The Space Operations Simulation Center (SOSC) and Closed-Loop Hardware Testing for Orion Rendezvous System Design

Zoran Milenkovic
C. S. Draper Laboratory, Houston, TX

Christopher D'Souza
Johnson Space Center, Houston, TX

Zachary Wilson, David Huish, John Bendle and Angela Kibler
Lockheed-Martin, Denver CO

Introduction

The exploration goals of Orion / MPCV Project will require a mature Rendezvous, Proximity Operations and Docking (RPOD) capability. Ground testing autonomous docking with a next-generation sensor such as the Vision Navigation Sensor (VNS) is a critical step along the path of ensuring successful execution of autonomous RPOD for Orion.

This paper will discuss the testing rationale, the test configuration, the test limitations and the results obtained from tests that have been performed at the Lockheed Martin Space Operations Simulation Center (SOSC) to evaluate and mature the Orion RPOD system. We will show that these tests have greatly increased the confidence in the maturity of the Orion RPOD design, reduced some of the latent risks and in doing so validated the design philosophy of the Orion RPOD system.

This paper is organized as follows: first, the objectives of the test are given. Descriptions of the SOSC facility, and the Orion RPOD system and associated components follow. The details of the test configuration of the components in question are presented prior to discussing preliminary results of the tests. The paper concludes with closing comments.

Objectives of the Orion Testing in the SOSC

The initial objectives for Orion testing in the SOSC were two-fold: first to characterize the facility resources (motion control, target environment, lighting, etc.) as providing a space-like environment for future Orion testing, and second to develop a core capability to execute a closed-loop Orion rendezvous and docking simulation in the SOSC. The focus of this paper will be on the latter objective, closed-loop rendezvous capability within the test facility. The first objective is discussed extensively by Christian et alia [1].

The main objective of these latter tests were to demonstrate the effectiveness of the VNS operating in a closed-loop with other elements of the RPOD system, including the navigation filter, the guidance system, and the control system. These tests were limited to the range of 15 m to dock with ISS (subsequent tests will evaluate the system from 60 m to dock). In addition, off-axis tests were conducted; these tests evaluated the ability of the system to handle trajectory dispersions.

The Orion program conducted testing in the SOSC in December 2011 in order to execute a closed-loop rendezvous simulation using Guidance, Navigation, and Control (GN&C) flight software (FSW) algorithms developed by the Orbit MODE Team (OMT). The test included executing the GN&C FSW algorithms closed-loop with the VNS unit that was flown on the Space Shuttle (STS-134) Development Test Objective (DTO) earlier in the year in STORRM (Sensor Test for Orion Relative navigation Risk Mitigation). While the DTO tested performance of the Orion Relative Navigation (RelNav) hardware in a space environment, it did not test the overall performance of the Orion RelNav system. The purpose of testing in the SOSC facility was to take the DTO one step further and characterize the performance of the entire Orion RelNav system in various Orion docking profiles. This round of tests was only the beginning of what will be required to develop rendezvous capability for the Orion vehicle, but it gave the team the opportunity to integrate the software and hardware components of the RelNav system for the first time.

The testing to date has demonstrated that the proximity operations and docking phase of the mission, which has been tested in the SOSC, is rapidly maturing. Extensive tests have increased confidence in the functionality, performance and robustness of the design.

Description of the SOSC

The Space Operations Simulation Center at the Lockheed Martin (LM) Waterton Campus in Littleton, Colorado is a dynamic test environment focused on Autonomous Rendezvous and Docking (AR&D) development testing and risk reduction activities. The SOSC supports multiple program pursuits and accommodates testing GN&C algorithms, hardware testing and characterization, as well as software and test process development.

The focal component of the SOSC during this test was the rail-mounted 6DOF robot that has a range of linear motion of 60 meters longitudinally, and 15.2 meters in the lateral and vertical directions each. The robot can translate, and rotate about all three axes via commands from the robot control station. The commands to the robot mechanism are given in facility coordinates.

Additionally, the facility supports several mock-ups to aid in simulating real-world rendezvous and docking scenarios: the International Space Station (ISS) mock-up including the APAS-To-LIDS Adapter System (ATLAS), the Low-Impact Docking System (LIDS) mock-up, a full-size, lightweight mock-up of the Orion Crew Module (CM), and an asteroid mock-up. The ISS mock-up is a static target that has a fixed location within the SOSC high bay, whereas the Orion CM mock-up is a mobile unit that can be mounted to the large robot mechanism when real-world simulations are desired and the asteroid mock-up can be mounted on the fixed base robot to simulate a moving body. The ATLAS is integrated as part of the ISS mock-up when required and is the adapter that allows the LIDS to be deployed and useful as part of the ISS docking system. The ISS mock-up is composed of three modules representing a portion of the ISS: the National Aeronautics and Space Administration (NASA) Harmony module (Node 2), the

European Space Agency (ESA) Columbus module, and the Japan Aerospace Exploration Agency (JAXA) Kibo module. In addition, the Harmony module has a docking port attachment called the Pressurized Mating Adapter (PMA-2) with a LIDAR target mounted at its center.

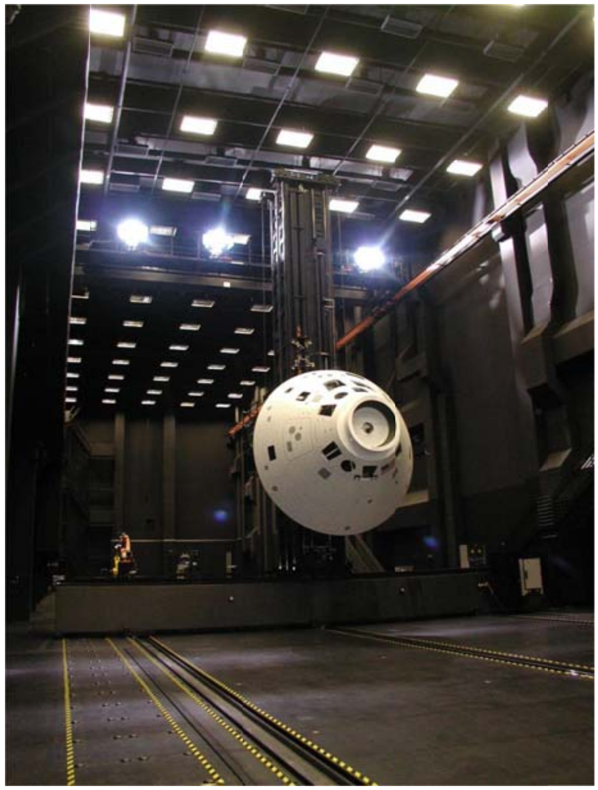


Figure 1: Large Robotic Mechanism (with existing Orion Crew Module mounted) located in SOSC High Bay

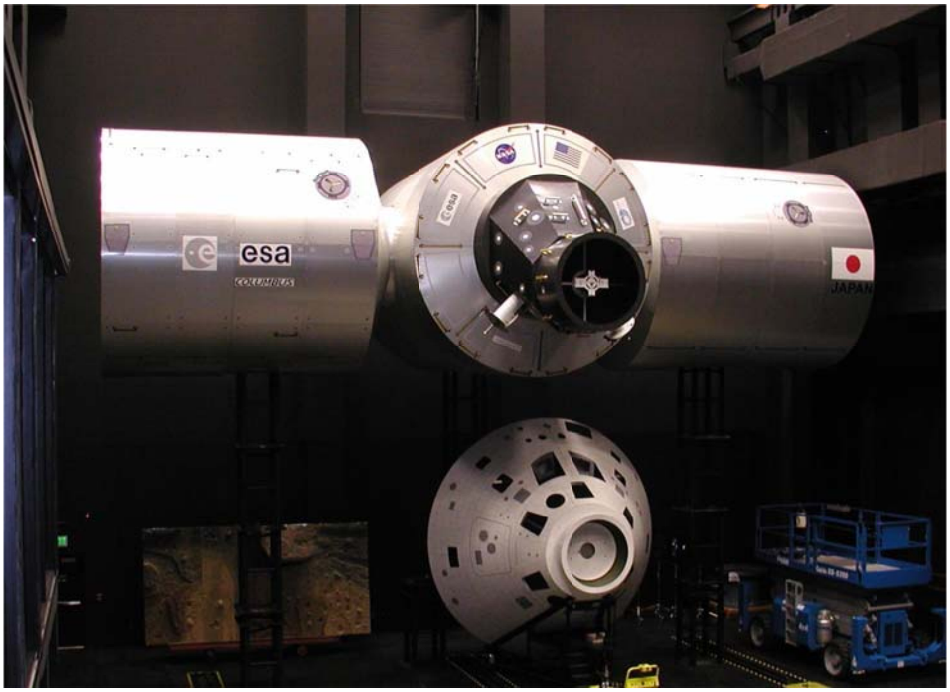


Figure 1: Full-sized mock-ups in SOSC High Bay: Left to Right – ESA Columbus Module, NASA Harmony Module with PMA-2 attached on forward end, JAXA Kibo Module, and CM mounted on cradle below

The Orion Rendezvous, Proximity Operations and Docking System

The Orion Rendezvous, Proximity Operations and Docking system has been matured greatly over the past five years. The first versions of the flight software have been developed and tested in simulations. Regardless of how much testing the RPOD software is subjected to, these software-only closed-loop tests have limitations. The VNS tested on STORRM is being tested with the image processing software (centroiding, reflector identification, pose estimation) and the RPOD GN&C flight software (the controller, the guidance system and the relative navigation filters), along with the delays/latencies associated with the throughput limitations of the systems used to emulate the image processing software. The components used during the closed-loop testing of the RPOD system at the SOSC are discussed in this section. The final flight configuration will differ significantly from the one described below, but integrated testing of many of the components represents significant step towards flight-readiness.

Vision Navigation System

The VNS is a next-generation, technology-defining, flash LIDAR built by Ball Aerospace. The VNS unit used during testing had previously flown in a relevant environment: it was flown and tested on Space Shuttle's STS-134 mission as part of the STORRM DTO documented again by Christian [2]. During the test, the VNS was placed on the 6DOF robotic platform that moved based on relative motion between the ISS and

Orion as driven by a simulation. The VNS provided images of its field of view at approximately 11 frames per second to a centroiding program designed to decipher the images.

Centroiding

The basic idea behind centroiding algorithms is to process the raw image from the VNS and determine whether strong returns in the pixel map represent noise, or actual LIDAR reflectors. LIDAR reflectors are purposely placed on the cooperative target (in this case, the ISS mock-up in the SOSC) to provide a stronger return signal than other structural elements on the target. By identifying the reflectors in the pixel map, the information can further be processed to generate pose measurements for processing by the RelNav Kalman filter. In the testing facility, the centroiding program ran on a stand-alone Linux machine. The inputs to the program were VNS images, while outputs were locations of centroids. It should be noted that the centroiding algorithm used for this test was developed by LM and is distinct from the Ball-developed algorithms presented by Gravseth [3] and the NASA-developed algorithms shown by Christian [1].

Vision Processing Unit

The vision processing unit (VPU), as a stand-alone unit with embedded algorithms, did not exist at the time of the testing. Instead a Windows computer using Simulink algorithms for processing centroids with was used as a substitute for the VPU. The VPU accepted centroids and generated pose measurements that were fed to the filter. At farther ranges, the centroids were converted to range and bearing measurements. However, when more than two centroids became visible and identifiable, the VPU used the RANdom SAmple Consesus (RANSAC) algorithm [4] to compute the relative attitude between the docking target mounted on the ISS and the VNS. The docking target mounted on the ISS mock-up contained five laser-reflective patches placed in a known pattern that the VPU algorithms can attempt to match. For the test, the execution rate of the VPU was set to 5 Hz, which allowed approximately two centroid maps to be weighted together to come up with a more accurate solution at each execution step.

GN&C Executive

The Orion flight software is comprised of many units associated with various Orion capabilities. For RPOD testing however, the critical components were relative navigation, orbit guidance, and Orion service module control. The each of these modules performs a pre-defined set of tasks with particular functions, e.g., navigation estimates the states of the vehicle, guidance determines the desired state at some point in the future, while control determines the thruster commands necessary to achieve the desired state. Holding all of the components together is the GN&C executive. The executive represents the decision-making software of Orion. It is responsible for activating different pieces of software and hardware at different times in the mission. A total set of objectives to be accomplished by Orion flight software are grouped together in a scenario and placed in a scenario file. A subset sequence of objectives is grouped in a segment. Fundamental sets of concurrent objectives for navigation, guidance, and control are specified separately, but integrated via a flight software activity. An activity is a basic building block of automation for Orion. Each activity contains a configuration file that in turn calls parameter files that are loaded at activity initialization. When an activity completes all of its tasks, it triggers a transition to the next activity via a set of transition criteria grouped in a transition file. The activity transition criteria consist of conditions tested by the GN&C executive program within the FSW. The transition criteria allow the executive to make autonomous decisions to move from one activity to the next. The executive is responsible for executing the correct scenario by calling the correct segments and activities at the correct time, while seamlessly transitioning and stepping through the segment list specified in the scenario. More information about the GN&C executive has been presented by King [5].

The Relative Navigation Filter

The Orion Relative Navigation Filter is an Extended Kalman Filter (EKF) which processes measurements from the IMU, GPS Receiver, Star Tracker, and the VNS. The Orion relative navigation translation filter keeps two inertial states: one for Orion and one for the target vehicle. The states are: Orion position vector, Orion velocity vector, Orion attitude vector (consisting of Modified Rodrigues parameters), IMU bias and misalignment states, Star tracker misalignment, and the VNS biases. The Orion angular rate is not a member of the state-space because attitude integration is done in the OIMU (Orion IMU). In the event that sensor data is lost, the filter can continue to propagate the two vehicles' states via dead reckoning so long as the OIMU data is available. The EKF processes 200 Hz data at 40 Hz intervals. Since there are multiple boxes associated with each sensor type, the sensor level Fault Detection, Isolation, and Recovery (FDIR) function selects the particular box to be used for that sensor type. The Star Tracker (ST) is the only (purely) bearing sensor used during Orion rendezvous operations. However, for SOSC testing, since it involves in operations from 15m to dock, the ST was not used. In practice the Relative Navigation EKF will initialize the Orion state from the selected) the Absolute Navigation filter slaved to a particular IMU. The target state will be initialized based upon an uploaded target ephemeris, with the assumption that the target is not a maneuvering vehicle.

The translation relative navigation filter differs from a generic EKF in two important ways. First, the Orion attitude states are handled in the manner of a multiplicative EKF (MEKF) **[6].** The second is that measurements that are received by the filter at the same time (though the time-tags of the measurements can vary due to measurement and timing latencies) are processed in the manner of a linear Kalman filter (KF), with a state update in the manner of an EKF occurring only after all of the aforementioned measurements are processed. The linear update was adopted due to the well-known difficulty that arises in the use of an EKF when combining some very precise measurements with measurements that are less precise (like inertial measurements). The nonlinearity of the relative measurements results in different filter solutions for a different ordering of measurement processing, and can even cause the EKF to diverge. This order dependency is mitigated by using a hybrid linear/extended Kalman Filter, whereby groups of measurements (inertial and relative) which are received at the same time, are processed as in a linear KF, after which the propagated state is updated in the familiar EKF sense.

The filter utilizes a second-order under-weighting process to account for the nonlinearities inherent in accurate range and bearing measurements. A Sigma-Point Kalman Filter (SPKF) was considered but was not further pursued due to the fact that it yields the same benefit as the second-order filter, while requiring more computations. The flow diagram of the Translation Relative Navigation Filter is provided in figure 3.

Finally, there is a Relative Attitude Filter that operates when the two vehicles are closer than 15 meters apart. This filter is a 9-state filter containing the relative attitude of the target, the attitude rate of the target and three misalignments. The sensor measurement is the attitude component of the pose measurement. Unlike the relative Navigation translation filter, this filter is very simple having to only process the pose obtained from the VNS.

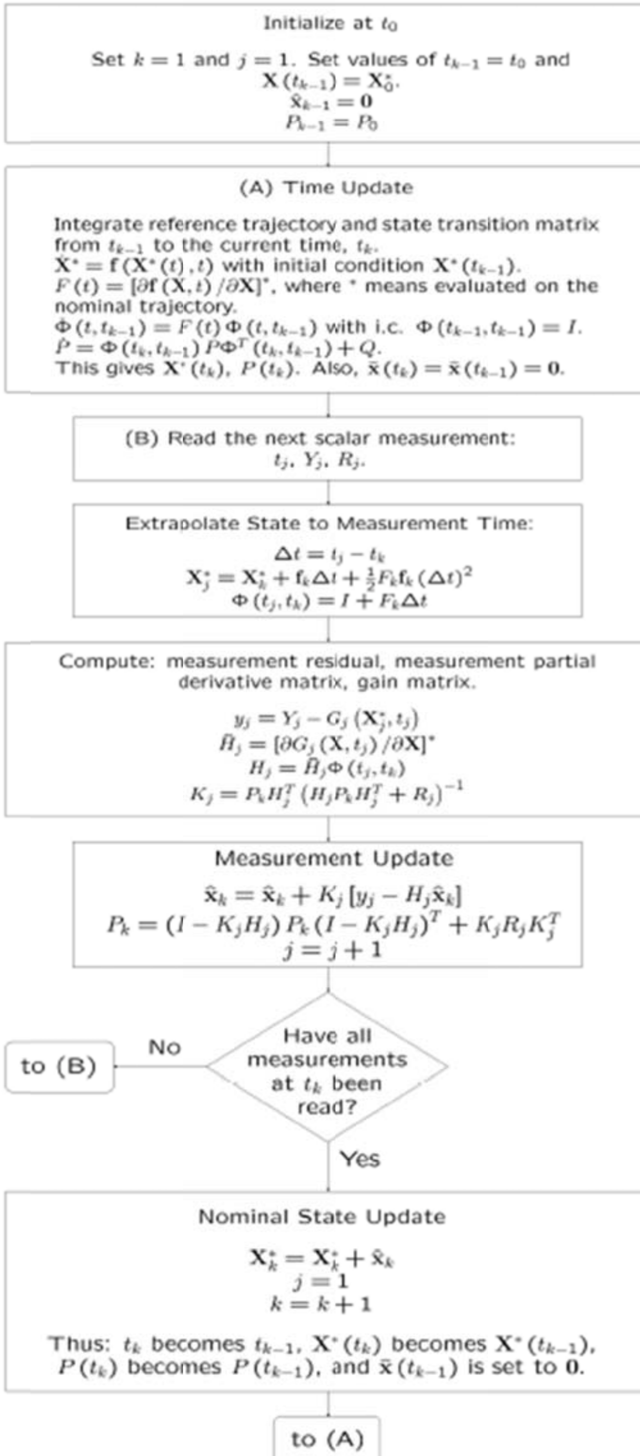


Figure 2: Orion Relative Navigation Translation Filter

Guidance

The Orion on-orbit guidance collection consists of several guidance and targeting algorithms packaged into separate units. While many algorithms are quite complicated, the unit used at the SOSC is one of the simplest, both algorithmically and computationally. The docking axis guidance module supplies a desired state (position and velocity) with respect to the target docking port (TDP) frame to the control system. An algorithm subject to linear dynamics only was selected to perform this function. For example, in case of station keeping, the desired position is always fixed at a constant, while the desired velocity is zero. For final approach a constant closing rate in the direction normal to the docking port plane can be selected, while zeroing out the TDP-relative lateral rate. The desired position at each time is computed by taking the current X in TDP coordinates and decreasing it by the desired rate multiplied by the guidance time step.

Strictly speaking, generating the desired attitude is performed within the control system by attitude pointing functions, but functionally it can be grouped with guidance. For station-keeping and final approach along the docking axis, attitude pointing commanded the docking attitude of Orion, such that the normal vectors of the docking port planes of Orion and ISS are axially opposed to each other, while maintaining zero yaw, pitch, and roll offsets with zero relative attitude rates.

Orion Service Module Control

The Orion service module controller is a hybrid proportional-derivative linear controller with an embedded phase plane. The control system errors are calculated by differencing the desired states with the current relative navigation estimates of the position and velocity for translation, while the desired rotational states are differenced with the IMU-derived attitude and attitude rates. The errors are converted to delta-V and delta-omega commands and passed to the control optimal group thruster selector, which minimizes the jet on-time while executing the given commands as derived by Glandorf [7]. On the Orion vehicle, the commands are passed to the service module propulsion unit for execution.

Osiris Simulation Software

While not strictly a part of the RPOD system, the Osiris simulation was critical during development, testing and configuration of flight software. Osiris is a high-fidelity simulation of Orion and ISS built for both analysis and software development. The role of Osiris in the development and execution of the test was also multifold. First, Osiris was configured to take inputs from the VPU machine, format it properly, and pass it to the flight software. Second, it simulated a multitude of vehicle models to mimic the Orion and ISS. The vehicle models included everything necessary to simulate a spacecraft from environment models such as atmosphere and gravity, to vehicle hardware such as sensor, thrusters (jets), and etcetera. The Osiris thruster models were able to take commands (jet on times) from the flight software, convert them to forces and torques, and apply them to the simulated Orion vehicle. Another role of Osiris was to format the output of the sensor models passed on to the flight software. Due to the highly configurable nature of Osiris, the simulation could shift from using VNS and VPU hardware-in-the-loop to using software models by simply setting a few flags in the initialization file, allowing the integration of hardware and software to proceed in parallel.

Test Configuration Overview

Figure 4 represents graphically the integrated test configuration. The VNS was mounted on the 6 DOF robot. The VNS data were passed the centroiding program running on a Linux machine. Once centroids were determined, they were passed to the VPU software running on a Windows machine. VPU outputs consisted of pose measurements that interfaced with the Osiris simulation. Osiris software ran on a second Linux machine (Linux2) and populated the entire input interface for the flights software. The pose measurements are represent a small fraction of this interface, as each piece of hardware that interacts with GN&C fills a part of this interface. Once the interface was populated the data were sent to the flight software running on the same Linux machine as Osiris. The flight software performed GN&C and returned thruster on-times to Osiris. The Osiris Orion service module model accepted the thruster on-times and generated forces and torques that combined with all other forces and torques acting on Orion to provide an updated state at each 0.1-second simulation time step. The ISS state was also integrated, however, ISS thruster firings were not modeled in Osiris. The inertial states were then passed to the robot controller station that computed new facility frame coordinates for the robot, effectively moving the robot to the new commanded position.

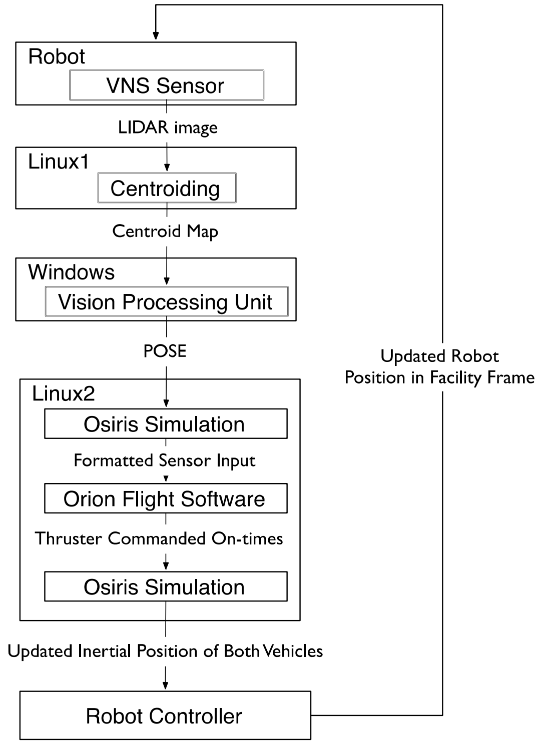
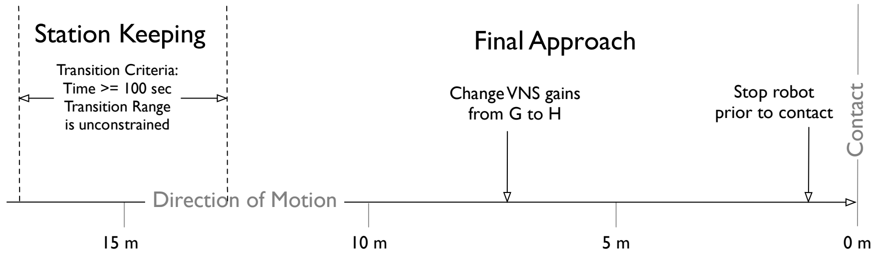


Figure 4: Connections and information flow between test components

Osiris Configuration
The ISS vehicle model utilized assembly-complete mass properties. The ISS attitude control system was not actively used. Rather, the ISS was placed at -7.5 degrees pitch with respect to LVLH and held attitude without any error. Gravity effects of degree 8 and order 8 (8x8) were in place. Aerodynamic effects were not considered. The Orion environment was set up exactly like the ISS, with an 8x8 gravity model and without any atmosphere. Flex vehicle dynamics were not used. The inertial measurement units (IMUs) produced perfectly accurate data that matched the truth model of the vehicle. Orion mass properties were based on the Lockheed Martin simulation data book, with crew module (CM) mass properties from "606D Completion of Coast to TPI" and service module (SM) mass properties based on "606D Orion LEO Check and Deploy" sections of the simulation data book. The thruster configuration utilized was the 607A.

Flight Software Configuration
The scenario given to the Orion flight software was given to perform three successive activities autonomously: initialization, station keeping at an Orion docking port (ODP) to TDP relative distance of 15 meters, and final approach from 15 meters to 0.5 meters. Since docking contact without damaging the hardware in the SOSC was not possible, the tests were stopped at relative distance of approximately 0.5 m. The initialization activity takes only one second; the second and third activities are neatly summarized in figure 5.



Station Keeping @ 15 m

Guidance:
Desired position and velocity guidance.
Desired velocity Vtdp = [0, 0, 0].
Desired position Rtdp = [-15, 0, 0].
Translational Control:
Desired position and velocity from Guidance.
Control errors computed based on Rel Nav.
Attitude Control:
Maintain R=172.5, P=180, Y=0, with respect to LVLH.
Control errors are calculated based on perfect IMU model.
Attitude Navigation:
Estimate relative attitude from VPU pose quaternion.
Translational Navigation:
Process VPU pose range and bearing.

Final Approach

Guidance:
Desired position and velocity guidance.
Desired velocity Vtdp = [-0.03, 0, 0].
Desired position Rtdp is derived from current position by moving forward along the Xtdp projection, with zero lateral offset.
Translational Control:
Desired position and velocity from Guidance.
Control errors computed based on Rel Nav.
Attitude Control:
Maintain R=172.5, P=180, Y=0, with respect to LVLH.
Control errors are calculated based on perfect IMU model.
Attitude Navigation:
Estimate relative attitude from VPU pose quaternion.
Translational Navigation:
Process VPU pose range and bearing.

Figure 5: Orion Flight Software Activities for Closed-Loop RPOD Test in SOSC

Initialization

The purpose of the first FSW activity (initialization) was to simply initialize the absolute navigation system, which was set to produce states equal to the truth model in the Osiris simulation. Internally, RelNav is initialized from the absolute navigation state state, thus forcing a one-second activity for absolute navigation initialization at the beginning of the test. No other flight software was initialized or used in this activity. Since the initialization activity performed a very simple task, less than a second was required to complete it. The executive ran at 40 Hz and, in the case of the initialization activity for SOSC testing, ended the activity and transitioned to the next at one second of simulation time.

Station keeping at 15 meters

The second activity (station keeping) aimed to station keep Orion at an ODP-to-TDP relative distance of 15 meters. The software components of RelNav used in the test were VPU interface, the delta-V accumulation function, the translational filter, the rotational filter, and the user parameter processor. While the filter has the capability to process four different types of measurements, for the given experiment the filter was configured to process range and bearing measurements from VPU only. The IMU delta-V supplied to the filter did not have any error, i.e., the value corresponded exactly to the velocity changes of the simulated Orion. GPS was not used in the test. Relative attitude estimation was performed by updating the state with the relative attitude quaternion generated by the VPU.

Guidance supplied a desired state (position and velocity) with respect to the docking port frame to the control system. For station keeping, the desired position was fixed at Xtdp = -15 m, Ytdp = 0 m, Ztdp = 0 m, while the desired velocity is zero. The desired attitude was the docking attitude of Orion, such that the docking ports of Orion and ISS are axially opposed to each other, while maintaining zero yaw, pitch, and roll offsets with zero relative attitude rates.

Given a set of parameters that defined the size of allowable position, velocity, attitude, and attitude rate errors (6DOF control), the control system selected minimum on-times for the service-module thrusters using optimal jet select. For the test, the commands were processed by the Osiris simulation, which generated forces and torques on the simulated vehicle. The control system maintained the same mode of operation and same parameters for the length of the test. For anything outside the scope of the testing that has already been done, different modes and parameters will be required.

The station keeping activity ended when the transition criteria for the activity were met, i.e., elapsed time in the activity reached 100 seconds. At that time, the GN&C executive transitioned to the third and final FSW activity.

Final Approach

During the third and final activity (final approach) all domains except for GDO maintained their previous modes and parameters. Meanwhile, guidance provided a different desired state at each execution time step. The desired velocity corresponded to a constant closing rate of -0.03 m/s in the direction normal to the docking port axis, while zeroing out the TDP-relative lateral rate. The desired position at each time is computed by taking the current Xtdp and decreasing it by the desired rate multiplied by the time step. The flight software never transitions out of the final activity; rather, the robot is mechanically stopped when it reaches a minimum distance from the target. The test run is considered finished when this hardware stop is reached.

Results of the Closed-Loop Tests in the SOSC

Seven different test cases with varying initial conditions were run to completion during the formal SOSC testing. The tests were dispersed about the nominal position and attitude, but not about the initial zero velocity and zero attitude rate. Each run concluded at the 0.5 m VNS to target relative distance. The runs required approximately 800 seconds to run to completion. The total run-to-run turn around time was in the vicinity of 30 minutes. While the meat of data collected from the tests is yet to be analyzed, some preliminary analysis was done during execution of the tests to ensure that the components are performing within the expected envelope. The following discussion encapsulates the results of that preliminary analysis.

Functionally, the software and hardware each performed their assigned functions with little problems. The GN&C executive was able to correctly step through the list of activities provided in the scenario file.

The VPU outputs were used to determine the noise signature of the VNS. The test results show that the bearing measurements are accurate to within the VNS specification, while the range noise appears to have a linear relationship with the relative range between the VNS and the target.

A simple centroiding scheme that applied a range calibration and intensity-based thresholding performed with better-than-expected accuracy. It is unknown how this scheme will work with a target capable of producing spurious reflections.

The VPU itself performed well for most of the tests run. However, when accepting more than five centroids, VPU was not able to converge on a pose solution. This issue will be addressed in future iterations of the algorithm.

The RelNav system performance can be judged by the filter performance in terms of covariance and residual performance, and in terms of absolute error of the solution. The VNS position in the facility frame is known within a tolerance measured on the scale of millimeters partly thanks to the accuracy of the Leica positioning system available in the SOSC, and partly due to the many hours of painstakingly calibrating out the misalignments of the robotic system. Differencing the Leica-based position of the VNS with a relative VNS position derived from the Kalman filter provides a method of calculating the error in the filter state. Early test results filtered solution was accurate to meet docking tolerances for LIDS.

Given guidance's simple algorithm, anything less than predictable commands to control would be surprising, and most-likely a result of interface problems rather than algorithmic deficiency. Indeed, guidance did not disappoint providing nominal results.

The results of tests the showed well-controlled trajectories in the translation channel with position errors of up to 0.06 m and velocity errors of 0.05 m/s. The attitude errors were maintained within 0.25 degrees, with minimal attitude rates. All of these parameters are well within the docking envelope. While performance of the control system in terms of errors was quite satisfactory, it came at the expense of unusually frequent jet commands. More analysis is required to ascertain whether such frequency of firings is predictable and acceptable.

Summary and Conclusions

A judgment on the success of the closed-loop test with the minimal data analysis is incomplete at best. However, the preparations for and execution execution of a closed-loop hardware and software in the loop test is no small feat. Exercising many algorithms as FSW prototype components necessary for RPOD while running in near real time is a significant step in maturing the system. Defining and correctly implementing interfaces along the way was an important component of successful functional testing. Obtaining the kind of performance noted above is a significant success that could not have been obtained unless:

1) the test setup included successful calibrating of hardware mounting misalignments,
2) accurate tuning of Kalman filters for the particular sensor and test environment was performed, and
3) a good understanding of the sequencing and latency issues that arise when combining several complex hardware and software components.

While this is the first set of many more tests to come, it represents a fundamental building block for any future integrated of tests. The testing identified areas of concern for algorithms, software, and hardware, and clarified the forward path for future development and testing.

References:

[1] Christian, J.A., Patangan, M., Hinkel, H., Chevray, K., and Brazzel, J., “Comparison of Orion Vision Navigation Sensor Performance on STS-134 and in the Space Operations Simulation Center,” AIAA Guidance, Navigation, and Control Conference, Minneapolis, MN, 13-16 Aug 2012.

[2] Christian, J.A., Hinkel, H.D., D’Souza, C.N., Maguire, S.S, Patagan, M., "The Sensor Test for Orion RelNav Risk Mitigation (STORRM) Development Test Objective.” AIAA GNC Conference, 5-11 Aug 2011, Portland, Oregon. JSC-CN-24126.

[3] Gravseth, I., Rohrschneider, R., and Masciarelli, J., “Vision Navigation Sensor (VNS) Results from the STORRM Mission,” AAS Guidance and Control Conference, Breckenridge, CO, 3-8 February 2012.

[4] Martin A. Fischler, and Robert C. Bolles (June 1981). "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". June 1981. Comm. of the ACM **24** (6): 381–395.

[5] King, E., Hart, J., and Odegard, R., "The GN&C Data-Driven Flight Software Architecture for Automated Sequencing and Fault Recovery". January 21, 2010. *IAAAC #1358.*

[6] Lefferts, E., Markley, F., and Shuster, M., “Kalman Filtering for Spacecraft Attitude Estimation,” Journal of Guidance, Control, and Dynamics, Vol. 5, No. 5, Sept.–Oct. 1982, pp. 417–429.

[7] Glandorf, D.R, “An innovative approach to the solution of a family of linear programming problems.” Jan 1, 1987. NASA. NASA-CR-182987.