

Decisive Mission Analysis Capability: A Physics-Based Testing and Evaluation Framework for Space Domain Awareness

Cameron Harris, PhD., Alexander Cabello
EO Solutions Corp

Miguel Rodriguez, Virginia Wright
Air Force Research Laboratory

Jonathan Kadan, PhD., Justin Fletcher
Space Systems Command

ABSTRACT

End-to-end system Testing and Evaluation (T&E) is needed to assess and enhance autonomous space domain awareness (SDA) systems. This article presents a test environment for end-to-end evaluation of autonomous SDA sensor orchestration systems. The test environment wraps a system under test and plays through scenarios that are relevant to current SDA challenges, such as adversarial threat response evaluation, validation of expected day-in-the-life behavior, and other sensor orchestration performance analysis. An initial application of the methodology is demonstrated in guiding feature development for an operational sensor orchestration system called MACHINA. In a direct-ascent anti-satellite threat scenario, the T&E approach validated MACHINA's reactive sensor tasking and identified work areas in mission planning components for its prospective ballistic object tracking capability. These diagnostic findings show the utility of end-to-end T&E in strategically improving and hardening operational systems. Overall, the results demonstrate that the proposed test environment, instrumented with insightful and intentional analytics, provides a framework to reason about and accelerate development of complex SDA systems prior to real-world deployment.

1. INTRODUCTION

Space Domain Awareness (SDA) operations require robust decision-making systems that orchestrate networks of heterogeneous sensors and respond rapidly to on-sky events. Testing such autonomous mission systems in live environments is challenging, given the scarcity of opportunities (e.g., data from actual threat events) and the high stakes of live exercises. Correspondingly, comprehensive and repeatable Test and Evaluation (T&E) has become a desirable capability for SDA operations and is a growing topic in the SDA community [8].

Recent work on relevant test environments has emphasized the importance of high-fidelity, end-to-end simulation frameworks for validating intelligent systems in complex, dynamic settings. Platforms such as AirSim [19] and Gazebo [12] are standard tools for testing multi-agent coordination and perception systems. While these platforms are not domain-specific to space systems, they offer generalizable strategies for integrating and evaluating system components—including task scheduling, sensor processing, and state estimation—in realistic and controllable environments. Generally, there exist many agent-based simulation studies with proposed methods for testing, but many techniques focus on testing functional requirements rather than “society-level behavior” [5].

The test environment overviewed in this work provides a framework for evaluating SDA sensor orchestration systems, where the system is evaluated end-to-end rather than as individual functional components. Motivating this work is the premise that interactions between functional components of such systems, as well as the time-dependence of these interactions, is significant to meaningful performance analysis and assessment of operational-readiness. The test environment is intentionally architected such that developers and analysts may perform scalable and repeatable scenarios of rare but important events and systematically judge performance. The architecture of the test environment is illustrated in Figure 1.

As shown, the sensor orchestration system under test sends tasking to sensors, which return raw sensor data. The data produced by the sensors is generated from the simulation engine, where the engine curates data based upon a scenario

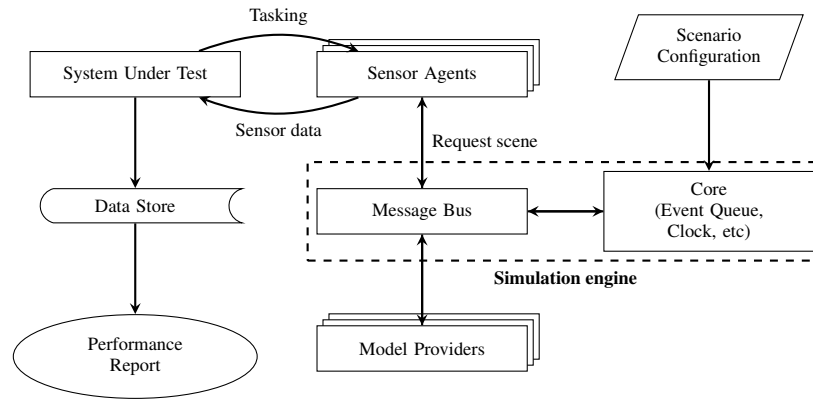


Fig. 1: High-level architecture of the test environment. The System Under Test sends tasking orders to the Sensors throughout the duration of a scenario. Data returned by the Sensors is processed by the System Under Test and used to produce new tasking in a closed loop. Data is extracted from the System Under Test for analysis at the scenario conclusion.

configuration and sensor model. The system under test processes the sensor data and produces new tasking based on the extracted information. The scenario concludes based on exit criteria, which is typically a prescribed end time. After the conclusion of a scenario, data is extracted from the data storage of the system under test for analysis, and a performance report is generated. The test environment must be made coherent with the system data storage to enable analysis – correspondingly, the hooks invoked to aggregate and analyze the data are configurable.

Fundamental to this effort are performance measures for each of the core components of sensor orchestration, allowing data inputs and outputs from functional components to be controlled and analyzed. The high-level functional component diagram expected by the test environment is illustrated in Figure 2.

This paper presents an approach to and underlying design philosophy of end-to-end testing of autonomous sensor network orchestration systems. As a case study, the test framework is used to exercise a sensor orchestration system called MACHINA to guide development for imminent capabilities. MACHINA is an SDA mission software platform that autonomously tasks sensors, collects observations, and updates orbital estimates in real time [7]. As the space environment and operator sensing objectives change, MACHINA dynamically re-plans tasking across a heterogeneous network of sensors to satisfy information needs. By mounting MACHINA into the test framework, we can perform controlled “what-if” scenarios, especially those that are impractical or infeasible to conduct live. Such scenarios include red team/blue team exercises or continuous multi-day operations with dozens of sensors.

As detailed in the 2025 Space Threat Assessment, anti-satellite (ASAT) weapons are an existing and growing threat to space operations [22]. Adversarial nations, such as China, are known to possess ASAT weapons launched from Earth, referred to as direct ascent anti-satellite (DA-ASAT) weapons [2, 22]. China’s tests of DA-ASAT capabilities in low Earth orbit (LEO) are well-documented [13, 11, 17, 10]. China has conducted more recent launches of ballistic objects that are capable of reaching higher orbit regimes, such as an apparent DA-ASAT test in 2013 [20, 9]. Regarding the 2013 test, a spokeswoman for the Pentagon stated “The launch appeared to be on a ballistic trajectory nearly to geosynchronous Earth orbit” [18], indicating that DA-ASAT weapons can threaten all near-Earth orbital regimes. Currently, the tracking and custody of DA-ASAT weapons is a high priority capability highlighted by the United States Space Force (USSF) leadership [1].

The contributions presented in this article are twofold:

1. A walkthrough of the methods, approach, and architecture for the end-to-end sensor orchestration system T&E
2. Demonstration of the T&E methodology in a DA-ASAT scenario modeled from available information

The article is organized as follows: Section 2 provides an overview of the simulation architecture. Section 3 details the evaluation metrics used to assess system performance. Section 4 describes the experimental scenario in which the

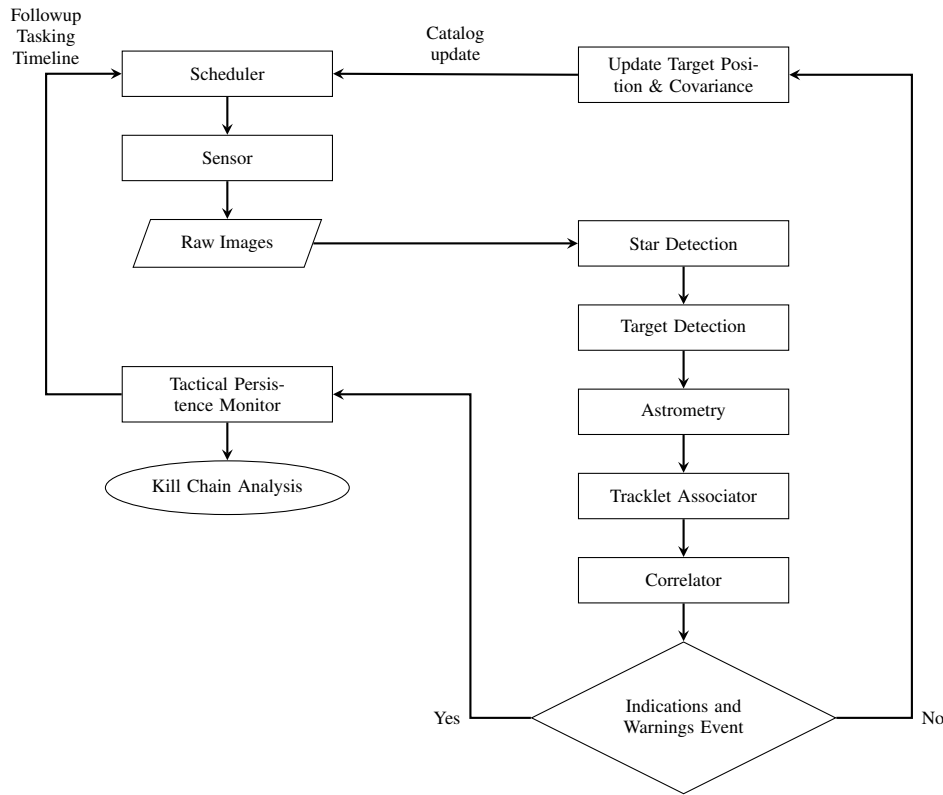


Fig. 2: An example architecture of a system under test, intentionally designed to capture an abstract logic flow common to all space control systems. Systems or components that fit this architecture may be compatible with the test environment.

test environment was applied to test MACHINA’s performance against DA-ASAT launch events. Finally, Section 5 concludes with a summary and outlook toward future work.

2. TEST FRAMEWORK ARCHITECTURE AND METHODOLOGY

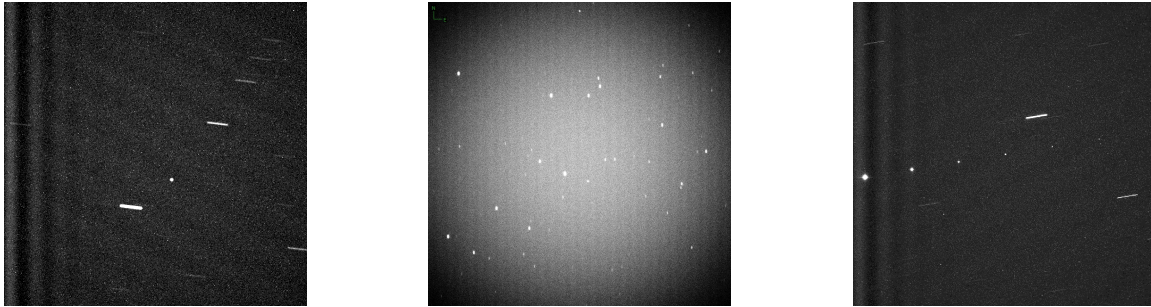
At the core of the test environment is a distributed simulation infrastructure designed to orchestrate scenario participants in a time-stepped, synchronized manner. The simulation engine advances a single, global scenario clock in discrete steps, ensuring that every simulated actor (sensors, satellites, and decision logic modules) receives events and updates in temporal alignment. This centralized clock provides the necessary temporal alignment for accurately recreating time-dependent system operations. By distributing actors across networked nodes and coordinating their interactions via an event queue and messaging system, the framework supports large and complex scenarios while maintaining deterministic behavior.

An important aspect of end-to-end testing of sensor orchestration systems is the interaction with sensor actors. We leverage SensorKit, a software package for telescope/dome/sensor automation, in the test environment to provide a standard abstraction for sensor interfaces [21]. The standard interface makes it possible to inject real, simulated, or hybrid (e.g., hardware-in-the-loop) sensor data into the system under test, provided that the system is compatible with SensorKit. This approach aims to facilitate the integration of diverse sensor systems and phenomenologies (optical telescopes, radars, etc.). For example, digital twins of a Raven-class small telescope or a GEODSS observatory can be deployed in the simulation to emulate the performance characteristics and data flows of those systems [23]. The sensor abstraction enables the integration of new sensor models or actual sensor feeds without requiring changes to the core testing logic.

With an established sensor interface, it is possible to create digital twin sensor models that behave like real sensors.

The sensor actors synchronize with the simulation backend and request simulated scene data (e.g., images or tracking observations) at specified times, just as physical sensors would collect real data. Synthetic data is required for end-to-end assessment of the system under test, as absolute knowledge and control of the "ground truth" is only possible with synthetic data. Specifically, absolute knowledge of ground truth is necessary for accurately assessing the precision and recall of image processing.

In this work, simulated optical sensors use SatSim to generate optical sensor data [3]. SatSim utilizes physics-based algorithms to generate high-fidelity unresolved imagery that is compatible with operational imagery processors. Example imagery generated by SatSim is provided in Figure 3.



(a) A sample SatSim image with pixel non-uniformity. (b) A sample SatSim image with high gain and a short exposure. (c) A sample SatSim image with a string of pearls.

Fig. 3: Three sample SatSim images with different viewing conditions and system parameters.

Rather than using a simulation of, or proxy for, MACHINA, the test environment hosts the as-built solution, using the same artifacts that are used for live deployments of the system. MACHINA issues sensor tasks and processes returned sensor data as if it were orchestrating real sensor hardware. Additionally, the test environment facilitates substitution of MACHINA components with a theoretical or idealized model to experiment with algorithm improvements in isolation. The modularity allows testing of different architectural variants and control strategies under repeatable scenario conditions, enabling consistent comparisons of various algorithms against relevant kill chains.

In summary, the architecture uses a simulation engine to synchronize orchestration with SensorKit's common interface, creating a modular, end-to-end system simulation environment. All sensor actors, autonomous control systems, and external data/model services are advanced in unison through scenario events. This ensures that causal relationships in the scenario (e.g., a satellite maneuver triggering new sensor tasking, or a fast target causing blurred images) unfold as they would in operations and can be analyzed in detail.

3. TEST METRICS FOR SYSTEM EVALUATION

This section defines a suite of standardized test metrics to evaluate system behavior, covering core SDA functions of the system under test. These metrics offer quantitative performance benchmarks that may be compared across different systems or algorithms. Additionally, these measures of core SDA functions are designed to provide explanatory power for when a system does or does not achieve its mission, informing and accelerating development.

The metrics are organized into three groups corresponding to major processing categories: *Sensor Data Processing*, *Observation Data Processing*, and *Sensor Tasking & Scheduling*. The authors note the list of system performance metrics continues to grow and evolve as more tests are performed and feedback is received. The schema of the data used for these tests adheres to or extends the Unified Data Library (UDL) formats where possible, with the aim of minimizing discrepancies from other operational systems. Each metric provides a measure of performance for a specific component or algorithm. Some metrics are specific to the DA-ASAT scenario for the purposes of this paper, but all analysis metrics are configurable within the test environment, depending on the scenario. The metrics should provide insightful quantitative comparison across scenarios and algorithms, and to facilitate identification of system strengths and areas for improvement. Many of the metrics are straightforward to derive from available data; however, equations are provided for metrics that are not apparent from the accompanying description. Guidance for expectations for metric performance is also noted where applicable.

3.1 Sensor Data Processing Metrics

This first category examines the system's ability to extract information from raw sensor data.

- **Target detection:** This test addresses the accurate detection and centroiding of targets in sensor data. Missed detections or inaccurate centroids severely degrade situational awareness and state estimates. The objective is to ensure that sufficiently detectable targets are observed within acceptable error.
 - **M1: Centroid error (EO only)** (Number): Average centroid error in pixels. A conservative expectation for centroid error is not to exceed one pixel.
 - **M2: Precision** (Percent): Percent of true positive detections of targets brighter than the minimum detectable threshold. The minimum detectable threshold is typically expressed as a limiting visual magnitude for EO systems, or expressed as a minimum signal-to-noise ratio.
 - **M3: Recall** (Percent): Percentage of true positive detections from all possible positive detections.

In the case of an optical observation, multiple targets may appear in a single frame, or false alarms detections may result from image artifacts, presenting a data association problem. To distinguish positive detections, missed detections, and false alarms, a bipartite matching problem is solved for the detections and sources in each frame [6]. Let C be a two-dimensional cost matrix, where $C[i, j]$ is the distance in pixels between the i^{th} source and the j^{th} detection. Further, let the two-dimensional match matrix X be a boolean matrix, such that $X[i, j] = 1$ if the i^{th} source is assigned to the j^{th} detection. Then, the matching problem is solved by Equation 1.

$$\min \sum_i \sum_j C_{i,j} X_{i,j} \quad (1)$$

Any unmatched sources result in missed detections, and unmatched detections are false alarms. It is advisable to set an upper bound on the allowable cost of a source-detection pairing. The bound is set to five pixels in the current test environment. Pairings that exceed this bound are discarded, resulting in both a missed detection and a false alarm.

- **Observation association:** This test concerns the accurate association of observations within a collect to form a tracklet, denoted as the observation-to-observation association problem [16]. Mis-associated observations degrade correlation, so the objective is to ensure that tracklet observations originate from a single target.
 - **M1: Single source association rate** (Percent): Percentage of tracklets associated to a single source.
- **Tracklet correlation:** This test evaluates the association of observations to catalog objects. Mis-tags and cross-tags degrade state estimates and expend sensing resources. Objectives include correctly identifying cataloged targets and tagging Uncorrelated Tracks (UCTs).
 - **M1: Correct correlation rate** (Percent): Percentage of observations correlated to the correct catalog target.
 - **M2: False UCT count** (Percent): Percentage of tracklets incorrectly tagged as UCTs.
- **Metric measurements:** This test assesses metric position errors in satellite state measurements (optical and radar), measured in inertial space to capture astrometry and relativistic correction errors. Measurement errors degrade target state estimates; therefore, the objective is to ensure that tracklet metric errors do not exceed allowable thresholds.
 - **M1: RMS tracklet metric error** (Number): Root Mean Square (RMS) error of a given tracklet for optical and radar sensors.
- **Tracklet correlation:** This test evaluates the association of observations to catalog objects. Mis-tags and cross-tags degrade state estimates and can waste sensor time. Objectives include correctly identifying cataloged targets and tagging Uncorrelated Tracks (UCTs).

3.2 Observation Data Processing Metrics

This category covers the downstream exploitation of observations, including orbit determination and threat evaluation. Relevant data models for inputs include EOObservation and RFObservation.

- **Orbit determination:** Tests for orbit quality apply to both Initial Orbit Determination (IOD) and maintenance Orbit Determination (OD) data. Issues include IOD estimates diverging before custody is maintained, and methods taking significant time to reach steady state. The objective of these metrics is to ensure orbit solutions are sufficient for custody.

- **M1: Local shelf life** (Number): The local shelf life of an orbit measures the duration of target custody from the perspective of the sensor that produced the observations used for the orbit solution. IOD solutions are unstable but are expected to be sufficient to perform follow-up collections.

To compute the local shelf life from a given orbit estimate $\hat{\mathbf{x}}$, begin by propagating the estimate until a stopping condition is met, using the dynamics model of choice, $f(\hat{\mathbf{x}})$, as expressed in Equation 2.

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}), \quad \mathbf{x}(0) = \hat{\mathbf{x}} \quad (\text{propagate until stopping condition is met}) \quad (2)$$

For example, under two-body gravitation, the dynamics model $f(\mathbf{x})$ is provided in Equation 3 [24].

$$f(\mathbf{x}) = \begin{bmatrix} \mathbf{v} \\ -\mu \frac{\mathbf{r}}{\|\mathbf{r}\|^3} \end{bmatrix} \quad (3)$$

The stopping condition may be computed from the angle subtended by the estimate slant range $\hat{\mathbf{p}}$ and the true slant \mathbf{p} from a terrestrial observer at position $\mathbf{r}_{observer}$, as shown in Equation 4.

$$\theta(t) = \cos^{-1} \left(\frac{\hat{\mathbf{p}}(t) \cdot \mathbf{p}(t)}{\|\hat{\mathbf{p}}(t)\| \|\mathbf{p}(t)\|} \right) \quad (4)$$

where $\mathbf{p} = \mathbf{x} - \mathbf{r}_{observer}$.

The observer's position $\mathbf{r}_{observer}$ is a fixed site. The shelf life is judged against a typical field of view of an SDA sensor, approximated to be one square degree. Therefore, the shelf life metric captures the minimum amount of time necessary for the estimated state to diverge half of a degree from the true state, as shown in Equation 5.

$$t^* = \min \{t \mid \theta(t) > 0.5^\circ\} \quad (5)$$

This shelf life metric captures the duration of time in which the observing sensor could perform a follow-up collection and still acquire the target in-frame.

- **M2: Global shelf life** (Number): The global shelf life of an orbit measures the duration of time an orbit solution is valid for tasking from any arbitrary sensor site. The metric follows the same approach as Equation 5, but the observer position is chosen to be at nadir of the estimated state, such that

$$\mathbf{r}_{observer}(t) = \frac{\hat{\mathbf{r}}}{\|\hat{\mathbf{r}}\|} R_{Earth} \quad (6)$$

For a network of sensors, the global shelf life metric provides insight into the duration of time that any sensor in the network could task on and acquire the target. As a rule of thumb, a stable OD solution is expected to have a shelf life of at least 24 hours.

- **M3: Average Mahalanobis distance** (Number): Average Mahalanobis distance of the post-orbit solution for all observed targets [14]. All solutions should be within 3 standard deviations of the true value if uncertainty is characterized correctly. In order to compute the Mahalanobis distance, a covariance, P , associated with the state estimate, $\hat{\mathbf{x}}$, is required. Covariance characterization may not be possible for IOD or unstable OD solutions, so this metric is not generally applicable to all orbit solutions. The Mahalanobis distance is defined in Equation 7.

$$d = \sqrt{\tilde{\mathbf{x}}^T \mathbf{P}^{-1} \tilde{\mathbf{x}}} \quad (7)$$

where the residual $\tilde{\mathbf{x}}$ is defined as the difference between the estimate state $\hat{\mathbf{x}}$ and the true state \mathbf{x} , shown in Equation 8.

$$\tilde{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{x} \quad (8)$$

- **Conjunction Prediction (if applicable):** The system is expected to propagate targets to predict possible conjunctions and produce associated Indications and Warnings (I&W) events. The objective is to ensure possible conjunctions are identified in a timely manner, based on the estimated trajectory.
 - **M1: Conjunction target identification** (Boolean): True if the correct satellite is identified that the DA-ASAT model will conjunct.
 - **M2: Time to produce conjunction I&W** (Number): Time elapsed before an I&W event is produced by the system.

3.3 Sensor Tasking/Scheduling Metrics

This group of metrics assesses the system’s ability to generate a schedule for a sensor network based on high-level system objectives and the current understanding of the world state. Relevant UDL data models for inputs include State Vector, Maneuver, Element Set (Elset), Ephemeris, and Conjunction.

- **ASAT Acquisition:** This test evaluates the system’s ability to acquire the un-cataloged target given an external trajectory estimate, without relying on threat priors like Earth observation data. The ASAT must be acquired promptly to initiate kill chain analysis.
 - **M1: Time to acquire ASAT trajectory** (Number): Time elapsed to acquire the ASAT trajectory.
 - **M2: Average observation gap** (Number): Time elapsed between collections on the ASAT.

For convenience, Table 1 summarizes the test measures across these three categories.

Table 1: Summary of Test Measures

Category	Test	Measure	Unit/Type
Sensor Data Processing			
	Target detection	M1: Centroid error M2: Precision M3: Recall	Number Percent Percent
	Observation association	M1: Single-source association rate	Percent
	Tracklet correlation	M1: Correct correlation rate M2: False UCT count	Percent Percent
	Metric measurements	M1: RMS tracklet error	Number
Observation Data Processing			
	Orbit determination	M1: Local shelf life M2: Global shelf life M3: Average Mahalanobis distance	Number Number Number
	Conjunction prediction	M1: Conjunction target identification M2: Time to produce conjunction I&W	Boolean Number
Sensor Tasking/Scheduling			
	ASAT acquisition	M1: Time to acquire launch M2: Average observation gap	Number Number

With the architecture and evaluation metrics in place, the next section describes how the framework was applied to a direct-ascent ASAT threat scenario to assess the performance of the MACHINA system.

4. CASE STUDY: DEVELOPMENT OF BALLISTIC ASCENT CUSTODY CAPABILITY

The first application of the test environment was in support of feature development for MACHINA. To address the DA-ASAT kill chain, MACHINA is scheduled to deliver a ballistic ascent custody objective in the near future. Using the proposed test environment, scenarios were designed to incorporate two fundamental elements of a ballistic object launch:

1. An un-cataloged high-velocity object appears in LEO
2. The object is on a trajectory towards GEO, where it may conjunct with an existing satellite

This type of scenario reflects a stressing operational challenge: a roughly ten hour window to detect, track, and maintain custody of an object with no prior orbital history. The goal of this case study is to evaluate MACHINA's performance in the context of the DA-ASAT scenario. The results support reasoning about the complex interactions among system components and to inform feature development aimed at addressing the specific challenges posed by the DA-ASAT kill chain.

4.1 Scenario Overview and Test Setup

Each simulated scenario begins with an un-cataloged space object appearing in low Earth orbit (LEO). The object ascends through LEO toward the GEO belt, on a GEO transfer orbit (GTO). The scenario concludes when the object reaches GEO altitude, where it may conjunct with other assets depending on the scenario configuration. As a result, the end time of each scenario may vary. The combination of an un-cataloged target, a GTO trajectory, and possible conjunctions at GEO are the model we used for a potential direct-ascent ASAT with publicly available information.

Trajectories and kinematics are modeled using a Two-Line Element set (TLE) and SGP4 propagator in the simulation backend. MACHINA was connected to the test environment as the command-and-control system under test. To initiate sensor tasking in the scenario, noisy observations are injected of the ASAT vehicle as it reaches LEO. The injected observations are routed to MACHINA's orbit determination component, where an ephemeris is generated from the observations. In this configuration, MACHINA operates exactly as it would in the real world: monitoring sensor feeds, issuing tasking commands against ephemeris, and attempting to detect and track an un-cataloged target. Initially, MACHINA recognizes that the observations do not correlate to a known object and begins scheduling observations to acquire custody.

A network of ground-based optical sensor actors were connected to MACHINA in each scenario. The available sensors vary by scenario, but generally, multiple geographically distributed sensors were available for tasking, where each sensor had line of sight with the un-cataloged object at some point during the scenario. Using SensorKit, representative sensor systems were instantiated as digital twins. As the ASAT launch unfolds in the simulation, MACHINA generates sensor tasking orders via its standard interface. These tasking messages are routed to the appropriate SensorKit digital twin, prompting the simulated Raven sensors to "collect" imagery at the specified time and pointing. Model providers then generate the requested synthetic sensor data. The parameters used to simulate the Raven telescopes may be found in Nishimoto et al [15].

The synthetic data produced by the Raven digital twin is a star-field image that covers the specified sky patch and includes any simulated objects in view (notably, the ASAT, if present). As stated previously, SatSim was used to render these images with high physical fidelity. The synthetic image is returned to MACHINA through the same interface as a real sensor image would be. MACHINA's autonomous processing pipeline ingests the image, performs astrometric calibration and object detection, and checks for any new detections that could correspond to the object if present. The closed-loop tracking—MACHINA commanding simulated sensors and analyzing the returned data—continues throughout the engagement timeline. A test campaign of 10 runs of this scenario was executed. The key scenario inputs are provided in Table 3 in Appendix A.

4.2 Outcomes and Roadmap

Across all simulated runs of the direct-ascent ASAT scenario, the test environment captured performance data on MACHINA's tasking decisions and data processing outputs. The analysis of these results yielded actionable insights into both MACHINA's current capabilities and development roadmap for tracking a ballistic target. The key performance metrics are highlighted and analyzed in this section.

4.2.1 Sensor Tracking Performance

The astrometry component struggled to plate solve imagery captured by the Raven digital twin models while tracking the target in the LEO regime. Across all scenarios, no observations were recorded for the target while in LEO. Although the target was within the sensor field of view (confirmed by the synthetically generated ground-truth imagery), observation data cannot be extracted without a plate solution.

The GTO trajectory of the ASAT results in velocities faster than typical LEO satellites, which is a condition rarely encountered in nightly operations. The high target velocity results in a complex tradeoff during collection – a short integration time risks producing target centroids too dim to be detected, while a long integration time risks stars streaking off frame. The limited field of view and resolution of the Raven sensors further constrains the parameters necessary for collecting sufficient imagery. The optimal integration time may require precision to hundredths of a second, and will change as the target gains altitude. In this case study, the imagery collected by the Raven sensor models was unsuitable for detecting stars when tracking the DA-ASAT target model. Astrometric measurements were not extracted from the imagery as a result, and there was insufficient information to produce a predicted conjunction. Although not initially anticipated, this result is understandable in retrospect, as a high velocity LEO target is one of the most challenging cases for an optical sensor from a physics perspective.

This outcome highlights a key consideration for the use of heterogeneous sensors for SDA. Precisely tuned integration times are required for Raven-like telescope imagery to produce observations in such scenarios. More exquisite optical systems likely require less tuning for LEO observations but face the same challenges. Alternatively, active radar or other radar frequency (RF) spectrum systems excel in the LEO regime, and these sensors provide a compelling case for observation collection when an un-cataloged target appears in LEO. Based on this result, future work will be done to support RF phenomenology sensors for DA-ASAT scenario analysis. It is expected that the suggested alternatives will successfully yield consistent detections in the LEO-regime of the ascent trajectory, and MACHINA’s planning logic for ballistic object custody will incorporate these findings accordingly.

4.2.2 Sensor Orchestration and Coverage

MACHINA demonstrated competent sensor orchestration under the conditions tested. In each scenario, MACHINA successfully generated and distributed collection requests to the available sensors. Within the roughly 10 hour duration of each scenario, an average of 62 collections were scheduled, meaning the system succeeded in producing tasks that satisfied all observability constraints approximately every 10 minutes. Across all scenarios, the average time to the first scheduled collection was about 8.36 minutes.

An example schedule is shown in Figure 4, depicting the scheduled tasks across four available sensors for the duration of the scenario.

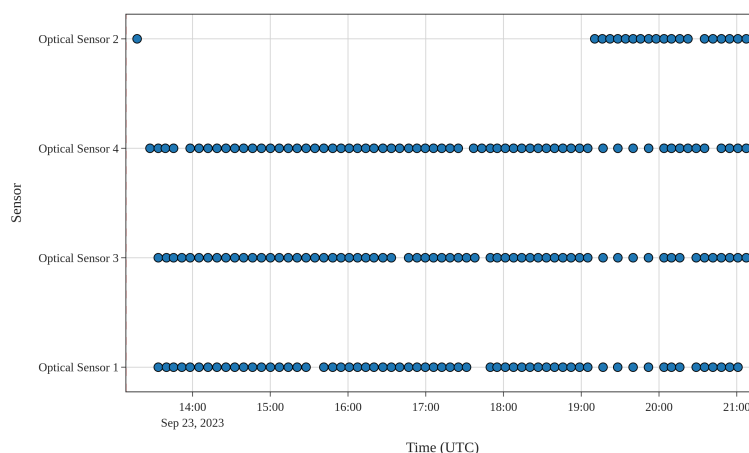


Fig. 4: A sample sensor schedule, showing that many tasks were generated for the available optical sensors in the scenario. Each point along the timeline indicates a collection request, which generates 10 individual frames.

The simulated telescopes responded to MACHINA’s tasking and generated imagery at the requested times. Notably, MACHINA’s scheduling algorithm successfully allocated multiple observation opportunities during the ascent of the object through LEO.

Overall, the system’s planning logic (which leverages visibility predictions and sensor availability) ensured that observations were attempted in a timely manner. In qualitative terms, MACHINA’s coordination of sensors and its use of the distributed network were successful: the right sensors were looking at the right place at roughly the right time. This positive result underscores that the concept of autonomous sensor management reduces timelines for coverage of emerging events, as MACHINA incurred minimal delays between receiving the launch cue and re-tasking the sensor.

4.2.3 Orbit Solution Quality

Orbit determination is a key component of object custody. Because the scenario begins with the injection of noisy, angles-only observation data, this section analyzes the quality of the IOD solution resulting from the injected observations. The “local shelf life” and “global shelf life” test metrics for the IOD solutions in each scenario are provided in Table 2. MACHINA’s IOD algorithm does not produce a covariance associated with the orbit solution, so no Mahalanobis distance was computed for the orbit quality metrics.

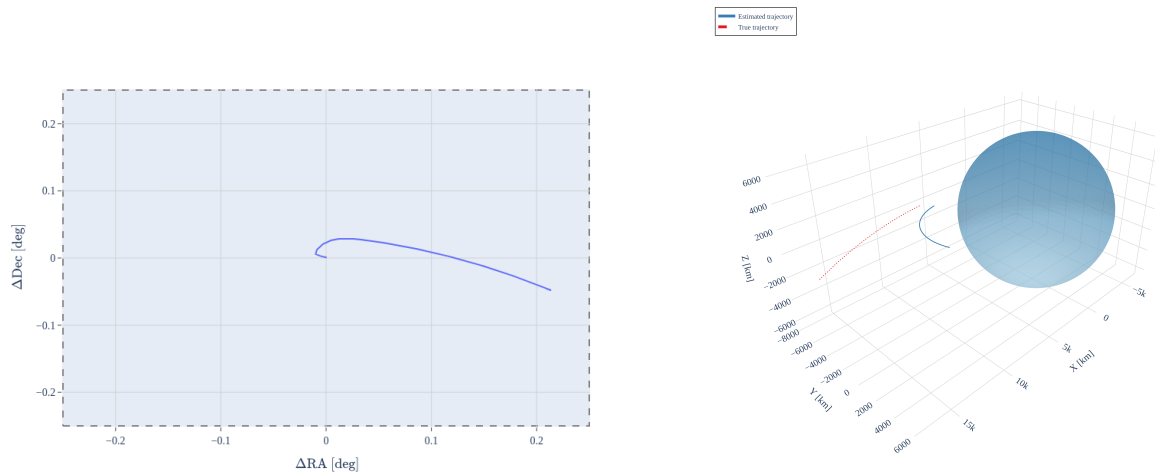
Table 2: Orbit Quality Test Measures of IOD Solutions

Scenario 1		Scenario 2	
Global Shelf Life	80.04 minutes	Global Shelf Life	0.0 minutes
Local Shelf Life	627.75 minutes	Local Shelf Life	47.53 minutes
Scenario 3		Scenario 4	
Global Shelf Life	0.0 minutes	Global Shelf Life	0.0 minutes
Local Shelf Life	15.4 minutes	Local Shelf Life	179.41 minutes
Scenario 5		Scenario 6	
Global Shelf Life	0.0 minutes	Global Shelf Life	0.0 minutes
Local Shelf Life	55.32 minutes	Local Shelf Life	14.51 minutes
Scenario 7		Scenario 8	
Global Shelf Life	0.0 minutes	Global Shelf Life	0.0 minutes
Local Shelf Life	25.75 minutes	Local Shelf Life	47.53 minutes
Scenario 9		Scenario 10	
Global Shelf Life	0.0 minutes	Global Shelf Life	0.0 minutes
Local Shelf Life	10.19 minutes	Local Shelf Life	30.69 minutes

With the exception of the first scenario, MACHINA’s IOD of the injected ASAT observations resulted in poor global shelf-lives. However, the local shelf-lives resulted in valid follow-up collections from the initial observing sensor for at least 10 minutes, and up to several hours in some cases. Previous analyses of MACHINA’s IOD capability exhibit equal or superior performance when compared to traditional methods [4], so the use of other methods is not expected to produce significantly different results.

Figure 5 illustrates how an IOD solution appears from a local frame and an inertial Earth-centered frame, to provide intuition for the results for local and global shelf life. Both solutions are propagated for 30 minutes.

This result provides a key insight into effective ballistic object custody for MACHINA. While MACHINA is capable of orchestrating a network of sensors to cooperatively collect observations, results indicate that successful follow-up observations may only be possible by the sensor that first produced a track on the ballistic target. Notably, the local shelf-lives exceed the average time to the first scheduled collect of the ASAT at 8.36 minutes, providing merit that MACHINA effectively orchestrates sensors to respond to a newly-launched ballistic object.



(a) The field of view of a sensor tracking an IOD solution with high local shelf life. The field of view is centered on the IOD solution, while the curve shows the drift of the true orbit relative to the IOD solution.

(b) The inertial frame view of an IOD solution with low global shelf life. While the object may appear close to the true trajectory from the observing sensor's point of view, a sensor at a different geographical site would likely not observe the object when tracking the IOD solution.

Fig. 5: Visual comparison of local and global shelf life metrics for an IOD solution.

5. CONCLUSION

A novel testing approach has been presented for a simulation-driven framework to evaluate and improve autonomous SDA systems. This work demonstrated the successful end-to-end testing of scheduling, image processing, and orbit determination components unified into a sensor orchestration system. The system was deployed within the test environment and evaluated against operationally relevant conditions. Through a detailed case study involving a simulated DA-ASAT model, it was demonstrated that the test approach not only provides performance analysis but also supports feature development for kill chains prioritized by USSF leadership. Exercising MACHINA within the test environment facilitated the development of a concrete roadmap towards a ballistic object custody capability. Key factors for success were identified in the DA-ASAT scenarios, which will guide the feature development:

1. Tracking high-velocity targets in LEO is challenging with Raven-like sensors and may be more suitable for alternative sensors, such as exquisite optical systems or RF-spectrum sensors
2. Follow-up collections on UCT targets in LEO are likelier to succeed when re-tasking the same observing sensor, instead of tasking sensors at other geographical sites
3. Reactive follow-up tasking in 10 minutes or less mitigates risk of lost target custody after initial detection

Beyond the DA-ASAT scenarios examined in the provided case study, the test environment architecture may be scaled from brief, high-tempo engagements to extended, complex operations spanning days and involving numerous actors. Whether it is testing continuous day-to-day surveillance workflows, integrating with a marketplace of commercial sensors, or simulating advanced orbital pursuit threats, the test environment supports a variety of applications. By using a distributed, time-stepped simulation engine and a sensor abstraction layer, the framework maintains a high degree of realism and repeatability, which is considerable for both scientific analysis and mission rehearsal. The work presented in this paper is part of an ongoing effort to bridge the gap between simulation and real-world SDA operations.

Planned work will incorporate additional emerging SDA technologies into the test environment to evaluate their potential impact in a controlled environment. These technologies include alternative orbit determination schemes and

sensing phenomenologies, such as RF-spectrum sensors. Further refinements to the fidelity of simulations will continue, including more sophisticated space environment effects and adversary behavior models, to challenge space control systems like MACHINA against cutting-edge space capabilities.

Future applications of the test environment aim to explore scenarios beyond the DA-ASAT profile. One such planned application is the evaluation of MACHINA (and related algorithms) against co-orbital engagements. By simulating this within the test environment, the aim is to identify any latent issues in the system’s long-term planning or data association algorithms that may not be apparent in the more abrupt direct-ascent scenario. This planned experiment highlights the broad forward-looking applications of the proposed methodology. Before such events ever occur in reality, they can be rehearsed in simulation, and capabilities and features can be improved and accelerated. Ultimately, by providing a comprehensive and scalable proving ground, the proposed test environment shortens the innovation cycle for SDA, allowing concepts to be vetted, matured, and optimized in simulation before they are deployed to real-world operations.

6. ACKNOWLEDGEMENTS

The views expressed are those of the author and do not necessarily reflect the official policy or position of the Department of the Air Force, the Department of Defense, or the U.S. government.

A. SCENARIO INPUT CONDITIONS

Table 3: Input Conditions for Each Scenario: Satellite TLE, Sensor Locations, and Time Bounds

Scenario 1	Scenario 2
TLE: Line 1: 1 99999U 000307B 23266.54640000 +.00000000 +00000-0 +00000-0 2 0001 Line 2: 2 99999 043.6667 333.8615 7431620 344.6613 1.5250 02.120620480 0001 Start Time: 2023-09-23T13:08:55.000000Z End Time: 2023-09-23T21:16:55.766281Z Sensor Locations: - Lat: 35.331, Lon: -105.654 - Lat: 38.963, Lon: -108.237 - Lat: 37.07, Lon: -119.413 - Lat: 13.441, Lon: 144.778	TLE: Line 1: 1 99999U 000454B 23266.41340000 +.00000000 +00000-0 +00000-0 2 0001 Line 2: 2 99999 000.0414 165.5609 8784913 274.3613 4.0544 01.595156420 0001 Start Time: 2023-09-23T09:59:18.000000Z End Time: 2023-09-23T22:55:44.971999Z Sensor Locations: - Lat: -29.69, Lon: 17.818 - Lat: -23.088, Lon: 29.913
Scenario 3	Scenario 4
TLE: Line 1: 1 99999U 000356B 23266.13750000 +.00000000 +00000-0 +00000-0 2 0001 Line 2: 2 99999 004.2447 066.2052 7556697 165.4235 1.3434 02.030261210 0001 Start Time: 2023-09-23T03:38:18.000000Z End Time: 2023-09-23T12:14:54.100850Z Sensor Locations: - Lat: -30.471, Lon: -70.765 - Lat: 29.348, Lon: -82.119	TLE: Line 1: 1 99999U 000467B 23266.42530000 +.00000000 +00000-0 +00000-0 2 0001 Line 2: 2 99999 000.0135 153.3375 8633182 116.6988 4.2204 01.712330810 0001 Start Time: 2023-09-23T10:16:20.000000Z End Time: 2023-09-23T22:04:00.188658Z Sensor Locations: - Lat: 35.331, Lon: -105.654 - Lat: 38.963, Lon: -108.237 - Lat: 37.07, Lon: -119.413 - Lat: 20.667, Lon: -156.399 - Lat: 29.348, Lon: -82.119 - Lat: 30.588, Lon: -104.081

Scenario 5

TLE:

Line 1: 1 99999U 000326B 23266.54070000
+.00000000 +00000-0 +00000-0 2 0001

Line 2: 2 99999 000.0345 211.8250 9184829
247.6160 4.5403 01.858475680 0001

Start Time: 2023-09-23T13:02:36.000000Z

End Time: 2023-09-23T23:51:53.996384Z

Sensor Locations:

- Lat: -34.661, Lon: 19.537

- Lat: -23.088, Lon: 29.913

Scenario 6

TLE:

Line 1: 1 99999U 0 23266.73790000 +.00000000
00000 0 00000 0 0 0001

Line 2: 2 99999 16.0218 162.9586 6238784
340.2834 3.6912 3.896194790 0001

Start Time: 2023-09-23T18:03:11.000000Z

End Time: 2023-09-24T06:38:52.338580Z

Sensor Locations:

- Lat: 30.237, Lon: -5.608

- Lat: 35.07, Lon: 25.973

- Lat: 37.501, Lon: -2.409

- Lat: 44.408, Lon: 5.515

Scenario 7

TLE:

Line 1: 1 99999U 000614B 23266.51440000
+.00000000 +00000-0 +00000-0 2 0001

Line 2: 2 99999 002.8834 082.3168 8054819
143.5748 2.4695 01.572946090 0001

Start Time: 2023-09-23T12:32:38.000000Z

End Time: 2023-09-24T01:20:30.833035Z

Sensor Locations:

- Lat: -34.269, Lon: 140.342

- Lat: 13.441, Lon: 144.778

Scenario 8

TLE:

Line 1: 1 99999U 000301B 23266.19780000
+.00000000 +00000-0 +00000-0 2 0001

Line 2: 2 99999 007.8819 088.0044 7419428
348.6868 1.1281 02.136216440 0001

Start Time: 2023-09-23T05:16:20.000000Z

End Time: 2023-09-23T12:48:54.839069Z

Sensor Locations:

- Lat: -31.82, Lon: 117.281

- Lat: -23.767, Lon: 133.915

- Lat: 13.44, Lon: 144.778

Scenario 9

TLE:

Line 1: 1 99999U 000159B 23266.48680000
+.00000000 +00000-0 +00000-0 2 0001

Line 2: 2 99999 002.8214 192.2364 7320849
181.3408 359.8674 02.283091330 0001

Start Time: 2023-09-23T12:01:00.000000Z

End Time: 2023-09-23T23:04:00.188658Z

Sensor Locations:

- Lat: 29.348, Lon: -82.119

Scenario 10

TLE:

Line 1: 1 99999U 000162B 23266.77190000
+.00000000 +00000-0 +00000-0 2 0001

Line 2: 2 99999 006.9328 055.2478 7296241
180.6536 359.9297 02.278188150 0001

Start Time: 2023-09-23T18:44:50.000000Z

End Time: 2023-09-24T01:39:04.639509Z

Sensor Locations:

- Lat: -34.661, Lon: 19.537

- Lat: -29.69, Lon: 17.818

REFERENCES

- [1] Sda tap lab. <https://sdataplub.org/>. Accessed: 2025-07-15.
- [2] Annual threat assessment of the u.s. intelligence community 2025. Technical report, Office of the Director of National Intelligence, 2025.
- [3] Alexander Cabello and Justin Fletcher. SatSim: a synthetic data generation engine for electro-optical imagery of resident space objects. In Genshe Chen and Khanh D. Pham, editors, *Sensors and Systems for Space Applications XV*, volume 12121, page 1212107. International Society for Optics and Photonics, SPIE, 2022.
- [4] Alexander Cabello, Jeff Houchard, Cameron Harris, J Zachary Gazak, Jonathan Kadan, and Justin Fletcher. Learned initial orbit determination from simulated electro-optical observations. In *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, page 75, 2024.

- [5] Andrew G Clark, Neil Walkinshaw, and Robert M Hierons. Test case generation for agent-based models: A systematic literature review. *Information and Software Technology*, 135:106567, 2021.
- [6] David F Crouse. On implementing 2d rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, 2016.
- [7] Justin R. Fletcher and Jonathan E. Kadan. Autonomous interactive agents for global telescope network orchestration. In Jorge Ibsen and Gianluca Chiozzi, editors, *Software and Cyberinfrastructure for Astronomy VIII*, volume 13101, page 131012U. International Society for Optics and Photonics, SPIE, 2024.
- [8] I Goumiri, Luc Peterson, Ashley Cocciadiferro, Ryan Lee, Jason Bernstein, and S Allen. A common task framework for testing and evaluation at the space domain awareness tools, applications, and processing lab. Technical report, Lawrence Livermore National Laboratory (LLNL), Livermore, CA (United States), 2024.
- [9] Mike Gruss. U.s. state department: China tested anti-satellite weapon. *SpaceNews*.
- [10] Shirley Kan. China’s anti-satellite weapon test. Technical report, 2007.
- [11] TS Kelso. Analysis of the 2007 chinese asat test and the impact of its debris on the space environment. In *8th Advanced Maui Optical and Space Surveillance Technologies Conference, Maui, HI*, volume 7, 2007.
- [12] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ international conference on intelligent robots and systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154. Ieee, 2004.
- [13] James Mackey. Recent us and chinese antisatellite activities. *Air & Space Power Journal*, 2009.
- [14] Goeffrey J McLachlan. Mahalanobis distance. *Resonance*, 4(6):20–26, 1999.
- [15] Daron Nishimoto, David Archambeault, David Gerwe, and Paul Kervin. Satellite attitude from a raven class telescope. 2010.
- [16] Alejandro Pastor, Manuel Sanjurjo-Rivo, and Diego Escobar. Track-to-track association methodology for operational surveillance scenarios with radar observations. *CEAS Space Journal*, 15(4):535–551, 2023.
- [17] Phillip C Saunders and Charles D Lutes. China’s asat test: motivations and implications. 2007.
- [18] Marc V. Schanz. Chinese anti-satellite test? *Air & Space Forces Magazine*.
- [19] Shital Shah, Debadepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics: Results of the 11th international conference*, pages 621–635. Springer, 2017.
- [20] Andrea Shalal-Esa. U.s. sees china launch as test of anti-satellite muscle -source. *Reuters*.
- [21] Ryan Swindle, Eric Blackhurst, Kevin Iott, Shelby Stubbe, Matthew Phelps, J Zachary Gazak, and Justin Fletcher. A mobile, off-grid, robotic observatory for space domain awareness. In *Ground-based and Airborne Telescopes X*, volume 13094, pages 866–879. SPIE, 2024.
- [22] Clayton Swope, Kari A Bingen, Makena Young, and Kendra Lafave. Space threat assessment 2025. Technical report, Center for Strategic and International Studies (CSIS), 2025.
- [23] Paul F Sydney, John L Africano, Amy Fredericks, Kris M Hamada, Vicki Soo Hoo, Daron L Nishimoto, Paul W Kervin, Steve Bisque, and Matthew Bisque. Raven automated small telescope systems. In *Imaging Technology and Telescopes*, volume 4091, pages 237–247. SPIE, 2000.
- [24] William E Wiesel. *Spaceflight Dynamics*. CreateSpace, third edition, 2010.

Approved for public release; distribution is unlimited. Public Affairs release approval #AFRL-2025-4360.