AFRL-RV-PS-
TR-2022-0045

AFRL-RV-PS-
TR-2022-0045

SPACE DOMAIN AWARENESS COLLABORATIVE RESEARCH INFRASTRUCTURE

Roberto Furfaro

The University of Arizona
888 N Euclid Ave
Tucson, AZ 85719

28 February 2022

Final Report

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

AIR FORCE RESEARCH LABORATORY
Space Vehicles Directorate
3550 Aberdeen Ave SE
AIR FORCE MATERIEL COMMAND
KIRTLAND AIR FORCE BASE, NM 87117-5776

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research which is exempt from public affairs security and policy review in accordance with AFI 61-201, paragraph 2.3.5.1. This report is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RV-PS-TR-2022-0045 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//SIGNED//
JAMES FRITH
Program Manager

//SIGNED//
SEAN PHILLIPS
Tech Advisor, Space Control Technologies Branch

//SIGNED//
JOHN BEAUCHEMIN
Chief Engineer, Spacecraft Technology Division
Space Vehicles Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
28-02-2022	Final Report	27 Mar 2018 – 28 Feb 2022

4. TITLE AND SUBTITLE
Space Domain Awareness Collaborative Research Infrastructure

5a. CONTRACT NUMBER
FA9451-18-2-0105

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62601F

6. AUTHOR(S)
Roberto Furfaro

5d. PROJECT NUMBER
3000

5e. TASK NUMBER
EF130964

5f. WORK UNIT NUMBER
D0D2

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The University of Arizona
888 N Euclid Ave
Tucson, AZ 85719

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Space Vehicles Directorate
3550 Aberdeen Avenue SE
Kirtland AFB, NM 87117-5776

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RVSW

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-RV-PS-TR-2022-0045

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Space Domain Awareness Collaborative Research Infrastructures

14. ABSTRACT
The funded cooperative agreement established a collaborative effort between Air Force Research Laboratory (AFRL) Space Vehicle Directorate and University of Arizona. Such collaboration enabled the development of a dedicated cyber-infrastructure named VerSSA which has been developed to provide the overall Space Domain Awareness community with a platform for data management and sharing as well as algorithm development and sharing.

15. SUBJECT TERMS
Cooperative agreement, collaboration, development, cyber-infrastructure, VerSSA, Space Domain Awareness

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			James Frith
Unclassified	Unclassified	Unclassified	Unlimited	46	19b. TELEPHONE NUMBER *(include area code)*

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. 239.18

(This Page Intentionally Left Blank)

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

Page

(This Page Intentionally Left Blank)

1.0 SUMMARY

The funded cooperative agreement established a collaborative effort between Air Force Research Laboratory (AFRL) Space Vehicle Directorate and University of Arizona. Such collaboration enabled the development of a dedicated cyberinfrastructure named VerSSA which has been developed to provide the overall Space Domain Awareness community with a platform for data management and sharing as well as algorithm development and sharing. VerSSA is powered by CyVerse, and NSF-funded platform that enables large collaborative projects and scalable high-performance computing between life science research teams. VerSSA has been developed to serve a similar purpose within the SDA community and it is developed and demonstrated across a set of projects across the three-year of performance. In year 1, we demonstrated that VerSSA can be employed to develop collaborative workflows that accounts for physics-based models and data-driven models within the same framework. Machine learning and physics-based algorithms can be integrated to process data in and end-to-end fashion to solve SDA problems in a new and effective fashion. A use-case comprising the simulation of the Sentinel-1A experienced anomaly detection (solar panel impacted by a small debris) has been considered to showcase the power of VerSSA and demonstrate that anomalous spacecraft behavior can be detected using ground-based sensor and dedicated machine learning systems. In year 2 and 3, VerSSA has been further developed to provide a platform for international collaborative efforts within the FVEYs defense organization. Algorithms performing a variety of functions including astrometry, photometry, orbit determination have been contributed by the participation nations and integrated ("dockerized") in a comprehensive VerSSA workflow and made it available to the FVEYs community for data storage, processing and sharing. The VerSSA workflow has been employed in two FVEYs campaigns (Phantom Echoes 1&2) devoted to observational characterization and tracking of the Northrop Grumman Mission Extension Vehicles 1&2. VerSSA has demonstrated to be an effective platform for large and scalable collaborative enterprises in SDA efforts.

Finally, VerSSA has been proposed as platform for algorithm development and data sharing for effective Cislunar SDA (XDA). As part of the cooperative agreement, we have proposed and started the development of a set of tools and processes that aims at building the first academic XGEO catalog. The later leverages forty years of experience of UArizona in developing and deploying planetary defense methods and tools for asteroid tracking and characterization.

2.0 INTRODUCTION

In March 2018, AFRL signed a Cooperative Agreement (CA) with University of Arizona (UArizona) for a three years contract (2018-2021, ~$3.5M) with University of Texas as Subcontractors. This Cooperative Agreement grant is intended to fund the development of a dedicated Cyberinfrastructure for Space Situational Awareness (SSA, now Space Domain Awareness or SDA). University of Arizona has been leader in developing and managing large-scale cyberinfrastructure. Indeed, since 2008, UArizona has been home to National Science Foundation (NSF) BIO's largest investment in cyberinfrastructures (~ $115M), resulting is the development and deployment of a cyberinfrastructure named CyVerse (Cyber + Universe). Currently, CyVerse accounts for more than 70,000 users across multiple life-science disciplines. Notably, CyVerse has been employed as computation infrastructure to process the Event Horizon Telescope data which led to the first black hole imaging. The Cooperative Agreement has been leveraging such ancillary infrastructure to create VerSSA, a cyberinfrastructure capable of supporting SSA specific analysis and data management. VerSSA is has been conceived as a platform for data sharing and algorithms development and deployment on large stream of SSA collected data. Currently, VerSSA supports national and international efforts with allied nations. Indeed, the UArizona has been working with the FVEYS Nations to developing and execute a VerSSA-based coordinated SSA campaigns, leading to the Phantom-Echoes 1&2 coordinated observational campaign. Additionally, VerSSA has been employed as basis to start the development of infrastructures for space domain awareness in cislunar space (XDA).

This report is organized as follows. In section 3.0 the development and the features of the VerSSA system is described, including major features inherited by the NSF CyVerse cyberinfrastructure. In section 4.0, the research and development program executed in year 1 to evaluate the power and the flexibility of VerSSA in SSA-specific use-cases is described. Results and analysis are reported for the Sentinel 1 anomaly detection use-case. Section 5.0 report the development of the integrated VerSSA workflow/pipeline development to support the FVEY Phantom Echoes 1 observational campaign. Contribution from the UArizona SSA dedicated sensors is reported. Section 6.0 shows the completion of the VerSSA system and deployment to support the FVEY Phantom Echoes 2 campaign. Finally, Section 7.0 reports the initial effort of the UArizona team to support the development of infrastructures and methodologies for effective ground-based XDA.

3.0 METHODS, ASSUMPTIONS AND PROCEDURES

3.1 Description of the VERSSA System

The VerSSA system (contraction of CyVerse and SSA) has been proposed as premiere cyberinfrastructure that can serve as data management system and algorithm integration sharing platform for SSA users, see Figure 1. It relies on the foundational effort funded by NSF to develop a dedicated cyberinfrastructure capable of data and algorithms management and sharing.

3.1.1 What is CyVerse.

Funded by the National Science Foundation (NSF) from 2008, with primary mission to design, deploy, and expand a national cyberinfrastructure for Life Sciences researchers, and to train scientists in its use. Built with matures and battle tested open sources components and with over 70,000 registered users, CyVerse provide core computational infrastructures for multiple disciplines from Astronomy to Earth Sciences as the Cyber Infrastructure components are domain agnostic, allowing community of computational, data and observational scientists to effectively and securely collaborate on large scale projects.

Platforms, tools, datasets Storage and compute Training and support

Figure 1. CyVerse System

cyberinfrastructure (also known as CI or computational infrastructure) provides solutions to the challenges of large-scale computational science. Just as physical infrastructure such as laboratories makes it possible to collect data, the hardware, software, and people that comprise cyberinfrastructure make it possible to store, share, and analyze data. Using cyberinfrastructure, teams of researchers can attempt to answer questions that previously were unapproachable because the computational requirements were too large or too complex.

3.1.2 CyVerse Philosophy and Summary of Capabilities.

The overall CyVerse development philosophy can be summarized as follows:

- Strive to provide the **CI Lego blocks**
- Danish 'leg godt' - '**play well**'
- Also translates as '**I put together**' in Latin
- If desired functionality is not available, the community can craft their own by using and extending CyVerse CI components (like lego blocks)
- Through these extensible and customized platforms create an ecosystem of interoperable tools that benefit the broad community (and <u>not few lab groups</u>)
- Provide the tools to allow community to manage their digital assets (cloud, HPC etc.)
- Improve Computational Productivity

Importantly, CyVerse has a set of extensive capabilities that can be readily adapted for the SSA community (see section 3.2):

- CyVerse manages data and metadata by iRODS (integrated Rule Oriented Data Systems)
  - Highly customizable to search and discover via queries, events and policies
  - Driven by a set of rules that can be defined based on requirements
- Cyverse Discovery Environment:
  - Enables the definition of Apps that can be executed on any platform (either locally or using CyVerse inherent massive parallel computing capabilities)

- Apps are powered by the Docker technologies
    - Employs portable container for platform independent computing
- Apps can be public or private or shared among collaborators as needed
- Uses the REST API to facilitate the autonomous execution of tasks upon direct availability of events

- ITAR Control: CyVerse-in-a-Box:
    - Access a Cyverse isolated section via VPN

3.2 VerSSA Overview

VerSSA is a contraction of two acronym, i.e. CyVerse & SSA. VerSSA is conceived to extend CyVerse Cyberi-nfrastructure capabilities to support SSA specific analysis and data management. VerSSA makes it easy for data generators and algorithms and methods developers to collaborate by easily sharing tools and datasets. Importantly, VerSSA make it easy for communities to share best practices in a reliable, reproducible manner. Indeed, it allows tools/ methods developed in different programming languages and technology stacks to exist and promotes reuse e.g. visualization developed with open source web technologies like CesiumJS can be securely utilized by any pipeline. Additionally, VerSSA provides access to contemporary data science tools and platforms such as Apache Spark, computational notebooks such as Zepplin, Jupyter in a integrated manner to support reproducible and scalable analysis. For registered users, VerSSA features 1) Data Access & Retrieval; 2) Data Storage; 3) Data Processing and Algorithm Development. Figure 2 shows a high-level workflow of the VerSSA system.

Figure 2. Overview of the VerSSA System

3.2.1 VerSSA General Workflow and Data Upload.

VerSSA general approach to data upload and general analysis workflow is described in Figure 3. The process includes 1) ability to upload raw images from optical SSA sensors, 2) storage in dedicated directories, 3) automatic & semi-automatic data processing for astrometry and photometry, 4) data visualization; and 5) processing of data via dedicated orbit determination workflows.

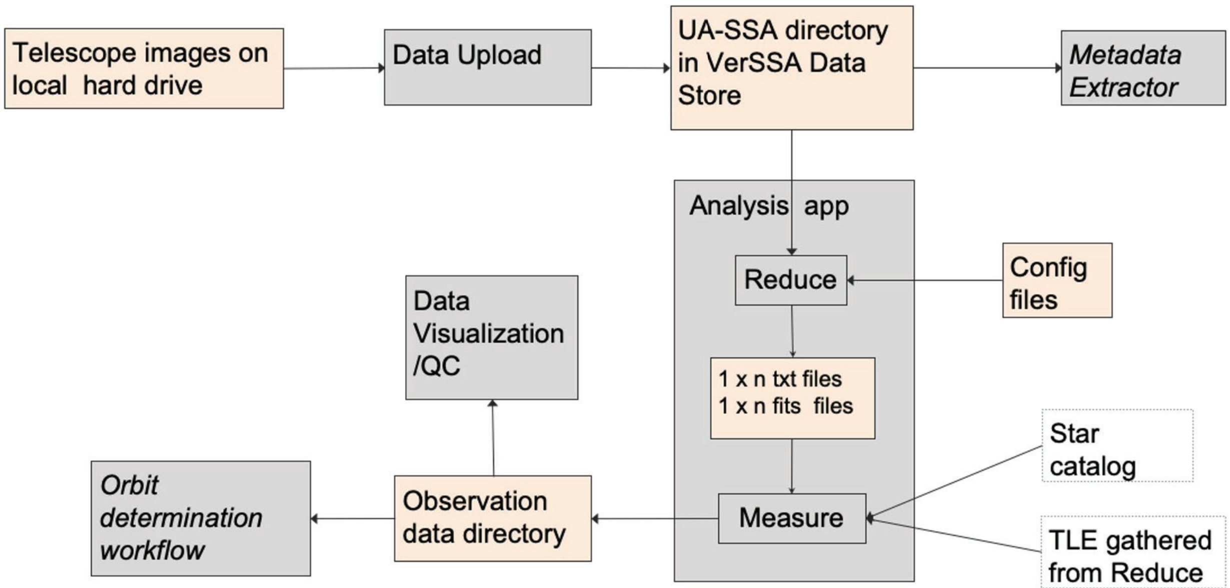


Figure 3. General VerSSA Workflow and Data Analysis

VerSSA data upload can be run manually or automatically after data collection. Uses IRODS tools to efficiently upload data. Once data are uploaded, meta data is applied and analysis can start. Generally, extracted meta data is searchable (i.e. NORAD ID, Telescope, Date, etc.). Importantly, uploaded data can be selectively shared with collaborators or only visible to the uploader.

Although any data can be uploaded and shared on VerSSA, but standardization speeds analysis and improves discoverability. As such, a specific file naming for **FITS files** has been conceived and generally has the following format:

- noradID_yyyymmdd_exposureTime_binning_seriesNumber.fits

Also, the following general directory structure has been adopted:

/iplant/home/shared/

-SSA-Arizona

--Organization or Observatory_name (or codes for vetted observatories)

---telescope_name

----night (YYYYMMDD.tar.gz) -- one tarred file per night per telescope

More details of the development and progresses of the overall VerSSA workflow are provided in Section 4.0.

3.2.2 VerSSA-R (Restricted).

VerSSA-R (VerSSA + Restricted access) enable restricted access for customized data controls. VerSSA-R provides ITAR and NIST 800-171 compliance for public VerSSA capabilities. It is housed-in dedicated AWS.gov (Amazon Web Services) infrastructure, with controls provided and managed by University of Arizona InfoSec compliance office. VerSSA-R inherits the same level of flexibility and sharing capabilities as public VerSSA. However, VerSSA-R users undergo additional steps to obtain access to this infrastructure and connect through dedicated VPN client with MFA (Multi Factor Authentication). Here, approved users can share data and analysis methods (pipelines) within VerSSA-R user community and projects.

VerSSA-R application catalog for analysis pipelines is identical to public VerSSA, additionally it includes specific restricted access methods and datasets. VerSSA-R users and applications have access to shared SSA data housed in public VerSSA. All approved AWS.gov applications (Machine Learning, IoT etc.) are available to pipelines. VerSSA-R can be used for data curation and validation to publish content and applications to public VerSSA after review.

4.0 RESULTS AND DISCUSSION

4.1 Project Year 1: VerSSA System Demonstration in Selected Use-Case

The first year was dedicated to the development of the VerSSA system as well as demonstration of its capabilities. The overall goal was to show how a combination of machine learning and AI methods with physics-based models, integrated in a modern computing and data management cyber platform as VerSSA can provide more efficient solution to the problem of spacecraft anomaly detection and behavior characterization. The following research questions related to Science & Technology (S&T) as well as Research & Development (R&D) were developed:

– **Research Question #1 (S&T)*:*** How can data-driven methods rooted in deep learning enhances the spacecraft anomaly detection and behavior characterization.

– **Research Question #2 (S&T)*:*** How can deep learning algorithms be effectively trained using physics-based models to learn patterns that account for the physical interaction of the spacecraft with the physical environment.

– **Research Question #3 (R&D)*:*** What is the most effective method for deploying and operating machine learning approaches in conjunction with physics-based models and visualization methodologies that can provide effective decision support to human operators.

– **Research Question #4 (R&D)*:*** How can modern computing infrastructure and data management systems (e.g. VerSSA) can be employed to effectively deploy machine learning and physics-based tools for the contemporary and future SSA challenges.

A specific use-case consisting of a simulation of a specific spacecraft anomaly detection has been considered to provide answers to the research questions above.

4.1.1 Use-Case Spacecraft Anomaly Detection: General Description.

The use-case is specifically conceived to tackle the problem of spacecraft anomaly detection. The general goal is to show that data analytics (e.g. deep learning) and physics-based models can work together to effectively provide timely analysis of observations, as well as trigger observations that can provide actionable intelligence, i.e. detect anomalous spacecraft behavior from telemetry data, trigger observations, classify specific mode of spacecraft behavior, explain mode of behavior and visualize content and information in real-time. The goal is to show that such a system is superior to a physics-based system only and that can provide decision support to a human operator.

Generally, observations need informed understanding of sensor operations and sensitivity as well interaction with physics of sensed environment (medium and targets) to provide the most effective (accuracy, precision, computation, timeliness) solution for prediction, characterization or attribution. Data storage needs knowledge of how data being used for efficient retrieval and alleviate improper overwrites as well as formatting for use and visualization. Data processing requires methods that are space domain relevant and exploit domain knowledge and acquired data for adapting solution and providing prediction ability to weight cost of observation/computation/sensing with decision making timelines and goals. Importantly, we will show that modern computing infrastructures such as VerSSA platform, enable efficient and real-time processing, computing and visualization of the desired SSA workflow for spacecraft anomaly detection and behavior characterization.

Described with specifics in the following, all for developing methods to be experimented on for data collection, sharing, and processing for prediction and attribution. There are S&T (basic) and R&D (applied) questions for object/environment prediction and understanding, sensing, visualization, storage, processing, retrieval, testing, and interaction between government-industry-academia for (re-) train/learn/tool/teach workforce.

4.1.2 Use-case Description: Set-up & Challenge.

The following scenario is considered:

- Take real world example of Sentinal-1A (LEO, ESA bird) where while monitoring telemetry operators noticed a drastic drop in solar array output and seemed to see some attitude and orbital effects in the telemetry stream: [1, 2]
  - i) Resolution: using on-board cameras that looked out the solar arrays, they were able to determine that a SA was impacted by approx 5mm debris (via counting damage cells in the array)
  - ii) European Space Agency, "Copernicus Sentinel-1A Satellite Hit by Space Particle," 31 August 2016. [Online]. Available: http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-1/Copernicus_Sentinel-1A_satellite_hit_by_space_particle
  - iii) Krag, H. et al, “A 1 cm Space Debris Impact Onto the Sentinel-1A Solar Array”, *Acta Astronautica*, Vol 137, August 2017, pp. 434-443.

The following challenges are anticipated:

- Identifying information sources and provide monitoring and prediction (trajectory/attitude/system telemetry) algorithms and processing appropriate architecture to provide automated detection and resolution of Sentinel-1A anomaly AS IF the cameras on the solar array were not available
- Tap multiple information sources without necessarily storing data all in one place but having access to the data
  - i) Space Weather feeds
  - ii) Telemetry Feeds
  - iii) Amateur/academic observation network feed /data

- Apply and refine orbit and attitude prediction and models to provide possible causes as well as identify follow-up observations and/or attitude slews necessary to identify cause

4.1.3 VerSSA Approach to the Use-Case.

The overall goal is to create a VerSSA integrated demo that demonstrates the power of VerSSA for the proposed use-case Sentinel-1A anomaly detection. UArizona and University of Texas at Austin (UTA) worked synergistically to provide innovative solutions to the spacecraft anomaly identification and behavior identification. The integrated demo demonstrates the research approach and show how combined machine learning & Physics-based methods help solve the problem. Overall contribution is articulated as follows:

- UA provide VerSSA platform , Machine Learning and AI methodologies, Visualization methods and Project Leadership
- UTA provides high-fidelity simulations physics-based OD/AD determination algorithms and astrodynamics expertise

The combined team is synergistic in nature: UTA provides physics-based training points from their simulations whereas UArizona works on designing, training and validating Deep Networks. All algorithms are combined in the VerSSA environment to combine both physics-based OD/AD and they are executed in a demo. The UA/UTA team evaluates performances and reaches conclusions and define a path forward. We devised the following general approach:

- Create a high-fidelity simulation of Sentinel-1A scenarios for both nominal and off-nominal behavior
    - Simulate trajectory and attitude motion, sensor and actuators (RWs, IMUs, Solar Panels and Power profile)
    - Simulate ground observations,i.e. astrometric and photometric data as collected by EO sensors
- Define API with VerSSA for astrometric and photometric data, as well as telemetry data
    - Flow of data into VerSSA (storage of data and meta data for intelligent search, email alert)
- Define data processing algorithms
    - Define workflow and appropriate algorithms (OD, AD for status of the system)
    - Define data analytics (Space Object Bayesian Networks, Deep Networks for anomaly detection)
- System outputs: Sentinel-1A anomaly detection

4.1.4 Science Rationale.

The overall approach Deep learning methods are integrated with physics models to cover gaps in understanding S/C behavior. More specifically:

- Physical methods provide estimation of S/C trajectory and attitude but does not provide information on S/C behavior. Deep learning methods use data-driven approach to capture S/C behavior unmodeled by physics (e.g. human-in-the-loop)
- Deep Learning models: We employ Recurrent Networks(Long-Short Term Memory - LSTM) with Convolutional Networks (CNN) within the framework of Meta-Learning(Learn to Learn)
  - Training on a range of physically-based models to classify behavior and adapt to incoming data (few-shot learning)
  - Meta-Deep Learning based on LSTM-CNN improves upon conventional physics-based estimation by learning trends in the data and adapt to behavior
- LSTM, CNN, Generative Adversarial Networks (GAN) within conventional learning and meta-learning are state of the art in ML community – SSA current shortcomings in integrating of physics models with deep learning that can be addressed by VerSSA
  - Enabling research at scale using modern computing techniques, including 1) Dockerization and deploying of complex algorithms; 2) support curated models; 3)train on new data using GPUs, 4) publish models curated/validated by community

4.1.5 Use Case Spacecraft Anomaly Detection: Implementation.

4.1.5.1 Concept of Operations

For the operational scenario, a spacecraft is simulated as operating nominally, and at a specific time, it is impacted by a space debris object. The subsequent behavior analysis is divided into four phases.

Phase I. Simulation data posing as spacecraft onboard telemetry are streamed to the VerSSA environment. The data are then processed by deep networks for anomaly detection to confirm when/if the spacecraft enters an anomalous state. The results can also be visualized in 3D using the ISV system. This then triggers further simulated observations which are also streamed to VerSSA.

Phase II. The simulated observational and telemetry data are processed through several algorithms to determine the event that occurred. Observations are autonomously reduced in the VerSSA environment and can trigger OD/AD algorithms, including UKF, MUKF, and batch least squares. The results are then processed by another deep network for spacecraft behavior characterization. Filtered trajectory and attitude as well as telemetry are retrieved and visualized using the ISV 3D system. A heads-up display (HUD) is implemented to display the behavior results from the machine learning algorithms, overlaid on the filtered spacecraft motion.

Phase III. The same scenario is run using only physics-based models and visualization. Data is processed and used for OD/AD algorithms. Spacecraft behavior is analyzed in this phase without the assistance of deep networks and compared to the behavior determined in Phases I & II.

Phase IV. Document and report findings, conclusions, and recommendations.

A visual representation of the complete CONOPS can be seen in Figure 4.

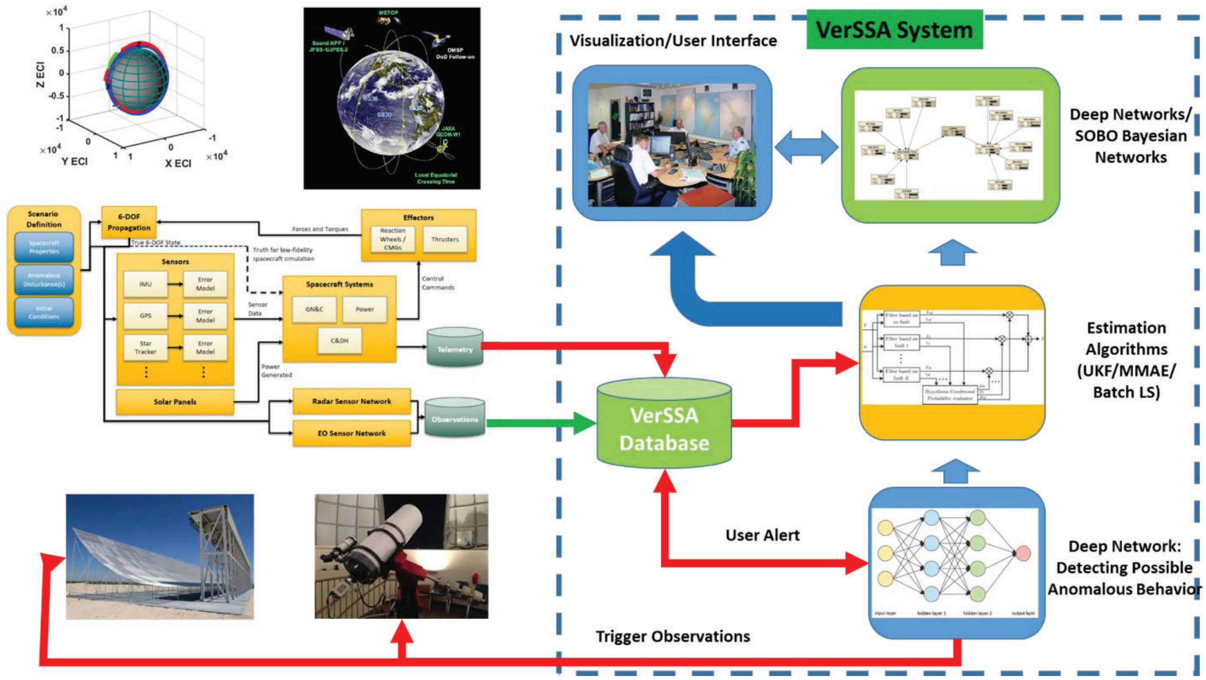


Figure 4. Complete CONOPS

4.1.5.2 Simulation Architecture

The simulation was designed to replicate the Sentinel 1A space debris impact event. As a result of the impact, the spacecraft experienced a power drop and a perturbation from nominal orbit. The event was discovered after seeing the power drop during preparations for a maintenance maneuver.

The attitude GNC system lost valid star tracker quaternions 1 second after the event, and the attitude control system recovered about 4 minutes after the event. Therefore, a control system was implemented to differentiate between internal (such as a GNC anomaly) and external (impact) events. Simplicity was prioritized over fidelity for this simulation. A simple LVLH attitude hold was used for guidance, and a simple control logic with no optimality assumption was employed. Translation GNC was not in use for the spacecraft, so it is only included in the architecture for completeness and in case of future development. However, no software has been implemented at this time.

A permanent 280 W power loss was detected due to solar array damage after the impact event. This was modeled in the output of the “Solar Panels” module. A power module in the spacecraft was included to add complexity later if needed.

A C&DH module was also included to assemble spacecraft telemetry in the specified format, fidelity, and frequency.

Figure 5 & Figure 6 provide a visual representation of the simulation architecture below.

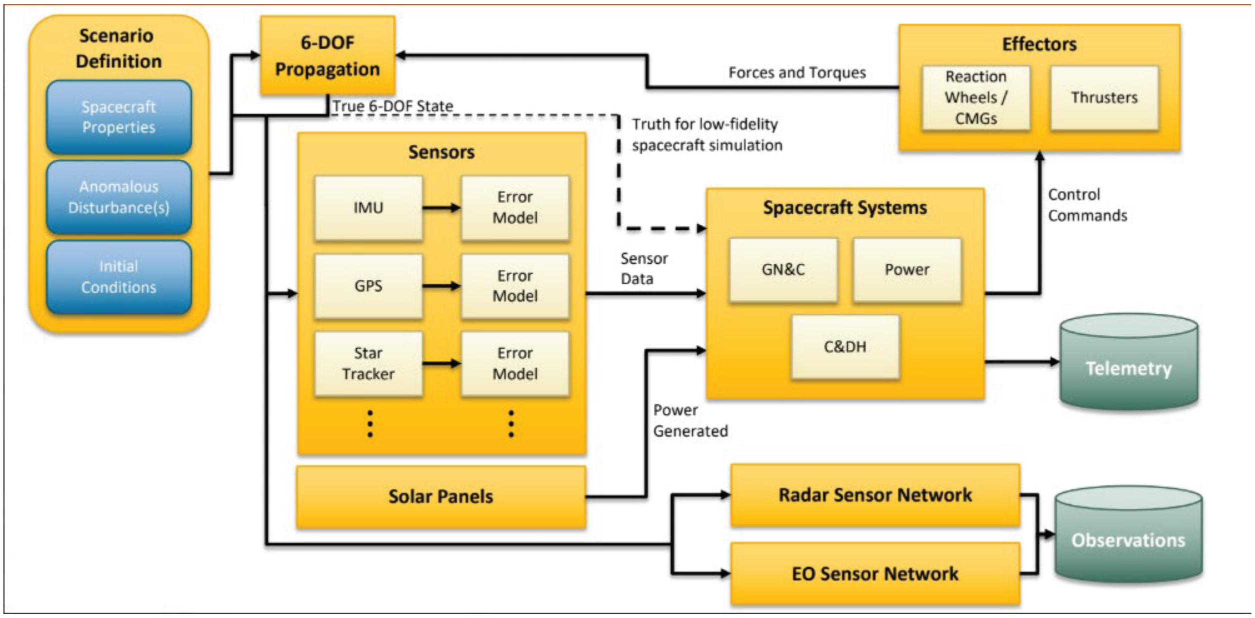


Figure 5. Overview of spacecraft simulation

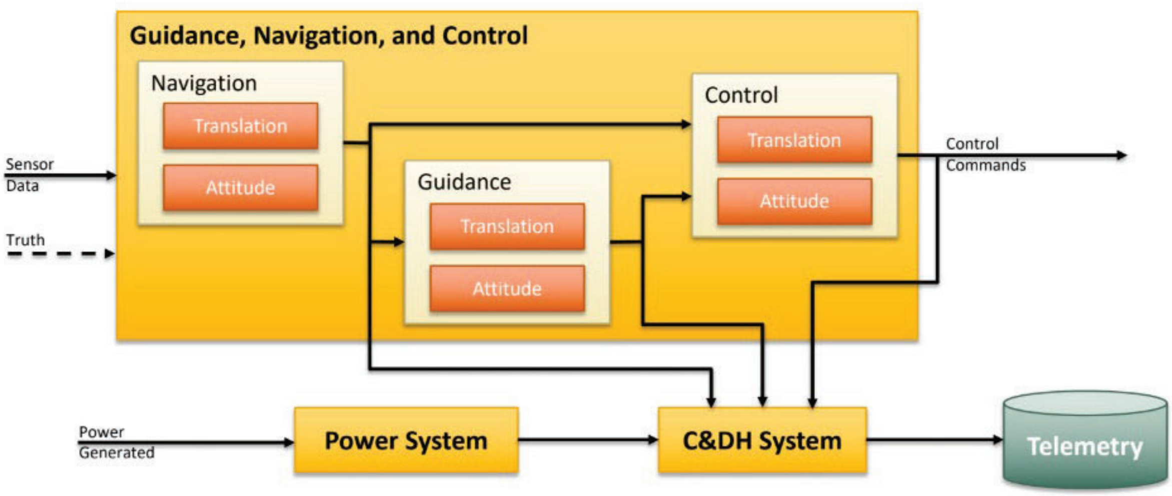


Figure 6. Detailed spacecraft systems simulation

4.1.5.3 OD/AD Tools and Rationale

- Translation State UKF
- Attitude State UKF
- Coupled Translation/Attitude State UKF

These filters were chosen to provide an efficient approach with an improved covariance realism. Process noise can be easily adjusted to maintain custody and easily identify point of divergence in the solution (real-time data integrity/fault monitoring). UT had previously implemented the translation state UKF, so it is straightforward to extend to attitude estimation. Additionally, coupled and de-coupled filters help to isolate anomalous events given differences in their responses.

- Translation State Batch Estimator

This estimator was chosen since the software is readily available and has an interface similar to the UKF. Depending on the configuration (whether data editing is enabled/disabled), it will react differently from a sequential estimator for an anomalous event. It is also a first-order filter that will yield a different solution from the UKF.

- Multiple Model Adaptive Estimator (MMAE)

This provides a bank of filters that each consider different physics-based models. It allows for flexibility in hypotheses considered at the OD/AD algorithm level. It can also identify a dynamics model that best describes the spacecraft trajectory given the available data.

4.1.6 Algorithm Design and Dockerization.

4.1.6.1.1 Translational Batch Filter Design

The batch filter attempts to estimate the initial translational state of the spacecraft, as well as its error covariance, by minimizing the weighted root-mean-square (RMS) of the measurement residuals over all the measurements. This process is repeated to refine the estimate until convergence is achieved or the filter diverges. The translational state is defined simply as

$\boldsymbol{x} \equiv [\boldsymbol{r}^T\ \dot{\boldsymbol{r}}^T]^T$, the position and velocity of the vehicle.

4.1.6.1.2 Filter Algorithm

<u>Prediction:</u> An initial augmented state vector is constructed by appending the columns of the initial state transition matrix (STM) (i.e., identity) to the *a priori* initial state estimate $\boldsymbol{\bar{x}}_0$ and nonlinearly propagated to each measurement epoch by integrating the system of differential equations:

$$\begin{cases} \dot{\boldsymbol{\bar{x}}}(\boldsymbol{t}) = \boldsymbol{f}(\boldsymbol{t}, \boldsymbol{\bar{x}}(\boldsymbol{t})) \\ \dot{\boldsymbol{\phi}}(\boldsymbol{t}) = \boldsymbol{A}(\boldsymbol{t}, \boldsymbol{\bar{x}}(\boldsymbol{t}))\boldsymbol{\phi}(\boldsymbol{t}) \end{cases} \tag{1}$$

where $f(\cdot,\cdot)$ is the dynamics model, $A(\cdot,\cdot)$ is its Jacobian, $\boldsymbol{\bar{x}}(\cdot)$ is the estimated reference trajectory, and $\phi(\cdot)$ is the STM from the initial time.

Measurement Processing: For each measurement z_i received at each measurement epoch, the predicted measurement based on the propagated reference trajectory is computed by:

$$\bar{\boldsymbol{z}}_i = \boldsymbol{h}_i\big(\boldsymbol{t}_i, \bar{\boldsymbol{x}}(\boldsymbol{t}_i)\big) + \boldsymbol{b}_i \tag{2}$$

where t_i is the measurement epoch, $h_i(\cdot,\cdot)$ is the corresponding sensor's measurement model, and b_i is its bias, if known. The measurement residual is then given by $\boldsymbol{y}_i = \boldsymbol{z}_i - \bar{\boldsymbol{z}}_i$ and its covariance by:

$$\boldsymbol{S}_i = \boldsymbol{R}_i(\boldsymbol{t}_i) + \boldsymbol{H}_i\big(\boldsymbol{t}_i, \bar{\boldsymbol{x}}(\boldsymbol{t}_i)\big)\boldsymbol{\phi}(\boldsymbol{t}_i)\bar{\boldsymbol{P}}_0\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{t}_i)\boldsymbol{H}_i^T\big(\boldsymbol{t}_i, \bar{\boldsymbol{x}}(\boldsymbol{t}_i)\big) \tag{3}$$

where $R_i(\cdot)$ is the covariance of the sensor's measurement uncertainty, $H_i(\cdot,\cdot)$ is the Jacobian of its measurement model, and $\bar{\boldsymbol{P}}_0$ is the *a priori* initial estimate error covariance. After all measurement residuals and their covariances have been processed, those not satisfying the following condition, for some constant ε, are removed:

$$\sqrt{\boldsymbol{y}_i^T \boldsymbol{S}_i^{-1} \boldsymbol{y}_i} < \boldsymbol{\varepsilon} \tag{4}$$

This is done to improve robustness in the presence of outliers, which can seriously degrade filter performance. Finally, the estimated initial covariance is updated by:

$$\widehat{\boldsymbol{P}}_0^{-1} = \bar{\boldsymbol{P}}_0^{-1} + \textstyle\sum_{i=1}^{N} \boldsymbol{H}_i^T(\boldsymbol{t}_i)\boldsymbol{R}_i^{-1}(\boldsymbol{t}_i)\boldsymbol{H}_i(\boldsymbol{t}_i) \tag{5}$$

And the estimated initial state deviation is given by:

$$\boldsymbol{\Delta}\widehat{\boldsymbol{x}}_0 \equiv \widehat{\boldsymbol{x}}_0 - \bar{\boldsymbol{x}}_0 = \widehat{\boldsymbol{P}}_0\big(\bar{\boldsymbol{P}}_0^{-1}\boldsymbol{\Delta}\bar{\boldsymbol{x}}_0 + \textstyle\sum_{i=1}^{N} \boldsymbol{H}_i^T(\boldsymbol{t}_i)\boldsymbol{R}_i^{-1}(\boldsymbol{t}_i)\boldsymbol{y}_i\big) \tag{6}$$

where $\Delta\ \bar{\ }_0$ is the *a priori* estimated initial deviation, which is zero in the first iteration.

Reinitialization: To initialize the next filtering iteration, the prior estimated initial state is incremented by the updated estimated deviation while the prior estimated initial deviation is decremented by the same amount, and the prior covariance is kept at its original value:

$$\begin{aligned} \bar{\boldsymbol{x}}_0 &\leftarrow \bar{\boldsymbol{x}}_0 + \boldsymbol{\Delta}\widehat{\boldsymbol{x}}_0 \equiv \widehat{\boldsymbol{x}}_0 \\ \boldsymbol{\Delta}\bar{\boldsymbol{x}}_0 &\leftarrow \boldsymbol{\Delta}\bar{\boldsymbol{x}}_0 - \boldsymbol{\Delta}\widehat{\boldsymbol{x}}_0 \\ \bar{\boldsymbol{P}}_0 &\leftarrow \bar{\boldsymbol{P}}_0 \end{aligned} \tag{7}$$

Exit Conditions: The weighted RMS of the measurement residuals is given by:

$$\mathbf{RMS} = \sqrt{\frac{\boldsymbol{yR^{-1}y}}{\boldsymbol{N}}} \tag{8}$$

If this value changes by less than some small constant from one iteration to the next, the filter is considered to have converged and returns the estimated initial state and covariance ($\boldsymbol{x}\hat{}_0$ and $\widehat{\boldsymbol{P}}_0$) If, instead, the norm of the state deviation exceeds some large constant, the filter is considered to be diverging. Finally, if all measurements are rejected by the test in Equation (1), the filter is considered to have failed to converge.

4.1.6.2 Multiple Model Adaptive Estimation Framework Design

Multiple model estimation techniques facilitate the estimation of model parameters that are not well-suited to inclusion in the estimated state. Multiple model adaptive estimation (MMAE) attempts to select the most likely model given a discrete set of system models. To achieve this, a bank of independent filters is initialized, each assuming a different model. The filters propagate their state estimates from one measurement epoch to the next and perform their measurement update. The probability of each model is then updated using its respective filter's measurement residuals and residuals covariance. Finally, the individual filters' estimates are fused to produce a single estimate of the state PDF, and the next iteration begins.

Model Probability Update: Given the prior model probabilities $\mu_k^{(i)}$ at time t_k for a bank of N filters with model indices $i \in \{1, \dots, N\}$, the updated model probabilities are given by:

$$\boldsymbol{\mu}_{\boldsymbol{k+1}}^{(\boldsymbol{i})} = \frac{\boldsymbol{\mu}_{\boldsymbol{k}}^{(\boldsymbol{i})} \boldsymbol{L}_{\boldsymbol{k}}^{(\boldsymbol{i})}}{\sum_{\boldsymbol{j=1}}^{\boldsymbol{N}} \boldsymbol{\mu}_{\boldsymbol{k}}^{(\boldsymbol{j})} \boldsymbol{L}_{\boldsymbol{k}}^{(\boldsymbol{j})}} \tag{9}$$

Where $L_k^{(i)}$ is the model likelihood, $p\left(z_k \middle| \hat{x}_k^{(i)}\right)$.

Output Fusion and Attitude Estimation: The state estimate output by the MMAE estimator at each time step is the average of the state estimates $\hat{x}_k^{(i)}$ produced by each filter, weighted their respective by model probabilities:

$$\widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\mu}_{\boldsymbol{k}}^{(\boldsymbol{i})} \widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})} \tag{10}$$

$$\widehat{\boldsymbol{P}}_{\boldsymbol{k}} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\mu}_{\boldsymbol{k}}^{(\boldsymbol{i})} \left(\widehat{\boldsymbol{P}}_{\boldsymbol{k}}^{(\boldsymbol{i})} + (\widehat{\boldsymbol{x}}_{\boldsymbol{k}} - \widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})})(\widehat{\boldsymbol{x}}_{\boldsymbol{k}} - \widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})})^{\boldsymbol{T}} \right) \tag{11}$$

This arithmetic averaging breaks down when attitude states are included in the state vector definition. To account for this, Markley et al.'s method for computing the scalar-weighted average quaternion can be used for the attitude portion of the state. In this method, the estimated quaternion is given by the unit eigenvector of the following matrix corresponding to its maximum eigenvalue:

$$\boldsymbol{M} \equiv \sum_{i=1}^{N} \boldsymbol{\mu}_k^{(i)} \hat{\boldsymbol{q}}_k^{(i)} \hat{\boldsymbol{q}}_k^{(i)T} \tag{12}$$

The angular rates and other state components can still be averaged as usual. The covariance fusion in Equation (2) above can then be replaced by:

$$\hat{\boldsymbol{P}}_k = \sum_{i=1}^{N} \boldsymbol{\mu}_k^{(i)} \left(\hat{\boldsymbol{P}}_k^{(i)} \boldsymbol{\delta x}_k^{i} \boldsymbol{\delta x}_k^{(i)T} \right) \tag{13}$$

Where the error vector δx_k^i is defined as:

$$\boldsymbol{\delta x}_k^{i} = \begin{bmatrix} \hat{\boldsymbol{r}}_k^{(i)} - \hat{\boldsymbol{r}}_k \\ \dot{\hat{\boldsymbol{r}}}_k^{(i)} - \dot{\hat{\boldsymbol{r}}}_k \\ \boldsymbol{\delta} \hat{\boldsymbol{\theta}}_k^{(i)} \\ \hat{\boldsymbol{\omega}}_k^{(i)} - \hat{\boldsymbol{\omega}}_k \end{bmatrix} \tag{14}$$

Where $\delta\hat{\boldsymbol{\theta}}_k^{(i)}$ is the vector of approximate error angles between the mean quaternion and the estimate of filter i.

4.1.6.3 Unscented Kalman Filter Design

The unscented Kalman filter (UKF) uses the unscented transform (UT) to sequentially estimate the state of the spacecraft subject to nonlinear dynamics and measurement models. Initially, the state will be defined as the translational state of the vehicle, $\boldsymbol{x} \equiv [\boldsymbol{r}^T\ \dot{\boldsymbol{r}}^T]^T$, but Section 4.3.3 will discuss modifications to the UKF to allow estimation of the full 6-DOF state including attitude and angular rates. This filter uses an α, β, κ parametrization to determine the sigma point weights for the UT, with α = 1, β = 2, κ ≡ 3 – *N*, where *N* is the number of elements in the augmented state vector.

<u>Prediction</u>*:* For prediction, an augmented state vector is formed by appending a realization of the process noise vector $\boldsymbol{w}_k$ to the dynamic state, with a block-diagonal augmented covariance with the prior estimated error covariance $\hat{\boldsymbol{P}}_k$ in the upper block and the process noise covariance $\boldsymbol{Q}_k$ in the lower block. This augmented PDF is then propagated through the dynamic model using the UT:

$$\dot{\boldsymbol{x}}_{\boldsymbol{k}}^{(i)} = \boldsymbol{f}\left(\boldsymbol{t}_{\boldsymbol{k}}, \boldsymbol{x}_{\boldsymbol{k}}^{(i)}\right) + \boldsymbol{w}_{\boldsymbol{k}}^{(i)} \tag{15}$$

where $\boldsymbol{f}(\cdot,\cdot)$ is the dynamics model and:

$$\boldsymbol{\chi}_{\mathbf{k}}^{(i)} \equiv \begin{bmatrix} \boldsymbol{x}_{\boldsymbol{k}}^{(i)} \\ \boldsymbol{w}_{\boldsymbol{k}}^{(i)} \end{bmatrix} \tag{16}$$

is the i^{th} sigma point of the augmented PDF.

Measurement Update*:* The measurement update is performed similarly, but with the state augmented by the addition of the measurement noise vector $\boldsymbol{v}_k$, which is assumed to be zero-mean Gaussian, with covariance $\boldsymbol{R}_k$:

$$\boldsymbol{z}_{\boldsymbol{k}}^{(i)} = \boldsymbol{h}\left(\boldsymbol{t}_{\boldsymbol{k}}, \boldsymbol{x}_{\boldsymbol{k}}^{(i)}\right) + \boldsymbol{b}_{\boldsymbol{k}} + \boldsymbol{v}_{\boldsymbol{k}}^{(i)} \tag{17}$$

$$\boldsymbol{\chi}_{\mathbf{k}}^{(i)} \equiv \begin{bmatrix} \boldsymbol{x}_{\boldsymbol{k}}^{(i)} \\ \boldsymbol{v}_{\boldsymbol{k}}^{(i)} \end{bmatrix} \tag{18}$$

is the i^{th} sigma point of the augmented PDF.

Modifications to Support Attitude Estimation: The UT model is made more complicated when attitude states are included in the estimated state vector. To account for them, first redefine the estimated state vector as $\boldsymbol{x} \equiv [\boldsymbol{r}^T\ \dot{\boldsymbol{r}}^T\ \boldsymbol{q}^T\ \boldsymbol{\omega}^T]^T$**,** with the attitude and angular rates appended to the translational state vector. For the purposes of performing the UT and defining the state error covariance, the error state vector will be used instead, defined as $\delta\boldsymbol{x} \equiv [\Delta\boldsymbol{r}^T\ \Delta\dot{\boldsymbol{r}}^T\ \delta\boldsymbol{\theta}^T\ \Delta\boldsymbol{\omega}^T]^T$, where $\delta\boldsymbol{\theta}$ is the vector of error angles between the quaternion $\boldsymbol{q}$ and the current estimated quaternion $\boldsymbol{q}^{-}{}_k$ (pre-update) or $\boldsymbol{q}^{\wedge}{}_k$ (post-update), while the other sub-vectors are defined as $\Delta\mathbf{r} \equiv \mathbf{r} - \bar{\mathbf{r}}_{\mathrm{k}}$ (pre-update) or $\Delta\mathbf{r} \equiv \mathbf{r} - \boldsymbol{r}^{\wedge}{}_{\mathrm{k}}$ (post-update). In practice, the error angles are approximated by the error generalized Rodrigues parameters (GRPs), multiplied by a scaling constant. In practice, the use of the error state only affects the handling of the attitude vector elements.

In the UT, the sigma points are generated with different realizations of $\delta\boldsymbol{\theta}$, which can be converted into an approximate error quaternion and used with the estimated quaternion as an input to the nonlinear transformation. If the output of the transformation contains another attitude quaternion, the approximate error angles are calculated again, now with respect to the mean sigma point $\boldsymbol{\chi}_k^{(0)}$. The error state sigma points are used to calculate the transformed mean and covariance, and the new estimated quaternion can be calculated by converting the mean of the error angles into an approximate error quaternion and using it to rotate the reference quaternion:

$$\boldsymbol{q} = \boldsymbol{\delta q}(\boldsymbol{\delta\theta})\boldsymbol{q}_{\boldsymbol{k}}^{(\mathbf{0})} \quad \textbf{(19)}$$

4.1.6.4 Recurrent Neural Network for Power Drop Detection

A recurrent neural network was developed in python to predict whether a power drop has occurred during the simulation. The network was trained using keras, which is a wrapper library for several different neural network programming backends. In this case, the tensorflow backend was used. The code outputs a Nx3 array text file to be used for the ISV system, along with precision and recall scores to evaluate the model performance. Whether or not a drop is detected is also stated, and where the drop occurs, if at all.

Test Data Format: The test data was formatted the following way as a *.pmg2* file: (Table 1.)

Table 1. Test data format

Byte 1	Char '0', '1', '2'; label for following data
Bytes 2-6	Char left zero padded value for number of samples 'd'
Byte 7	Char 'x'
Bytes 8-9	Char left zero padded row range of data 'h'
Byte 10	Char 'x'
Bytes 11-14	Char left zero padded column range of data 'w'
Bytes 15-4dw*h	Lsb 32-bit float of data

The above then repeats for each class until the end of the file

Data Description & Modification:

Training Data Description: The power history of 20,000 spacecraft was generated using an independent simulation from that used for evaluating use-case #1 and recorded for 700 seconds in 1 second intervals. The input training data shape for this network is 20,000x700x1. The labeling input training data shape is 20,000x700x2 defined as:

- If power is nominal at each second, label is: [1,0]
- If power drops at each second, label is: [0,1]

Only one power drop was simulated for each 700-second spacecraft sequence.

Modification: In order to modify the data in a usable way, it was split into 20 second intervals, each with a 19 second overlap as follows:

- Sequence 1 = [time step 1, time step 2, …, time step 20]
- Sequence 2 = [time step 2, time step 3, …, time step 21]
- The rest follow similarly

Each sequence was a 3D array with dimensions:

- Dimension 1 – length of 20,000 x 681 sequences of spacecraft time envelopes
- Dimension 2 – length of 20, each representing 1 second
- Dimension 3 – length of 1 for features and 2 for labeling

Power was also normalized to a value between 0 and 1 for training.

Test and Validation Data: To generate testing data, the power histories of 5,201 spacecraft were simulated and recorded for 700 seconds in 1 second intervals. The data were modified in the same manner as the training data.

4.1.6.5 RNN Design & Performance

Design: The RNN consists of the following layers and activations:

- 2 long-short-term memory layers with 20 neurons each
- 2 dropout layers where 20% of the previous layer's weights are dropped
- 2 standard feedforward layers with 20 neurons each
- Activation of the first used relu (rectified linear unit)
    - Activation of the last was softmax
    - Activation is not needed after the LSTM layers since the default is tanh

A design summary of the RNN can be seen below in Table 2.

Table 2. RNN Architecture

Layer (Type)	Output Shape	Parameter Number
lstm_1 (LSTM)	(None, 20, 20)	1,760
dropout_1 (Dropout)	(None, 20, 20)	0
dense_1 (Dense)	(None, 20, 20)	420
activation_1 (Activation)	(None, 20, 20)	0
lstm_2 (LSTM)	(None, 20, 20)	3,280
dropout_2 (Dropout)	(None, 20, 20)	0
dense_2 (Dense)	(None, 20, 2)	42
activation_2 (Activation)	(None, 20, 2)	0

Performance: Final epoch training statistics:

- Loss: 0.05
- Accuracy: 99.9%

Overall performance:

- Close to 100% of power data at any time step identified as nominal or drop were correct
- Close to 100% of power data identified as nominal were correct
- About 95% of the power data identified as a drop were correct

The network confusion matrix is also shown in Figure 7.

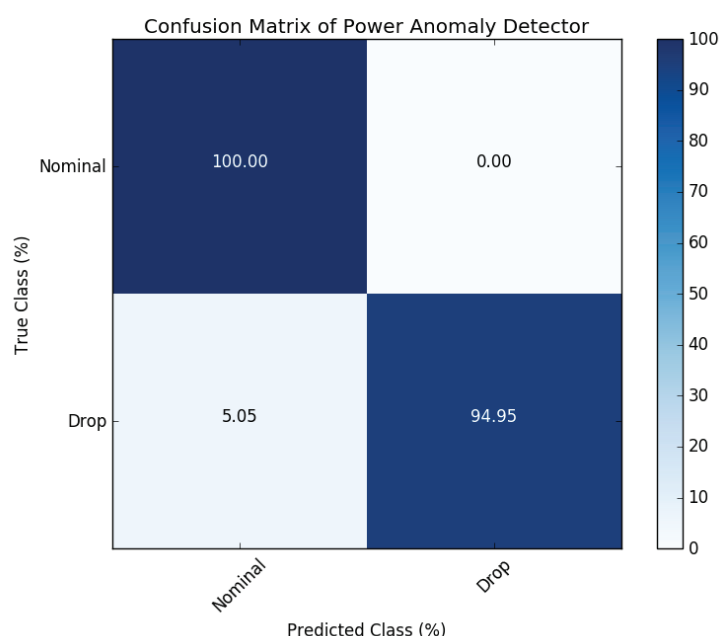


Figure 7. Confusion Matrix

Approved for public release; distribution unlimited.

4.1.6.6 Quaternion Anomaly Detection with Machine Learning

Training/testing data for 100,000 spacecraft simulated for 700 seconds in 1 second intervals was generated for this network. Each spacecraft will belong to one of the following spacecraft attitude behaviors:

- Nominal Case: Spacecraft is nadir tracking
- Perturbed Case: Spacecraft experiences a random, instantaneous change in attitude
- Slew Case: Spacecraft slews to a different attitude
- Tumble Case: Spacecraft gradually begins to tumble

The number of satellites belonging to each category can be found below:

- Nominal: 24,861
- Perturbed: 25,202
- Slew: 25,148
- Tumble: 24,789

Test and Validation Data: The data was then split into 20,000 training satellites per category, a total of 80,000. The remaining 20,000 satellites were used for testing. The number of satellites that belong to each category in the testing set are below:

- Nominal: 4,861
- Perturbed: 5,202
- Slew: 5,148
- Tumble: 4,789

CNN Design: The following is considered:

- Each training sample was a 4 x 700 "image" of quaternion time history
- 4x 2D convolution layers
- One densely connected layer
- One readout layer
- Three separate dropouts from 0.2-0.5 keep probability
- One max pooling layer
- Adam optimizer with softmax cross entropy loss

CNN Performance Evaluation: Model accuracy, loss, and confusion matrix are shown in Figure 8, Figure 9 and Figure 10 for performance evaluation.

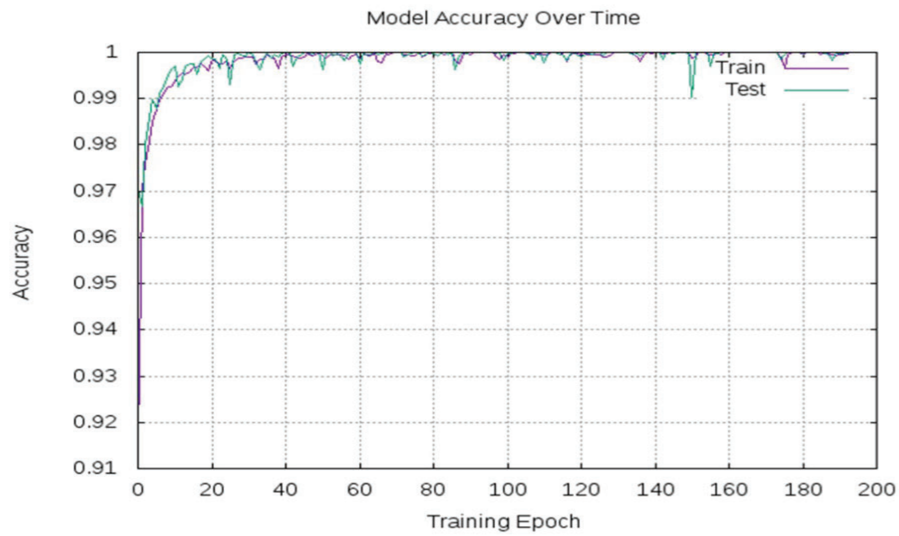


Figure 8. CNN Model Accuracy

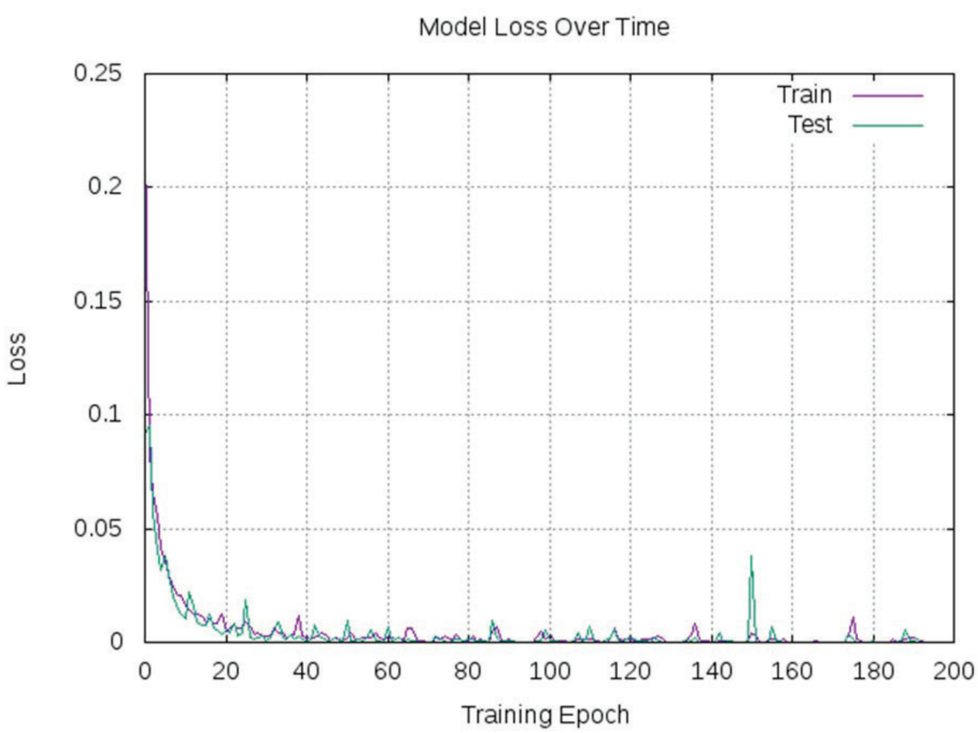


Figure 9. CNN Training Loss

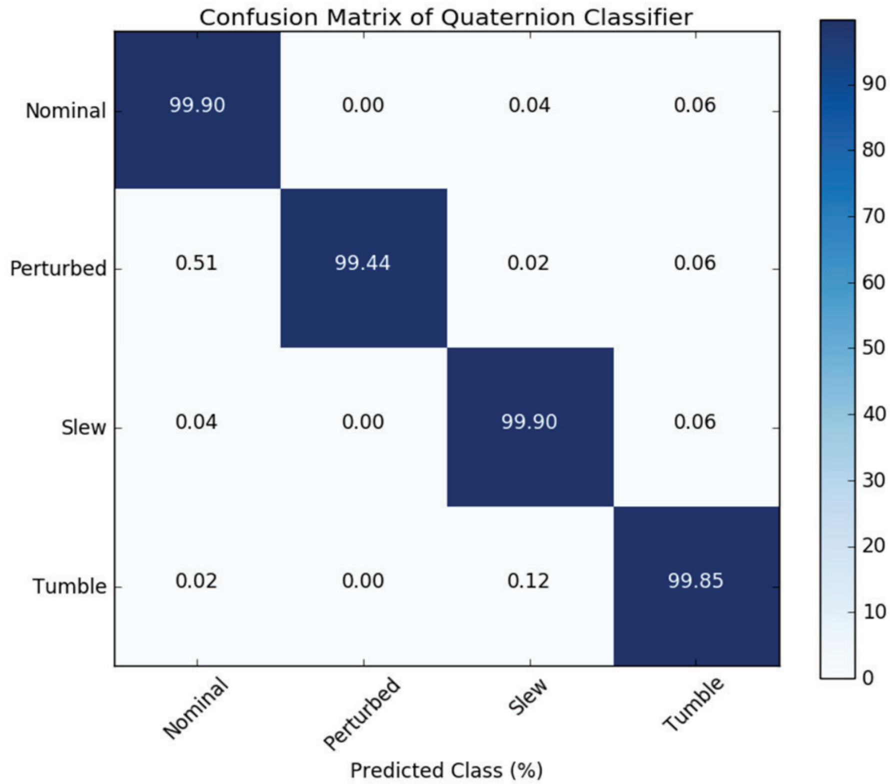


Figure 10. CNN Confusion Matrix

Testing statistics:

- 20,000 satellites classified with > 99% accuracy in < 30 seconds

4.1.6.7 Behavior Anomaly Detection

Training/testing data for 100,000 spacecraft simulated for 700 seconds in 1 second intervals was generated for this network. An 11 element spacecraft sate vector (**r**,**v**,**q**,b) was used for the training of each spacecraft.

- Spacecraft can belong to three of eight classes simultaneously:
  - Two classes for the translation state
  - Four classes for the attitude state
  - Two classes for the power state
- Each spacecraft belongs to one of the following attitude classes:
  - Nominal Case: Spacecraft is nadir tracking
  - Perturbed Case: Spacecraft experiences a random, instantaneous change in attitude
  - Slew Case: Spacecraft slews to a different attitude
  - Tumble Case: Spacecraft gradually begins to tumble

- Each spacecraft belongs to one of the following spacecraft power classes:
    - No change: Spacecraft does not experience any abnormal drop in power
    - Change: Spacecraft experiences a random, instantaneous change in power
- Each spacecraft belongs to one of the following translation classes:
    - Nominal: Spacecraft is translating "normally"
    - Maneuver: Spacecraft performs a translation maneuver

<u>Test and Validation Data:</u> The data was then split into 20,000 training satellites per category, a total of 80,000. The remaining 20,000 satellites were used for testing. The number of satellites that belong to each category in the testing set are below:

- Nominal: 4,861
- Perturbed: 5,202
- Slew: 5,148
- Tumble: 4,789

<u>CNN Performance Evaluation</u>: This network converged quickly and tested at > 99.7% prediction accuracy (Figure 11).

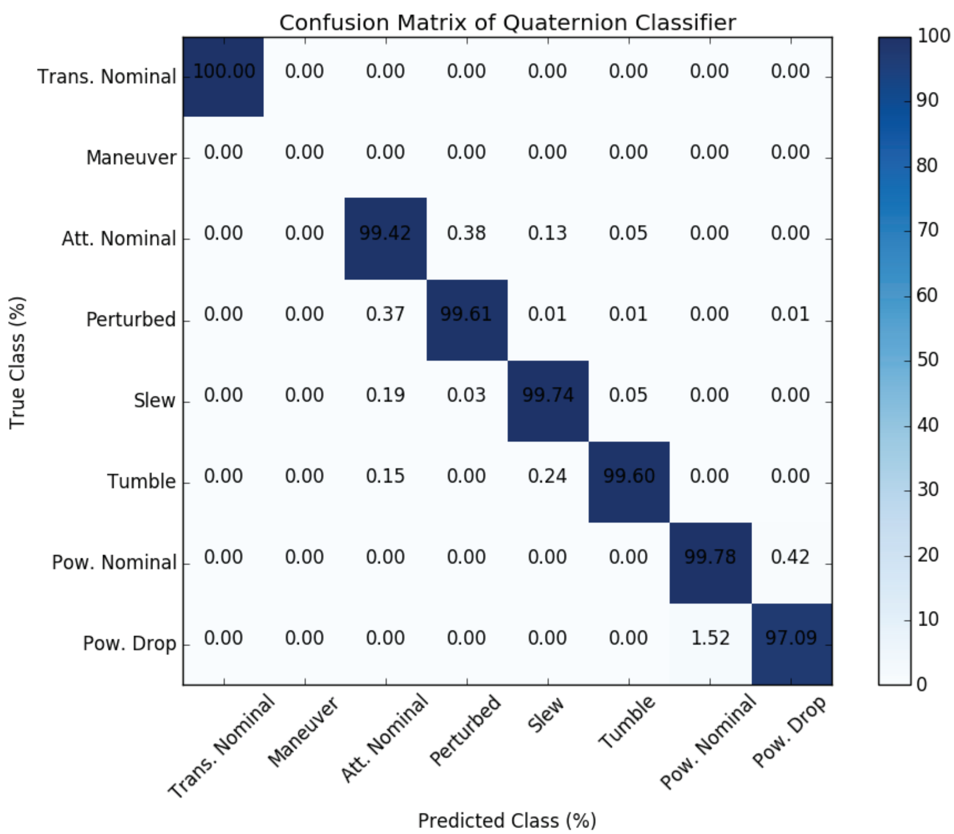


Figure 11. CNN Quaternions Confusion matrix

Note: There were no spacecraft's in the "Maneuver" class

4.1.6.8 Dockerization in the Discovery Environment

The below instructions provide a guide for creating an app in the CyVerse Discovery Environment (DE):

1. Create CyVerse account at https://user.cyverse.org/register

2. Identify the necessary software (e.g. dependencies, source code, etc.)

3. Test and build software in Docker
   a) Docker installation instructions are found here:
      i. Mac: https://docs.docker.com/docker-for-mac/install/
      ii. Windows: https://docs.docker.com/docker-for-windows/install/
      iii. Ubuntu or similar: https://docs.docker.com/install/linux/docker-ce/ubuntu/
   b) Create dockerfiles and push to dockerhub
   c) Identify test data and instructions to test the docker image

4. Push docker images to dockerhub

5. Define a "DE Tool" in the VerSSA Discovery Environment and input the requested information - Click "Apps->Manage Tools->Tools->Add tool…" (Figure 12).

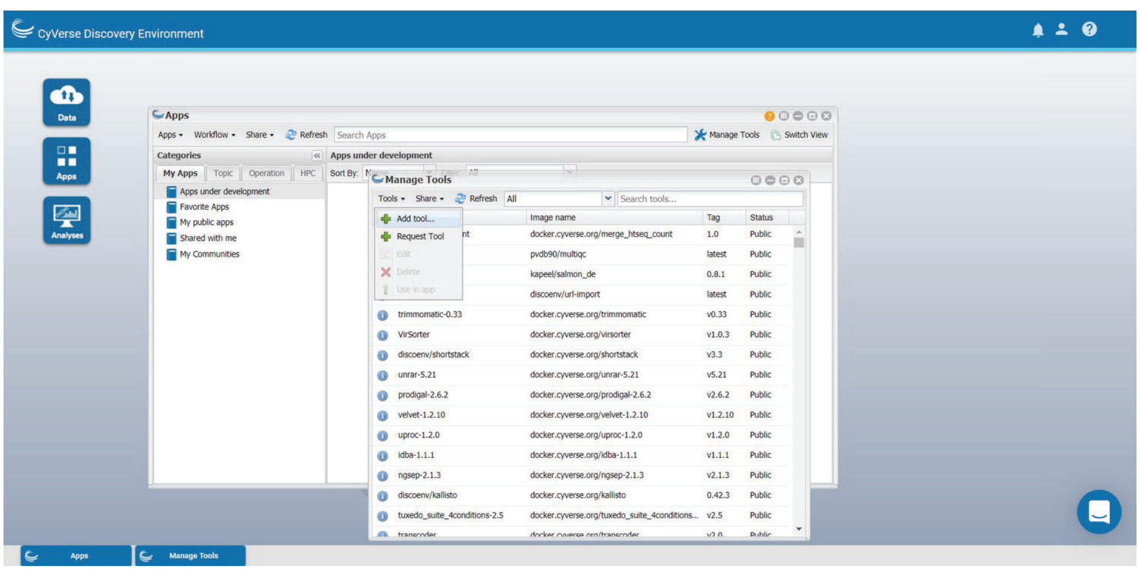


Figure 12. Discovery Environment Tool: Input requested information

6. Define a "DE App" in VerSSA Discovery Environment and input requested information - Click "Apps->Apps->Create new…" (Figure 13).

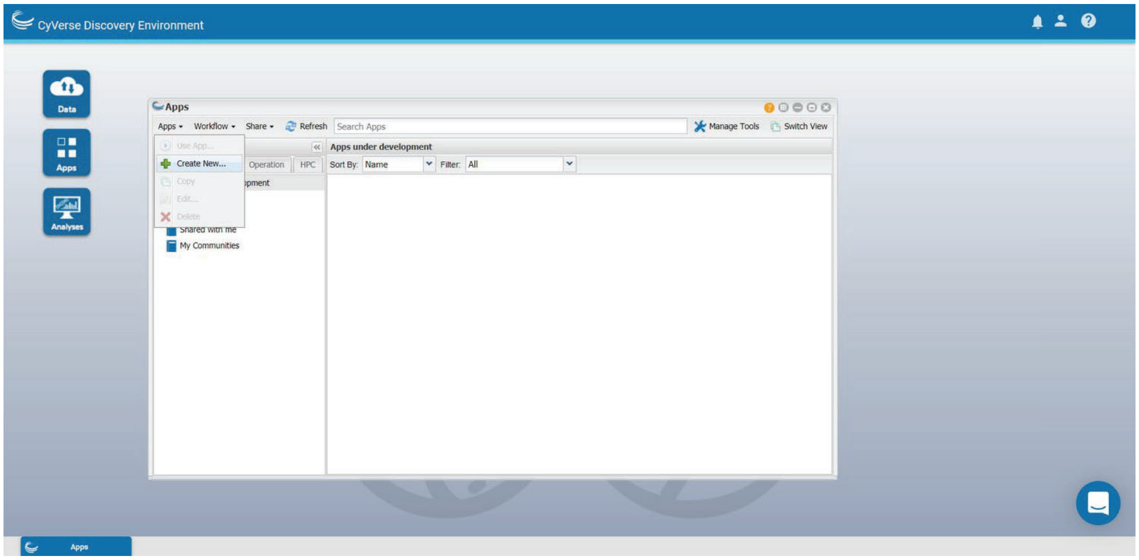


Figure 13. Discovery Environment App: Input requested information

7. Search for newly created app in the search bar.
8. The following link provides additional documentation regarding dockerization: https://docs.docker.com/get-started/

4.1.7 Interfaces and Automation.

4.1.7.1 VerSSA Automation Process

The automation process has been articulated according to the following steps:

1. The simulation generates spacecraft telemetry and observational data to be used by the filters and deep networks.

2. Data are then uploaded via Python script VerSSA for analysis

3. A daemon process scans for new data periodically, and initializes the appropriate apps to process the data (e.g. quaternion classifier, behavior network, etc.)

   - The current method of automation requires that each app to be run takes a unique file type as input. This is adhoc to our process, but can easily be adapted to suit any general app in VerSSA.
   - We have also set up a process lock file and processing log file so that the daemon does not run into itself and knows what data is already processed.

4. Once the filter output data is available, the daemon will reformat it and feed it to the behavior classification network.

5. Outputs from the filters and all deep networks are then available for the ISV system to ingest.

4.1.8 Results and Analysis.

4.1.8.1 ConOps Execution

An independent UA spacecraft simulation was used to generate training data for the neural networks. As described above, 100,000 satellites were simulated for 700 seconds in 1 second intervals to use for training, testing, and validation of the quaternion, power, and behavior neural networks. Once the networks were fully trained, with performance described above, the UT simulation of the Sentinel-1A impact event was run. Position, velocity, attitude, and power were all recorded over a 3,500 second interval in 700 second batches. The impact event occurred at 2,000 seconds where the spacecraft experienced a sudden sharp drop in power and an attitude shift.

The data from the UT simulation were uploaded to VerSSA for processing by the quaternion and power state classifiers. Once these completed their analyses, the behavior neural network was triggered by any anomalous event. Additionally, all outputs were available to be processed by the ISV-SSA system to show a 3D virtual reality representation of the simulation and network outputs. The user can go forward or backward in time and move around freely in the ISV environment to confirm the spacecraft behavior.

4.1.8.2 Spacecraft Behavior Analysis

The results for the Sentinel-1A impact use-case are shown below in five different 700-second batches. We discovered after the test was performed, that there was a fundamental difference between how the attitude information was stored between the simulated test data what the networks expected. There are two conventions for expressing quaternions in the literature, and during our analysis of our test results, we discovered that we each used different conventions. This is likely the reason that the attitude classification network insists that the spacecraft is "slewing" when it is simulated as nadir tracking and does not detect the minute attitude shift when the impact occurs. It is also important to note that the spacecraft states that are plotted (bottom right plot of each quartet) is the filtered state output. This is not the same as the attitude estimates that the attitude classification network sees which will, in general, be coarser and less smooth than the filtered states. This is also likely a contributing factor to the misidentification by the attitude network.

0-700 seconds: For the first 700 seconds of the simulation, the spacecraft is supposed to be orbiting normally while nadir tracking. As can be seen by the power state information in the last plot in the bottom right corner, the spacecraft emerges from the umbra of the Earth at around the 550 second mark (Figure 14).

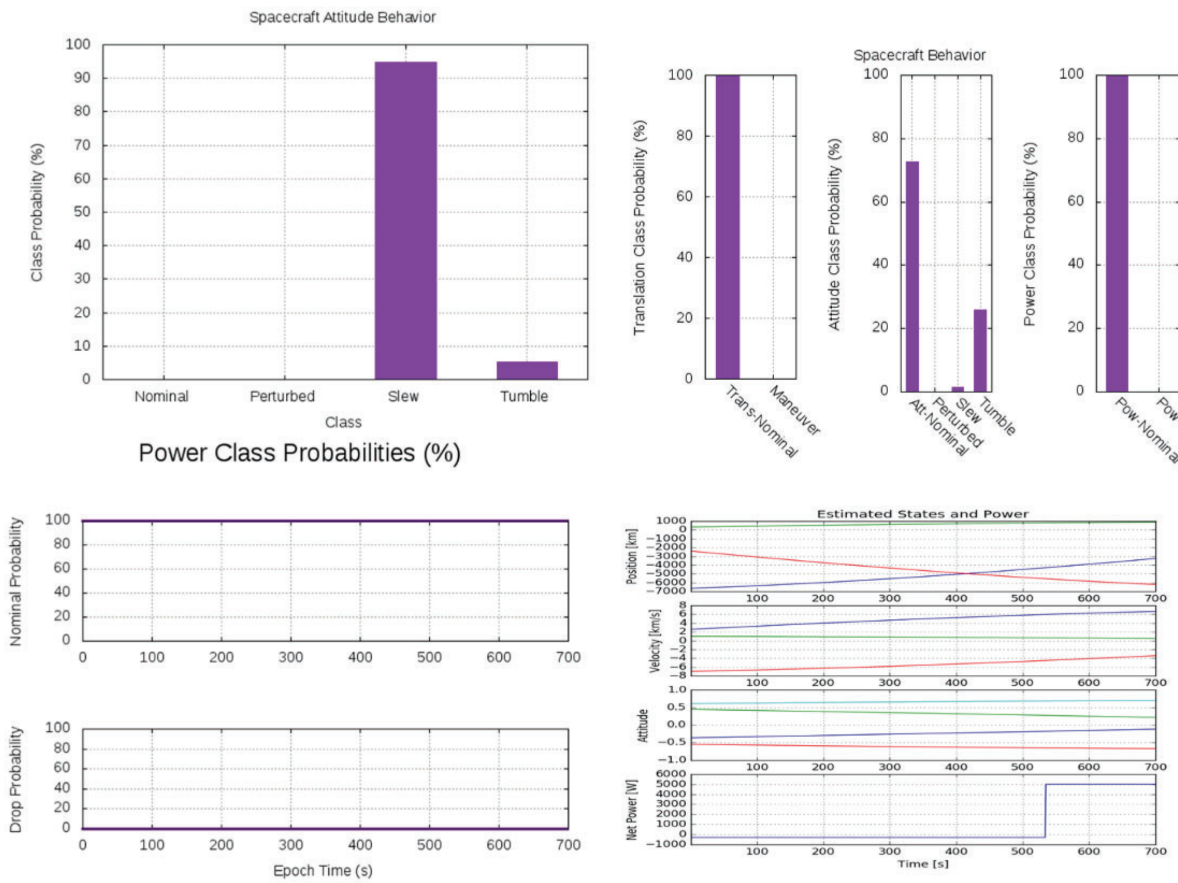


Figure 14. 0-700 sec Prediction Results

<u>700 – 1400 seconds:</u> During the next 700 seconds, we see that the power networks still strongly think the spacecraft is behaving normally, which it is, but we see some high frequency shifts in the classification probability. This is due to a training artifact where the power consumption data used to train the network was much smoother do to being battery discharge state and not direct power in minus power out. The data used to train the network had the benefit of being integrated and did not illustrate some of the staccato nature of direct power monitoring (Figure 15).

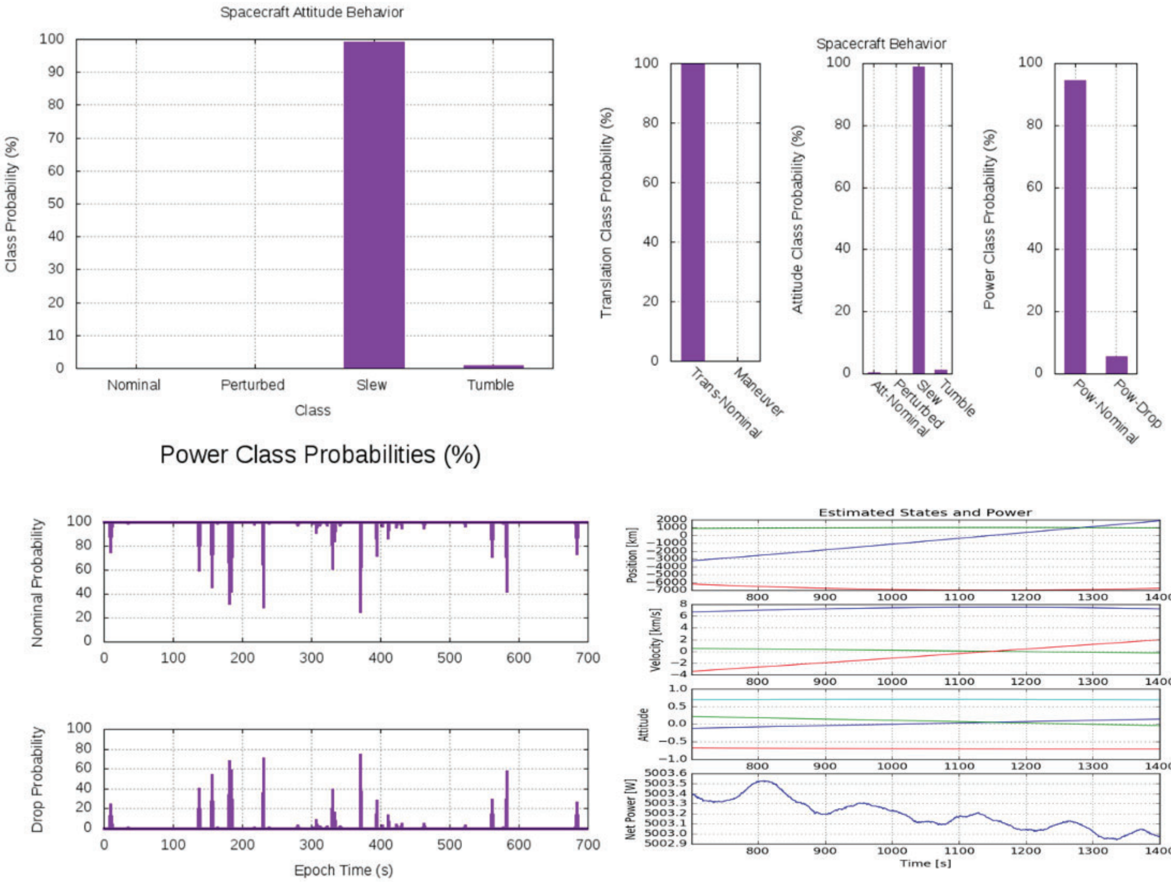


Figure 15. 700-1400 sec Prediction Results

<u>1400 – 2100 seconds:</u> It was during this time window that the impact event occurred. As we can see, the power network very clearly detects the drop in power and informs the user of the anomalous event (Figure 16).

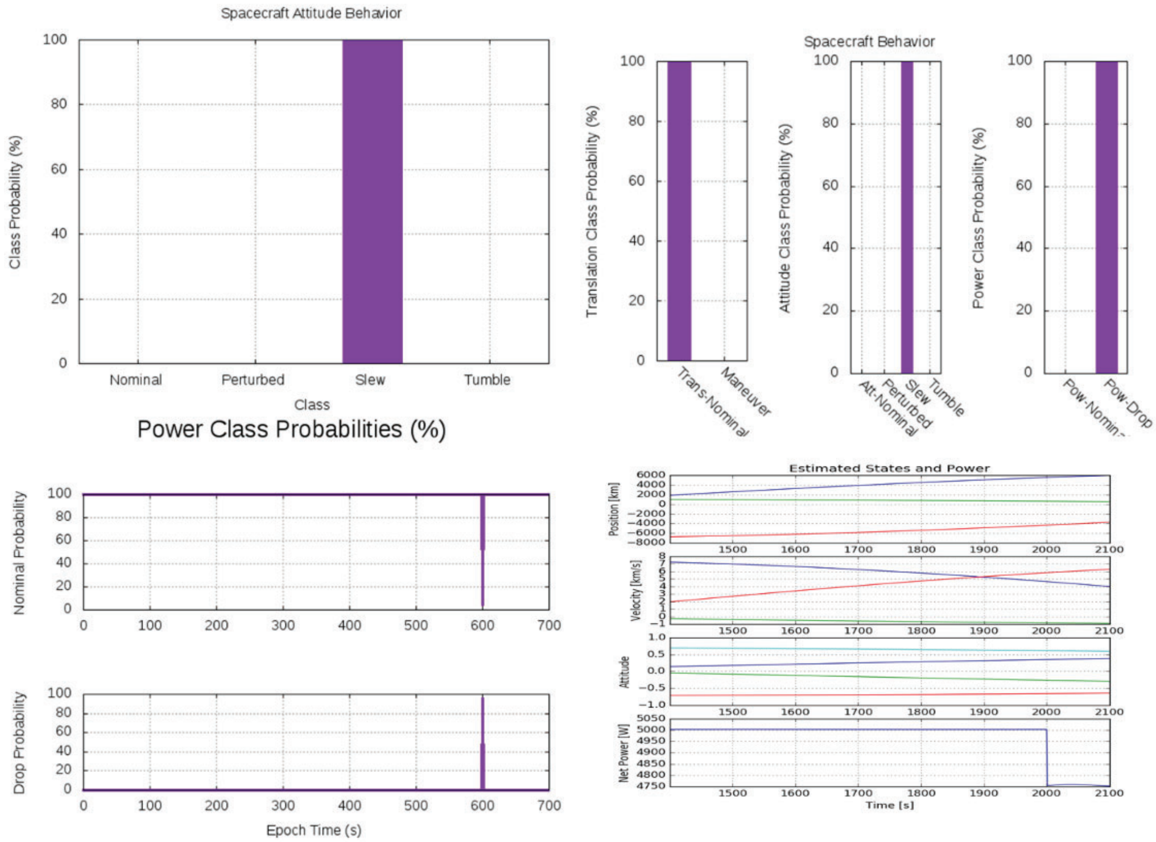


Figure 16. 1400-2100 sec Prediction Results

<u>2100 – 2800 seconds:</u> As the spacecraft returns to normal operation, we can see that the networks agree that the event is over, and the behavior classification network in the bottom left identifies that while the power state has returned to normal, it is damaged (Figure 17).

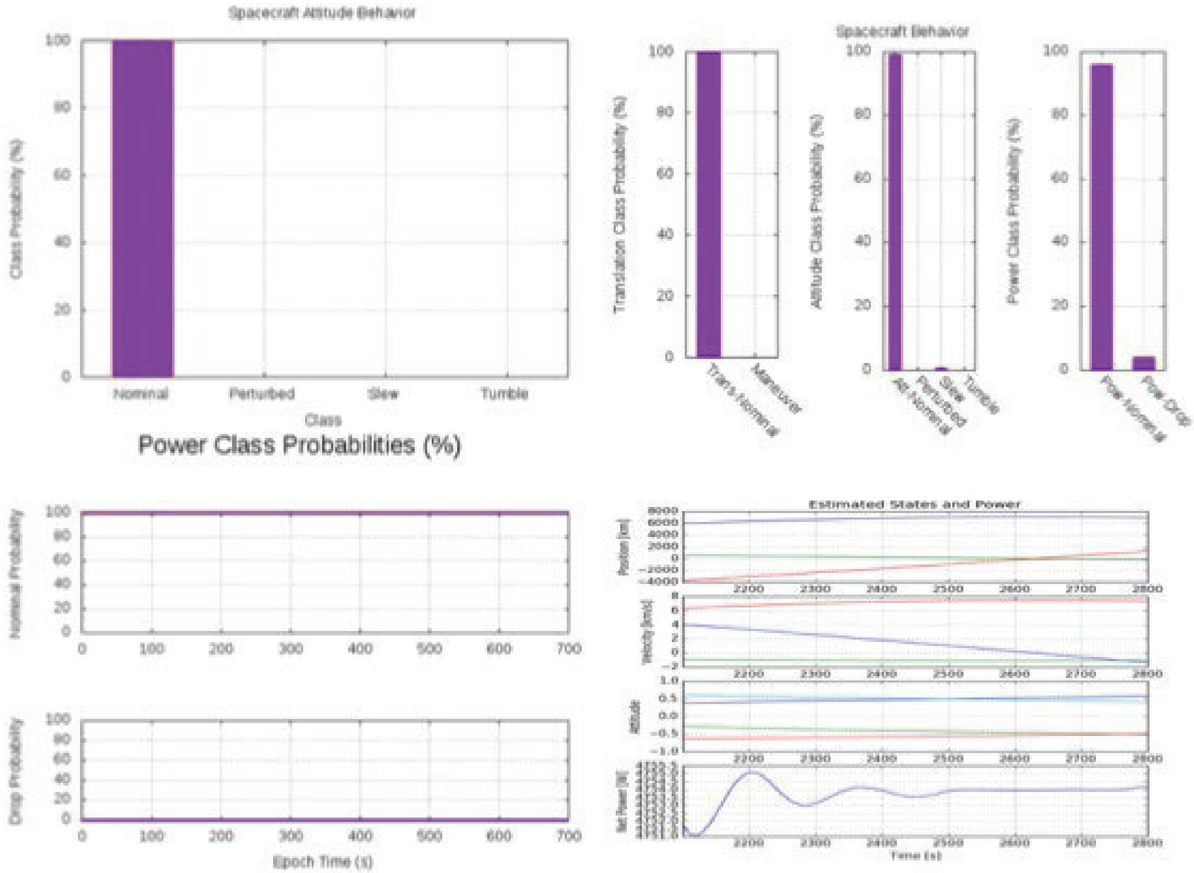


Figure 17. 2100-2800 sec Prediction Results

2800 – 3500 seconds: We can again see that the networks correctly identify the spacecraft as behaving normally, with slight damage to the power system. The same training artifact is apparent in the power system network, and this is easily remedied in future versions (Figure 18).

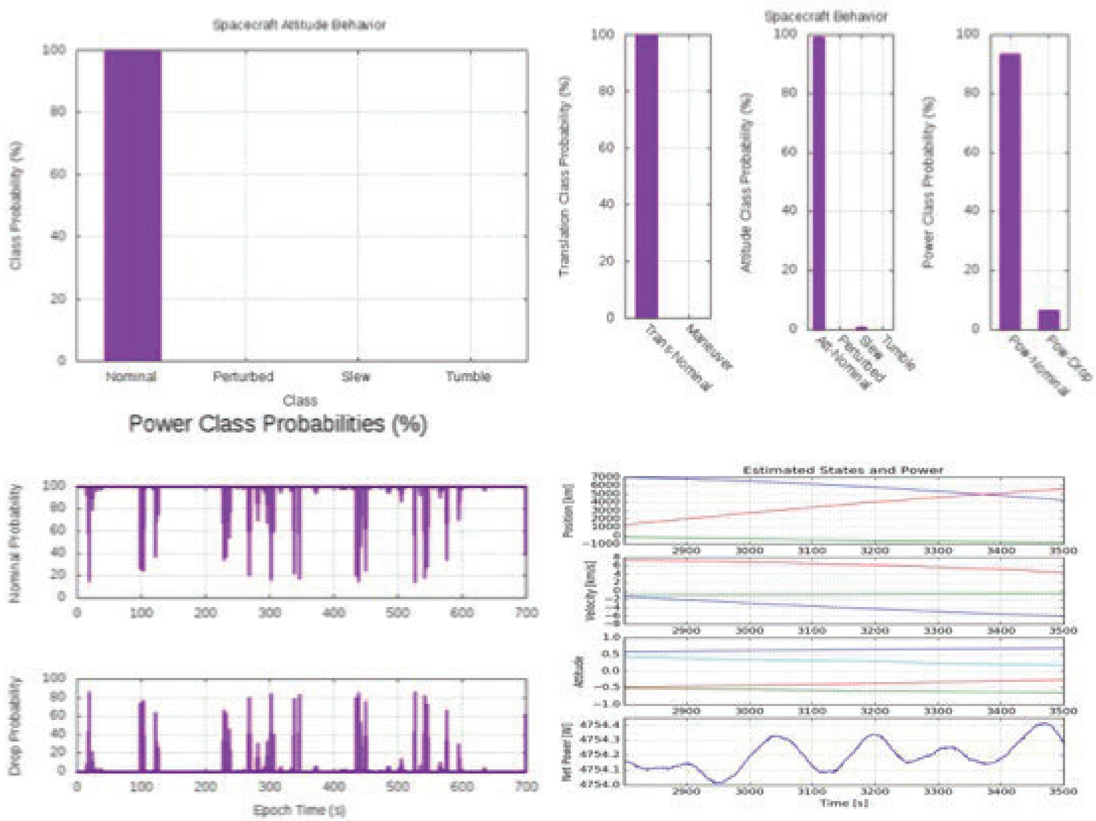


Figure 18. 2800-3500 sec Prediction Results

4.1.9 Conclusions.

Through the use of modern cyber infrastructures such as VerSSA and physics-based deep learning models, we were able to successfully show that we can indeed offer near real time anomalous spacecraft behavior detection and characterization. There are a few minor artifacts that we discovered while testing, but all are easily remedied with additional training of the neural networks. It is also apparent that absent the power system information (which is only available through telemetry) it would be almost impossible to detect any anomalous event occurred using only the physics-based filtered data. The neural networks, however, are able to recognize anomalies in the spacecraft telemetry much easier and then use this information to create a higher level classification of behavior.

4.2 Project Year 2: FVEYS Phantom Echoes 1

Phantom Echoes1 is a cooperative 5-eyes experiment to quantify benefits derived from a federated SSA capability to enhance decision making. The goal is to exploit both simulation and real-world events to test tools within operationally relevant scenarios (see Figure 19). More specifically: 1) focusing on GEO protection; and 2) Real-World tracking and characterization via the observation via optical assets distributed across the world of the Northrop-Grumman

Mission Extension Vehicle (MEV-1) rendezvous. VerSSA plays a major role by providing a platform for ingesting, processing and sharing data and algorithms across the FVEYs organizations.

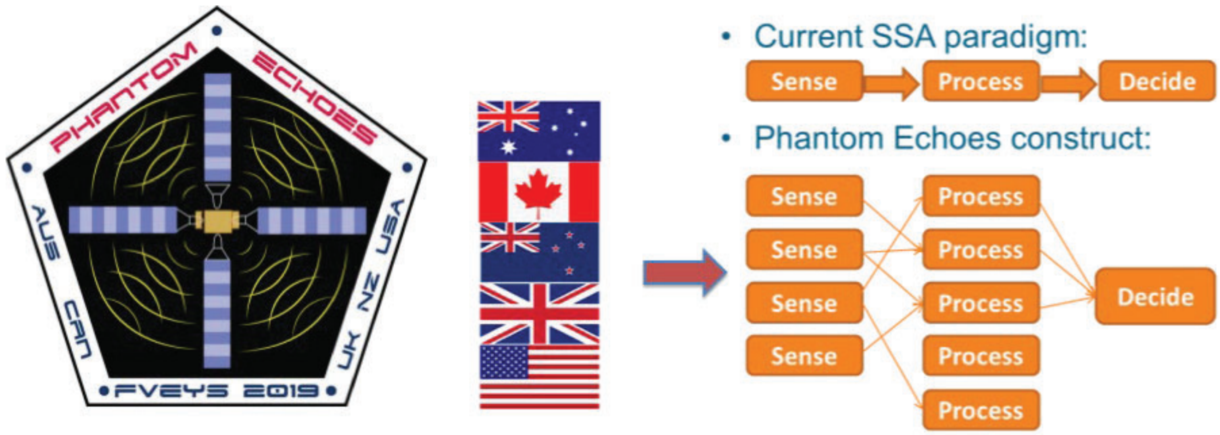


Figure 19. FVEYS Phantom Echoes 1 Paradigm

4.2.1 Mission Extension Vehicle 1 (MEV-1) Experiment Overview.

The goal of this FVEYs experiment is to jointly characterize a commercial satellite servicing mission called Mission Extension Vehicle 1 (MEV-1), On-Orbit-Servicing (OOS) and captivation of the Intelsat-901 geosynchronous satellite. SSA experimenters from the United Kingdom (Dstl), United States (AFRL, University of Arizona), Canada (DRDC), New Zealand (DTA) and Australia (DSTG) all contribute observational and computing resources to characterize a challenging space situational awareness experiment. Each nation collected a variety of observations on both the MEV-1 during its orbit raising, proximity flight and docking using ground and space-based telescopes. The experimenters were able to collect photometric and astrometric measurements on the vehicle pair and detected brightnesses for both targets spanned M_v 4 – 16 for Intelsat 901 and M_v 8-16 for MEV-1. Photometric and spectrometric observations were collected showing some identifying glint and wavelength resolved features in the pre and post docking characterization of the vehicles. National data and processing software were aggregated within a cloud-based storage and processing architecture referred to as VERSSA which enables individual national processing capabilities to be individually actioned within a collectively accessible IT architecture.

4.2.2 VerSSA Workflow and Data Management for Phantom Echoes 1.

A workflow with apps and algorithms provided by the FVEYs nations have been developed, implemented and tested in VerSSA to ensure cooperative and coordinated data management and processing for effective tracking and characterization of MEV-1. Figure 20 shows the workflow containing all the dockerized algorithms in development for manual, semi-automatic and automatic data processing (from raw images to orbit determination) developed for the Phantom Echoes campaigns. The original workflow was designed to ensure that every nation contributed a dedicated astrometric pipeline for their available sensors. Initial and Special Perturbation Orbit Determination algorithms together with analysis and threat assessment processes were included to provide end-to-end solutions to the coordinated observational campaign. Native algorithms (i.e. before dockerization, including, astrometry and photometry pipelines, initial and special perturbation orbit determination) have been provided by the contributing five nations. Importantly, the workflow included the development of dedicated external interfaces/API with Unified Data Library (UDL) and CUBRC Inc. for seamless automatic and semi-automatic data exchange and processing.

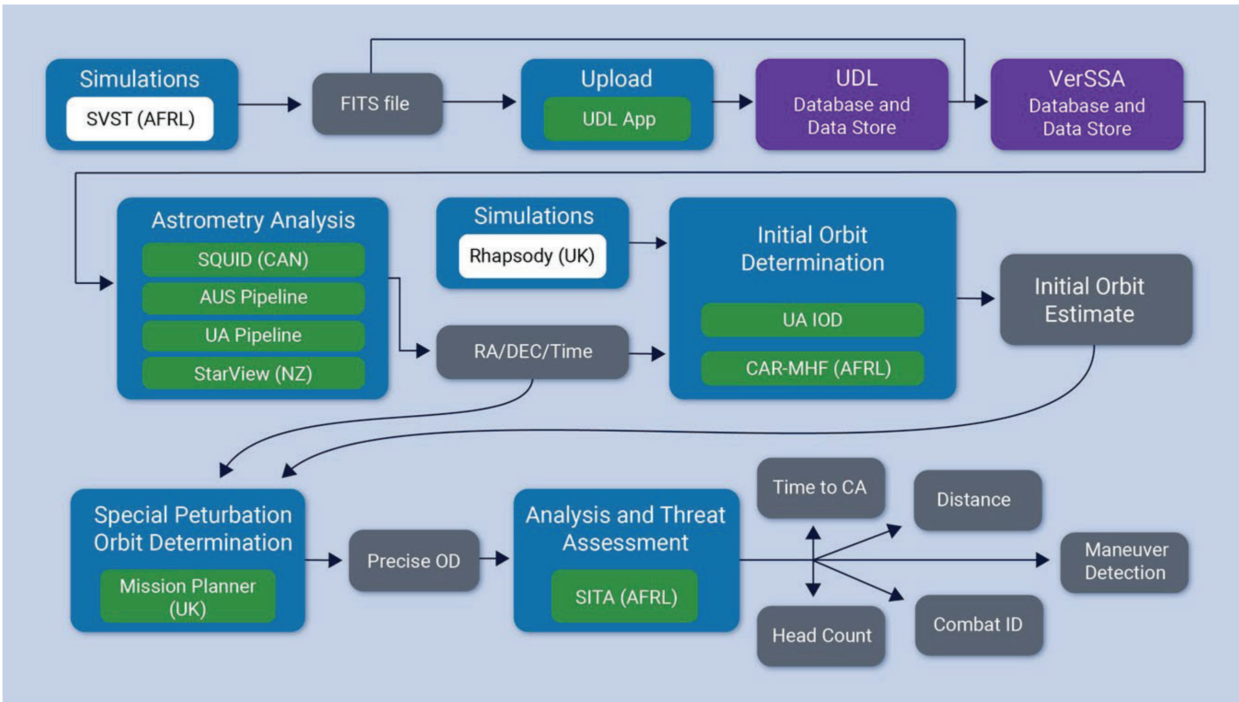


Figure 20. Phantom Echoes Workflow in VerSSA

Within the provided workflow, VerSSA has a set of algorithms implemented in sharable dockerized containers ("Apps") that can run in stand-alone and/or automatic fashion to process data end-to-end. Such algorithm include:

- Astrometry Apps specifically designed to ingest and process optical sensors images coming from multiple FVEYS countries to get the Right Ascension (RA) and Declination (DEC) of the Space Object. The set of pipelines include: 1) UArizone pipeline (USA), SQuid (CAN), StarView (New Zealand), AUS Pipeline (AUS), UK Pipeline (UK)
- Photometry App, developed by UArizona, processes optical sensors images to obtain light-curves
- Orbit/Attitude Determination Apps have been developed to obtain spacecraft trajectories from astrometry data, as well as to obtain attitude profile from on-board measurements. The apps include the following algorithms: 1) CAR-MHF (AFRL), 2) Initial Orbit Determination (UArizona), 3) Mission Planner (UK), 4) Extended Kalman Filter (US), and 5) Unscented Kalman Filters (US)

The UArizona VerSSA team has developed a set of external interfaces to upload and download data, including, FVEYs optical sensors distributed across the world (UArizona, CAN, AUS, NZ), Unified Data Library (UDL - BlueStaq), and Situation Identification and Threat Assessment (SITA – AFRL/CUBRC).

Figure 21 shows the Phantom Echoes 1 directory structure in VerSSA. The phantom_echoes_MEV1 folder contains one folder for each app plus an additional folder for all raw images. Importantly, raw images are organized by telescope. Under each app output folder, there is one folder for telescope.

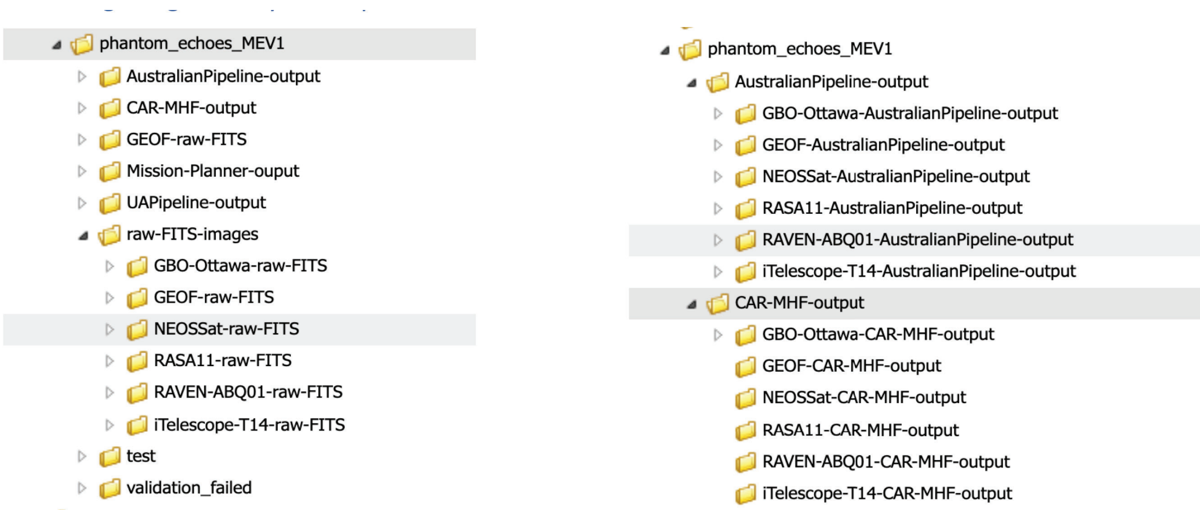


Figure 21. Phantom Echoes 1 directory Structure in VerSSA

Phantom Echoes 1 raw images undergo a Flexible Image Transport System (FITS) format data standardization. Table 3 shows the FITS keyword required to be processed by the workflow.

Table 3. Required FITS Keywords

Keyword	Type	Description	Format
DATE-OBS	String	Date and time of exposure start	YYYY-MM-DDThh:mm:ss.sss
EXPTIME	Float	Exposure time (s)	
OBJCTRA	String	Center RA	hh mm ss
OBJCTDEC	String	Center DEC	dd mm ss
SITELAT	String	Telescope latitude	dd mm ss
SITELONG	String	Telescope longitude	dd mm ss
SITEELEV	String	Telescope elevation (m)	
TELESCOP	String	Telescope name	
COUNTRY	String	Three letter country name	

Importantly, VerSSA can autonomously run workflows end-to-end, i.e. from sensor data to precise orbit determination. This is accomplished by special apps developed to link the individual pipelines in single automatic end-to-end processing system. Automation for the data processing is organized as follows:

- All image folders within Phantom_Echoes_MEV1/RawFitsImages/<Tel 1>/. will have meta data tag associated with them that specifies whether or not they have been processed
- The Auto script will periodically check each Tel folder for new image folders without this meta data tag
- When new images are detected desired workflows are started automatically on the new images
- Workflows correspond to the apps that are run sequentially:
    - Workflow1 (preconstructed) – UA Pipeline, CAR-MHF, and Mission Planner
    - Workflow2 (preconstructed) – UA Pipeline, UA IOD, and Mission Planner
    - SemiAuto workflow (flexible) – can run any sequence of Apps as specified by the user

VerSSA wiki with workflow description is currently available on-line at versa.atlassian.net (Figure 22).

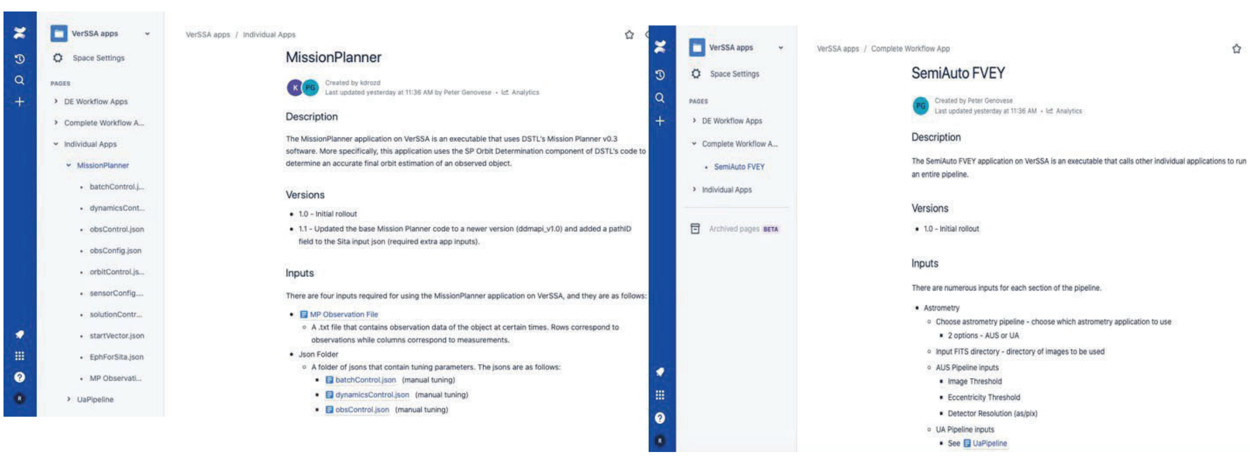


Figure 22. VerSSA Wiki for the Phantom Echoes workflow

4.2.3 UArizona Observational Contribution to Phantom Eachoes 1.

4.2.3.1 UArizona Observational Assets

The following observational assets are made available by University of Arizona for the project, i.e. A) LEO20; 2) RAPTORS (0.6m); 3) NASA IRTF; and 4) Large Binocular Telescope (LBT). Figures UA2 and UA3 report the telescopes characteristics (Figure 23, Figure 24).

Location: Tucson, Arizona, USA
Aperture: 0.5-meter
Focal Ratio: F/2.8
Field of View: 1.4 x 1.4 deg
Pixel Scale (1x bin): 1.22”/pixel
Focus: Multi-color photometry

Location: Tucson, Arizona, USA
Aperture: 0.6-meter
Focal Ratio: F/4
Field of View: 16 x 16 arc min
Pixel Scale (1x bin): 0.93”/pixel
Focus: Visible Spectroscopy

Figure 23. A) LEO20 and B) RAPTORS class telescopes

Location: Mauna Kea, Hawaii, USA
Aperture: 3.2-meter
Focal Ratio: F/38
Field of View: 60 x 60 arc secs
Wavelength Range: 0.65-2.55 µm
Spectral Resolution: R~100
Focus: Near-IR Spectroscopy

Location: Mt Graham, Arizona, USA
Aperture: 8.4-meter
Focal Ratio: F/15
Field of View: 30 x 30 arc sec
Instrument: LUCI-1 and IRTC (zJHK)
Spatial Resolution:
60 mas at K band (~10 meters at GEO)

Figure 24. C) NASA IRFT and D) Large Binocular Telescope (LBT)

4.2.3.2 Phantom Echoes 1 UArizona Observational Campaign

Table 4 through Table 7 report the set of collected observations during the MEV-1 campaign.

Table 4. LEO and RAPTORS Observations campaign (Phase 1)

Sensors/Location	Wavelength Range/Resolution	Type of Observation	UTC Dates of Observation	Target/Reduction Status
Leo20 0.5m F/2.8 Tucson, Arizona	0.41-1.0 µm	Sloan g', r', I', z' four color photometry	Jan-14-2020 Jan-24-2020 Jan-26-2020 Jan-27-2020 Jan-28-2020 Jan-29-2020 Jan-30-2020 Feb-01-2020 Feb-04-2020	Intelsat-901 MEV-1 Intelsat-901 Intelsat-901 MEV-1 MEV-1 Intelsat-901 MEV-1 & Isat-901 MEV-1 & Isat-901
RAPTORS 0.6m F/4 Tucson, Arizona	0.4-1.0 µm Resolution: R~30	Low-resolution visible spectra	Jan-26-2020 Jan-27-2020 Jan-28-2020 Jan-29-2020 Jan-30-2020 Feb-01-2020	Intelsat-901 Intelsat-901 MEV-1 MEV-1 Intelsat-901 Intelsat-901

Reduced Reduction ongoing

Table 5. NASA IRTF and LBT Observations Campaign (Phase 1)

Sensors/Location	Wavelength Range/Resolution	Type of Observation	UTC Dates of Observation	Target/Reduction Status
NASA IRTF 3m Mauna Kea, Hawaii	0.65-2.55 µm Resolution: R~100	Low-resolution near-infrared spectra	Jan-24-2020 Feb-01-2020	Failed to find Intelsat-901
LBT 2x8m Mt. Graham, Arizona	0.85-2.55 µm	Z', JHK resolved imaging	Jan-27-2020	Intelsat-901

Reduced Reduction ongoing

Table 6. LEO20 Observations Campaign (Phase 2)

Sensors/Location	Wavelength Range/Resolution	Type of Observation	UTC Dates of Observation	Target/Reduction Status
Leo20 0.5m F/2.8 Tucson, Arizona	0.41-1.0 µm	Sloan g', r', I', z' four color photometry	Jan-14-2020 Jan-24-2020 Jan-26-2020 Jan-27-2020 Jan-28-2020 Jan-29-2020 Jan-30-2020 Feb-01-2020 Feb-05-2020 Feb-06-2020 Feb-07-2020 Feb-08-2020 Feb-09-2020	Intelsat-901 MEV-1 Intelsat-901 Intelsat-901 MEV-1 MEV-1 Intelsat-901 MEV-1 & Isat-901 MEV-1 & Isat-901 MEV-1 & Isat-901 MEV-1 & Isat-901 MEV-1 & Isat-901 MEV-1 & Isat-901

Reduced Reduction ongoing

Table 7. RAPTORS and LBT Observations Campaign (Phase 2)

Sensors/Location	Wavelength Range/Resolution	Type of Observation	UTC Dates of Observation	Target/Reduction Status
RAPTORS 0.6m F/4 Tucson, Arizona	0.4-1.0 µm Resolution: R~30	Low-resolution visible spectra	Jan-26-2020 Jan-27-2020 Jan-28-2020 Jan-29-2020 Jan-30-2020 Feb-01-2020	Intelsat-901 Intelsat-901 MEV-1 MEV-1 Intelsat-901 Intelsat-901
LBT 2x8m Mt. Graham, Arizona	0.85-2.55 µm	Z', JHK resolved imaging	Jan-27-2020 Feb-08-2020	Intelsat-901 Anik F1R; F1

Reduced Reduction ongoing

4.2.3.3 Pre-Docking Characterization of Intelsat 901 (Winter 2019)

During the pre-docking phase, we employed LEO20, RAPTORS and LBT to support Intelsat 901 characterization. More specifically:

- LEO20 provides a multi-color characterization in the four sloan filters g',r',l',z' (four-color photometry
- RAPTORS provides low-resolution visible spectra
- LBT provides resolved imaging of Intelsat 901

LEO20 four-color photometry: Figure 25, Figure 26 and Figure 27 show the collected four-color apparent magnitude as function of the phase angle. From the graphs, an interesting glint in Intelsat-901 data in g' filter (401-550 nm). One key question to answer would be the following: What is the cause of this feature? Additionally, the glint is slightly apparent in the r' filter.

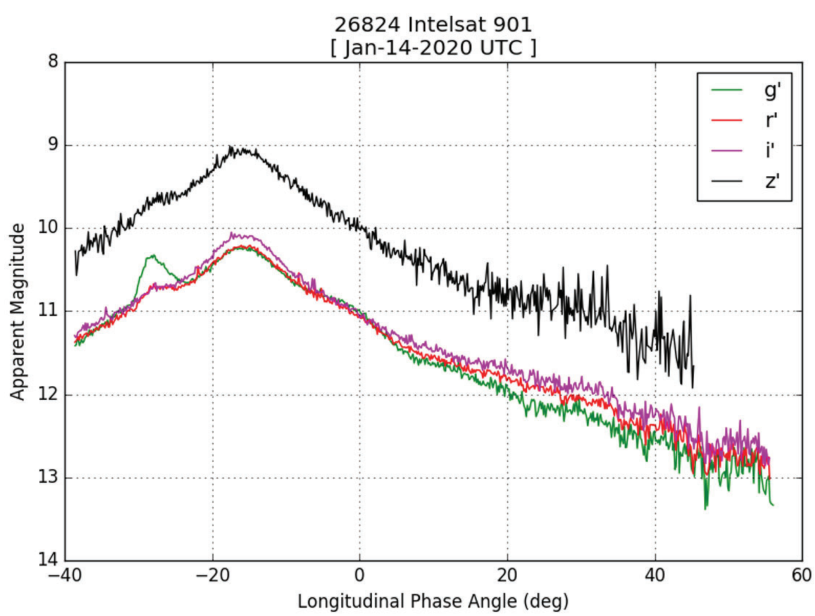


Figure 25. Four-color, phase-angle behavior of Intelsat 901 on January 14, 2020

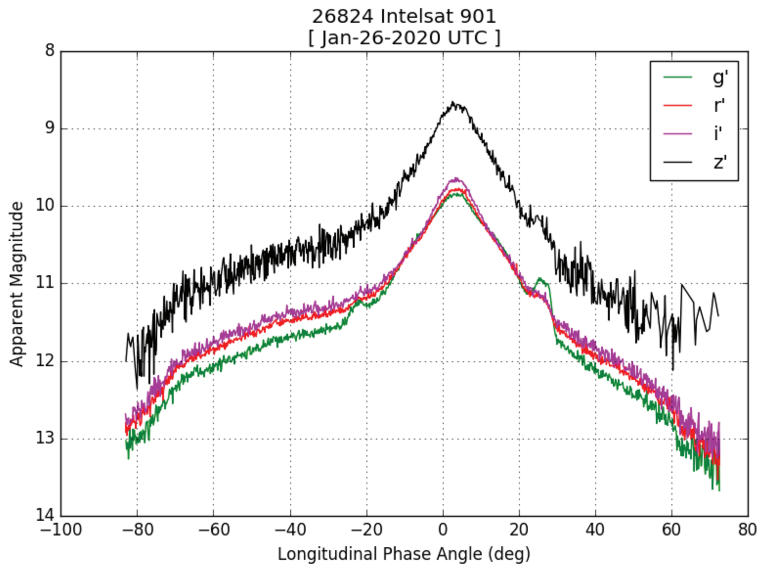


Figure 26. Four-color, phase-angle behavior of Intelsat 901 on January 26, 2020

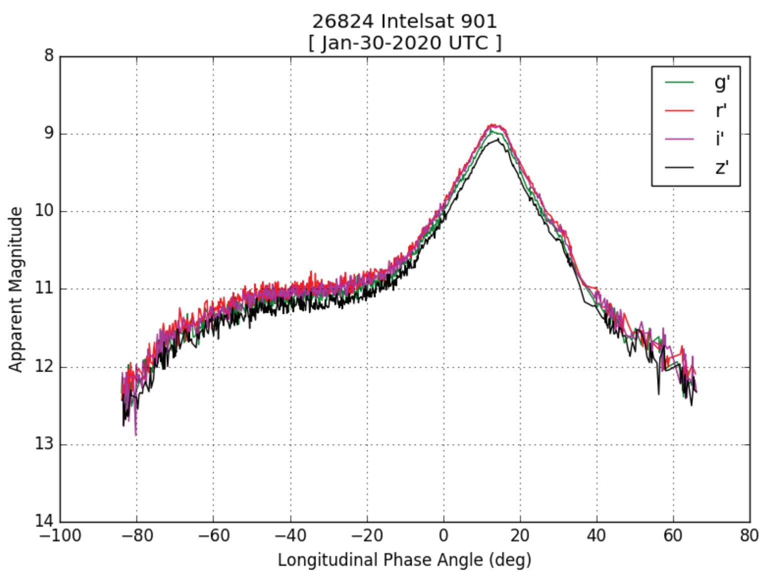


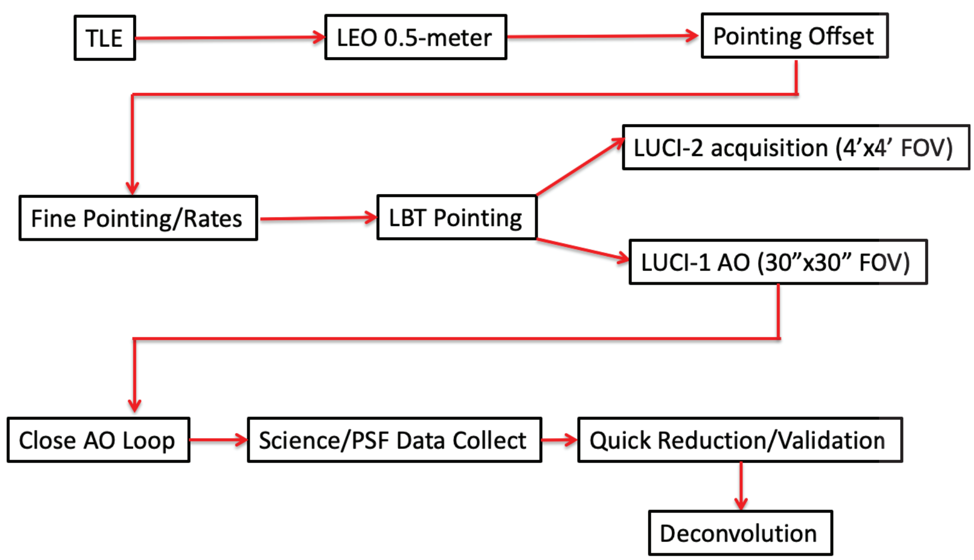
Figure 27. Four-color, phase-angle behavior of Intelsat 901 on January 30, 2020

<u>LBT Resolving Imaging Campaign:</u> GEO satellites are ideal targets for measuring adaptive optics performance of large ground-based telescopes such as the LBT. On the truth data, we have better constraints on the shape and dimensions of GEO satellites. Additionally, consistent observing conditions (airmass/elevation) and always available (all night/year). Importantly, cross calibration between AO observatories is possible. Table 8 shows the characteristics of the two instruments comprising the LBT system.

Table 8. LBT Instruments Set

Instrument	Wavelength Range/pixel scale/FOV	Type of Observation	Spatial Resolution	Targets
LUCI (LBT Utility Camera in the Infrared)	0.8-2.5 µm 0.015”/pixel 30”x30”	Diffraction Limited Sloan z’ and JHK imaging	~60 mas in K ~45 mas in H ~30 mas in J ~25 mas in z’	Intelsat 901
IRTC (Infrared Test Camera)	0.8-2.5 µm 0.010”/pixel 30”x30”	Diffraction Limited JH imaging	45 mas/pixel in H	Anik F1R Anik F1

Figure 28 shows the LBT observational process.



*Simultaneous GRIZ photometry and visible spectroscopy data were collected

Figure 28. LBT Observations

Figure 29(A) shows the set of collected images in the K-/H-/J-/Z-Bands before applying deconvolution Figure 29(B). The employed LBT deconvolution technique was the Multiframe Blind Deconvolution (MFBD). Observing conditions prevented observing PSF calibration star. Importantly, a Gaussian PSF was used for the initial estimate. Artifacts observed to both sides of the satellite are the first-order diffraction (Airy) rings.

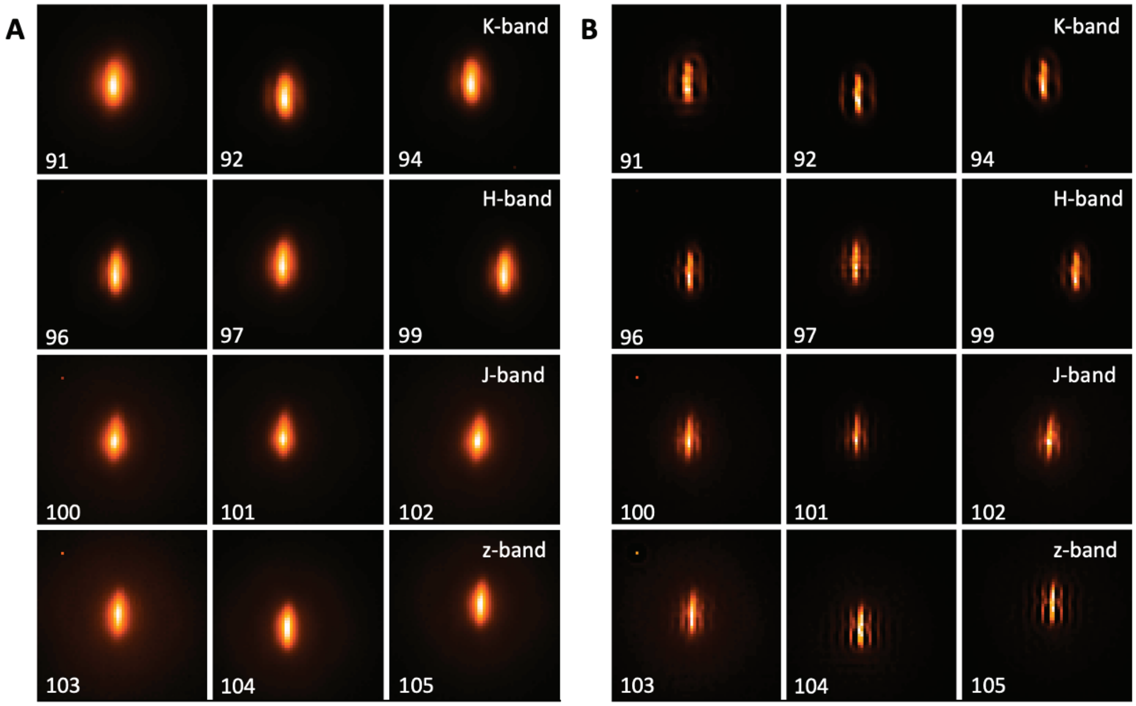


Figure 29. Intelsat 901. A) Images before deconvolution; B) Images after convolution

As evident from Figure 29, we successfully resolved Intelsat-901 using diffraction limited imager at LBT. The advantage of having two 8.4 meter mirrors is that one can be used as finder and the other for science. We achieved a spatial resolution ~30 mas (~5 meters at GEO) for Intelsat-901. Key components (Solar panels, Bus, Antenna?) can be directly identified.

4.2.3.4 Orbit Raising Period (October 2019 – January 2020)

During the orbit raising period, we employed LEO20 and RAPTORS to support MEV-1 characterization. More specifically:

1. LEO20 provides a multi-color characterization in the four sloan filters g’,r’,l’,z’ (four-color) photometry
2. RAPTORS provides low-resolution visible spectra

<u>**LEO20 four-color photometry:**</u> Figure 30, Figure 31, Figure 32 and Figure 33 show the collected four-color apparent magnitude as function of the phase angle for three different observational periods.

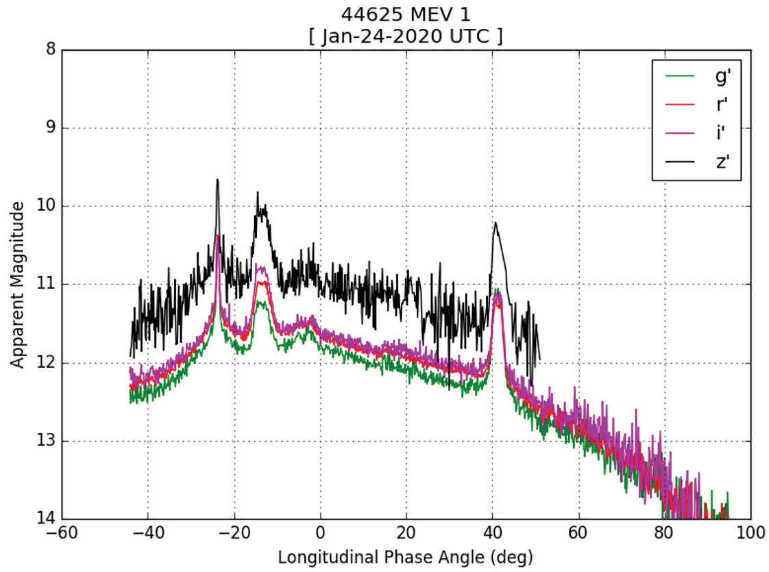


Figure 30. Four-color, phase-angle behavior of MEV-1 on January 24, 2020

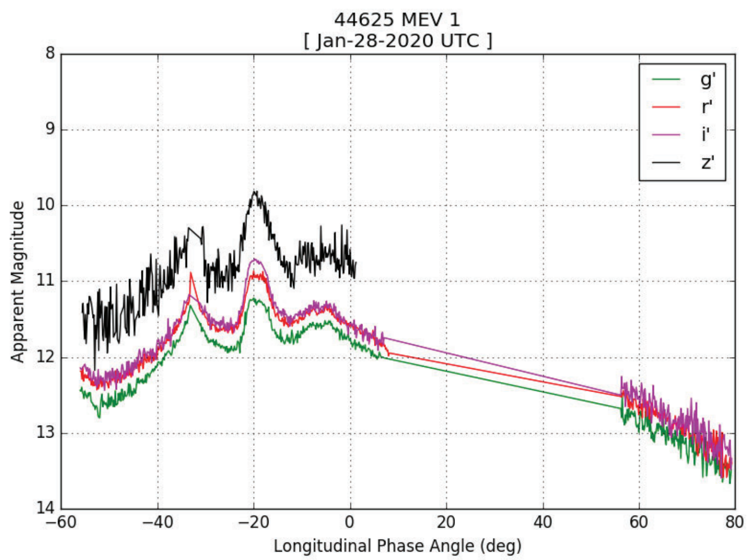


Figure 31. Four-color, phase-angle behavior of MEV-1 on January 28, 2020

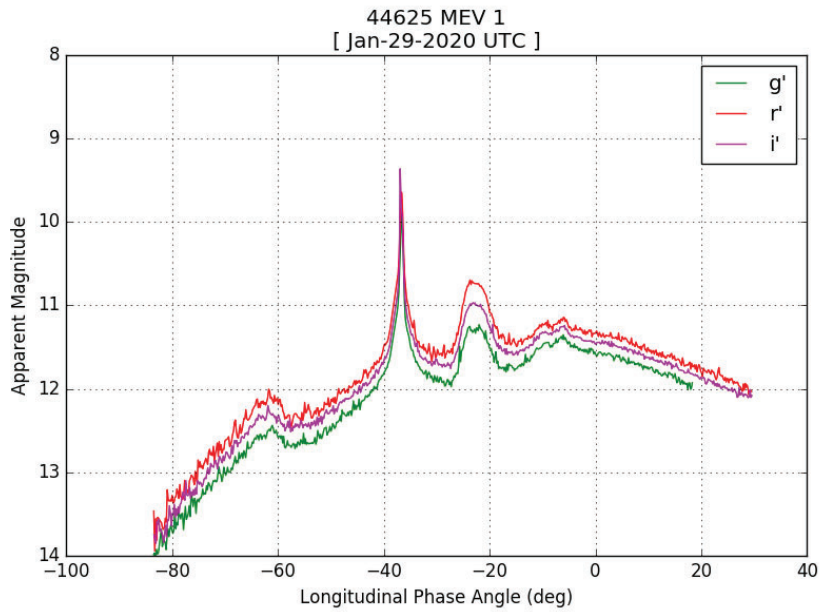


Figure 32. Four-color, phase-angle behavior of MEV-1 on January 29, 2020

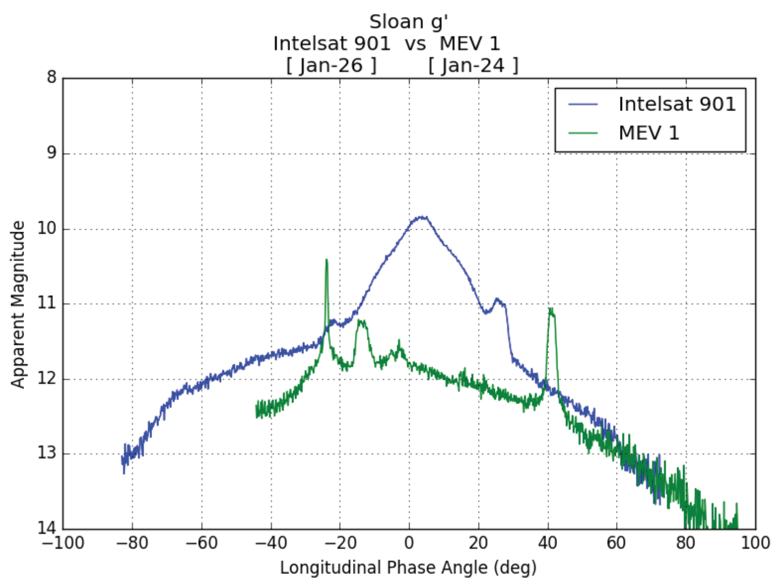


Figure 33. Four-color, phase-angle comparison, MEV-1 vs. Intelsat 901

Figure 34 shows a side-by-side comparison of MEV-1 versus Intelsat 901 in the sloan g' filter as function of the phase angle. From a photometric point of view, the following conclusions can be drawn. First, both objects can be clearly distinguished from their phase curves as they have different morphology and glints. Additionally, changing geometry leads to changes in phase curves. Hence phase curves cannot be used to distinguish two objects that are maneuvering.

<u>RAPTORS low-resolution spectra:</u> Figure 34 shows the collected spectral reflectance for both MEV-1 (collected Jan 18, 2020) and Intelsat 901 (collected Feb 1, 2020). We have observed

Intelsat-901 and MEV-1 on multiple nights to characterize the spectral phase effects. Initial analysis suggests visible wavelength spectroscopy would be a helpful tool to uniquely identify RSOs if systematics can be worked out.

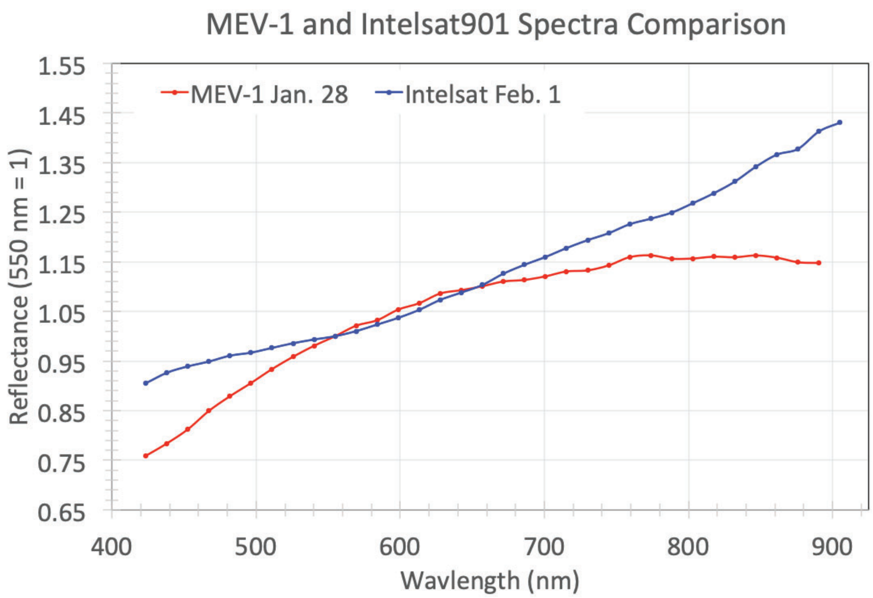


Figure 34. Spectral reflectance for MEV-1 and Intelsat 901

4.2.3.5 Proximity, Docking and Captivation

An observational characterization using the LBT system for the Anik F1cluster (proximity operations). Figure 35 shows the image data before (A) and after (B) deconvolution. The image data are obtained from the IRTC camera in H-band. There were 9 data sets each comprising of 50 frames. The image scale show in Figure 35 is 0.010"per pixel. The measured H-band Point Source Function (PSF) was saturated but a separate H-band. PSF was obtained from an earlier observation during the night and this was used for the initial PSF estimate. Deconvolution was applied to the 9 separate data sets yielding the 9 images shown here. The 9 images are very self-consistent: there are two distinct point-like reflections on the left of the image and the one on the right appears to be more elongated. Importantly, the apparent angular size of the satellite is ~ 0.20" x 0.015" (~33mx3m).

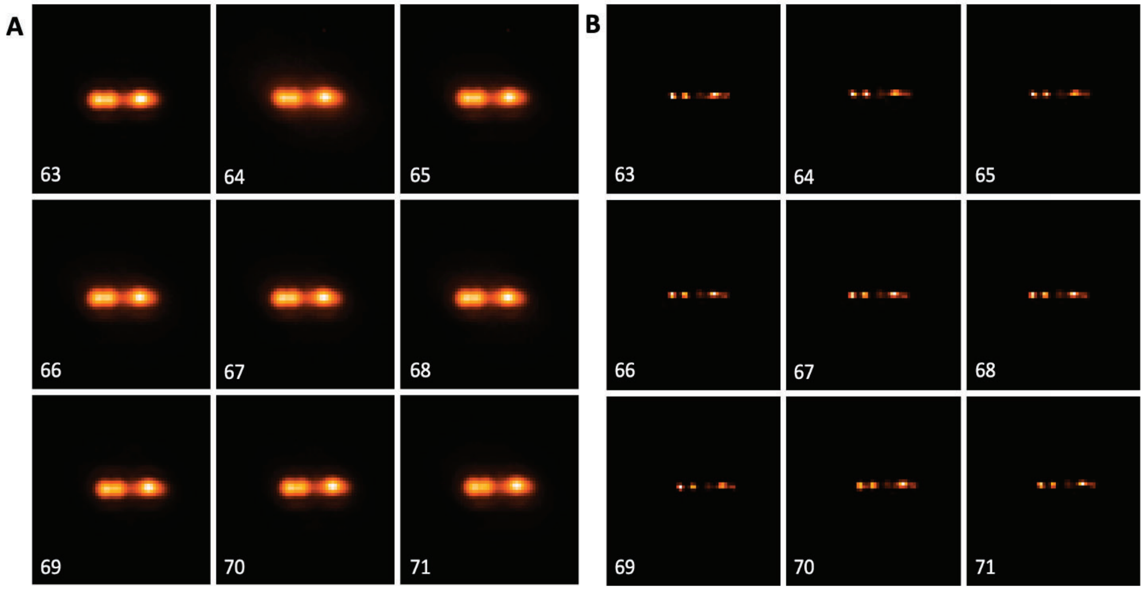


Figure 35. Anik F1, Band H, February 8, 2020. A) Images before deconvolution; B) Images after convolution

4.3 Project Year 3: FVEYS Phantom Echoes 2

A second FVEY campaign to track and characterize the Northrop Grumman Mission Extension Vehicle 2 (MEV-2) was conducted in year 2020. The Phantom Echoes 2 campaign was a natural project continuation where the workflow put together in VerSSA would be exercised to show live collaboration during real-time tracking and custody of MEV-1 across the FVEY network.

4.3.1 EPOR Tracking Campaign: Goal and Objectives.

The MEV-2 EPOR campaign was conceived to answer the following question: "**Can the Phantom Echoes FVEY community observe and maintain custody of *MEV-2* during EPOR, without using Space-Track.org?**". The goal is to test the cycle Observe-Orient-Decide-Act (OODA) loop (Figure 36) and verify the ability to maintain custody of MEV-2 during the raising phase orbit. The overall idea is to use Two-Line Elements (TLE) to generate indigenous orbit cues to re-acquire, update orbit state and maintain custody during ~2-week period. During this phase, all data processing conducted within VerSSA. The latter include 1) image reduction, 2) initial orbit determination, 3) special perturbation orbit determination to yield the state vector output to TLE. If all sensors cannot acquire using supplied cue, revert to NG predictions or SpaceTrack.org TLE. The following objectives have been identified:

- Successfully execute a distributed collection experiment on MEV-2 comprising FVEYs sensors and VerSSA
- Trial FVEYs S&T capability to observe challenging targets and process data to cue other sensors
- Assess gaps / limitations in custody maintenance of EP-enabled vehicles

- Collect unclassified, shareable data on *MEV-2* for submission into the international science community
- Evaluate utility of VerSSA-like architectures for future FVEY national applications
- Inform development of Concept of Operations (CONOPs) to tackle constantly-manoeuvring targets

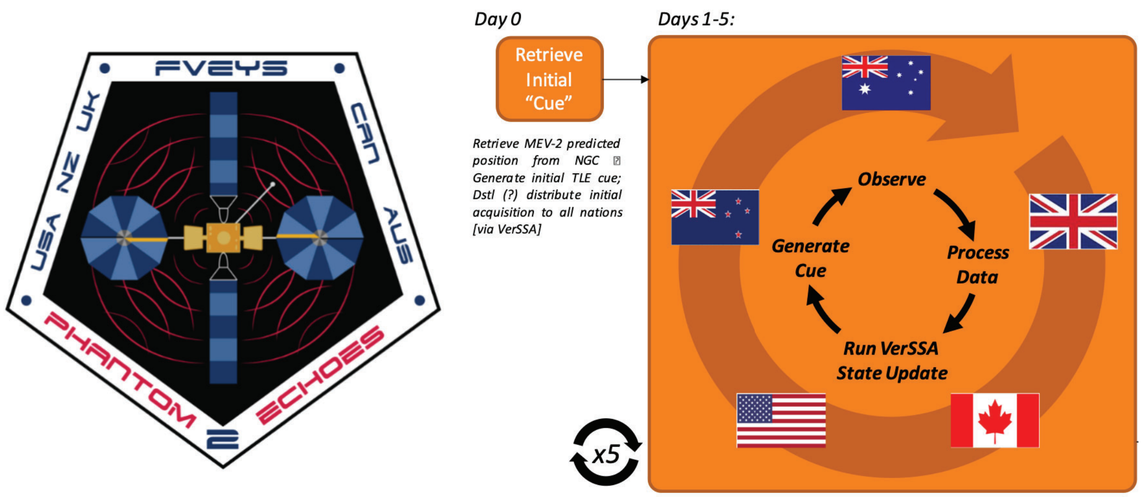


Figure 36. FVEY Phantom Echoes 2 Campaign

Two notional campaigns have been devised:

- De-risk trial: ***29th - 31st July***
  - Prepare CONOPs for full campaign
- Campaign window: ***28th September to 11th October**** tbc
  - Week 1: **28th to 2nd October** (+ Hotwash review)
  - Week 2: *5th October to 9th October,* with the following goals
    - Observe MEV-2 in anger; exercise OODA loop
    - Generate sovereign UK/allied TLE cues: use to generate indigenous orbit cues to re-acquire, update and maintain custody during 1-week "sprints"
    - Apply a "Follow-the-Sun" approach to observation, processing & cueing across nations

4.3.2 Phantom Echoes 2 CONOPS.

A notional CONOPS for the MEV-2 observational campaign has been devised. The nominal per-sensor process is illustrated in Figure 37.

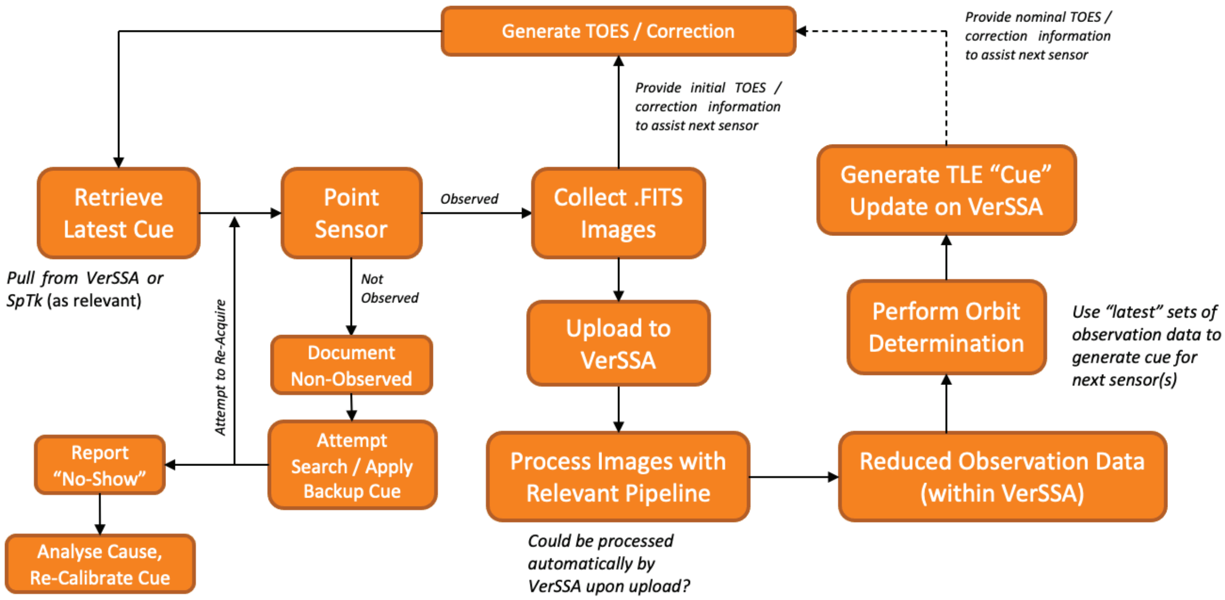


Figure 37. Nominal Per-Sensor Process for the Phantom Echoes 2 Campaign

The ideal solution is iterative in nature and includes the following steps:

- Collect observations of sensor
- Near real time upload to VerSSA/UDL
- Data reduction using VerSSA workflows
- Orbit update using CARMHF/Mission planner with ability to estimate ΔV
- Use estimated ΔV to refine propagation
- Behavior analysis to predict burn pattern/times
- Schedule sensor coverage
- Generate accurate cues for next sensor

4.3.3 VerSSA set up to support Phantom Echoes 2 MEV-2 tracking campaign.

The overall VerSSA workflow (see Figure 20) has been updated to supporting the MEV-2 real-time tracking with semi-automatic and automatic workflow. Additionally, VerSSA has been connected to a Slack channel to ensure real-time communication and update to the FVEY Phantom Echoes community.

4.3.3.1 VerSSA Directory Set-up & Data Format for automated processing

The VerSSA discovery environment has been connected to a new directory structure support real-time upload and data processing & analysis. Uploading can include both raw optical data (images) and astrometric solutions. This option has been provided to avoid lags in uploading and transferring a large number of raw images. The uploading of astrometry includes the following steps:

- Within a telescope directory, one creates a new directory for each astrometry solution uploaded
- The directory needs to be named exactly: Pipeline_Name-#
- 'Pipeline_Name' is one of: "AUS_Pipeline", "UA_Pipeline", "SQUID", "StarView", "LEO_Pipeline"
- '#' is any unique identifier one desires to place. VerSSA team suggested YYYMMDD of observations
- Importantly, do not use spaces anywhere in directory or file names
- A possible example: UA_Pipeline-20201104

An example of directory structure in the VerSSA discovery environment (illustrated for telescope LEO-20) is shown in Figure 38.

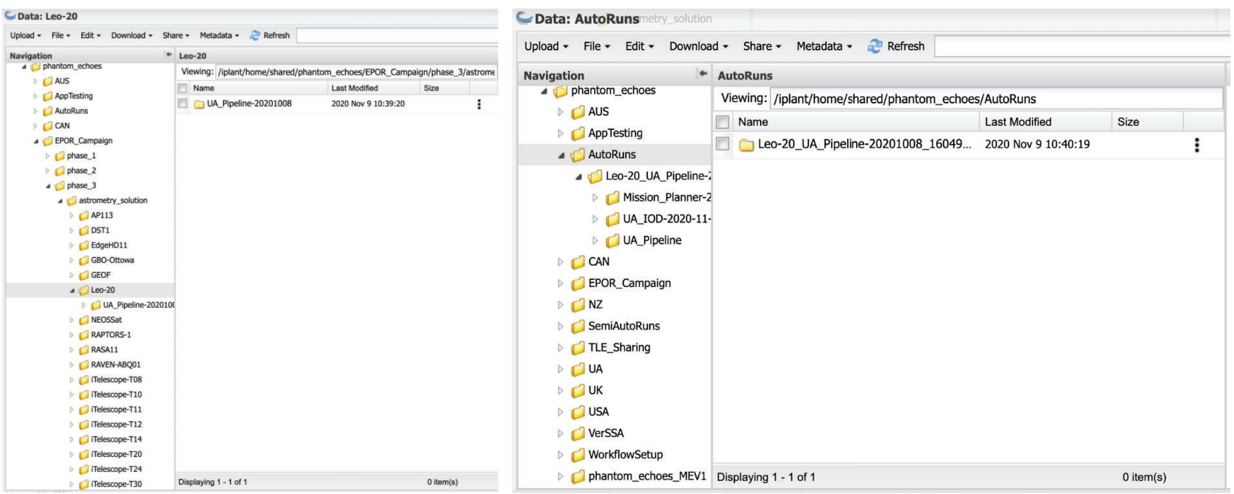


Figure 38. VerSSA directory structure for Phantom Echoes 2

The automated process currently supports the natively exported formats from each of the astrometry pipelines listed above. Importantly, the format of the astrometry should match the directory name, regardless of telescope used. A couple of examples are provided:

- E.x: UA_Pipeline-20201104/somefilename.uao
- E.x. AUS_Pipeline-20201030/filename.obs

Uploading other formats is possible, but the VerSSA team recommended the .uao format for ground-based sensors. The later is a 8 columns, whitespace delimited: SATID JD Exp Xloc Yloc RA DEC Mag.

4.3.3.2 VerSSA Automated Workflow

The automated and semi-automated process has been developed to enable end-to-end processing of raw images and/or astrometric data. The development included the development of dedicated parsers that enabled the correct interfaces between the various apps and new scripts to generate workflow directory structure for users employing the automated process. Two pre-made auto-app sequences have been made available to the Phantom Echoes community to generate precise orbit determination: 1) Workflow1, which processes raw images and executes the sequence UA pipeline-CARMHF-Mission Planner and 2) Workflow 2, which processes raw images and executes UA pipeline-IOD-Mission Planner. A specialized app has been created to support the generation of customized automated pipelines. The automated process has been tuned to check for new data in phase_3 directory every 5 mins. Initial OD is executed using the UA IOD app, and the OD with MissionPlanner App in SGP4 mode (Figure 39). Results, comprising notification of completion and calculated TLE if available, is posted to the data processing Slack channel. Importantly, New directories will be made automatically for each "run" of the automatic app. The name for the output state vector is: Telescope_Pipeline-#_timestring ('timestring') is integer POSIX time of run start. The results from the automated processing results in the most rapid results, but not necessarily the most accurate. The solutions are posted in Slack Channel as they are, and they are not queued.

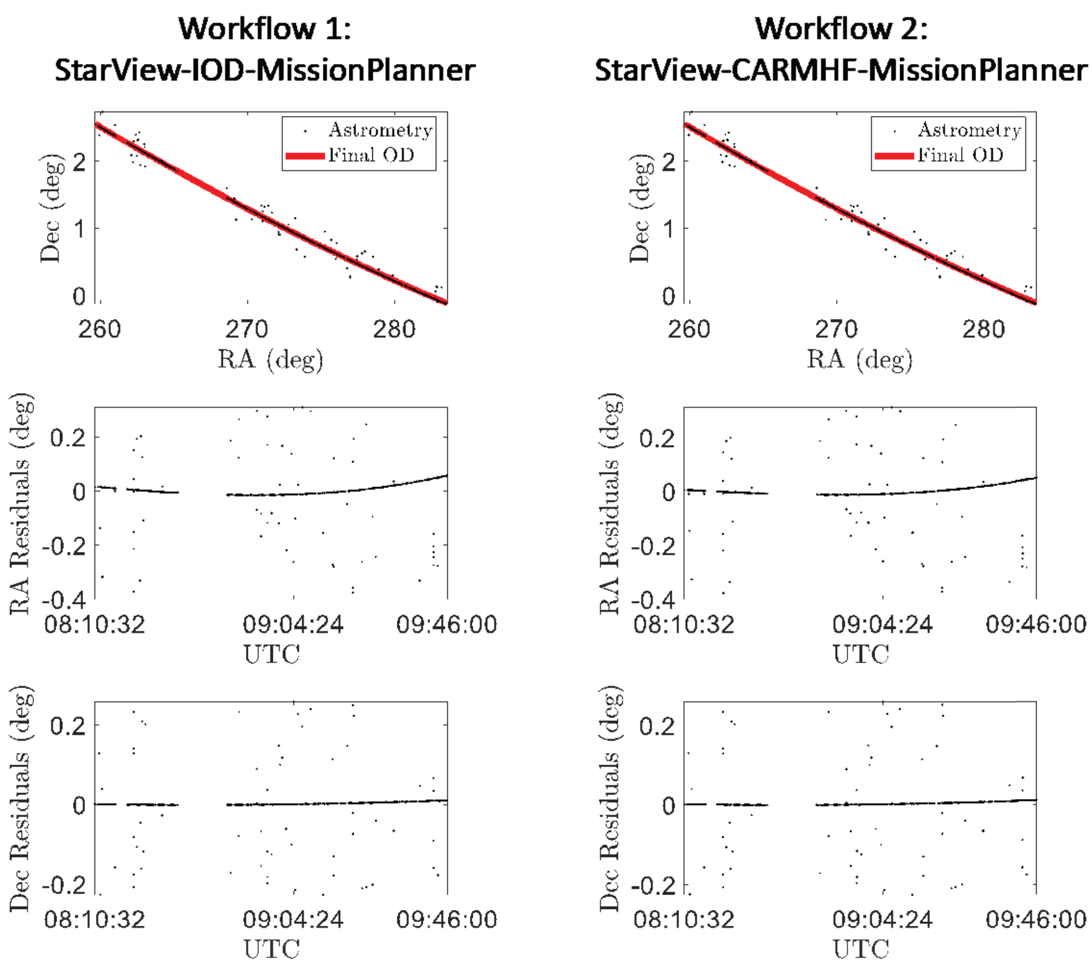


Figure 39. Automated Workflow for TRDS-10 Orbit Determination

As a demonstration, we report an example where we processed images coming from the New Zealand telescope tracking the TRDS-10 satellite and automatically processed the associated optical images within the VerSSA through StarView and UaPipeline workflows:

- /iplant/home/shared/phantom_echoes/NZ/calibration_images/TDRS_10

All outputs and results are located here on VerSSA:

- /iplant/home/shared/phantom_echoes/UA/TDRS_Analysis

The plots in Figure 39 are reported directly from the residuals.txt output of Mission Planner and show how well Mission Planner converges on an orbit in the end-to-end automated workflows.

4.3.4 VerSSA processing and Phantom Echoes Slack Channel.

VerSSA has been connected to dedicated FVEYs slack channels with automated posting of updates during the MEV-2 EPOR campaign (Figure 40). More specifically:

- Verssa-fvey-discuss: real-time discussion between Phantom Echoes-2 international teams
- Verssa-fvey-dataprocessing: fvey bot alerts team members in real-time when new data for MEV-2 in VerSSA are available for further data processing and/or data analysis

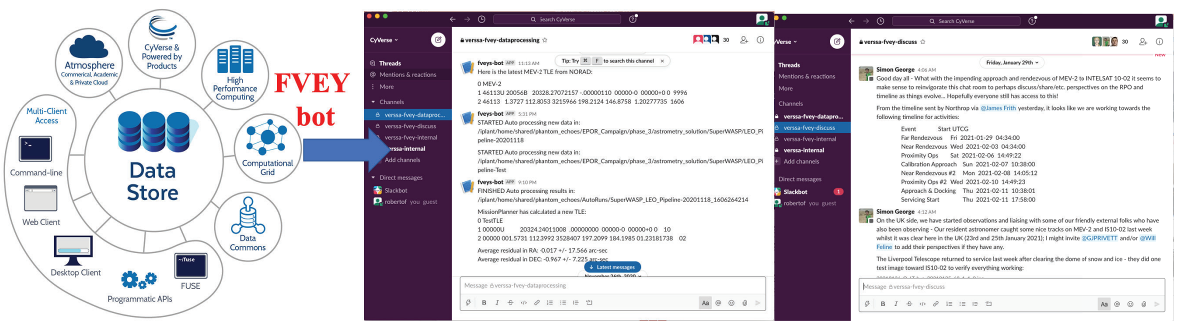


Figure 40. FVEY Slack Channel in VerSSA

4.4 Preliminary Analysis and Approach to XDA Problem

As humans plan to return to the Moon's surface as early as 2024, many nations are already deploying an increasing number of space objects in the Cis-lunar space. Defined as the volume comprised within the Moon's orbit, general awareness of cis-lunar space is critical to support unconstrained access and operations including both surface and orbital domains. However, legacy Space Domain Awareness (SDA) systems were not designed to detect, track and catalog space objects transiting in such environments. This has given rise to a new discipline named X-Geo Space Domain Awareness (XDA) that covers the cis-lunar space. XDA's fundamental goal is to build and maintain a catalog of space objects transiting and residing in the cis-lunar space using a combination of ground-based and dedicated space-based platforms. XDA is extremely important for many reasons. Currently, most activities in cis-lunar space are going unmonitored and only self-reported. Continuous detection, tracking, and monitoring of such space objects is highly desired to 1) avoid strategic surprises; 2) maintain strategic and tactical high ground; 3) support US allies and partners; and 4) protect humans in space by ensuring safe access to the lunar surface. Effective XDA is nevertheless challenging. Indeed, comprehensive coverage from ground-based optical telescopes is limited by the moon's brightness. This difficulty of imaging space objects within ~15 *deg* from the Moon's center defines the so-called "Cone of Shame." Conversely, active radar systems may require a prohibitive amount of power to illuminate space objects at lunar distances. A combination of dedicated space-based and ground-based sensors and dedicated set of innovative algorithms for data processing capable of being deployed in modern cyberinfrastructures will be required for XDA catalog building and maintenance.

4.4.1 UArizona team initial approach to the XDA problem.

The initial UArizona team approach to the XDA problem is to leverage UArizona's 40 year experience in planetary defense from both Earth (Spacewatch, Catalina Sky Survey) and space-based assets NEOWISE and NEO Surveillance Mission). We propose the development of a VerSSA-powered dual-track which aims at 1) developing and validating a prototype Cis-lunar catalog based on existing planetary defense data and new collected observation (UArizona in-house assets); and 2) Mature and integrate basic research (new tools) into the VerSSA framework for rapid deployment (Figure 41).

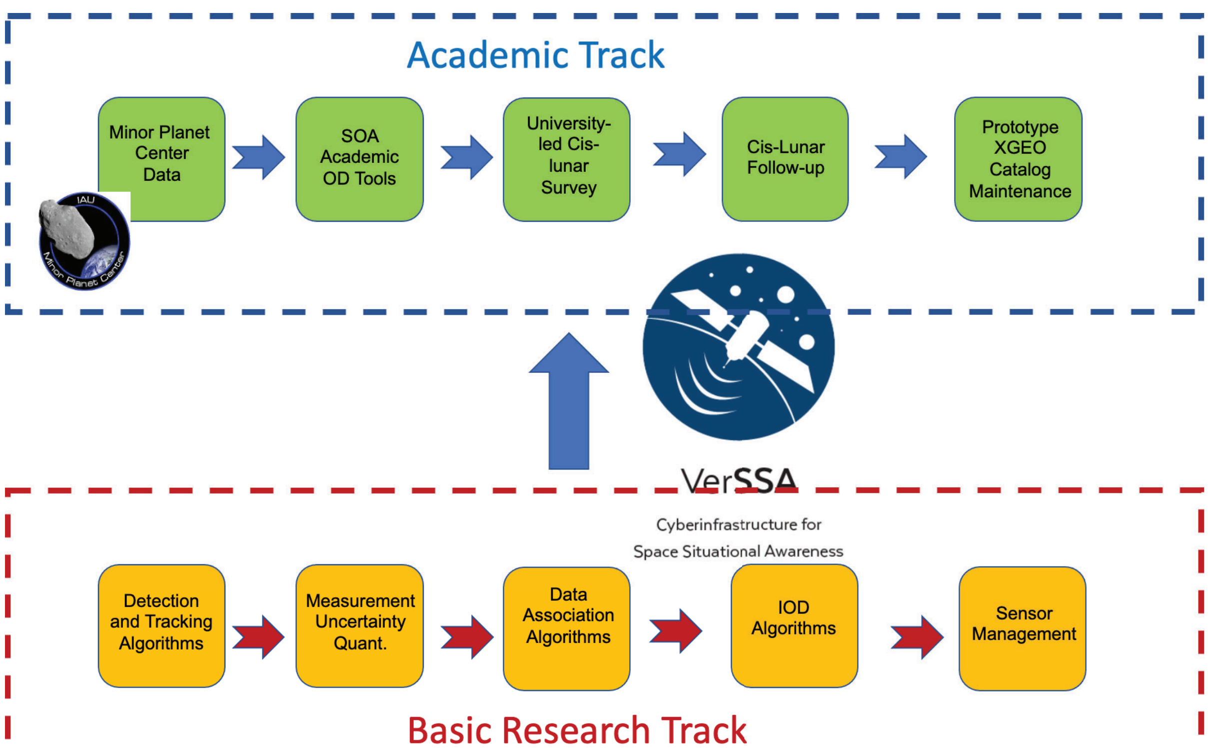


Figure 41. Proposed XDA Approach

4.4.2 XDA Problem: Overall Plan.

The initial goal is to build and maintain a prototype Space Object Catalog in XGEO via the VerSSA cyber infrastructure. The overall approach is to use observations of XGEO objects sent to the Minor Planet Center (MPC) by planetary defense surveys to construct a basic catalog. Importantly, this gives us an opportunity to test and validate existing tools for robust Cis-Lunar catalog maintenance, i.e. execute a full evaluation of planetary exiting planetary defense orbit determination tools. As second step the objective is to identify objects that need follow-up observations to maintain the catalog and deploy UARIZONA assets to collect these observations. A notional workflow of the proposed approach is shown in Figure 42.

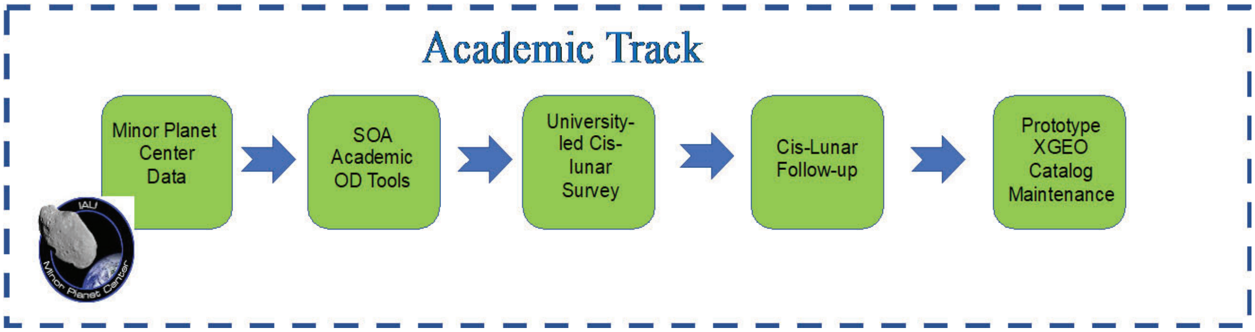


Figure 42. UArizona Academic XDA Notional Development Approach

The development of tools for academic demonstration may include the following:

- Cis-lunar XDA Survey planning tool
    - Projection of cislunar space into the sky
        - Project different orbital regimes (e.g. Halo orbits, Manifolds, HEO/XGEO)
        - Object characteristics (albedo, size)
    - Planning approach with Sensors
        - Given an observation cadence, determine what the sensor sees
        - Given an object in a specific regime, determine the optimal cadence
    - Leverage tools currently developed for NASA NEO Surveyor Mission
- Cis-Lunar Follow-up tool
    - Develop a tool for follow-up and tracking of critical objects that can’t be maintained using just planetary defense data
        - AI system currently developed for NASA NEO Surveyor Mission
- Deployment of all R&D tools (existing/modified/new)
    - All tools developed by academic entities will be dockerized and deployed on VerSSA for demo in operational environment
        - Leverage FVEYs Phantom Echoes experience for the collaborative environment

5.0 CONCLUSIONS

The Cooperative Agreement between AFRL Space Vehicle Directorate and UArizona has shown that synergy between the two research organization is critical to develop new methods, approaches, algorithms and tools for SDA and XDA. Over the past three-year, the UArizona team made a focused effort to develop and deploy the VerSSA cyber infrastructure as a platform for data management and algorithm development and sharing in the SDA community. VerSSA has been successfully deployed and tested on real-world campaigns such as FVEY Phantom Echoes 1&2 and demonstrated to be an ideal system for large and scalable collaborative SDA efforts. VerSSA has also been proposed and initially tested as a platform for the first XGEO academic catalog.

6.0 REFERENCES

[1] European Space Agency, "Copernicus Sentinel-1A Satellite Hit by Space Particle," 31 August 2016. [Online]. Available: http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-1/Copernicus_Sentinel-1A_satellite_hit_by_space_particle

[2] Krag, H. et al, "A 1 cm Space Debris Impact Onto the Sentinel-1A Solar Array", *Acta Astronautica*, Vol 137, August 2017, pp. 434-443.

LIST OF ACRONYMS & GLOSSARY

AD – Attitude Determination

AFRL - Air Force Research Laboratory

AI – Artificial Intelligence

API – Application Protocol Interface

AWS – Amazon Web Service

CA - Cooperative Agreement

CAR-MHF – Constrained Admissible Regions, Multiple Hypothesis Filter

C&DH – Command and Data Handling

CI – Computational Infrastructure

CNN – Convolutional Neural Network

CONOPS – Concept of Operations

CyVerse – Cyber Universe

DE – Discovery Environment

DEC - Declination

ESA – European Space Agency

FITS – Flexible Image Transport Systems

FVEY – Five-Eyes defense organizations

GAN – Generative Adversarial Network

GEO – Geostationary Earth Orbit

GNC – Guidance Control & Navigation

GPU – Graphic Processing Unit

GRP – Generalized Rodriguez Parameters

HEO – High Earth Orbit

IMU – Inertial Measurement Unit

IOD – Initial Orbit Determination

IoT – Internet of Things

iRODS – integrated Rule Oriented Data Systems

ITAR – International Traffic Army Regulations

LBT – Large Binocular Telescope

LEO – Low Earth Orbit

LSTM – Long-Short Term Memory

LIST OF ACRONYMS & GLOSSARY (continued)

LVLH – Local Vertical Local Horizontal

MEV – Mission Extension Vehicle

MFA – Multi Factor Authentication

MFBD – Multi-Frame Blind Deconvolution

ML – Machine Learning

MMAE – Multiple Models Adaptive Estimator

MPC – Minor Planet Center

MUKF – Multiple Unscented Kalman Filter

NASA – National Aeronautics and Space Administration

NEO – Near Earth Object

NG – Nortrop Grumman

NSF – National Science Foundation

OD – Orbit Determination

PDF – Probability Density Function

RA – Right Ascension

R&D – Research & Development

RMS – Root Mean Square

RNN – Recurrent Neural Network

RW – Reaction Wheels

S/C - Spacecraft

SDA – Space Domain Awareness

S&T – Science and Technology

SSA – Space Situational Awareness

STM – State Transition Matrix

TLE – Two-Line Elements

UArizona – University of Arizona

UDL – Unified Data Library

UKF – Unscented Kalman Filter

UT - Unscented Transform

UTA – University of Texas at Austin

VerSSA - CyVerse + SSA

VerSSA-R – VerSSA Restricted

LIST OF ACRONYMS & GLOSSARY (continued)

VPN – Virtual Private Network

XDA – Cislunar Space Domain Awareness

XGEO – X-time Geostationary Earth Orbit

DISTRIBUTION LIST

DTIC/OCP 8725 John J. Kingman Rd, Suite 0944 Ft Belvoir, VA 22060-6218	1 cy
AFRL/RVIL Kirtland AFB, NM 87117-5776	1 cy
Official Record Copy AFRL/RVS/James Frith	1 cy

(This Page Intentionally Left Blank)