SUB-MINIMUM IMPULSE ATTITUDE/RATE CONTROL OF SPACECRAFT

John P. McCullough, III[*] Steven L. Hough,[†] Keith R. Clements[‡], and Robert A. Hall[§]

In one of the future NASA exploration missions utilizing the Space Launch System (SLS), the Orion spacecraft separates from the Exploration Upper Stage (EUS), reorients itself end over end, and then autonomously or manually docks with a Co-manifested Payload (CPL) that is attached to the forward end of the EUS. During Rendezvous, Proximity Operations, and Docking (RPOD), the mated EUS/CPL acts as the passive vehicle for docking, yet must actively minimize body rates and translational velocity at the docking interface to accommodate International Docking System Standard (IDSS) requirements and more stringent Orion manual docking handling qualities. The initially designed EUS closed loop autopilot (i.e., flex filters, classically based phase plane, etc.) with the vehicle's mass properties and its baseline, twelve, non-throttle-able hydrazine thruster configuration is insufficient for achieving the initial docking state allocations.

Subsequently, more complex control modes are developed and analyzed to help minimize rates at the docking interface. These software modes include a minimum impulse mode, a feed forward state estimator mode, and a sub-minimum impulse mode. Monte Carlo analyses results are shown for these cases to illustrate the time domain performance in dispersed conditions for anticipated mission objectives. The sub-minimum impulse algorithm provides the lowest body rates of the analyzed control modes, and is the only one that meets the initial allocations.

Refinements to the RPOD requirements for the impacted missions allows several of the algorithms to robustly achieve increased docking rate allocations. Therefore, the sub-minimum impulse is not needed for the baseline design. However, sub-minimum impulse shows promise where further reductions in body rates are desired for a particular vehicle configuration, with only a modest impact to complexity over the standard autopilot design.

[*] Aerospace Engineer, Jacobs Space Exploration Group (JSEG), NASA Marshall Space Flight Center (MSFC), Huntsville, AL 35812.
[†] In-Space Controls Team Lead, Dynamic Concepts Inc., JSEG, NASA MSFC, Huntsville, AL 35812.
[‡] Control Systems Analyst, NASA MSFC, Huntsville, AL 35812.
[§] GN&C Subject Matter Expert, Mclaurin Aerospace, JSEG, NASA MSFC, Huntsville, AL 35812.

INTRODUCTION

The Space Launch System (SLS) is being developed to launch the Orion capsule around the Moon in 2021 on the Artemis I mission followed by several subsequent lunar missions, and potentially Mars missions in the future. For the first three Artemis missions, NASA plans to use the Block 1 configuration of the SLS consisting of two 5 segment solid rocket boosters, a core stage with 4 RS-25 liquid fueled engines, and an Interim Cryogenic Propulsion Stage (ICPS), a modified Delta Cryogenic Second Stage.[*] Artemis I will use this Block 1 configuration to launch the Orion Multi-Purpose Crew Vehicle (MPCV) consisting of a European Service Module (ESM), a launch abort system, encapsulating service module panels, and an unmanned Orion capsule around the Moon. Artemis II is planned to launch a crewed mission around the moon with the same configuration. Artemis III is expected to launch a crew to the Moon's proximity to rendezvous with landers and other support equipment and land on the Moon.

Block 1B is planned for use beyond Artemis III[†] to be able to launch even larger payloads or capsules to the Moon and beyond.[‡] Block 1B consists of the familiar rocket boosters and core stage, but it has a larger Exploration Upper Stage (EUS) with larger propellant tanks and more engines for launching larger payloads to the moon or other destinations.

SLS EUS CONFIGURATION

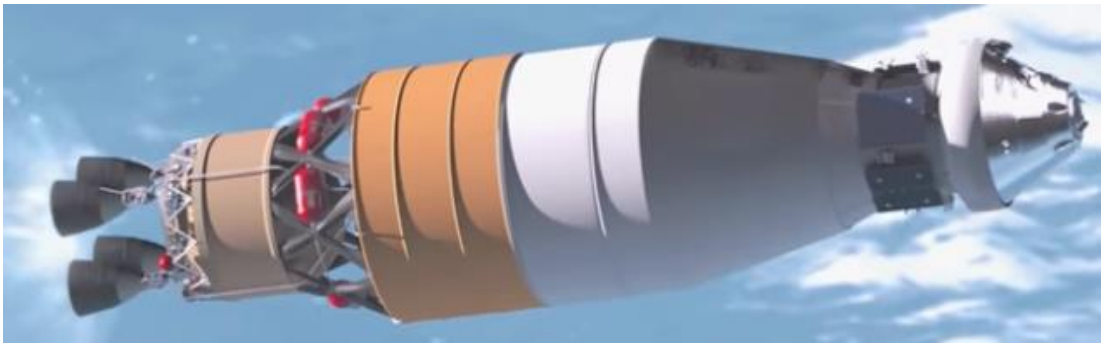

Figure 1. SLS EUS with USA, Spacecraft Adapter, Service Module, and Orion.

The first version of the EUS is planned to launch an Orion capsule with other co-manifested payloads utilizing a Universal Stage Adapter (USA) and Spacecraft Adapter. Figure 1 shows the configuration of the EUS, adapters, and Orion in the configuration to send the capsule to the Moon.

One of the potential co-manifested payloads to be launched along with the Orion capsule could be a lander and/or ascent vehicle. Those would reside internal to the USA. Since the Orion capsule is oriented in the vehicle stack with its docking port facing forward, it would need to separate from the EUS and dock with the co-manifested payload after USA jettison. At the time of these analyses, a simulated docking test was planned for the first EUS mission and docking for the second EUS mission. For these docking scenarios the EUS remains "stationary" or passive in an attitude hold with limited rotational rates, while Orion moves actively to match up its docking port with that on the co-manifested payload still attached to the EUS. The EUS control system is used to maintain the attitude hold of the EUS and its co-manifested payload, referred to together as the EUS, henceforth. The Orion capsule and ESM together will be referred to as Orion.

---

[*] NASA, (2017), "Getting to Know You, Rocket Edition: Interim Cryogenic Propulsion Stage," Retrieved from https://www.nasa.gov/sls/interim_cryogenic_propulsion_stage_141030.html.

[†] NASA, (2019), "NASA Commits to Future Artemis Missions With More SLS Rocket Stages," Retrieved from https://www.nasa.gov/feature/nasa-commits-to-future-artemis-missions-with-more-sls-rocket-stages.

[‡] NASA Marshall Space Flight Center, (2019), "Space Launch System Fact Sheet," FS-2019-10-067-MSFC, Retrieved from https://www.nasa.gov/sites/default/files/atoms/files/0080_sls_fact_sheet_10162019a_final_508.pdf.

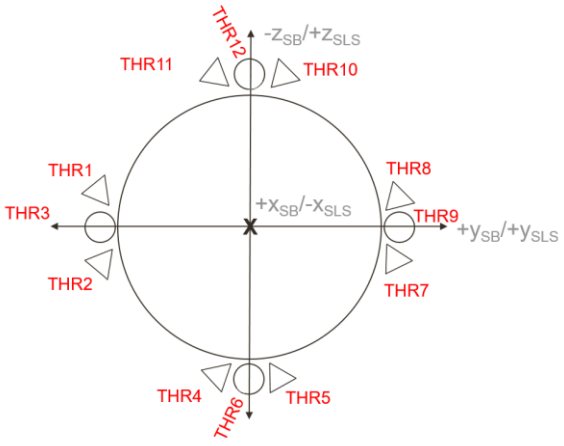

Figure 2. EUS RCS Thruster Configuration, Aft Looking Forward.

For roll, pitch, and yaw attitude control, the EUS is configured with eight Reaction Control System (RCS) thrusters that have already been chosen by the prime contractor based on NASA specifications. An additional four thrusters are on the EUS for axial thrusting maneuvers such as propellant settling, collision avoidance maneuvers, and EUS disposal into a heliocentric orbit. All thrusters are located behind the center of mass of the combined spacecraft. Figure 2 shows the configuration of the thrusters, which provide loss-of-thruster fault tolerance. Thrusters are generally fired in pairs to achieve a particular maneuver. For example, a positive rate about the y_{SLS} axis can be achieved by firing thrusters 11 and 10 or 1 and 8.

BASELINE CONTROL ALGORITHM AND LIMITATIONS

In the original SLS EUS baseline design, each axis is independently controlled with a typical phase plane controller using switching lines defined in terms of attitude and body rate errors which encompass a dead band zone where no thruster firings occur.[1] Outside those switch lines commands required to keep the EUS stationary are issued from each controller channel and sent through thruster/jet selection logic to fire appropriate thrusters to obtain the desired adjustments to body rates.

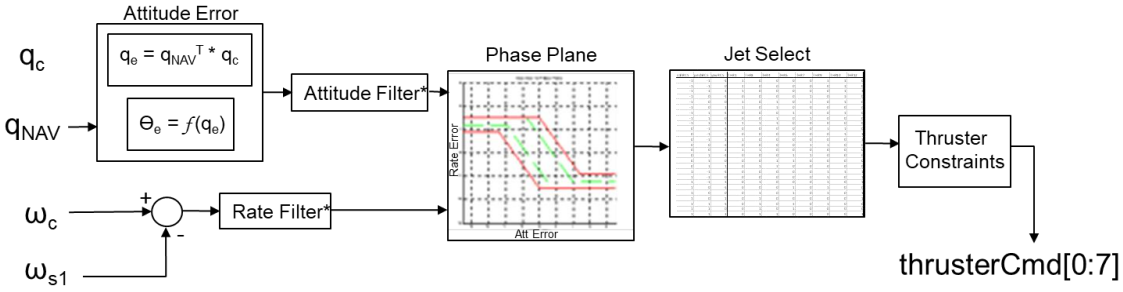

Figure 3. SLS EUS General Control System Algorithm.

In Figure 3, the quaternion error, q_e,is generated from the commanded quaternion, q_c, and the navigation measured quaternion, q_{NAV}. Quaternion to Euler angle conversions provide the attitude error, θ_e, while subtracting the measured angular body rates, ω_{s1}, from the commanded body rates, ω_c, provides the rate error. Attitude and rate errors are each then filtered through two 4th order filters, 8th order total, to filter out flex frequencies. The frequency separation between the rigid body and elastic dynamics allows filtered control of the rigid body attitude and rates and removes the potential for adverse influence of flexible body bending dynamics on the spacecraft while maintaining sufficient phase margin. The filtered attitude and rate errors feed into three independent phase plane controllers, one for each body axis. Jet selection logic translates

the desired roll, pitch, and yaw commands from the phase plane controller into commands for each thruster. Further thruster hardware constraints, such as "minimum on time" before a thruster can be turned off and restrictions to avoid water hammer in the RCS, are applied before sending the thruster commands, *thrusterCmd*, to the eight thrusters.

Phase plane designs are specified for the mission in time based segments, as certain parts of the mission have varying levels of required pointing accuracy. For the missions utilizing an EUS, a Rendezvous, Proximity Operations, and Docking (RPOD) phase is defined for docking algorithm testing or for actual docking. This is after Orion separation and after USA jettison. For the simulations conducted, the RPOD phase duration is just over 1,400 sec.

The initial RPOD attitude and rate requirements were based upon the International Docking System Standard.[2] This standard requires that the vector sum of pitch and yaw rates is less than 0.2 deg/s and the roll rates be less than 0.2 deg/s relative between the two bodies. The vector sum of pitch and yaw relative misalignment must be less than 4.0 deg, and the roll relative misalignment must be less than 4.0 deg. Relative lateral misalignment must be less than 0.328 ft (0.1 m) and relative lateral rate must be less than 0.131 ft/s (0.04 m/s). Due to thruster capability, mass properties, and manual control handling qualities of the Orion spacecraft, the initial rate limits imposed upon the EUS passive vehicle were significantly more demanding than those of the International Docking System Standard.

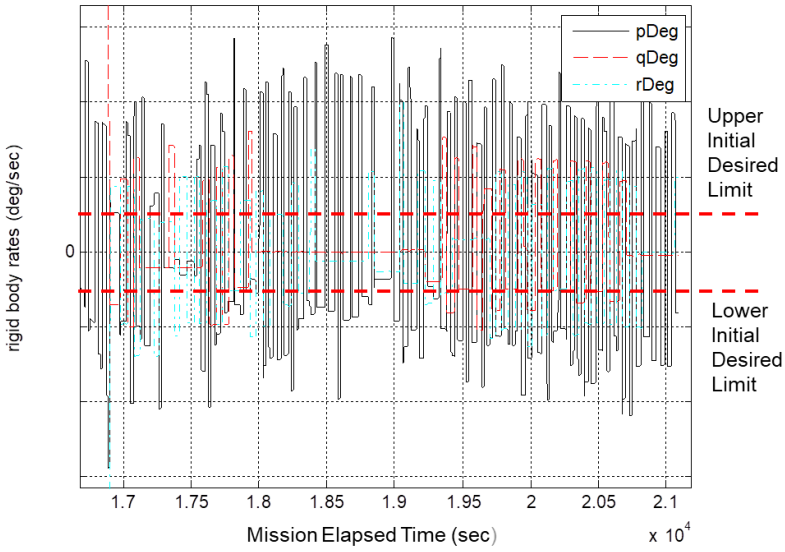

Figure 4. EUS Initial Baseline Rigid Body Rates.

Concern arose after hand calculations, based on the initial estimates of the capabilities of the EUS thrusters and vehicle configuration, showed that minimum impulse firings provided roll rate changes and pitch/yaw rate changes greater than initial allocated rates. Translational motion was not a concern, primarily because the vehicle center of mass is located between the thrusters and the docking mechanism, hence docking interface motion is smaller when compared to that of the center of mass. Six Degrees of Freedom (6 DoF) simulation of the EUS in the Marshall Aerospace VEhicle Representation in C (MAVERIC) with a high fidelity thruster model showed similar results as seen in Figure 4.

Investigation of the control algorithm indicated a couple of things. The baseline algorithm relies on feedback from sensors to determine angle and rate errors and the corresponding thruster on/off commands. Thus the minimum rate changes achievable from a thruster firing are typically limited by the maximum of two constraints: 1) the thruster minimum impulse or firing duration and 2) the control loop latencies such as sensor latencies, bus delays, Flight Control System (FCS) / Flight Software cycle times, hardware controller response times, flex filters, etc.

The thruster minimum impulse is typically constrained by the hardware itself. As thruster pulses get shorter, uncertainty increases in the amount of thrust produced due to incomplete propellant oxidation. To reduce uncertainty and ensure consistency of the thruster firings, a minimum impulse threshold is usually specified.

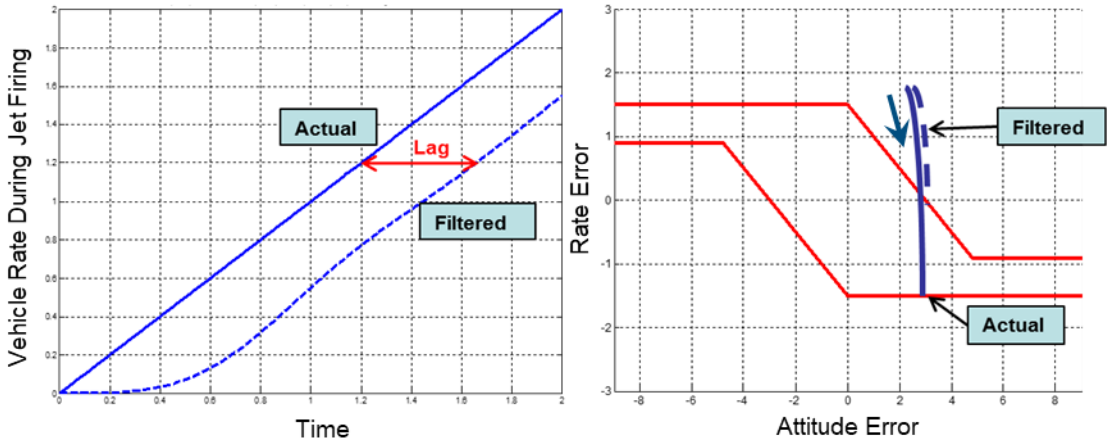

Figure 5. Latency effect on Vehicle Rate with a Phase Plane Controller.

The other limiting factor is avionics latency. As mentioned, the EUS has two cascaded 4th order flex filters on the attitude and rate errors, before they are sent to the phase plane controller. Figure 5 shows the effect in the time domain and in the phase plane.[1] In the right portion of Figure 5, both the filtered, dashed blue curve, and unfiltered signals, blue curve, are initially outside the dead band red lines, and thus firing is commanded, driving the state closer to the dead band. The actual unfiltered signal enters the dead band first, but firing continues until the filtered signal, which the controller is using, hits the edge of the dead band. By the time the filtered signal reaches the dead band, the unfiltered signal or actual rate has changed much more than needed to reverse the trajectory back into the phase plane switching boundaries.

Thus, minimizing the latency in the system and minimizing the thrust duration or impulse is important in reducing the body rates of the vehicle, yet are often irreducible beyond a certain level due to system tradeoffs or hardware limitations. Note that system tradeoffs occur between minimizing rotation rates and increasing thrust to provide better manual control handling qualities. In the sections that follow, control algorithm solutions are proposed that account and compensate for constrained levels of latency and selected hardware characteristics. These can allow larger thrusters for better handling qualities, yet still provide minuscule rotation rates if needed for fine attitude/rate control or docking.

CONTROL ALGORITHM SOLUTIONS

In order to minimize latency in the SLS EUS control system and/or minimize the thrust duration/impulse, several control algorithms, and combinations thereof, were developed and/or evaluated utilizing time domain Monte Carlo (MC) analysis in planned mission environments. Options included improvements to the flex filter phase lag to reduce sensitivity to latency, implementation of a Feed Forward State Estimator (FFSE), implementation of a minimum impulse algorithm, development of a new sub-minimum impulse algorithm, and select combinations of those. These methods were developed, implemented in 6 DoF simulations, and tested in parallel by the Authors.

As an initial attempt to reduce body rates, the flex filter was redesigned from having 180 ms of latency down to 70 ms of latency based on the phase lag at the rigid body phase margin crossover frequency. The original 8th order filter was reduced to a simple 2nd order low pass filter, while still providing sufficient flex amplitude reduction. This improved the body rates somewhat, but was not alone capable of the reductions required to achieve the initial EUS requirements.

The FFSE was implemented based on a second order Luenberger State Observer. Typically, not all vehicle states are directly measured; hence an "observer" is used to estimate unmeasured states. For example, Apollo[3] and Shuttle[4] on-orbit RCS control had only an attitude measurement input, so the software estimated rotational rate and rotational acceleration. The approach here, derived from Apollo and Shuttle, selects filter gains based on stability and performance. The state estimator implementation works to take the place of "lag-causing" bending filters. The basic premise is that the expected/estimated thruster firing torque is fed forward in the control loop to minimize latency, while the state estimator removes error due to uncertainty in vehicle mass properties and thrust, attenuate flex (with supplemental filter), and better estimate rate.

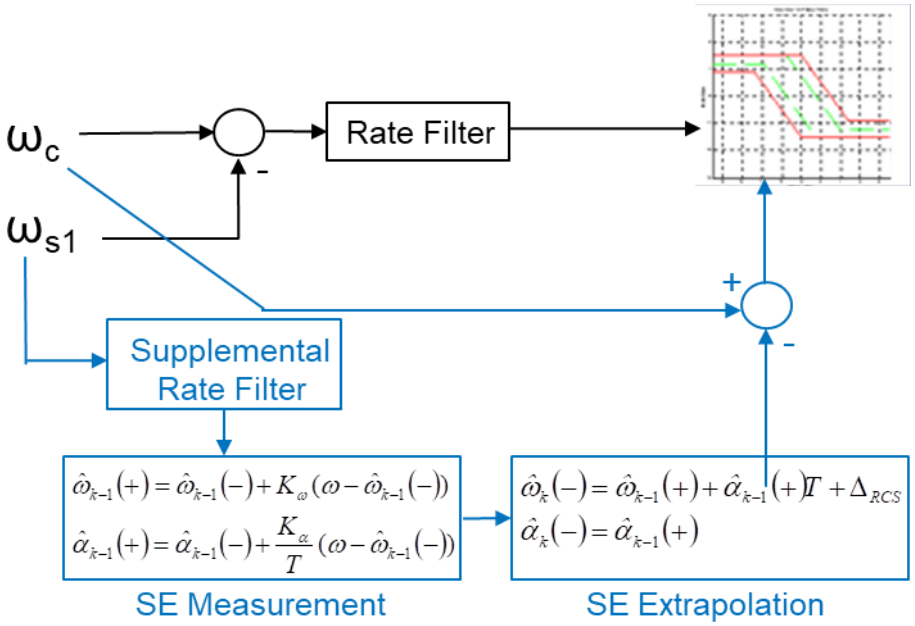

Figure 6. Feed Forward State Estimator (SE) Implementation.

Figure 6 shows the state estimator in blue replacing the original flex filter on the rate channel in black. The state estimator is written as a state estimation extrapolation and the state estimate measurement.[5] The updated estimated measurement states of the body rate, $\hat{\omega}_{k-1}(+)$, and the estimated angular acceleration measurement, $\hat{\alpha}_{k-1}(+)$ are computed from the previous estimated values, $\hat{\omega}_{k-1}(-)$ *and* $\hat{\alpha}_{k-1}(-)$, plus gains, K_ω, and K_α/T, where T is the time step, times the residual error, $\omega - \hat{\omega}_{k-1}(-)$. Those updated estimated measurement states are substituted into the extrapolation equations. The current estimated rates, $\hat{\omega}_k(-)$ are computed from the updated estimated rate measurement and the updated estimated acceleration measurement times the time step plus the feed forward signal, the anticipated change in rate due to the thruster firing, Δ_{RCS}. The current FFSE implementation uses a priori data for thrust, thruster locations, and inertia estimates to compute the feed forward signal. Last the updated estimated acceleration becomes the current estimated angular acceleration $\hat{\alpha_k}(-)$. Those signals are then used to calculate the error for use in the phase plane. Note that future investigations could include a higher fidelity a priori thrust profile and potential use of RCS tank pressure measurements in the FFSE. Also a slightly modified FFSE was investigated and implemented by two of the Authors for thruster Failure Detection Isolation and Recovery (FDIR).

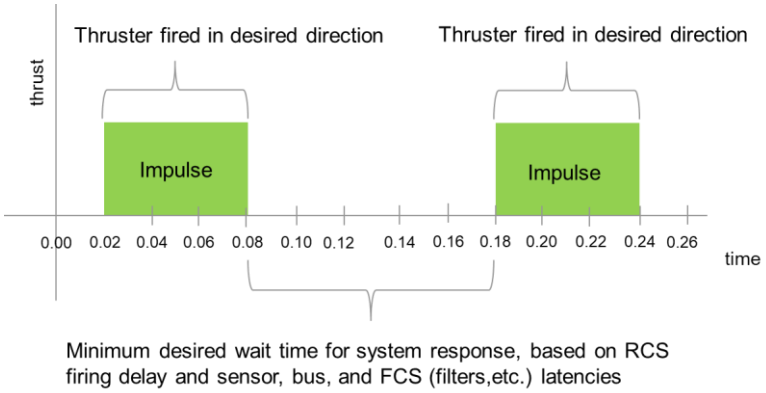

Figure 7. Minimum Impulse with a Wait Concept.

An algorithmic solution to the baseline control architecture's latency sensitivity is denoted here as the minimum impulse mode, whereby the logic issues minimum impulse firings with a predetermined wait time between each firing. The premise is to issue thruster firing commands at a fixed duration for the minimum firing time possible from the thrusters and then wait for a given time based on system latencies before firing again, as shown in Figure 7. This allows the filtered/delayed rate measurement to converge on the true vehicle rate before subsequent commands are issued. However, the pulse lengths can only be shortened to the capability of the thrusters. As pulses get shorter, uncertainty increases in the amount of thrust produced and timing due to incomplete propellant oxidation, valve response, and other factors. Minimum impulse with a wait also introduces a reduction in control authority, as pulses are limited to a fraction of the 100% duty cycle. Thus, this implementation also currently includes logic to exit this mode when large excursions from the phase plane switch lines are observed.

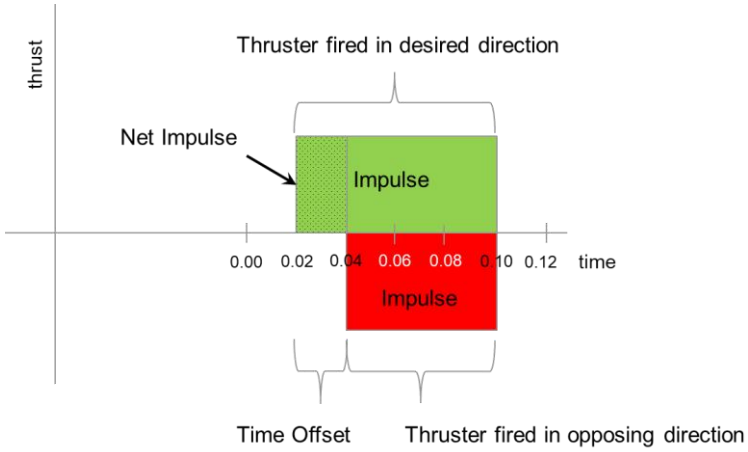

Figure 8. Sub-Minimum Impulse Concept.

In a brainstorming session to figure out how to meet the angular rate requirements imposed on the EUS, the idea for a sub-minimum impulse method was proposed. Figure 8 illustrates the concept via a thrust timeline. Sub-minimum impulse is an enhancement to minimum impulse where opposing thrusters are commanded to fire with a prescribed time offset such that the net torque/impulse on the vehicle is below the minimum impulse level of any given thruster by itself. Thus the net impulse is only dependent on the time offset of the opposing thruster firings or the lengths of the two pulses if they are started at the same time and fired for different durations. In the EUS case, the net impulse is only limited by the FCS execution rate. For example, for a positive pitch maneuver, $+y_{SLS}$ direction in Figure 2, *Thr1* and *Thr8* are fired in the desired direction, and one FCS time step later *Thr2* and *Thr 7* are fired. All four thrusters are com-

manded off after four time steps from the start of the original firing. Longer thrusts could be used, but using minimum impulse firings, plus the firing of one thruster pair for the time offset, minimizes the amount of propellant used. Note that if shutting off multiple thruster at a time is not allowed due to water hammer, the thrust and opposing thrust could be started at the same time and shut off at different times.

Similar to the minimum impulse with wait algorithm, the wait time is also implemented in the sub-minimum impulse method to allow the system measurements to catch up to the actual rates. The results in the proceeding sections will show that the sub-minimum impulse method uses more propellant than minimum impulse, but much less than the EUS baseline phase plane case, while providing reduced body rate changes per commanded thruster firings.

Combinations of the above methods and different wait times were executed using the MAVERIC time domain simulation. A total of 10 different nominal undispersed cases were evaluated. Six of those were evaluated using a MC approach in planned mission scenarios. All minimum impulse cases without the state estimator were combined with the 70 ms lag flex filter. Results for the original baseline phase plane controller (case 1), the FFSE (case 3), the minimum impulse with 200 ms wait (case 4), and sub-minimum impulse with 200 ms wait (case 6) are presented in the next section. The reduced latency flex filter (case 2) by itself did not significantly improve results. The FFSE with its own supplemental flex filter was combined with the minimum impulse mode (case 8), as it is similar to the "Alt Mode" flown on the Space Shuttle.[6] That case produced better results than the FFSE by itself (case 3), but still not as good as the minimum and sub-minimum impulse cases. Two cases not shown used a shorter wait time of 120 ms with cases 4 and 6. That wait time still provided enough time for the system measurements to catch up to the actuals before additional firings, and thus did not significantly impact the results. The remaining cases used 0 ms, 80ms, and 160ms wait time with the FFSE with minimum impulse. None of those wait time produced different results, because the feed forward term in the FFSE eliminated the sensitivity to the wait times.

NOMINAL AND MONTE CARLO RESULTS

In nominal, undispersed simulations, only the sub-minimum impulse test cases meet all the specified initial requirements. The sub-minimum impulse propellant consumption was marginally higher than the minimum impulse cases and the FFSE cases, but was greatly reduced from the baseline case and the case with the reduced 2nd order filter.

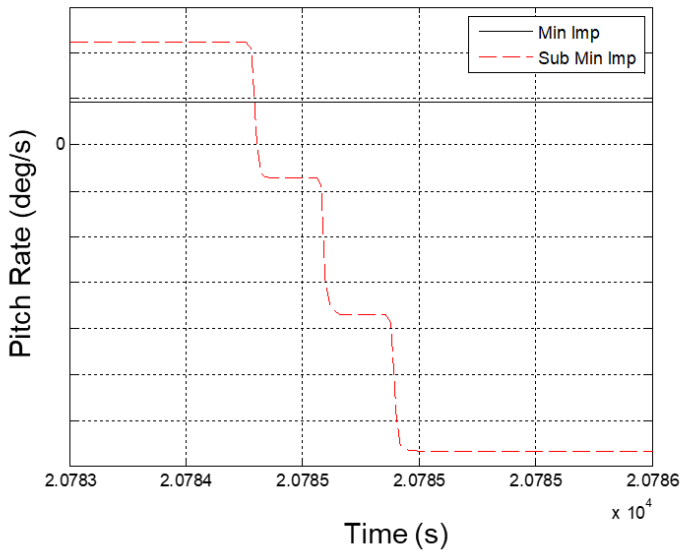

Figure 9. Sub-Minimum Impulse Time Response Showing Potential Flex Interaction.

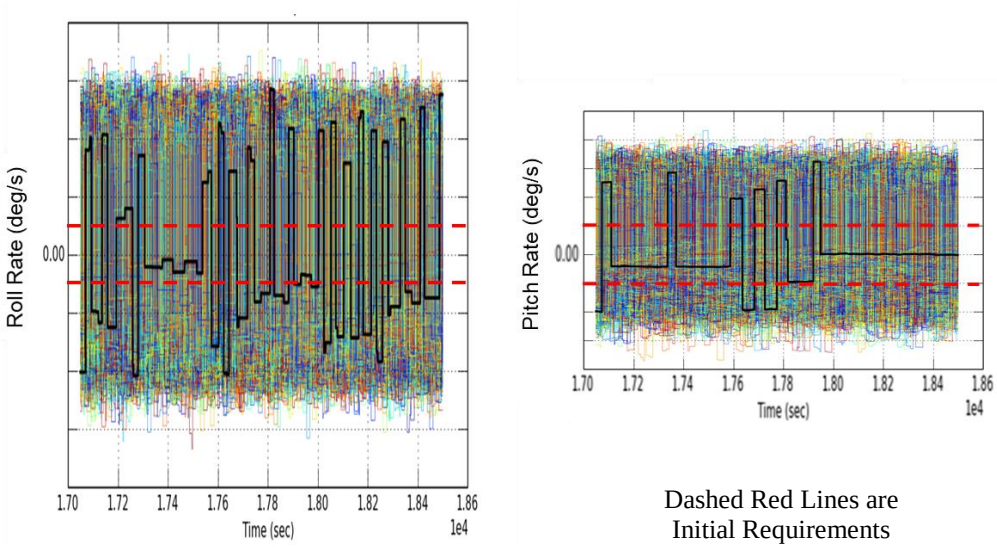

Figure 10. Case 1 - Roll and Pitch Rate MC Results for the Original Baseline Phase Plane.

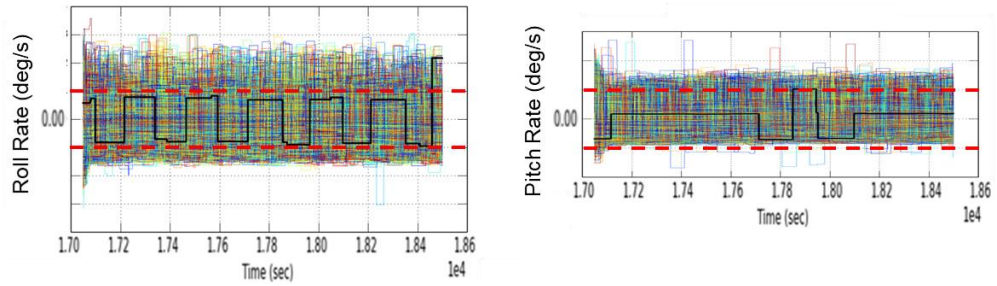

Figure 11. Case 3 - Roll and Pitch Rate MC Results for the Feed Forward State Estimator.

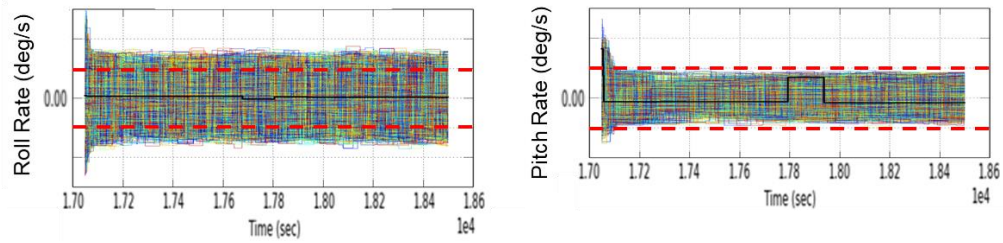

Figure 12. Case 4 - Roll and Pitch Rate MC Results for Minimum Impulse w/ 200 ms Wait Time.

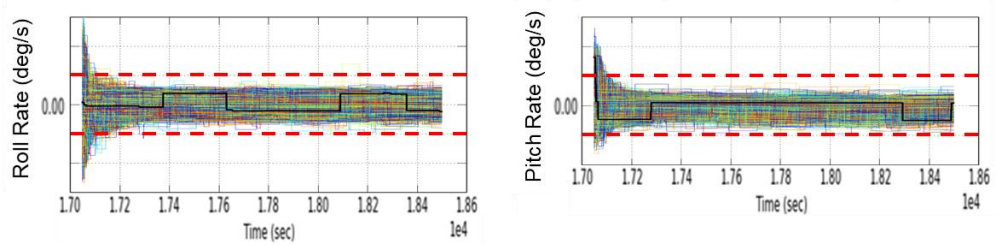

Figure 13. Case 6 - Roll and Pitch Rate MC Results for Sub-Minimum Impulse w/ 200ms Wait Time.

Nominal results showed improved performance of sub-minimum impulse over minimum impulse in the roll axis, but they both show the same performance in the pitch/yaw axes. The reason for this is that the pitch/yaw value reported for sub-minimum impulse is actually three consecutive thruster firings close together, as seen in Figure 9. When viewed over a large time scale these appear to be one firing. These consecutive firings were attributed to flex interacting with the algorithm, discussed below.

To address the use of the non-linear phase plane controller, uncertainties in the EUS system properties and parameters, and potential for flex interactions, MC runs of several of the cases were performed. 2000 runs of each case were done with over 1500 dispersed variables in each run. Examples of some of the variables dispersed were mass properties (masses, inertias, center of masses), flex (frequencies, damping, gains), slosh (masses, frequencies, damping), sensors (alignment, locations, parameters), stage alignments, venting (force, direction, flow rate), and thrusters (alignment, pressure drops, max thrust, thrust factor).

Figures 10 through 13 show the roll and pitch rate MC results for the standard baseline option (case 1), the FFSE (case 3), minimum impulse (case 4), and sub-minimum impulse (case 6) cases respectively. Those plots are scaled similarly for comparison from one figure to the next. Yaw results were similar to pitch results except the sub-minimum impulse case showed even less variation in the MC results than the pitch case. Yaw MC time histories are not shown. Note again that MC runs for the cases with different wait times were not executed, since those cases showed limited impact on the nominal results. Table 1 summarizes the Monte Carlo results for the cases in which they were run. Red indicates where allocations were not met, while green indicates allocations that were met. Except for propellant used, values are based on the maximum dispersed values over the docking phase, expressed as percentages of the allocations. So if the maximum roll rate during the docking phase for a particular MC case is half the roll allocation, the table shows 50%, and if it is twice the roll allocation, it shows as 200%.

Table 1. Summary of Maximum Values of Monte Carlo Results

Case	% of Initial Roll Rate Allocation	% of Updated Roll Rate Allocation	% of Initial Pitch/Yaw Rate Allocation	% of Updated Pitch/Yaw Rate Allocation	% of Initial Translational Velocity Allocation	% of Updated Translational Velocity Allocation	% of Baseline Propellant Used
1-Baseline	700%	233%	440%	176%	182%	87%	100%
2-Reduced Lag Filter	500%	167%	310%	124%	120%	57%	31%
3-FFSE	300%	100%	270%	108%	94%	45%	11%
4-Min Impulse (200 ms wait)	170%	57%	100%	40%	92%	44%	7%
6-Sub-Min Impulse (200 ms wait)	95%* / 140%	47%	95%* / 105%	42%	81%	39%	22%
8-FFSE w/ Min Impulse (0 ms wait)	180%	60%	140%	56%	89%	42%	7%

*without 2 outliers in roll and 1 in pitch due to flex, sub-minimum impulse also met the initial requirements. Outliers can be reduced with an adaptive phase plane algorithm.

These MC results assume there are limited disturbances (only gravity gradient), thruster performance as predicted by the high fidelity thruster model, and mass properties are bounded by the configuration/dispersions analyzed. Note that there were two outlier runs in the sub-minimum impulse case in roll and a single one in pitch. All the rest of the runs in that case were below that threshold. As mentioned, these consecutive firings were attributed to flex. A minimum impulse thruster firing with wait times between pulses at the left or right edge of the phase plane slightly excited flex modes. An adaptive phase plane method provided some improvement in results, although MC results in previous figures showed there was not much of an impact from flex interactions. With the exception of the three outlier runs, the sub-minimum impulse method showed the best MC angular rate performance against the initial RPOD rate allocations.

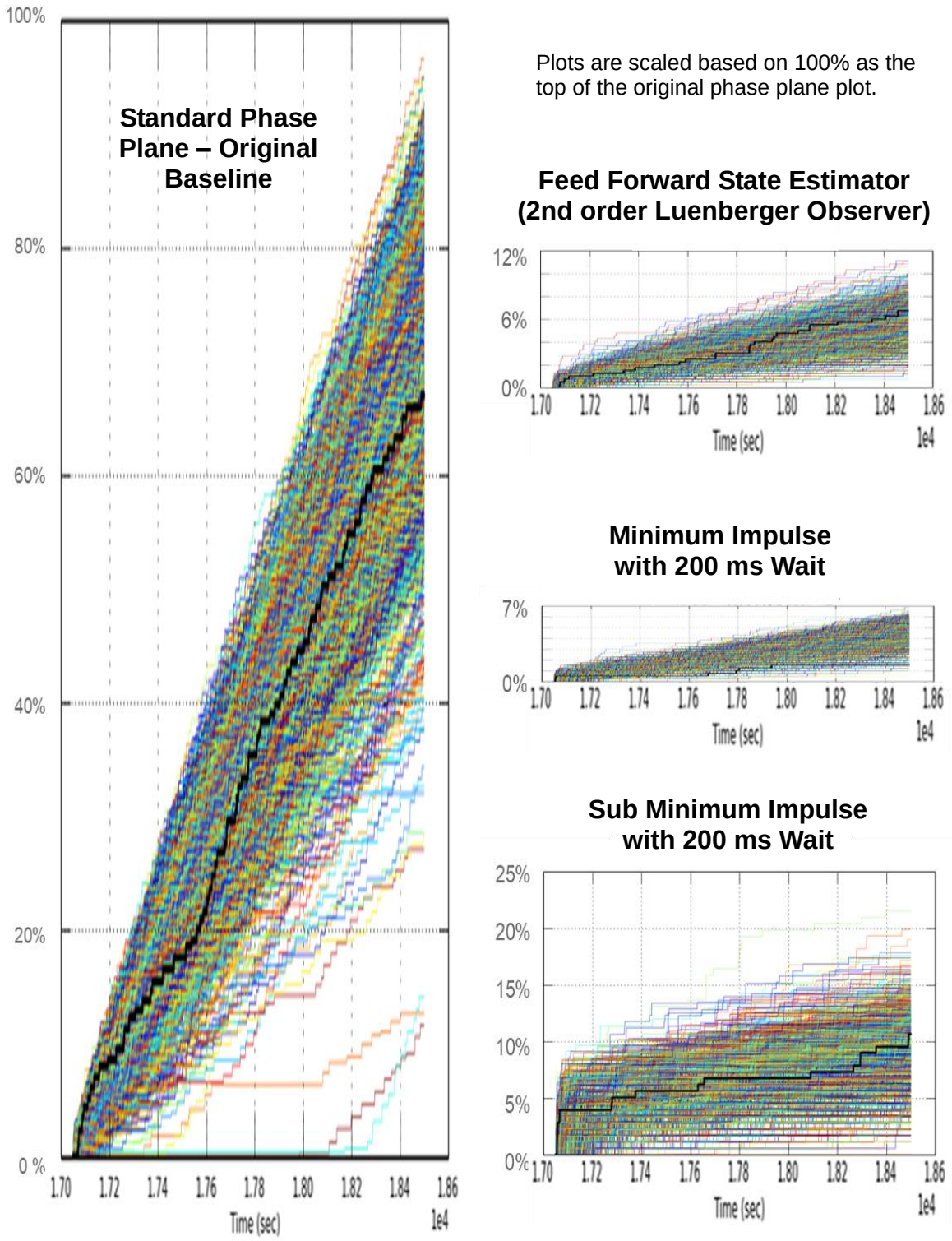

Figure 14. Propellant Usage Comparison of four of the Control Options Considered.

By the time the MC runs were processed, some analysis of the docking capability of Orion had been completed by Johnson Space Center personnel, and the allowed rotation rates on the attitude hold for the EUS during docking were subsequently increased. With the MC simulation and mission scenario assumptions above and the updated constraint values, the minimum impulse, sub-minimum impulse, and FFSE combined with minimum impulse cases all met the updated requirements as seen in Table 1. This included the aforementioned outlier runs.

With the relaxed constraints on the allowable EUS rotation rates during docking, the FFSE with Minimum Impulse and 0 ms wait time was selected as the new baseline for the EUS design. It is similar to the "Alt Mode" flown on the Space Shuttle, and thus has a higher technology readiness level than the sub-minimum impulse algorithm. It provides similar performance to basic minimum impulse mode for achieving tight rate control. However, the FFSE with minimum impulse minimizes the wait period between firings which increases the disturbance rejection capability. Thus, that algorithm was baselined as the algorithm employed during proximity operations for the EUS missions.

As mentioned, some concerns with the minimum impulse and especially the sub-minimum impulse method are propellant usage, disturbance rejection / reduction in control authority, stability, and rates near filtered signal noise limits or flex interaction causing extra firings. While all of those were investigated and analyzed, propellant usage will be addressed here, because it is the next key metric for assessing vehicle control performance, second to the rotation rates provided by the algorithms.

A potential concern with the sub-minimum impulse method is propellant consumption, since two pairs of offsetting thrusters are fired nearly simultaneously. Figure 14 shows the dispersed propellant usage for four of the cases investigated. The plots are scaled relative to the original baseline case such that 100% encompasses that case's propellant usage. While the maximum propellant usage by sub-minimum impulse on average was approximately twice that of the FFSE or minimum impulse, it was still less than about 25% of that used by the original baseline phase plane. Note that the sub-minimum impulse mode shows larger steps in propellant usage at each firing due to firing desired and opposing thrusters. However, on average it appears that there were fewer instances of firings occurring due to reduced body angular rates and the vehicle state taking longer before it drifts out of the phase plane dead bands. Despite the use of opposing thruster firings in the sub-minimum impulse method, its ability to achieve tighter control in the low disturbance environment yields a significant reduction in propellant consumption over the baseline phase plane case.

CONCLUSION

With the original phase plane control design and vehicle properties, the SLS EUS could not meet the desired body rates during RPOD. Implementation of a minimum impulse algorithm, a sub-minimum impulse algorithm, and a FFSE algorithm all provided promising algorithmic solutions for achieving reduced body rates. The sub-minimum impulse algorithm provided the lowest body rates of the analyzed control modes in both nominal and MC cases. It also met the initial desired body rates allocated to the EUS at the cost of decreased control authority and slightly increased propellant consumption compared with the minimum impulse method.

The FFSE with minimum impulse algorithm was chosen as the new baseline for EUS RPOD, because it meets the revised rotational rate requirements, exhibits better disturbance rejection capability, yields minimal propellant usage, and has Shuttle flight heritage. However, with only modest impacts to software/algorithm complexity, the sub-minimum impulse algorithm shows huge promise for providing an increased range of capabilities for a given set of thrusters on a particular vehicle for RPOD or spacecraft station keeping, in particular, where desired rotation rates for a period of time are less than that of minimum impulse provided by the thrusters.

REFERENCES

[1] Hall R., Hough S., Orphee C., and Clements K., "Design and Stability of an On-Orbit Attitude Control System Using Reaction Control Thrusters." AIAA SciTech GN&C Conference, January 4-8, 2016.

[2] NASA, International Space Station (ISS) Multilateral Control Board (MCB), "International Docking System Standard (IDSS) Interface Definition Document (IDD) Rev. E." October 2016, Table 3.3.1.1-2.

[3] Kubiak, E., "Performance Comparison of Apollo and AAP CSM RCS DAP State Estimation Schemes." Project Apollo Internal Note, November 25, 1969, MSC-EG-69-46.

[4] Zimpfer, D., Kirchway, C., Hanson, D., Jackson, M., and Smith, N., "Shuttle Stability and Control of the STS-71 Shuttle/MIR Mated Configuration." AAS/AIAA Space Flight Mechanics Meeting, Austin, Texas, February, 1996, AAS-96-131.

[5] Gelb, Arthur (Ed.), *Applied Optimal Estimation*. The Analytic Sciences Corporation, Cambridge, MA: The M.I.T. Press. 16th printing, 2001, pp 102-119.

[6] Boeing Company, "Space Shuttle Orbiter Operational Level C Functional Subsystem Software Requirements (FSSR) Guidance Navigation and Control Part C, Flight Control Orbit DAP." December 14, 2000, STS-83-0009-30.