(Preprint) AAS XX-XXX

CHECK-CASES FOR LUNAR SIX DEGREE-OF-FREEDOM SIMULATIONS

Jason Neuhaus[*], Matthew Hawkins[†], Jesse Couch[‡], Heather Koehler[§], Bruce Jackson[**], Robert Shelton[††], Robert Bossinger[‡‡], Kyler Brazukas[§§], Trey Fiebelkorn[***], Edwin Crues[†††], Benjamin Margolis[‡‡‡], Tristan Hasseler[§§§], Justin Ganniban[****], Nicholas Olson[††††], Zu Quan Li[‡‡‡‡], Nathan Perreau[§§§§], Timothy Curry[*****], Dane Erickson[†††††], Matthew Andreini[‡‡‡‡‡], and Zhiqanq Zhou[§§§§§]

This effort expands upon a previous NASA activity that developed flight simulation benchmark check-cases to include new check-cases for the Cislunar domain, comparing multiple NASA simulation tools. The results of this effort describe the benefits of standardizing inputs, simulation comparisons and describe an interactive website that enables comparison of externally provided simulation data. Participating simulations improved their software and identified implementation errors. This activity elevated simulation credibility and provided a measure of validation for the simulations actively in use for NASA's Human Landing Systems (HLS).

INTRODUCTION

In 2012, the NASA Engineering and Safety Center (NESC) conducted an assessment, *Development of Verification Data for Flight Simulation*, to develop Earth-based benchmark check-cases for aerospace vehicle Six Degree of Freedom (6DOF) flight simulation verification (Reference 1).

[*] Senior Chief Scientist, Simulation Development and Analysis, NASA Langley Research Center, Hampton, VA.
[†] HLS Deputy GNC Discipline Lead, NASA Marshall Space Flight Center, Huntsville, AL.
[‡] Research & Development Engineer/Computer Scientist, Adaptive Aerospace Group Inc., Hampton, VA.
[§] NASA Technical Fellow Flight Mechanics, NASA Langley Research Center, Hampton, VA.
[**] Senior Research & Development Engineer, Adaptive Aerospace Group Inc., Hampton, VA.
[††] Lead Simulation Engineer, Commercial Crew Program, NASA Johnson Space Center, Houston, TX.
[‡‡] Research & Development Engineer, Adaptive Aerospace Group Inc., Hampton, VA.
[§§] Engineering Intern, Adaptive Aerospace Group Inc., Hampton, VA.
[***] Engineering Intern, Adaptive Aerospace Group Inc., Hampton, VA.
[†††] Lead, Crewed Human Interaction Piloting Working Group, NASA Johnson Space Center, Houston, TX.
[‡‡‡] Aerospace Engineer, Systems Analysis Office, NASA Ames Research Center, Mountain View, CA.
[§§§] Robotics Technologist, Robotic Manipulation & Sampling, Jet Propulsion Laboratory, Pasadena, CA.
[****] Simulation Engineer, Control Systems Design and Analysis, NASA Marshall Spaceflight Center, Huntsville, AL.
[††††] Simulation Engineer, Control Systems Design and Analysis, NASA Marshall Spaceflight Center, Huntsville, AL.
[‡‡‡‡] Aerospace Engineer, Simulation and Graphics Branch, NASA Johnson Space Center, Houston, TX.
[§§§§] Engineer, Simulation Development and Analysis, NASA Langley Research Center, Hampton, VA.
[*****] Simulation Engineer, Guidance, Navig. and Mission Analysis, NASA Marshall Spaceflight Center, Huntsville, AL.
[†††††] Aerospace Engineer, Guidance, Navig. and Mission Analysis, NASA Marshall Spaceflight Center, Huntsville, AL.
[‡‡‡‡‡] Flight Mechanics Engineer, Analytical Mechanics Associates, NASA Langley Research Center, Hampton, VA.
[§§§§§] Aerospace Technologist, Dynamics and Control Branch, NASA Langley Research Center, Hampton, VA.

It was noted in that assessment that many NASA Centers have independently developed preferred frameworks for flight simulation software, and that differences in model implementation and numerical approaches resulted in variations between simulations and resulting analyses.

Commercial companies are seeking to provide services for NASA in the cislunar region, providing transportation elements and/or deploying orbital and lunar surface assets. The trajectories and dynamics of these complicated systems must be simulated using validated simulations per NASA standards which are crucial to predicting space flight performance and enabling key design decisions.

Lunar-based check-cases support the Artemis Missions and commercial lunar landers as they seek to use high-fidelity flight mechanics tools to simulate complex integrated systems and demonstrate requirements and objectives are met through analysis.

BACKGROUND AND SCOPE

To understand how this assessment is a new expansion of the previous activity, it is helpful to summarize what was done previously. The previous assessment included three degrees-of-freedom (3DOF) and 6DOF comparisons of a variety of vehicles in orbit or in atmospheric maneuvering flight. The prior assessment used Earth-centered cases, while the current study is focused on the cislunar environment. The prior assessment included several complex cases using propulsive forces, active maneuvering, and a full Guidance, Navigation, and Control (GNC) model. For the current assessment, it was desired to avoid repeating complex cases that were already thoroughly examined, and to devise a smaller, focused set of cases that exercise new and unique features of the lunar environment. To maximize the usefulness of the assessment, it is recommended that users implement cases of interest from the previous assessment. The new cases presented here were selected to be representative of the variety of orbital conditions typically encountered during lunar mission simulation. As the Moon lacks an appreciable atmosphere, only different orbits need to be considered, and no atmospheric or aerodynamic modeling is required. This study targets a smaller scope; it is limited to space flight for two reference vehicle models. Some scenarios involve primarily long-duration orbital propagation without maneuvering or use of thrusters.

Similar to the previous effort which involved six NASA simulation tools and one open-source flight simulation tool, this effort involved eight NASA simulation tools. An additional similarity in both efforts is that each participating Center found opportunities to make changes which improved the quality of their simulation tool; thus, it is felt the study “paid for itself.”

As in industry, multiple simulation tools are developed within each NASA Center that are somewhat incompatible in terms of moving a spacecraft model between them. Various simulations provide for cross-comparisons, but programming language, variable naming, model structure, initial condition specification, and frames of reference incompatibilities limit and slows cooperative studies of any particular vehicle model. This effort seeks to compare multiple simulation tools while utilizing a commonly defined set of inputs; sharing models across simulation platforms was not in scope due to the limited time and availability of the team members to implement but some concluding thoughts are offered to improve this more complex problem.

Utilizing benchmarked check-cases improves the simulations being assessed, reduces errors, builds confidence in solutions, and serves to build credibility of simulation results per NASA Standard 7009A, Standard for Models and Simulations (M&S) (Reference 2). This Standard specifies that comparing simulations is a way to validate system performance, this standard states:

> *Once the computational model is available, the next step is empirical validation, which is the comparison of M&S results with a referent (generally, data from either an operational*

simulation, or a 'representative system'). In some instances, e.g., for the development of so-called 'surrogate models', the referent can be the results obtained from a higher-fidelity (and typically computationally expensive) model.

Often, validating a simulation against flight or test data can be challenging given the limited opportunities and resources to conduct flight tests, and demonstrating those tests in the desired flight environments and conditions. Cross-comparisons with different models and simulations with different implementations given the same inputs is a valid domain for simulation validation.

PARTICIPATING SIMULATIONS AND DESCRIPTIONS

Eight NASA simulation teams participated in this assessment. The simulation tools are described below and are developed and maintained from Ames Research Center (ARC), Jet Propulsion Laboratory (JPL), Johnson Space Center (JSC), Langley Research Center (LaRC), and Marshall Space Flight Center (MSFC). While some simulations are available to the public, some require licensing and authorization to use in direct support of NASA missions.

Condor Flight Vehicle Toolkit

Condor Flight Vehicle Toolkit is a flight vehicle simulation tool built using Condor (Reference 3), an open-source software library for the rapid development of mathematical models of complex systems in Python. Condor was developed in the ARC's Systems Analysis Office to address rapidly evolving analysis tasks across a wide range of applications, including conceptual aircraft design performance analysis and robust orbital trajectory design. Model templates are used to define models from different categories, e.g., systems of algebraic equations, table lookups, optimization problems, and trajectory analyses of systems of ordinary differential equations (ODEs) with events. These models can be used as building blocks to assemble more complex system models.

Condor was written to be modular, allowing alternate symbolic backends or solvers to be used with minimal additional effort. An emphasis on employing off-the-shelf, third -party, open-source software packages (widely used in Python's scientific computing and machine learning communities) reduces the development burden of the Condor team while leveraging the open-source software community's inherent advantages of efficient computing performance and verification and validation histories.

Dynamics Simulator for Entry, Descent, and Surface Landing (DSENDS-DARTS)

DSENDS – DARTS (Dynamics Algorithms for Real Time Simulation) is a multi-mission flight dynamics and simulation tool for closed-loop flight dynamics and atmospheric Entry, Descent, and Landing simulations. DSENDS-DARTS belongs to the family of spacecraft and robotics system simulation tools developed by the DARTS Lab (Reference 10) at NASA's JPL.

DSENDS-DARTS is part of the larger DARTS simulator family which provides a wide domain of applications including autonomous robots, ground vehicles (ROAMS-DARTS), and rotorcraft (HeliCAT-DARTS). This family of tools builds upon middleware like the DARTS rigid/flexible multibody dynamics engine and the DARTS Shell (Dshell) simulation framework. All DARTS simulators use C++ for speed but feature a rich and user-friendly Python interface. DSENDS-DARTS used the Dormand-Prince adaptive step size Runge-Kutta 45 integrator, absolute and relative error tolerances were used to control integration error instead of simulation frame rate.

GeneraLized Aerospace Simulation in Simulink (GLASS)

The GeneraLized Aerospace Simulation in Simulink (GLASS) is a 6DOF vehicle simulation developed at MSFC. GLASS is designed to be a modern, modular, user-friendly, and efficient

6DOF simulation tool for aerospace vehicles and is built within the MathWorks MATLAB® and Simulink environment.

The HLS GNC Insight team uses GLASS to test and validate GNC algorithms for the contractors working the HLS program. GLASS has modeled multiple vehicles and multiple GNC algorithms within the framework, showing its flexibility. GLASS has also been validated against the Earth-based NESC 6DOF Orbital check-cases prior to this new study. All the lunar check-cases were exercised in GLASS utilizing an Runge-Kutta fourth-order integration scheme (ode4) with 0.1-sec time steps.

JSC Engineering Orbital Dynamics (JEOD) Software Package

The Johnson Space Center (JSC) Engineering Orbital Dynamics (JEOD) Software Package (Reference 15) is an open-source simulation tool designed to work with the NASA Trick Simulation Environment that provides vehicle trajectory generation by the solution of a set of numerical dynamical models. These models are subdivided into four categories: environment models representing the conditions surrounding the vehicle; dynamics models for integrating the equations of motion; interaction models representing vehicle interactions with the environment; and a set of mathematical and orbital dynamics utility models. JEOD is designed to simulate spacecraft trajectories in flight regimes ranging from low Earth orbit to lunar operations, interplanetary trajectories, and other deep space missions. JEOD can be used to simulate a stand-alone spacecraft trajectory and attitude state, or it can be interfaced with a larger simulation space, e.g., coupling with spacecraft effectors and guidance, navigation, and control GNC systems. More than one spacecraft can be simulated about one central body or separate spacecraft about separate central bodies.

Langley Standard Real-Time Simulation (LaSRS++)

The Langley Standard Real-Time Simulation in C++ (LaSRS++) is an object-oriented framework for construction of aerospace vehicle simulations. The Simulation Development and Analysis Branch (Reference 16) uses LaSRS++ simulations to support desktop analysis, hardware in-the-loop simulations, and high-fidelity, human-in-the-loop simulators. Projects using LaSRS++ have modeled planetary landers, crewed spacecraft, launch vehicles, RPOD, planetary aircraft, advanced concept aircraft, commercial transport aircraft, military fighters, and unmanned aerial vehicles.

LaSRS++ was employed for the orbital check-cases. In all cases, the integration methods used were the same, but customized depending upon which state was being propagated. The selected LaSRS++ frame rate of 1000 Hz minimized the integration error differences between the LaSRS++ solution-data and the truth dataset for the first check-case. That frame rate was used for all ensuing check-cases. (LaSRS++ simulations used with human-in-the-loop real-time simulations are normally operated with frame rates of 100 Hz or less.)

Marshall Aerospace Vehicle Representation in C (MAVERIC)

Marshall Aerospace Vehicle Representation in C (MAVERIC) is a low-to-high fidelity 3DOF/6DOF vehicle flight simulation program developed at MSFC, written primarily in the C and C++ programming languages. MAVERIC was designed to be generic, and data driven and can provide for the rapid development of end-to-end vehicle flight simulations for a variety of launch or on-orbit scenarios. MAVERIC vehicle simulation models are layered upon a set of foundational software called TFrames. TFrames is a time-based differential equation solver environment. TFrames provides an environment for developing a dynamic simulation that insulates the simulation developer from the programming details associated with numerical integration, discrete data sampling, table look-ups, etc. High-level routines provide convenient interfaces between the simulation code and the numerical integration engine. MAVERIC was developed with data structures, code, and interfaces for standard vehicle components and is designed to easily incorporate and

interface with customized models. MAVERIC has been in active development at MSFC since the mid-1990s and has been used to model over a dozen different aerospace vehicles. Currently, MAVERIC is the primary 6-DOF simulation for the SLS ascent phase and GNC design.

Program to Optimize Simulated Trajectories II (POST2)

POST2 is a generalized 3/6/multi-DOF event-based trajectory simulation software that is descended from the original POST developed by Martin Marietta in 1970. POST2 provides the capability to simulate, target, and optimize point mass trajectories for multiple powered or un-powered vehicles near an arbitrarily rotating, oblate planet. POST2 has been used successfully to solve a wide variety of atmospheric ascent and re-entry problems, as well as exo-atmospheric orbital transfer problems. This flexible simulation capability is augmented by an efficient, discrete parameter-optimization capability that includes equality and inequality constraints.

POST2 include many generalized models for atmosphere, gravity, and propulsion that are used to simulate a wide variety of launch, orbital, and entry missions. Users may also provide custom models of varying fidelity, including flight software. POST2 can simulate multiple vehicles in a single simulation, each with independently defined environments, attracting body characteristics, and flight software. Thus, each vehicle can have its own GN&C system for completely independent, on-board autonomy. Additionally, effects of multi-body and interaction forces that depend on the relationship of one vehicle to another can be included. POST2 also supports 3DOF and 6DOF trajectories within the same simulation. POST2 has extensive flight heritage, and its core codebase is considered flight-validated via pre-and post-flight analysis of dozens of Earth and Mars EDL missions and ground/flight tests. For the lunar check-case effort, POST2 used a fourth-order Runge-Kutta integration method at 100 Hz.

Space Transportation and Aeronautics Research Simulation (STARS)

STARS is MATLAB/Simulink-based air, launch, and space vehicle dynamics simulation and GNC design software. STARS takes advantage of MATLAB/Simulink capability and flexibility to make the creation of the simulation and GNC design much faster. STARS enables the time domain simulations of the vehicle dynamics with the GNC system to be conducted. In addition, the Bode and Nichols plots of the frequency domain models can be obtained in STARS to conduct GNC design.

STARS has been used for the SLS vehicle dynamics simulation and GNC design. SLS STARS is aimed to predict the overall vehicle dynamics of the SLS launch vehicle stack, including rigid-body and flexible modes, as well as propellant slosh dynamics, dynamics associated with the engine actuation and gimballing, and sensor dynamics. Additionally, interactions between many of these effects are also modeled. All these effects are modeled in SLS STARS, and many of them can be enabled or disabled for a given run.

OVERVIEW OF CHECK-CASES

The basic cases were selected to stress environment models, numerical integration, coordinate frames and simple reference models. They follow a build up-and-out fashion, starting from a simple Keplerian low lunar orbit, then adding effects e.g., a detailed gravitational field and third-body perturbations. Typically, a minimal implementation of additional effects is used to best permit future simulation comparisons. This section gives an overview of the check-cases, including the reasoning for including each check-case. Implementation details can be found in the NASA Technical Memorandum (Reference 17).

The first cases (1, 2, and 3) use a near-circular low lunar orbit with a high inclination. The specific orbit was chosen to be like orbits that the participating simulation teams are regularly using

in support of the Artemis campaign Program to land near the lunar south pole. Case 1 is a simple Keplerian propagation, which allows comparison with a known analytical solution, and is referenced in Appendix B of Reference 17. Case 2 uses the Gravity Recovery and Interior Laboratory (GRAIL) gravity model to order and degree 8 (8x8 GRAIL), allowing testing of the actual gravitational model most lunar simulations will use. Case 3 uses a high-fidelity GRAIL gravity model to order and degree 320 (320x320 GRAIL), allowing testing of ingesting and using a complex gravitational model. Case 1 was created explicitly from the desire to exactly match the known analytical solution data at the start time. Case 1 is the only cases where simulations can be compared directly with the closed form solution.

The 8x8 GRAIL model is used for all cases after Case 3. For actual lunar simulations, a higher-order GRAIL model is generally recommended, with considerations made for the length of the simulation time and proximity to the Moon. However, for the purposes of this assessment, the 8x8 model is used to best allow simulation-to-simulation comparisons. As will be seen, simulation-to-simulation differences naturally grow as higher order models are implemented, so an 8x8 model ensures that a simulation is successfully implementing the GRAIL model while minimizing differences due to the GRAIL model itself. Only the spherical harmonic gravity model is used in this assessment. Detailed simulation of local terrain may require a different type of gravity model, for example one based on mass distribution. This assessment is limited to cases where a spherical harmonic model is appropriate.

Cases 4 and 5 use a near-circular high lunar orbit with an altitude of approximately 500 km. This is the altitude where perturbations from the Earth start to become important. Case 4 uses no third-body perturbations, establishing a baseline before third-body effects are introduced. Case 5 uses perturbations from the Earth and the Sun, allowing testing of these perturbation effects.

Only Earth and Sun perturbations are used in this assessment, as they are the two dominant perturbation effects. Although most simulations will have several third-body perturbations available, limiting the study to Earth and Sun is sufficient to test the implementation and minimizes the risk that a given simulation does not have a particular third-body perturbation available.

Cases 6 and 7 introduce a highly elliptical orbit, with initial perilune of approximately 250 km and initial apolune of approximately 9,385 km. These cases include a range of dynamics, including apolune with higher orbital velocity where the GRAIL model is most important, to perilune with lower orbital velocity where the third-body perturbations are at their most significant.

Case 8 utilizes a Near-Rectilinear Halo Orbit (NRHO) that is of interest for the Artemis program and presents interesting astrodynamics given the nature of this orbit as a periodic quasi-stable family of orbits around the Earth-Moon Lagrange points (Reference 13). The NRHO gets to within an altitude of 2000 km at closest approach, to nearly 70,000 km at its furthest point. The NRHO is well beyond the regime of two-body astrodynamics and approximations, and alternate orbital elements may need to be utilized.

Finally, Case 9 is a nearly polar orbit of similar size to the low lunar orbit. The two most common coordinate systems, Mean Earth (ME) and Principal Axis (PA), have slightly different equatorial planes, therefore the inclination of an orbiting vehicle is different in each system. The purpose of the polar orbit is to be near the origin of the GRAIL model, which is expressed in the PA system. This case is nearly polar in the PA frame.

Two different vehicle models were defined for the check-cases. The first vehicle is a simple cylinder, which matches the cylinder model in the original check-cases. The cylinder is relatively easy to implement and tests basic rotational dynamics and modeling. The second is the Apollo Lunar Module spacecraft, drawn from documented Apollo mass properties. The Apollo spacecraft

is asymmetric and has off-diagonal moment of inertia terms. The Apollo spacecraft tests more complex dynamics and provides a means to check sign conventions and coordinate systems.

Typically, the vehicle has an initial pitch rate that is matched to the orbital period. For circular orbits, the flight-path angle remains approximately constant, while for elliptical orbits the flight-path angle will deviate before returning after one orbit. Certain cases use other initial rates, as described in the following paragraphs.

Case 5 tests a high circular with third-body perturbations and includes the typical pitch rate for approximately constant flight-path angle. A case that tumbles about all three axes was desired, so Case 5A repeats Case 5 with initial rotation rates along all three axes. The additional rotations were added to expand the tests to include testing of signs on cross products of inertia.

Case 6 tests a highly elliptical orbit. The initial pitch rate is selected to match the orbital period, however since the orbit is elliptical, the flight-path angle nods up and down during one orbit with respect to the VO frame. A case that includes no inertial rotations was desired, so Case 6A repeats Case 6 with zero inertial rotation. Most simulations will likely propagate in an inertial frame, so Case 6A should, in effect, test integrating zero. In practice, simulation details vary for high-fidelity simulations, so very small differences are not unexpected.

Case 8 tests the NRHO, which is of great interest for the Artemis program. The NRHO is nearly stable, but spacecraft will eventually escape without station-keeping maneuvers. To explore the sensitivities of the NRHO, four additional cases were defined. Case 8A initializes at a true anomaly of 180 degrees, after approximately half of the orbital period. Case 8B initializes at a true anomaly of 0 degrees, after nearly one orbital period. Case 8C perturbs the initial conditions of Case 8 by adding 10 meters to the initial position vector. Case 8D perturbs the initial conditions of Case 8 by adding 0.1 m/s to the initial velocity vector.

Case 9 tests the polar orbit. Cases with a sensor station offset from the center-of-mass (CoM) were desired, so two sensor stations were defined. Case 9 uses a sensor station that is offset only in the body X-direction. Case 9A and 9B use a sensor station that is offset from the CoM in all three body directions. Additionally, Case 9B introduces an active moment profile. A moment profile was defined to apply to various combinations of body axes and specified as both a table and with pseudo-code.

In addition to the sensor stations, a case that helps identify any potential errors in implementation of the two planet-fixed frames (PA vs. ME) and a case that ingests a digital elevation model (DEM) was desired. Case 9 includes two test points, one fixed and one that moves as a function of time. Both test points contain outputs focused on planet-fixed frame differences. For the specified DEM, altitude, latitude, and longitude are output for the moving second test point, testing the ingestion of the DEM.

Cases 8 and 9 specify the vehicle orientation differently than the others. Cases 1 to 7 all specify the initial orientation of the body frame with respect to the VO frame. Cases 8 to 9 were modified to allow more of a focus on pure propagation differences by explicitly stating the initial quaternion with respect to MI to use for the body frame orientation. It is assumed that the underlying equations of motion (EOM) state is the attitude with respect to MI for all simulations. Cases 1 to 7 may include effects from differences in converting from VO to MI relative attitude at t=0 and propagate this difference throughout the run. Cases 8 to 9 should represent a purer comparison of propagation techniques.

Table 1 below summarizes the cases, including the orbit and additional information on the effects under study. Complete definitions of each case can be found in Section 7.6 of Reference 17.

Table 1. Summary of Effects Tested by Each Case

Case	Orbit	Vehicle	Sun/Earth Pertur-bations	Notes
1	LLO ~120x120km	Cylinder	No	Keplerian gravity, permits analytical solution
2	LLO	Cylinder	No	Introduces 8x8 GRAIL
3	LLO	Cylinder	No	320x320 high-fidelity GRAIL
4	HLO (high lunar orbit) ~500x500km	Apollo	No	Introduces Apollo vehicle model
5	HLO	Apollo	Yes	Introduces third-body perturbations
5A	HLO	Apollo	Yes	Body tumbles about all three axes
6	HEO (highly elliptical orbit) ~250x9385km	Cylinder	Yes	Re-visits cylinder model
6A	HEO	Cylinder	Yes	Zero inertial angular rotation
7	HEO	Apollo	Yes	Returns to Apollo vehicle model
8	NRHO	Apollo	Yes	Introduces NRHO orbit (radius ranges from ~2000km to ~70,000km)
8A	NRHO	Apollo	Yes	Re-initializes at a true anomaly of 180 degrees, after approximately one half of the orbital period
8B	NRHO	Apollo	Yes	Re-initializes at a true anomaly of 0 degrees, after approximately one complete orbital period
8C	NRHO	Apollo	Yes	Initial radius perturbed by +10 m relative to Case 8
8D	NRHO	Apollo	Yes	Initial velocity perturbed by +0.1 m/s relative to Case 8
9	LPO (lunar polar orbit) ~120x120km	Apollo	Yes	Polar orbit, includes a sensor station offset from the CoM in one direction; tests DEM ingestion
9A	LPO	Apollo	Yes	Includes sensor station offset from CoM in all three directions
9B	LPO	Apollo	Yes	Includes open-loop moment profile

Other cases considered, but ultimately not done due to insufficient resources and schedule, are listed here for consideration as forward work:

1. Landing (closed loop)
2. Rendezvous (closed loop)
3. Orbit raising (closed loop)
4. Ascent (open loop)
5. Apogee raising (open loop)

GROUND RULES AND ASSUMPTIONS OVERVIEW

All definitions, models and agreements, such as those for time references, use of open-source data, measurement frames, and data transmission formats are further described in Reference 17. There were 17 different scenarios for implementation with 51 simulation output variables for comparison and an additional 31 optional or scenario-specific outputs. This section will describe, at a high level, the types of information required to complete the check-cases. Two standard vehicle reference models were used for the check-cases, based on non-proprietary sources. Vehicles specified for this application are defined by their equivalent body coordinate systems. A simple cylinder of uniform density with mass properties and inertia is shown in Figure 1. The Apollo Lunar Module was also used as a model and shown in Figure 2, all detailed mass properties and inertias are described in Reference 12.

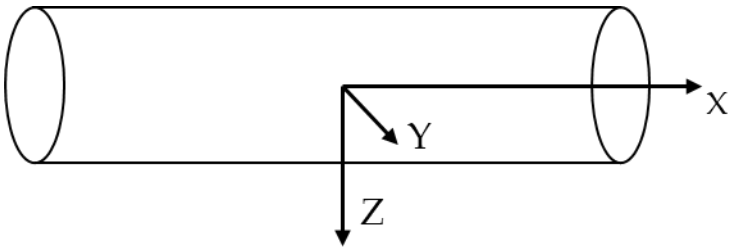

Figure 1. Cylinder Body Frame (not to scale)

Additional models for this assessment are briefly described here. Three Lunar Geodesy Models to describe the way in which the flight simulation tools model the surface of the Moon including a reference spheroid, as seen in Figure 3, and PA and ME planet-fixed frames. The PA and ME frames are unique to the lunar environment and care should be taken with these frames as use of one of the frames is sometimes assumed without explicitly stating which frame is used. This can result in errors that can be easily overlooked due to the small differences in the orientation of the two frames. This small difference in orientation can result in differences on the order of hundreds of meters on the lunar surface.

An additional frame was added for the orientation with respect to a Local-Vertical, Local-Horizontal (LVLH) frame and specifically the Vehicle-Carried Orbit-Defined frame (VO) is an LVLH frame used in this assessment. A singular vehicle body-axis system with axes and origin fixed with respect to a rigid vehicle were also used along with defining a Lunar Polar Stereographic (LPS) specific for the Artemis Missions.

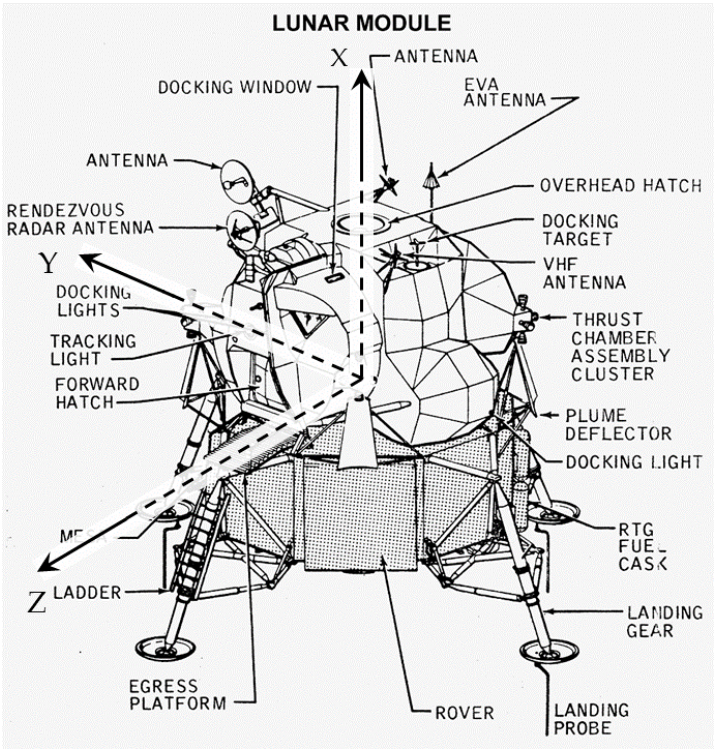

Figure 2. Apollo Lunar Module Body Frame (not to scale)

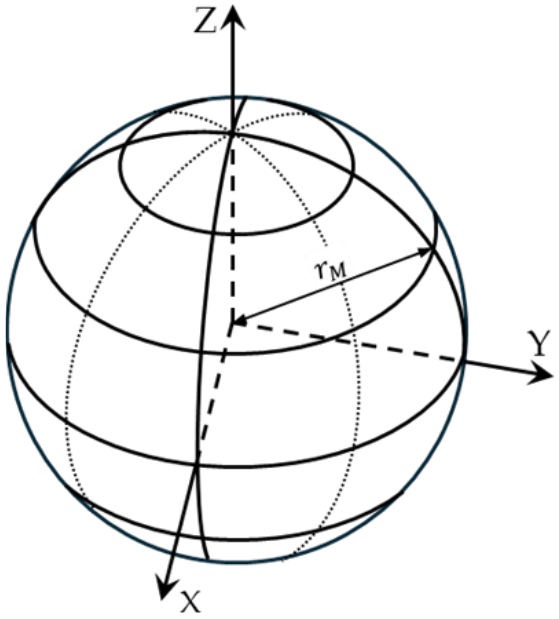

Figure 3. Reference Spheroid

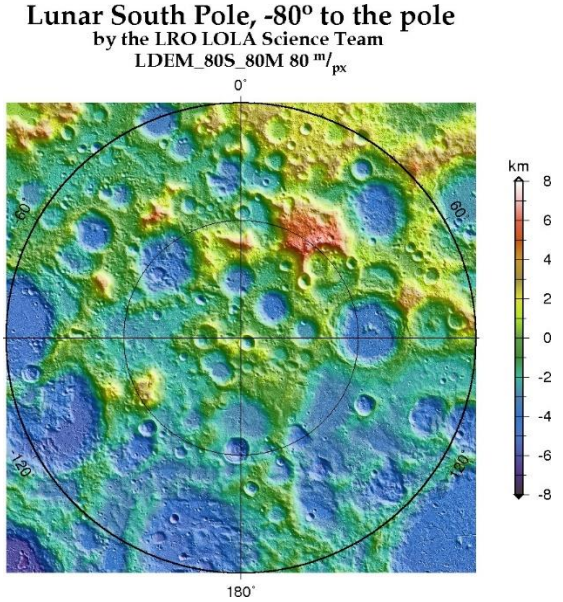

Figure 4. LOLA South Pole GDR, DEM Height Map

OUTPUT SPECIFICATIONS

The simulation output products are briefly described here but with exhaustive details captured in Reference 17. The output variable names followed the AIAA S-119 (Reference 14) naming convention. Simulation outputs are recorded every 60 seconds of elapsed time. Simulation data was requested to be recorded with 16 significant digits of precision into a Comma Separated Values (CSV) output format. The CSV format is broadly used in engineering and maintains all significant digits of precision. However, individual programs may alter data slightly depending on the implementation of a data read-in routine, or when using the copy-paste function. The CSV formatted file produced by the simulation tool is considered the authoritative output file.

For outputs of the transformations between frames, quaternions were ultimately chosen as the primary output product, along with optional Euler angle outputs, and post-processed Euler angles. Approximately half the participating simulations use left transformative and half use right transformative quaternions. Texts and the general literature most commonly use right transformative quaternions. Ultimately, the right transformative quaternions were chosen for the simulation output products.

Common Output Variables included variations in time, epoch, state position in XYZ, velocity in XYZ and Acceleration in XYZ components. Angular rates, angular accelerations, positions of center of mass relative to various frames, and positions in various coordinate frames were produced. Quaternions, altitudes, and a variety of orbital elements applicable to two and three body class orbits.

SIMULATION OUTPUT COMPARISON

A unique interactive web-based tool was developed as a means for the simulation groups to perform quick data comparison using interactive plots, access scenario specifications, and catalogue the results. This website is being migrated to a new public website under the NASA NESC Academy and will be published along with the data files and description of this report in Reference 17. The website was continuously developed throughout the assessment and was accessible to anyone with NASA Virtual Private Network. Each scenario had its own webpage that included a

scenario description, specification table, a list of the latest results, and interactive plots. With all cases and variables considered, several hundred comparison plots were generated.

The interactive plots were developed using the `plotly.js` JavaScript plotting library and were a valuable resource that avoided the need for sim groups to develop their own plotting routines for comparison to other submitted results. Through the web interface the user had the option between time histories and difference plots with respect to an average of all sim group results or an individual sim group. By default, all sim results were displayed on the plots, but the user had the option to select sims to omit. Interactive plot options included zooming to regions of the plot, hovering over the plot with the cursor to display the value at any point along each series, and hiding or showing specific plot traces. Most scenario webpages had three plots, and each could draw any of the defined output variables associated with the scenario as selected by the user. A feature was added to enable anyone with access to the website to load temporary data to compare to the existing delivered results. This allowed sim groups to compare newly generated results to the other sim groups without the need to provide a more formal delivery. It also gave the team the opportunity to compare results between difference scenarios. For example, data from Scenario 8 could be uploaded to any of the NRHO sensitivity subcases to evaluate the sensitivity inherit in a full NRHO period. A feature was added to the interactive plotting website to optionally highlight a 'family' of plot traces detected by a K-means clustering algorithm. The K-means algorithm iterates to identify clusters of similar data based on relative distance and variance of the data provided. Clusters of traces are identified using the final points in each plot trace and a primary 'family' is determined based on the largest cluster group. Once a 'family' is found, a shaded region is drawn based on the min and max values of the traces in the group. Quaternion outputs were modified in post processing by forcing the scalar element of the quaternion to always be positive, inverting the signs on the vector component as necessary. Consistent signs on the plotted quaternion elements across the sim traces made them easier to compare.

SIMULATION RESULTS INTERPRETATION

The primary output of the check-cases is a time history of each output variable, which can then be plotted with any data plotting software. For simulation comparison, the results from multiple simulations are plotted together. Most of the output data for different simulations looks to be nearly identical when plotted at the full scale of the data. A second type of plot, showing the difference between each simulation's outputs and the average value of that output, is typically more useful for comparing and contrasting data.

The simulations are presented anonymously, simply labeled SIM1 through SIM8. This ordering is independent of the order of the simulation descriptions. Appendix A of Reference 17 includes a wealth of output plots from the participating simulations. Both the full-scale outputs and difference plots are shown. The difference plots are scaled to easily show the differences on the plot regardless of the scale. The interactive website described in the previous section, *Simulation Output Comparison Website*, provides additional options to compare simulation data. Shown below are a series of figures to show the reader typical outputs produced, Figure 6-12 are examples of the comparisons and results obtained from specific cases described earlier in this paper. Several of the plots still show unresolved differences. While not all differences are described in this paper or Reference 17, possible explanations are offered below.

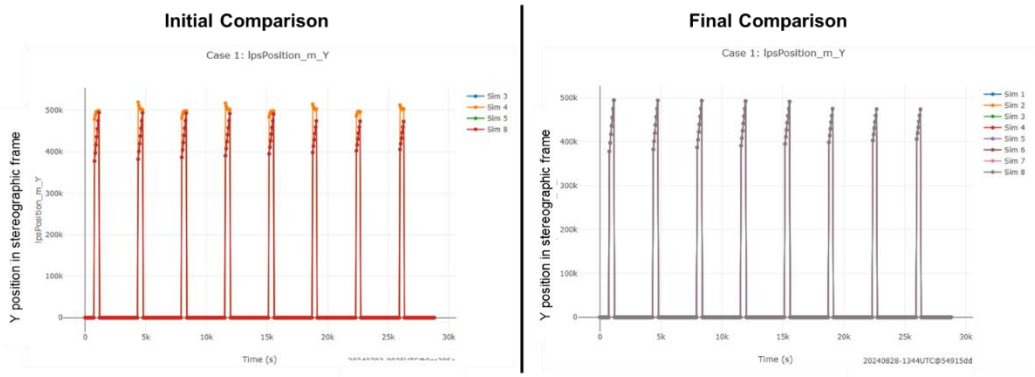

Figure 5. Ownship Position in Lunar Polar Stereographic Coordinates Initial and Final Comparisons

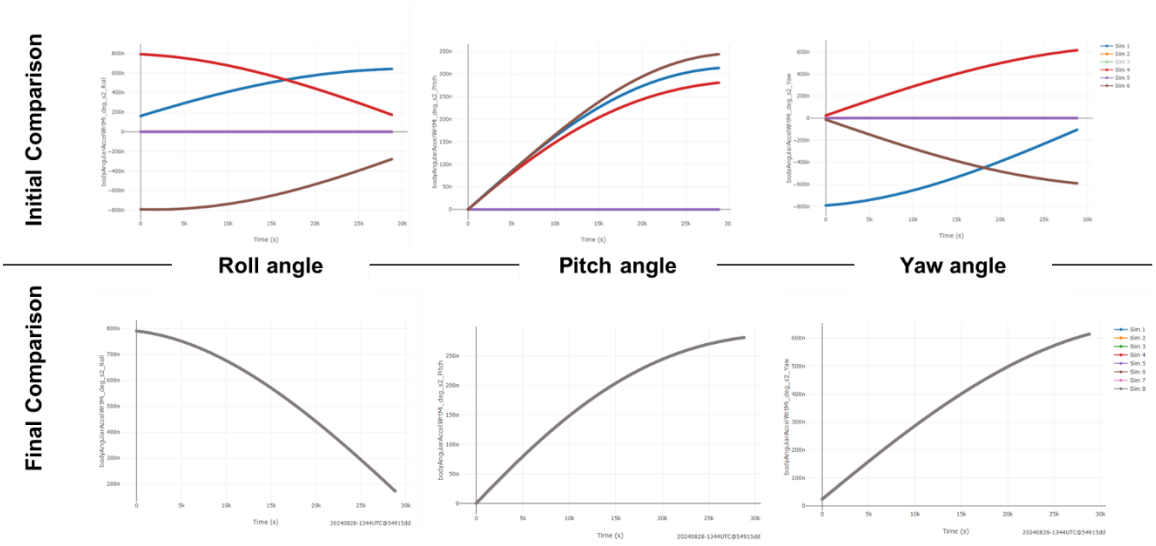

Figure 6. Case 4 (HLO, no third-body effects) Euler Angles Initial and Final Comparisons

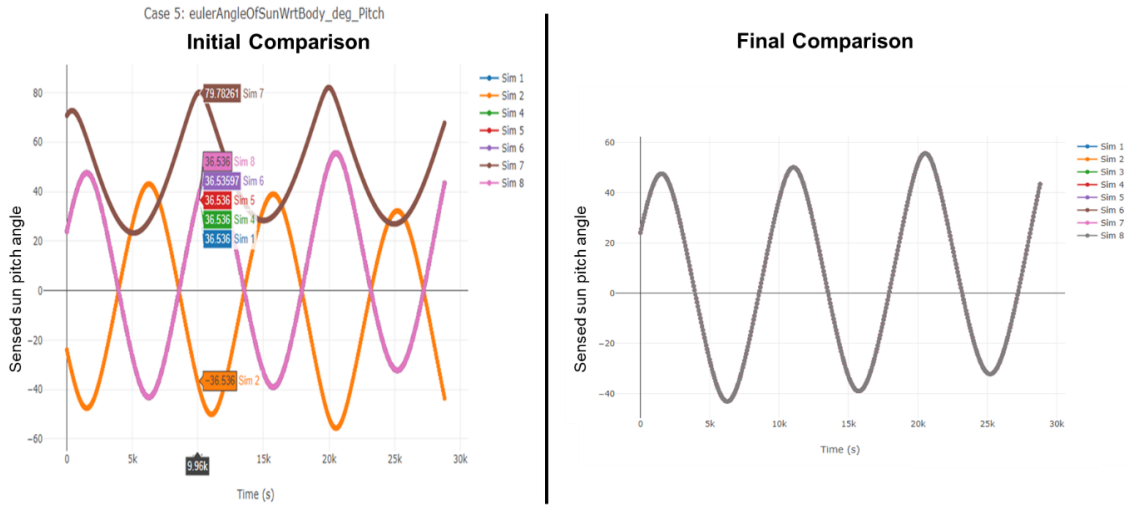

Figure 7. Case 5 (HLO) Sun-Pointing Angle (pitch component) Regarding Vehicle Frame Initial and Final Comparisons

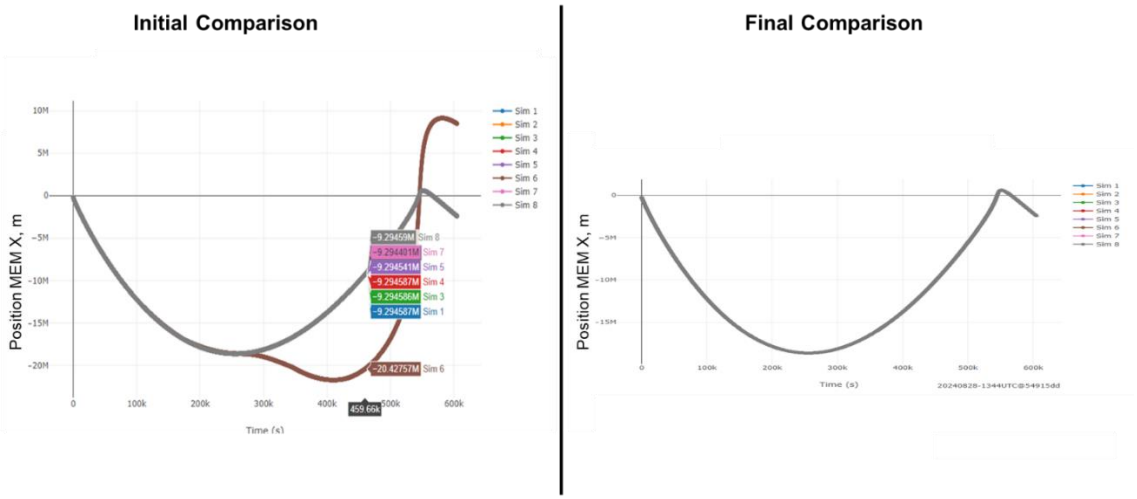

Figure 8. (NRHO) Position Regarding Moon-Earth X Axis Initial and Final Comparisons

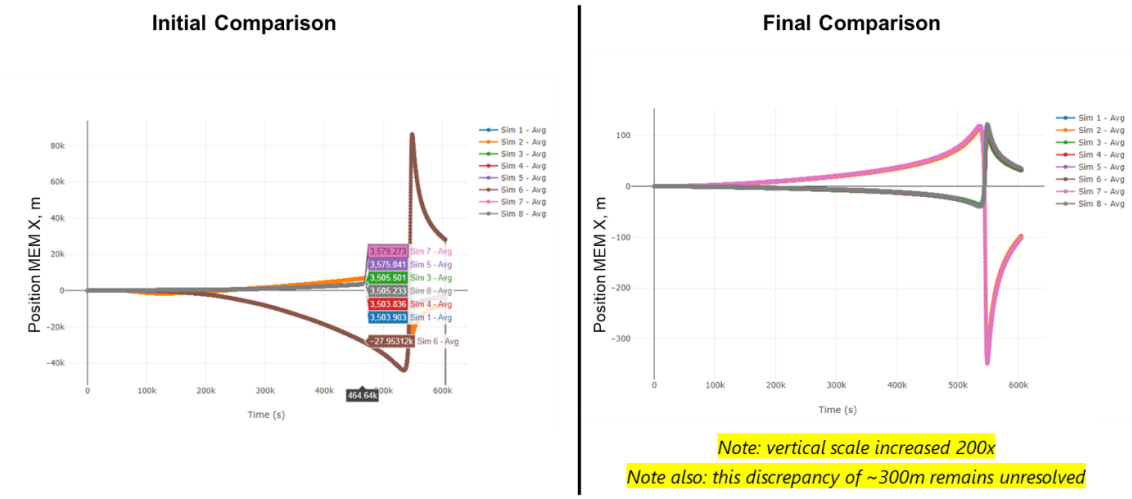

Figure 9. Case 8 (NRHO) Position Regarding Moon-Earth Z Axis Initial and Final Comparisons

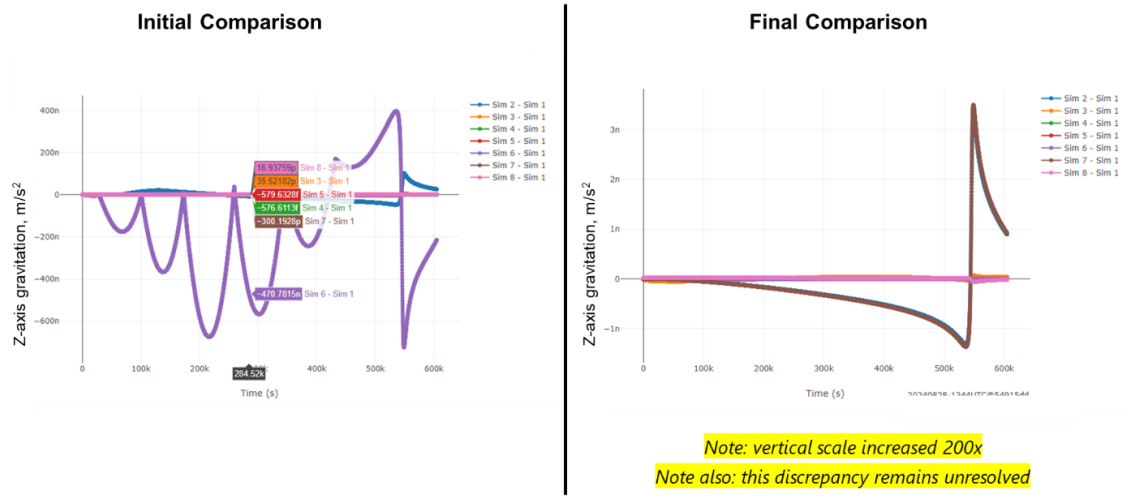

Figure 10. Case 8 (NRHO) Gravitation in Moon-Earth Z Axis Initial and Final Comparisons

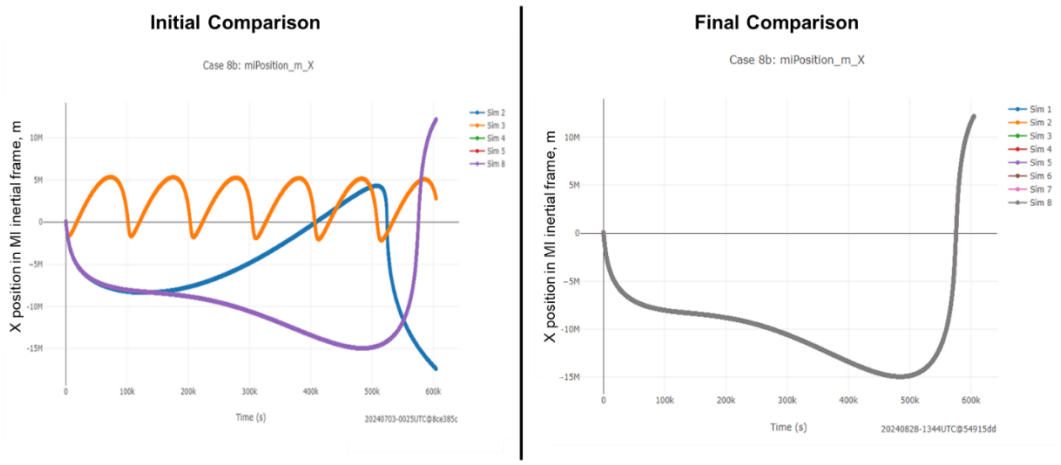

Figure 11. Case 8b (NRHO) Inertial X Position in Moon Inertial Frame Initial and Final Comparisons

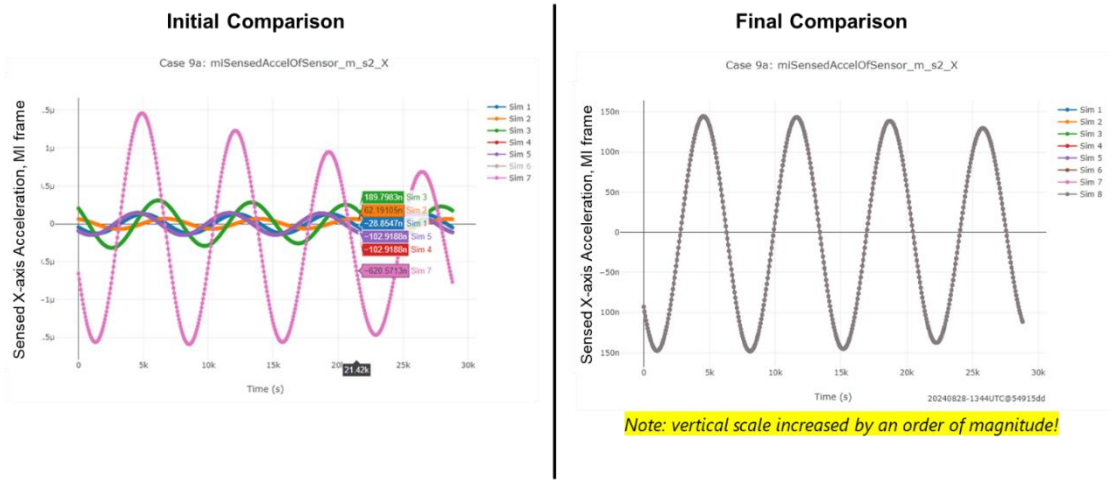

Figure 12. Case 9a (LPO) Sensed X-body Axis Acceleration Regarding Moon Inertial Frame Initial and Final Comparisons

The simulation outputs were compared for all plots. It was difficult to determine a matching criterion due to the intentional different purposes each simulation was developed to meet and their underlying differences in implementation of algorithms and numerical integration. Attempts were made to specify clear and unambiguous ground rules, assumptions, definitions, frames, sources of data and other relevant information. However, it is possible that some definitions remained unclear or misinterpreted as several plots reveal significant differences. These differences highlight the importance of continuous communication and documenting all aspects of the modeling and simulation process. NASA STD 7009 (Reference 2) states that uncertainty quantification is: *The process of identifying all relevant sources of uncertainties, characterizing them in all models, experiments, and comparisons of M&S results and experiments, and of quantifying uncertainties in all relevant inputs and outputs of the simulation or experiment.* It was beyond the scope and resources of this assessment to characterize the sources of all uncertainties and comparisons to show favorable comparisons. However, Section 4.1.3 of Reference 2 indicates: Programs must determine: *the critical decisions to be addressed with M&S and to determine which M&S are in scope.* Part of the acceptance criteria to use a Model or Simulation includes specification *of what constitutes a favorable*

comparison for the Verification Evidence, Validation Evidence, Input Pedigree Evidence and Use History level definitions in the Credibility Assessment Scale. While this assessment declined to assert a criterion for what is 'in family', the comparisons plots were mostly favorable across the simulations for many of the outputs. A feature was added to the interactive website describing simulations that were 'within family' as previously described. While no one simulation was described as 'truth', this feature is helpful to new simulation developers as a check if implementation of the check-cases matched other known sims. It is left as an exercise for the Simulation and Model provider and Program Office to determine the 'criteria' for favorable comparison to achieve the Verification and Validation goals specified.

Several plots show remaining differences after these efforts to resolve ambiguous inputs or scenario descriptions. Four important sources of potential differences are described next.

During the check-cases, comparison of the VO frame data showed up often as a source of differences. Differences were typically masked by the circular or nearly circular cases but became apparently in the VO frame output data for the elliptical orbits. This occurred due to two 'standard' definitions of VO or LVLH frames, one as used in this assessment where the X axis is in the direction of the velocity vector, and another where the X axis is coincident with the velocity vector. By the end of the assessment, it appeared the differences in this frame definition were resolved.

When originally comparing NRHO third-body gravitational acceleration differences, a difference in simulation was tracked down to at least one simulation (sim 8) including an 'indirect oblateness acceleration' term for the Earth gravitational acceleration contribution. This difference thus applies to all scenarios with third-body gravitational acceleration terms. No corrective action was taken. When comparing simulation time data, the sim teams discovered that at least two of the simulations (sims 7 and 8), advanced simulation time in the TDB time frame, and thus including relativistic effects, whereas most of the simulations advanced time in the UTC/TT time frame. No corrective action was taken.

The last important difference noted was for the DEM outputs from Case 9 specifically in an altitude parameter. The DEM output data was added late in the assessment and little time was available to the simulation teams to analyze and resolve differences, especially considering that some of the simulation tools had to implement new routines to handle the new output data. When originally added, it was noted that differences would likely be observed due to differences in interpolation routines. The time zero location of the test point was chosen explicitly to be a high sloped region of the south pole, and examining the plots, it is observed that indeed this is a location with some of the largest differences between the simulations.

CONCLUSION

The result of this effort was intended to provide external simulation developers a set of reasonable Lunar check-cases to support simulation validation and identify common areas where development 'bugs' can occur. Overall, simulation teams improved their tools, added new capabilities, found errors in implementation, and gained an improved understanding of the details of their simulation from participating in this activity. The external site for simulation results and output upload will be maintained under the NASA Flight Mechanics NESC Academy, https://nescacademy.nasa.gov/ repository as a useful resource for simulation developers (Reference 17).

ACKNOWLEDGMENTS

This effort required the use of extensive teams representing each simulation, developers and consultants. We would like to thank James Gentile, Marcus Lobbia, Abhinandan Jain, Nicholas

Olson, Alejandro Pensado, Wei Lin, John Penn, Curtis Zimmerman, Gregory Dukeman, Rafael Lugo, Juan Orphee, Jeremy Shidner and the HLS Leadership Team for supporting this work.

REFERENCES

1. NASA/Technical Memorandum-2015-218675: "Check-Cases for Verification of 6-Degree-of-Freedom Flight Vehicle Simulations
2. NASA Standard 7009A, *Standard for Models and Simulations*, 07-13-2016
3. Margolis, B. W. L., and Lyons, K. R.,"Condor," *New Technology Report ARC-18996-1*, NASA Ames Research Center, Nov. 2023..
4. B. W. L. Margolis, "SimuPy: A Python framework for modeling and simulating dynamical systems," *Journal of Open Source Software*, vol. 2, p. 396, September 2017.
5. B. W. L. Margolis and K. R. Lyons, "SimuPy Flight Vehicle Toolkit," *Journal of Open Source Software*, vol. 7, p. 4299, July 2022.
6. J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, p. 1–36, 2019.
7. A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker and C. S. Woodward, "SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers," *ACM Transactions on Mathematical Software (TOMS)*, vol. 31, p. 363–396, 2005.
8. E. Hairer, S. P. Norsett and G. Wanner, *Solving Ordinary Differential Equations* i. Nonstiff Problems, Springer Berlin Heidelberg, 1993.
9. B. W. L. Margolis, "A Sweeping Gradient Method for Ordinary Differential Equations with Events," *Journal of Optimization Theory and Applications*, vol. 199, p. 600–638, October 2023.
10. Dynamics Algorithms for Real Time Simulation (DARTS) Lab web site https://dartslab.jpl.nasa.gov.
11. Cameron, J.; Jain, A.; Burkhart, D.; Bailey, E.; Balaram, B.; Bonfiglio, E.; Grip, H; Ivanov, M.; Sklyanskiy, E: DSENDS: Multi-mission Flight Dynamics Simulator for NASA Missions. In *Proceedings of the AIAA SPACE Conference*, pp. 5421, Long Beach, CA, Sep. 2016.
12. National Aeronautics and Space Administration. 1971. "*CSM/LM Spacecraft Operational Data Book. Volume III: Mass Properties. Revision 3*" Pg. 2-15 and Pg. 3.1-10. April 1, 1971
13. Whitley, Ryan J. et al. "Earth-Moon Near Rectilinear Halo and Butterfly Orbits for Lunar Surface Exploration." AAS 18-406.
14. *Flight Dynamics Model Exchange Standard* (ANSI/AIAA S-119-2011)
15. JSC Engineering Orbital Dynamics (JEOD), JEOD GitHub Site: https://github.com/nasa/jeod, September 2023
16. NASA Langley Research Center Flight Simulation Facilities https://flightsimulation.larc.nasa.gov
17. NASA/Technical Memorandum-2024- TBD: "Expansion of Check-Cases for 6DOF Simulation"