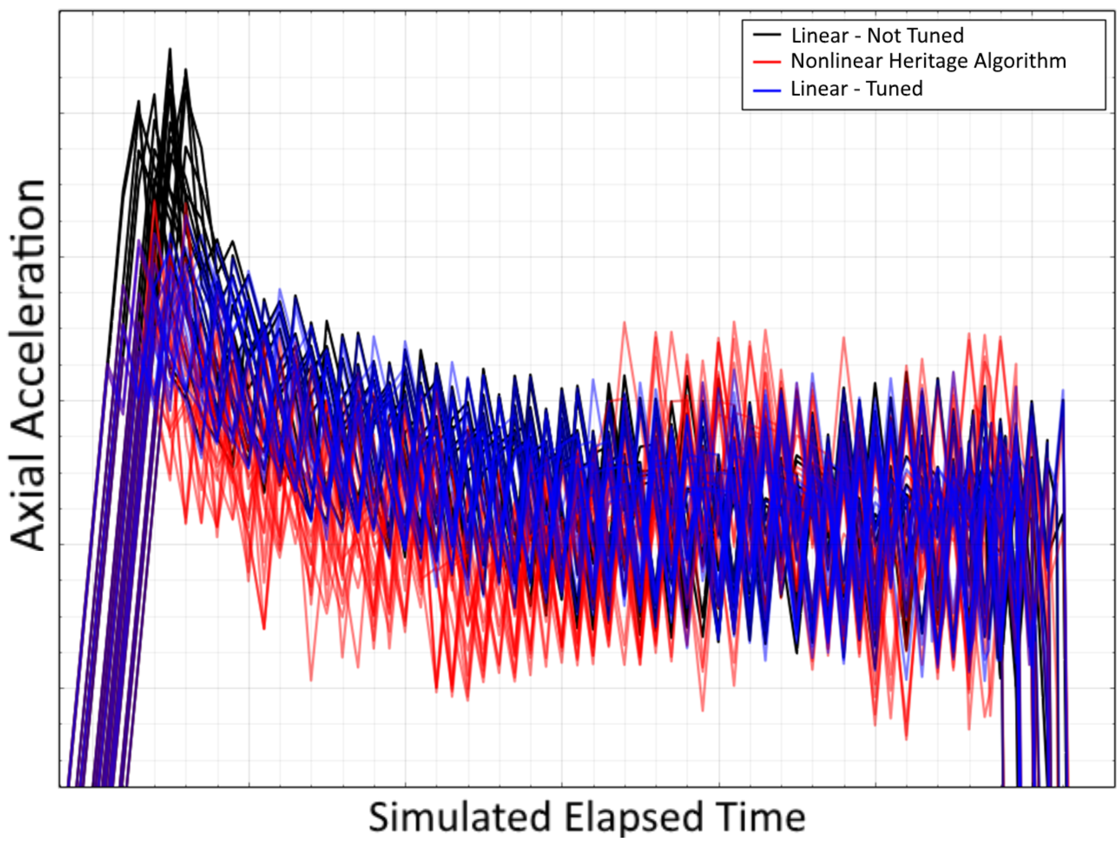

Figure 12: MAVERIC Preliminary 6-DOF Performance Results

Thus, as shown throughout this section and previous ones, the linearized algorithm brings advantages of stability enhancements and increased stability margin simply not possible with the nonlinear, multiplicative feedback implementation. The GN&C team at MSFC is continuing to run dispersion analysis on the possible benefits and disadvantages of the linearized architecture. The preliminary results show that a linearized g-limiting algorithm may provide a stable, simple alternative to the heritage g-limiting algorithm, containing physically representative and intuitive components. Following more comprehensive evaluations, the linear g-limiting algorithm may serve as a viable alternative to the heritage algorithm for future SLS flights.

CONCLUSION

Both nonlinear and linear glimiting algorithms perform adequately in the time domain and the differences in response between the two systems are hardly noticeable when command quantization is present. While the nonlinear system was flown throughout the Shuttle program and has performed well in many SLS simulations, the linear system provides an alternative formulation which appears to have increased overall robustness to system uncertainty and enables faster update rates. The linear system additionally provides a means to characterize robustness and deterministic system response using the well known classical LTI techniques, which can provide a level of assurance desirable for the certification of human spaceflight.

ACKNOWLEDGEMENTS

The authors appreciate the contributions of Greg Dukeman and Tom Fill on this work, especially their valuable inputs regarding the operation of and rationale for the Shuttle heritage approach to g-limiting.

REFERENCES

[1] 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.

[2] Space Shuttle Flight Software Detailed Design Specification. Volume III - Applications Part 1: Guidance, Navigation, and Control, USA001224, Feb 23, 2006

[3] Dukeman, G. ”Derivation of the Basic Shuttle-Heritage G-Limiting/Throttle Algorithm”, NASA MSFC EV42 Internal Document, Received April 26, 2019