# Predicting Custody of Objects in Cislunar Space

**Sean O'Neil**[1]
*MITRE*

## ABSTRACT

One of the primary goals of space surveillance is the establishment and maintenance of custody on resident space objects (RSO's) in orbit. For an RSO, custody at any point in time can be defined as a combination of two attributes—localization of an object with sufficient accuracy and compactness such that additional metric data could be gathered on it with minimal ambiguity using the sensors the user has at their disposal, along with a sufficient degree of confidence in the unique target label, such as a catalog or track number, that was applied to that RSO when its orbit was first estimated. Degradation of either metric accuracy or confidence in the label results in loss of custody.

Given the importance of custody, it would be useful to be able to predict the ability of a given set of sensors to maintain said custody against various types of space objects, and in varying regimes. We apply a recently-developed method for use on ground tracking to this domain. This approach includes an ability to predict the impact not only of the cadence/quality/gaps of metric obs on label confidence, but also the impact of so-called feature (i.e. non-kinematic) data, such as apparent brightness, fluctuations, spectral information, RF emissions, or other measurements of the object that aren't directly dependent on position or velocity on said label confidence.

We performed simulations in a Circular Restricted 3-Body Problem (CR3BP) environment using two orbiting angles-only sensors viewing a RSO in a periodic cislunar orbit. To this we added a feature-collecting telescope located on Earth that periodically collects data against the target. Simulations demonstrate the importance of feature data in maintaining target label confidence, and hence custody.

### 1. INTRODUCTION

One of the primary goals of space surveillance is the establishment and maintenance of a catalog of resident space objects. Such a catalog commonly contains not only regularly updated estimates of ephemerides for each object, but also some identifying information such as date and location of launch, and even possibly some information pertaining to the characteristics of the RSO itself. Implicit in the idea of a catalog is that objects are not maneuvering, at least not very often or deliberately (e.g., High Area to Mass Ratio (HAMR) objects), and that they are semi-permanent.

This is somewhat distinct from the way objects are viewed in other domain. For example, in the air domain objects are viewed as fleeting, and the main goal is "birth-to-death" tracking, meaning that the object becomes tracked shortly after it is in the air, or enters a particular airspace, and the track is terminated once it lands or exits the airspace. An important component of this is the goal of assigning a single track number to a particular target for the duration of its presence in the airspace. This is made simpler in the air domain by the presence of IFF that typically provides some kind of identification of the object, though that is not always present for every single airborne entity.

A more complicated version of this issue occurs in the ground domain. Ground targets move less predictably than air targets, the densities are much higher both in a literal sense and even in a normalized sense, meaning absolute density in the measurement space normalized by typical sensor resolution, and they can disappear and reappear quickly due to a variety of mechanisms, the chief being obscuration in all its forms. Ground targets also typically do not emit a signal that can be readily tied to the track on that target, so the goal of a single track number for a single target is usually very difficult if not impossible to achieve. Nevertheless, the idea of "birth-to-death" tracking is

[1] The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author.

equally important in the ground domain, and so practitioners continue to try and extend the length of time over which it is possible to keep a particular track number assigned to a particular target.

One approach that has been found useful in the ground domain is called feature-aided tracking (FAT) (also sometimes referred to in the literature as signature-aided tracking (SAT) or classification-aided tracking (CAT)). FAT is a technique most commonly-employed in radar systems, though it has also found use in optical tracking systems. In FAT, the normal surveillance mode is interrupted aperiodically with a radar mode that produces measurements that are only dependent on pose and speed of the target relative to the radar sensor. Modes such as Synthetic Aperture Radar (SAR) and High-Range Resolution Radar (HRR) are typically employed, as they produce signatures that are somewhat invariant once pose and LOS speed are taken into account. A detailed discussion of one implementation of this approach can be found in [1].

An easy way to understand the problem and how FAT can be used to help solve it is shown in Fig. 1. Two targets approach an intersection and end up close together for a time. Perhaps one or more of them stop at the intersection to let the other one pass unimpeded. For Ground Moving Target Indicator (GMTI) radar mode, which is the typical metric observation mode employed for ground surveillance radars, that means one or both targets disappear into the clutter while stopped. Even if they slow down sufficiently, they will disappear into the clutter. Sometime later the targets come to another intersection and exit via different paths. Since both targets may have stopped or at least moved slowly enough, and because they were close together for a time and the radar is only sampling the targets periodically rather than staring, on the basis of metric observations alone it is impossible to determine which target exited along which path. If FAT is being employed, then after the targets leave the intersection and are sufficiently separated (which depends on the sensor resolution), then an HRR mode would be employed and used to match against a signature library maintained for each target. Assuming the match performance is sufficiently different for the two targets, the ambiguity can be resolved and the proper track numbers can be reassigned if needed to those targets.

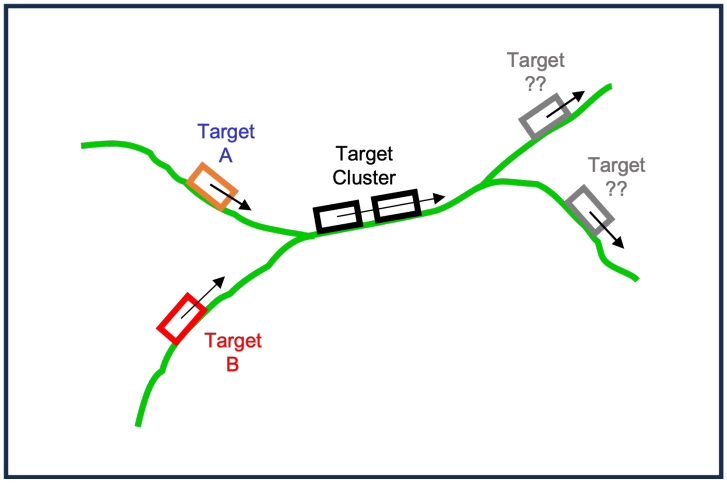



Figure 1: Targets A and B Encounter Each Other, Leading to Ambiguity After They Separate

In this paper we propose to begin to investigate the performance of that type of technique to RSO tracking or cataloging. One reason for this proposal is that, as we move into the cislunar domain, many of the assumptions underlying the current cataloging process are being stressed. In particular, the issues of orbit behavior and predictability, coupled with the challenges of making repeated metric observations on very distant objects, will stress the notions implicitly present in the current SSA process. Our belief is that FAT presents one realizable option for improving potential performance in this domain where objects difficult to collect on will increasingly behave in a dynamic and challenging-to-predict manner.

## 2. CUSTODY

Above we discussed, particularly for the ground tracking environment, how there is an ever-present issue of maintaining a unique and unchanging track number for objects within one's purview. For discussion purposes we will refer to this as the "track-labelling problem". This is one in which the challenge is to maintain a unique track label on each object within the scene, at the very least for the time that the object remains in the scene and potentially ad infinitum. We define a label as being an identifier that is unique to a particular target, though it is important to stress that that identifier need not provide any information beyond signifying which object this is. A simple example of a label is "the target that I first began tracking under track number XYZ". Thus, a label merely says that this object is Object A and not any other object, and Object A is understood to be unique. We adopt this nomenclature for clarity and to avoid unnecessary confusion with notions of ID and classification. Those can be viewed as additional information beyond the mere label. However, the importance of a label is that the ability to maintain it also a) preserves origin and behavior history information, and b) preserves whatever other target characteristic or identification information may have been gathered and associated with that track during its existence.

As shown in Fig. 1 above, one can easily construct situations where a label becomes ambiguous or confused for at least a period of time. Thus it is important to also introduce the notion of label confidence. This can be viewed as a measure of the probability the tracker assigns to a particular label being correct. When confusion arises, that translates to that measure becoming less than 1 for the label it had. It is also useful to imagine carrying along multiple potential labels for a target, each assigned a confidence, with said confidences summing to 1. As an example, consider the problem cited earlier where two targets meet at an intersection and become ambiguous due to sampling, resolution, and/or possibly obscuration. Our initial track labels for Track 1 and Track 2 on Targets A and B were unambiguous, so that initially:

$$TL(1) = \{A: 1.0\} \tag{1}$$

$$TL(2) = \{B: 1.0\} \tag{2}$$

Where $TL(1)$ represents the confidences associated with track 1, and $TL(2)$ represents the confidences associated with track 2. After the encounter, assuming that it occurred such that the labels were completely ambiguous (though often less than complete ambiguity results for many kinds of confusion events), then the new set of track label confidences would be,

$$TL(1) = \{A: 0.5, B: 0.5\} \tag{3}$$

$$TL(2) = \{A: 0.5, B: 0.5\} \tag{4}$$

Another useful notion is that of a catchall label category often referred to as "Other Target". This is some unspecified target or set of targets, usually representing the universe of unlabelled targets, that may potentially have become confused with our known labelled set of targets. For the example in Fig. 1, there might also be the possibility that one or more of the targets stayed stopped at the intersection or nearby, and a previously unknown target or targets started up in that vicinity. An example of how we would represent that is given as,

$$TL(1) = \{A: 0.45, B: 0.45, Z: 0.1\} \tag{5}$$

$$TL(2) = \{A: 0.45, B: 0.45, Z: 0.1\} \tag{6}$$

What the above is indicating is that Targets $A$ and $B$ are completely ambiguous and equally probable as belonging to either Track 1 or 2, but there is also a 10% chance for either track that it is some other previously unknown target, which we designate by the label $Z$. Note how the label confidences for each track number still sum to 1.

Unfortunately for tracking practitioners, the issue of how those confidences should be computed is sometimes overlooked or confused with other things, such as track score. In addition, the Mori formula [2] for correct association probability $P_{CA}$ for metric tracking is often used as a means of determining the track label confidence, by computing the cumulative product of $P_{CA}$ for each metric obs time. It should be noted that this method would be

valid if a single misassociation meant complete track swap or loss, but that is not true especially as the time between obs becomes smaller (i.e. faster revisits). However, the general result seen from applying this method is that track confidence is always dropping over time. This is inevitably true when only metric (i.e. kinematic-only) obs are being taken as there is mechanism for regaining confidence once it has been lost. Hence the need for a method such as FAT.

One tracking method that does permit ready computation of label confidences is a multi-hypothesis tracker, or MHT. Consider the unrealizable but useful example of an exhaustive MHT over all observation times. Such an MHT must consider the combinatorially expanding (as revisits accumulate) set of partitions of the data collected over all revisits, and explicitly evaluate their likelihood. By partition of the data we mean a grouping of subsets of it such that all data elements (in this case, obs) are in one and only one of the subsets. Each partition is a global assignment of all obs to labels, and its probability can be computed. Targets that are well-separated from other targets can be viewed as independent and their label probabilities can be summed independently. Targets that are not well-separated (i.e., are in a cluster) have to be treated in a coupled fashion.

As mentioned, the exhaustive MHT is unrealizable, but it is useful inasmuch as it provides the maximum a posteriori estimate of target state and label under a suitable set of assumptions. While it is unrealizable due to its combinatorial growth, much work has gone into the construction of approximations to it, which typically involve viewing the construction of the partitions as creating a tree, with branches of the tree continually pruned away or combined based on their probability measure and their degree of similarity. One of the main challenges is understanding how well an MHT will actually perform under a given set of circumstances.

Recently, an approach to predicting the performance of such MHTs for the ground tracking environment has been developed [3][4]. That approach is aimed at predicting the expected ability of an ideal MHT to maintain custody on a given object in a specific scenario. Before describing this further, some definitions are in order. First, custody is present on a target when a) the track on that target has sufficient accuracy in position and velocity, and b) the confidence in the correct target label for that track is sufficiently high. What those precise values for metric accuracy and label confidence need to be depends on the application and on user preferences, so a prediction method should aim at quantifying what those expected values are and then let the appropriate thresholds be applied to determine if custody is present. The second definition involves what we mean by "specific scenario". By this, we mean one in which the target trajectories have been defined, as have the sensor capabilities and trajectories. In addition, the sensor observation times have also been defined. The aim is then to take that information and predict how well an idealized MHT could be expected to maintain custody on any given target in the scenario.

The point of this technique is to develop a realistic bound on the level of performance that a well-studied approach to tracking, and one that is optimal under reasonable conditions, would achieve. We feel this technique is especially useful if one is trying to determine what sensor architecture will be sufficient for what kinds of target environments and scenarios. It is also useful in that it establishes an upper baseline for performance, which, if we are careful, should be closely approachable by a tracker instantiation if we allocate sufficient time and resources to it. In this way, such an approach provides a way of determining how much a given instantiation might be improved. Also, it means we have not bound our prediction to a particular suboptimal tracker, and do not need to worry about separating out elements of performance which are due to shortcuts or bugs in that tracker, which are due to fundamental information limitations. Furthermore, we have freed ourselves from many of the concerns over Monte Carlo evaluation using a tracker instantiation and the number of random variations on obs realizations required for statistical significance.

[3] and [4] in combination detail how such a calculation is carried out for the ground tracking environment. We shall provide a brief description to provide a sense to the reader of how that is being adapted for SSA. The idea behind this method is to do an efficient computation of the expected results of an MHT. This computation is kept efficient via two principal mechanisms but also acknowledges one irreducible complicating factor:

- Obs are treated as distributions according to their metric accuracy and according to the sensitivity of the sensor relative to the target (meaning $P_D$ and $P_{FA}$). The emulation of the evolution of track hypotheses proceeds according to these distributions and there is no need for multiple random number draws to generate detection/false alarm/clutter realizations.

- Hypotheses that we can foresee will be driven to zero probability as time progresses by the exhaustive MHT are discarded. These are typically "breakaway" hypotheses that separate from any modelled target by a certain distance.
- Targets are treated independently so long as they are well-separated. However, once they cluster, they must be considered jointly, and during target clustering that portion of the computation will increase combinatorially in complexity while the targets remain clustered

With that in mind, we now state how this process is carried out:

1. For a single well-separated target in clutter, for which we know the motion model, we begin with a PDF represented by a single mode (i.e. hypothesis).

2. That PDF evolves to a set of modes after a single revisit, which evolution we model with a Kalman filter computation using the known motion model, said set coming from mixtures of the target being detected or not, and a false alarm(s) occurring or not. Because we are taking an expectation over the distributions of where true and false detections may emerge, the modes representing those events are centered on the target location.

3. We lump elements of those distributions that stray too far from the target as belonging to a diffuse mode that surrounds the neighborhood of the target, and calculate the weight of that diffuse mode based on the logic that most of it will have probability driven to zero by the exhaustive MHT. Only those parts of it that recombine into the modes close enough to the target have a chance of surviving.

4. The modes close to the target are all centered around it, and are Gaussian or at least unimodal. We reapproximate this cluster of distributions at each step with a single Gaussian, which keeps our computation complexity bounded. As part of this, we also keep track of label confidence as it evanesces away, primarily through the chance that some other target has appeared in the track's vicinity, and also the chance that the target stopped or became obscured.

5. When two or more targets approach closely enough, we combine their states into a single joint state and execute this above process in a coupled manner until the targets separate. It is in this stage that we incur computations that grow combinatorially with the number of targets.

   We believe this is not a major concern for the following reason. The primary reason for doing this computation is to understand the degree of confusion that will result from multiple targets interacting. Many times this confusion is due to one-on-one encounters and is only partial, and that is where it is important to maintain fidelity in this computation. If so many targets are clustered such that the computation becomes untenable, it is also very likely that the confusion will be close to complete and so a simple heuristic on how to spread confidence throughout the cluster for the various track labels should suffice.

6. When a revisit occurs in which feature information is collected, we recalculate the confidences according to the confusion matrix parameterization of our feature-collecting sensor. This is the only place where confidences have a chance to increase.

7. Finally, in addition to the issue of modeled targets approaching, we include a statistical model of background targets based on density, which also results in additional evanescing of confidence away from a modelled target's label into the other target category for all tracks. The rate of evanescing is dependent on both the density of background targets and the metric uncertainty associated with that particular track's position-velocity (i.e., its covariance).

We modify this approach for SSA in one significant way, which is by eliminating the possibility that the target has stopped or become obscured. We defer discussion of other modifications that could be useful to the Conclusions section.

## 3. SIMULATION AND RESULTS

To test this approach out for SSA, we generated assumed that our targets and sensors are operating according to the dynamics of the Circular Restricted 3-Body problem (CR3BP). We believe that our results are mostly insensitive to the precise details of how we model the gravitational fields impacting objects in the cislunar domain, so the CR3BP provides an efficient framework for generating trajectories and analyzing results.

For generating targets, we largely followed the approach given in two related papers [5][6]. One difference is that we did not use the dimensionless synodic coordinate system for our targets but instead used the direct physical version of these equations:

$$\begin{aligned} \ddot{x} - 2\Omega\dot{y} - \Omega^2 x &= -\frac{\mu_1}{r_1^3}(x + \pi_2 r_{12}) - \frac{\mu_2}{r_2^3}(x + \pi_1 r_{12}) \\ \ddot{y} + 2\Omega\dot{x} - \Omega^2 y &= -\frac{\mu_1}{r_1^3} y - \frac{\mu_2}{r_2^3} y \\ \ddot{z} &= -\frac{\mu_1}{r_1^3} z - \frac{\mu_2}{r_2^3} z \end{aligned} \tag{7}$$

where the Earth has mass $m_1$, the Moon has mass $m_2$, they are separated by distance $r_{12}$, and the target is separated from the Earth by distance $r_1$ and from the Moon by distance $r_2$. Given those values, $\Omega$ is the magnitude of the inertial angular velocity of the Earth-moon reference frame about its barycenter, and $\pi_1$ and $\pi_2$ are convenient dimensionless quantities such that,

$$\pi_1 = \frac{m_1}{m_1 + m_2} \tag{8}$$

$$\pi_2 = \frac{m_2}{m_1 + m_2} \tag{9}$$

For simplicity sake, we also only considered three different orbits, borrowed from initial conditions specified in [5]. These are a Lyapunov-type orbit (Halo orbit), and two periodic but longer orbits labelled Orbit 1 and Orbit 2, respectively. The initial conditions for these orbits, and their resulting periods, can be seen in Table 1 below.

Table 1: Periodic Orbit Initial Conditions

| Parameter | Halo Orbit | Orbit 1 | Orbit 2 |
|---|---|---|---|
| $x_0$ (km) | 2.966368708667607e+05 | 3.885321513506661e+05 | 3.885321113594070e+05 |
| $y_0$ | 0 | -8.301568537662538e-04 | -0.014582211245233 |
| $z_0$ | 0 | -7.569405065509903e-04 | 0 |
| $\dot{x}_0$ (km/sec) | 0 | 0.001999205882491 | 0.003801177220001 |
| $\dot{y}_0$ | 0.514861132148840 | -2.074050417132850 | -2.156898385846957 |
| $\dot{z}_0$ | 0 | -1.187786439911839e-06 | 0 |
| Period (sec) | 1.771e+06 | 4.281e06 | 2.095e6 |

A better understanding of these orbits can be seen in the trajectories plotted for a single period of each orbit below in Fig. 2.

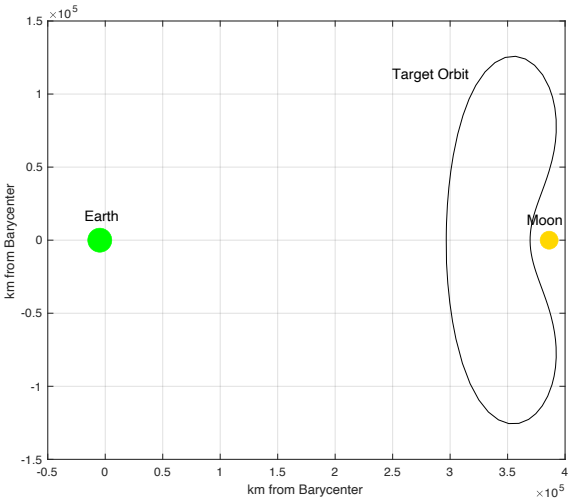
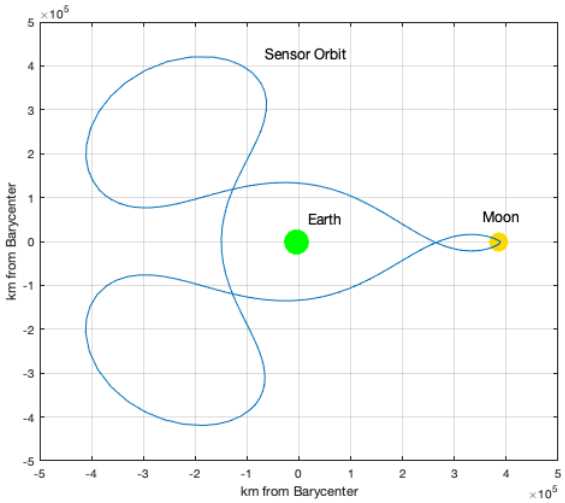



Figure 2a: Halo Orbit (used for Target RSO)




Figures 2b: Periodic Orbit 1 (used for Sensor Satellites)

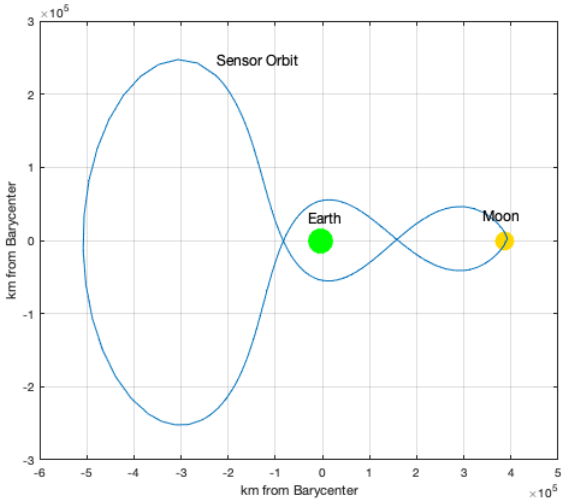



Figure 2c: Periodic Orbit 2 (used for Sensor Satellites)

We set our RSO target in all cases to be the Halo object, and we set the plant noise $q$ for our extended Kalman filter on that object to $4.0\text{x}10^{-2}$ $m^2$/sec (similar to the non-dimensionalized value in [6] when dimensionalized). Our observing sensors are telescopes on satellites that are in the periodic orbits represented in Orbit 1 and Orbit 2. For each run we only looked at having two observing satellites, and assumed both were following the same orbit, either Orbit 1 or Orbit 2, but with different phases. We examined phasing separations of $2.5\text{x}10^4$ sec (approximately 7 hours) and $1\text{x}10^5$ sec (28 hours). We only modeled target exclusions due to earth or lunar angle, and presumed we were in a period where solar angle would not apply, and also ignored any range exclusions for simplicity. We also modeled the effect of background traffic using a very simple unimodal Gaussian density centered around the Earth to see the impact that effect would have.

We also have a feature sensor (nominally a hyperspectral sensor, though its precise type does not matter) which we model with a confusion matrix for the set of targets under consideration. For this exercise we have restricted that set of targets to be just the Halo-orbiting RSO and one designated Other, representing all other possible targets out there. The confusion values utilized are shown in Table 2 below.

Table 2: Confusion Matrix for Feature Sensor

| | |
|---|---|
| 0.88 | 0.12 |
| 0.12 | 0.88 |

Fig. 3 shows the first scenario, with both observing satellites following Orbit 1 and looking at the target satellite following the Halo orbit, and making angular metric obs on its position.

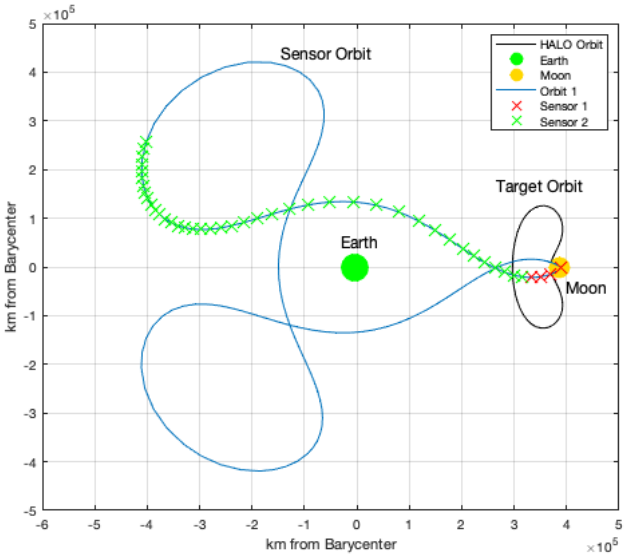
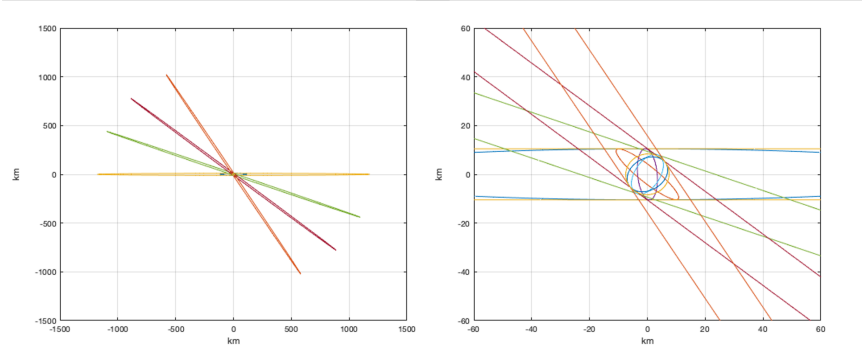


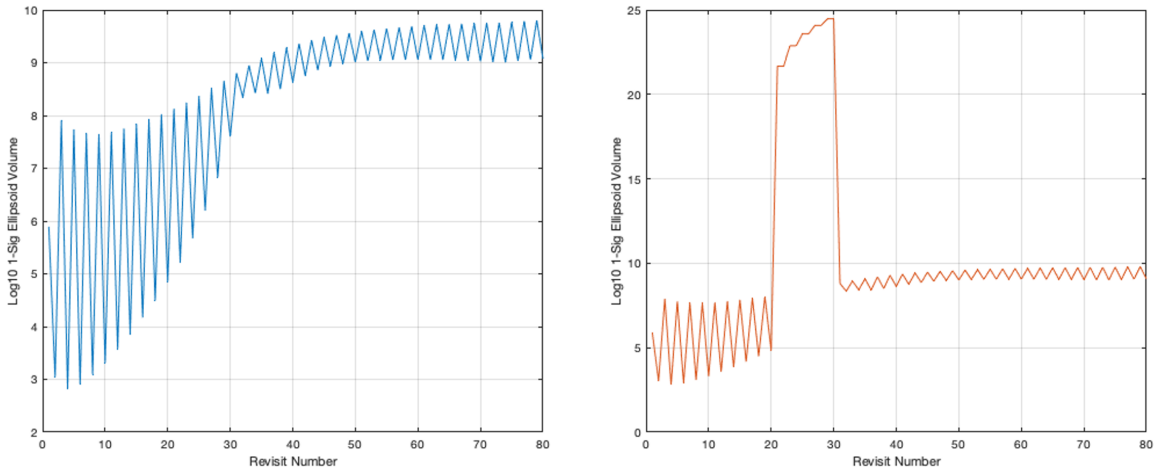

Figure 3: Sensor Satellites Positions in Time and Target Orbit

Both sensors are revisiting the target approximately every 2.5x105 seconds (almost 7 hours) and the resulting covariance ellipses for the first 10 revisits can be seen in Fig.'s 4a and 4b below.




Figures 4a and 4b. Covariance Ellipses and Zoomed-In View of Covariance Ellipses

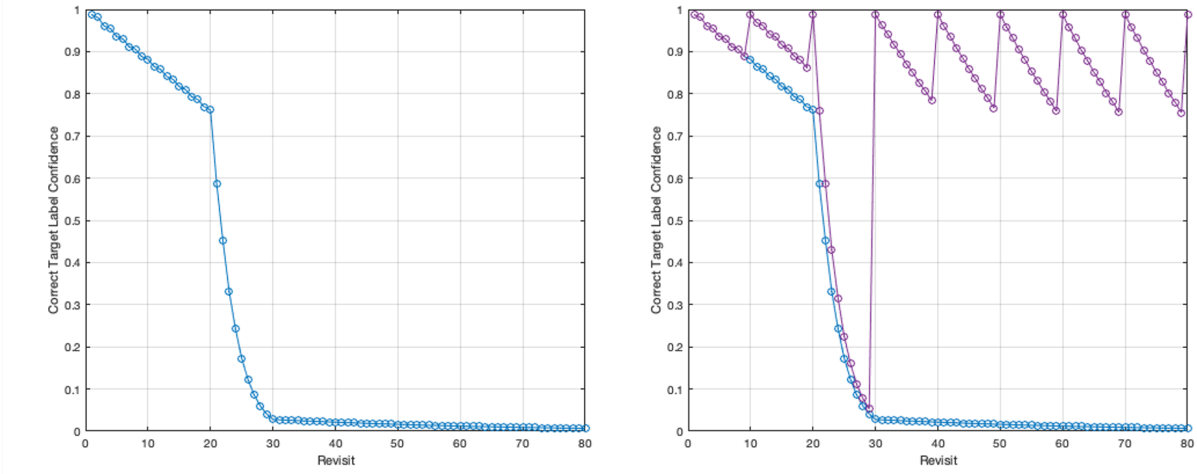
What is happening here is we have chosen to have both satellites view the target at almost the same time on each revisit. The target is localized to within 10km right after a pair of observations, but over 7 hours its uncertainty grows considerably due to the plant noise value. The first angular observation can only reduce the covariance significantly along the line of sight, but the next observation, which is typically coming at a slightly different angle, results in a significant covariance reduction down close to the initial 10 km. Of course, this is occurring early in the scenario when the sensors are relatively close to the Halo orbit. If we view the volume in the 1-sigma ellipse and how it changes over time for the full scenario, we see that the it exhibits this sawtooth behavior indicated by the figures above but also increases over time as the sensors move away from the target vicinity. This can be seen in Fig. 5 below




Figures 5a and 5b. Covariance 1-Sigma Volume Without Earth Exclusion (on left) and With Earth Exclusion (on right)

The figure on the left shows a of the covariance volume on a log scale, which gets larger by several orders of magnitude on average as the sensors move away from the target and as the look angle they are subtending diminishes for the most part. When we include the impact of Earth exclusion, which occurs near the early middle part of the scenario, we get significant dropouts in coverage and a very large growth in uncertainty until the exclusion period is over.

Putting this all together with the effect of background traffic, we can compute a confidence history for the track on the target in the Halo orbit. This is shown in Fig.'s 6a and 6b below, where we show two label confidence histories—one with kinematic (angular) obs only, and one augmented by the nominal hyperspectral feature sensor (which we model as being on the Earth and capable of viewing the target only during certain periods). In this case, we imagine the feature sensor revisits once every 3.3 days, and when it executes a revisit it takes two looks during the window it can look at it, to maximize its collection value.

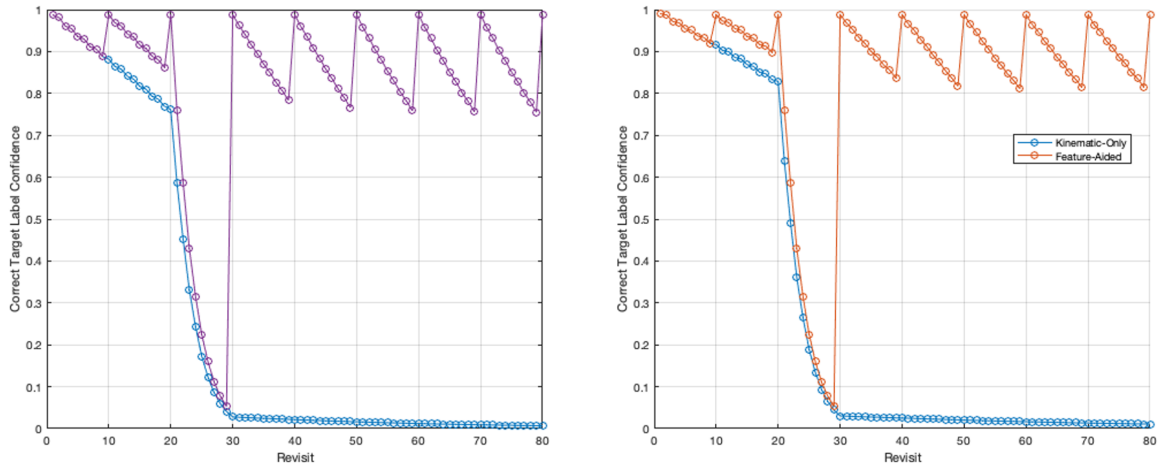



Figures 6a and 6b. Correct Label Confidence Using Angular Obs Only (left) and Correct Label Confidence (purple) Using Feature Obs Every 3 Days

There are some facets worth noting here. While the correct label confidence for the track on the RSO is dropping prior to revisit 20, once the Earth exclusion begins it drops very quickly towards zero as the covariance increases rapidly, as does the likelihood that it will become swapped with some other target. Note that the confidence, even after the exclusion period is over continues to drop, as there is no mechanism for confidence recovery with

kinematic only measurements. On the right, however, we can see the significant impact of making the pair of feature measurements every 3.3 days.

An interesting question is what happens if we increase the acuity of the sensor from 20 arc-sec (low acuity) to 5 arc-sec (high acuity). That can be seen in Fig.'s 7a and 7b, with Fig. 7a being a repeat of Fig. 6b for purposes of comparison.




Figures 7a and 7b: Confidence of Low Acuity Sensor (left) and Confidence of High Acuity Sensor (right)

The confidence values have improved slightly for the high acuity sensor, especially late in the period just prior to the next update from the feature sensor. However, the improvement is not very large compared to other variations.

In Fig. 8 we look at running both satellites on Orbit 2 with the same phase offset in time.

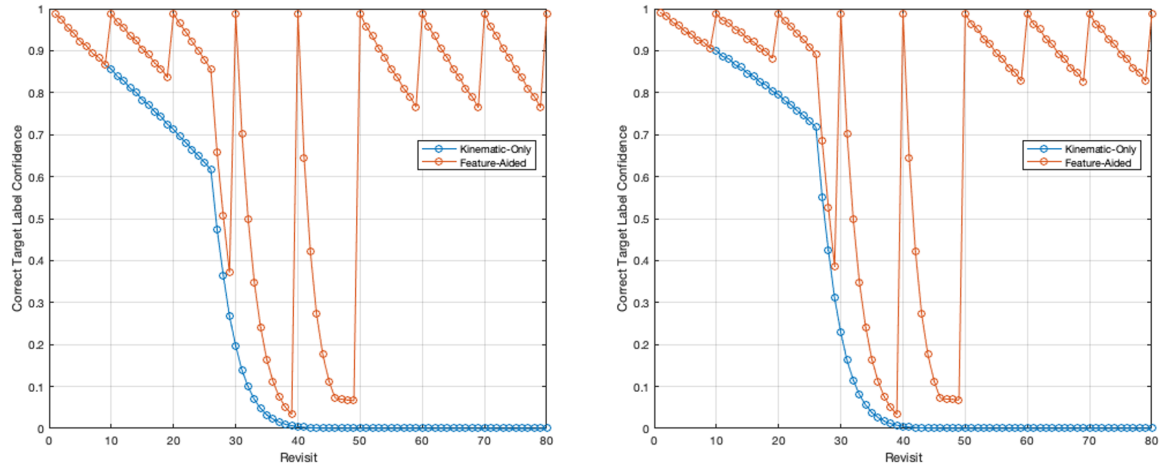



Figure 8: Sensors Following Orbit2, Confidence of Low Acuity Sensor (left) and High Acuity Sensor (right)

The primary difference we can notice is that both sensors experience a longer period of Earth exclusion (approximately revisits 28-48), with a concomitant faster drop in confidence during that time.

In Fig. 9 we have run the same two scenarios as above except the phasing between the two sensor satellites has been increased to $4x10^5$ sec, or 4.5 days, to see the impact of greater separation between them.

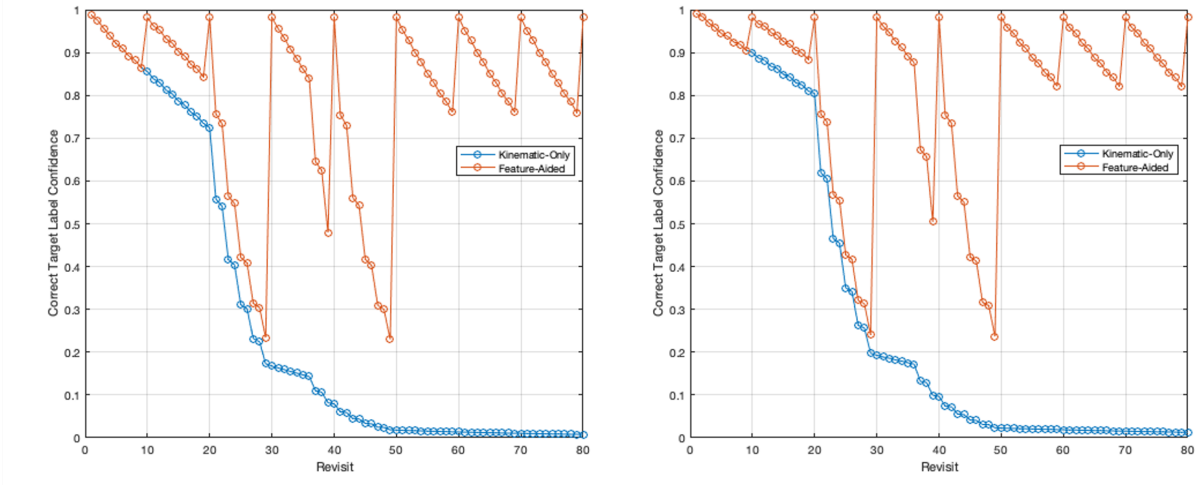



Figure 9: 4.5 Day Phase Offset, Confidence of Low Acuity Sensor (left) and Confidence of High Acuity Sensor (right)

The longer offset means that the exclusion times for the satellites are occurring during completely disjoint intervals (approximately revisits 20-30 and revisits 36-50). During those periods the track is only being updated with a single angles-only sensor, and the larger covariance that results can be seen in the more rapid drop in confidence than during periods (i.e., outside those intervals) when neither sensor has an exclusion.

In Fig. 10 we examine the impact of having only a single collection from the feature sensor at each revisit rather than two. In this case, both satellites are now following Orbit 1 with a phase offset of $1x10^5$ sec (as in Fig. 7).

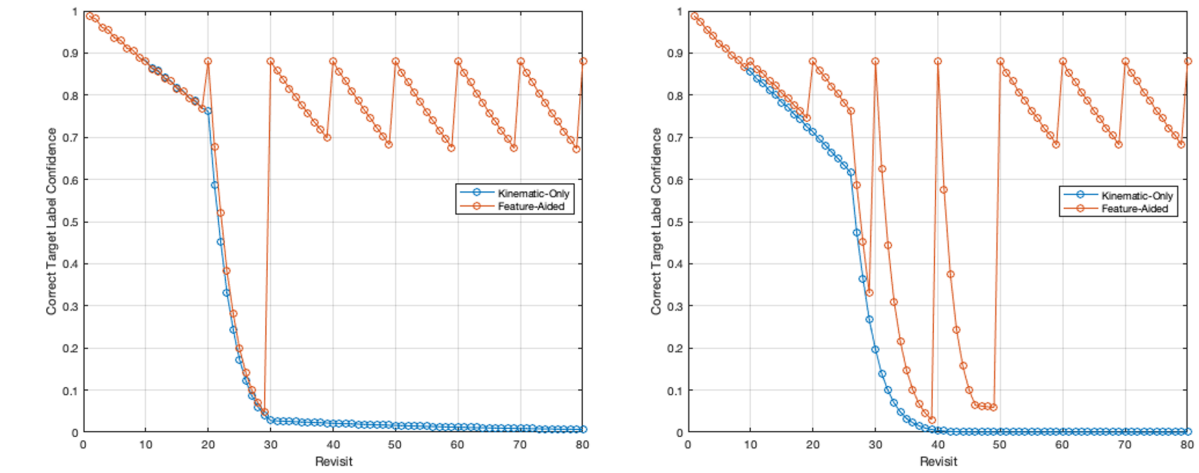



Figure 10: Single Feature Collection per Revisit, Confidence of Low Acuity Sensor (left) and High Acuity Sensor (right)

Note that the behavior is similar but our confidence is not restored to as high a value as from two collections per revisit in Fig. 7.

## 4. CONCLUSIONS

Given the many uncertainties in the modelling of an actual collection methodology, we have not yet explored the space of performance more fully. Rather, the primary intention of this work has been to highlight the concept of label confidence and note how it is a significant, perhaps the most significant, component of performance when considering custody. As such, the focus in the results section is on highlighting what parameter changes lead to the most notable differences in performance to use as a guide for further investigations. The most obvious result should be the dramatic difference between kinematic-only and feature-aided confidence. For the first, the confidence is

always decreasing, and any long period of sensor outage drives it down towards zero rapidly, while for the second, confidence recovers significantly whenever feature data is collected.

Looking at other parameter variations, the next largest set of differences between runs are due to differences in exclusions, which likely comes as no surprise. It is useful to note the boost in confidence given by performing two collections per revisit in the feature space, highlighting the value in collecting more feature data rather than less. It is also useful to note that, at this level of analysis, little substantive difference in performance is seen in label confidence when using the high-acuity sensor (5 arc-sec) vs using the lower-acuity sensor (20 arc-sec).

This last point deserves further analysis and consideration. One weakness of the modelling done in this analysis is that it assumes the feature sensor is able to perform a collection on the target RSO once every 3.3 days. That assumption may well become false as the uncertainty in the location of that RSO grows due to the likely more limited FOV of a feature sensor, and we would recommend (and intend) to redo this analysis with a more sophisticated FOV model coupled with a search strategy for the target RSO.

An additional caveat on this work should also be added relating to the kinematic constraints on RSOs. One facet of orbital dynamics we have not exploited in our MHT-based analysis is the constraint on the amount of diverting a real RSO could perform due to fuel limitations. Unlike other tracking domains, the total energy of the object is an important component that can be utilized to help distinguish one object from another in Space, and would represent a means of boosting confidence purely within the kinematic realm. Our recommendation and intention is to begin exploring this area as well.

## 5. REFERENCES


[1] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, Classification and Tracking of Moving Ground Vehicles, *Lincoln Laboratory Journal*, Vol 13, No. 2, 2002

[2] S. Mori, K.C. Chang, C.Y. Chong, Performance Prediction of Feature-Aided Track-to-Track Association, *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 50, No. 4, October 2014, pp. 2593-2603

[3] S. O'Neil, Performance Prediction Approach for Feature-Aided Tracking, *MITRE Working Note,* March 20, 2022

[4] B. Pugliese, Association Probabilities with Dissimilar Normal Distributions, *MITRE Working Note,* June 16, 2023

[5] A. P. Wilmer, R. A. Bettinger, and B. D. Little, Preliminary Viability Assessment of Cislunar Periodic Orbits for Space Domain Awareness, *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference,* Maui, HI, 2021.

[6] Joshua M. Block , David H. Curtis, Robert A. Bettinger, and Adam P. Wilmer, Cislunar SDA with Low-Fidelity Sensors and Observer Uncertainty, *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, Maui, HI, 2022.