(Preprint) AAS 25-804

SHOWCASE AND COMPARISON OF THREE METHODS FOR VISUALIZING NEAR-EARTH SATELLITE CONJUNCTION EVENTS

Erick H. White[*] and Luis G. Baars[*]

The field of satellite conjunction assessment, dating back to the early NASA Space Shuttle Program, is a cornerstone of modern space safety with the goal of keeping spacecraft safe from potentially catastrophic collisions, especially those that could produce debris that could lead to further collisions. This paper explores and compares three different methods for visualizing near-Earth satellite conjunction events (ellipsoid, bananoid, and point cloud), including mathematical derivations, analysis of use cases, and comparisons to currently employed methods (conjunction plane plots and wireframe ellipsoids). Parallels are highlighted between these visualization methods and the corresponding probability of collision calculation methods developed by NASA CARA. A discussion is included of future visualization considerations for one-cloud/one-surface views and for non-near-Earth (*e.g.*, cislunar) conjunctions.

I. INTRODUCTION

The field of satellite conjunction assessment, dating back to the early National Aeronautics and Space Administration (NASA) Space Shuttle Program, is a cornerstone of modern space environment awareness and safety, with the goal of keeping spacecraft in orbit safe from potentially catastrophic collisions, especially those which could produce debris that could further harm the orbital environment.[1] A conjunction event is a close approach between two orbiting objects (satellites, debris, *etc.*) with the potential for collision at the time of original identification in the screening. Assessing the risk of such an event is typically done using the probability of collision (P_c) metric, originating with Foster and Estes in 1992 and having since been greatly improved by Alfano, Hall, and others.[2,3,4] However, despite the utility of representing the likelihood of a collision as a single quantitative value, such a metric does little to enhance the understanding of satellite conjunction events, especially in cases where common P_c calculation assumptions are violated. Understanding conjunction events, especially high-risk ones, is critical to making informed mission decisions; additionally, knowing how satellite collision risk is evaluated, especially in edge cases, can help owner/operators (O/O's) better understand the risk to their missions.

A. Common P_c Calculation Assumptions

With the tremendous volume of objects NASA's Conjunction Assessment Risk Analysis (CARA) team has to analyze daily, it is necessary to develop fast algorithms for quickly evaluating satellite encounters and determining whether they merit closer analysis (possibly followed by a risk mitigation maneuver). While this analysis makes use of techniques such as screening volumes to detect potential high-risk events for analysis,

[*] Conjunction Assessment Research Analyst, Omitron Inc., 5450 Tech Center Drive, #400, Colorado Springs, CO, 80919.

such methods are outside the scope of this paper, which will instead focus on analyzing conjunctions that have already been detected.[1]

A common set of assumptions, applicable to approximately 93% of conjunctions with usable data analyzed by CARA, is that of rectilinear motion, negligible relative velocity uncertainty, and time-invariant relative position covariance matrices.[2] These assumptions are used in the 2D-P_c method of estimating collision probabilities (described further below), which is widely applicable due to the extremely short effective duration of the vast majority of events (on the order of fractions of a second).[5] However, there exist cases in which these assumptions are violated, such as those with low relative velocities, long propagation times, or high-eccentricity orbits; in such cases, the 2D-P_c method no longer produces accurate P_c estimates and more computationally intensive methods must be utilized instead.[4] Critically for this analysis, these assumptions also affect the appearance of the conjunction event in regards to the shapes of the point clouds and/or surfaces used to represent it, as discussed later in the paper.

B. The P_c Multistep Algorithm

The P_c Multistep algorithm, explained in detail by Hall *et al.* (2023)[*], derives a method for progressively relaxing layers of simplifying assumptions as required to converge on an accurate P_c.[4] The assumption-by-assumption method of the P_c Multistep algorithm closely follows the visualization use case flow covered in this paper and is therefore described here in some detail. This paper will also follow the same general scheme of assumption removal, starting with the most assumptions (ellipsoid representation) and then removing some ("bananoid" representation) and nearly all (point cloud representation) on an as-needed basis.

The first step of the multistep algorithm uses the 2D-P_c method using the assumptions given above to simplify the problem of P_c calculation to a single numerical integral as derived by Alfano in 2005 (in turn from the original two-dimensional integral derived by Foster and Estes in 1992).[2,3] Assuming the incoming data is of usable quality (true for approximately 93% of conjunctions analyzed by CARA), the multistep algorithm first attempts to use 2D-P_c as it is the most efficient of the estimation methods.[4] The algorithm checks for usage violations for this method (*e.g.*, rectilinear assumption violations) and, if it finds none, returns the P_c from this calculation. If a 2D-P_c method violation is found, the algorithm proceeds on to the 2D-N_c method described by Hall *et al.* (2023), which performs a two-dimensional numerical integration over the unit sphere to calculate P_c values. If a 2D-N_c method violation is found, the algorithm then resorts to the more accurate 3D-N_c method, described by Hall (2021), which requires an additional integration over time.[4,6] According to Hall *et al.* (2023), approximately 5% of encounters with valid data (in the 2021-2022 time period covered) require the use of 2D-N_c and only around 0.3% require the use of 3D-N_c.[4] It is, however, worth noting that some of these are due to conservative estimates applied when evaluating usage violations; in reality, some of the conjunctions that use these semi-analytical methods report values very close to those given by the 2D-P_c method.[4] For encounters which violate assumptions for all three of these semi-analytical methods, the P_c Multistep algorithm resorts to using Monte Carlo methods in one of two forms: Simplified Dynamics Monte Carlo (SDMC), which propagates from the time of closest approach, and Brute Force Monte Carlo (BFMC), which propagates from epoch.[4,7] Under a hundredth of a percent of encounters with valid data require Monte Carlo methods (although they are used optionally for high-interest events where obtaining very accurate results is critical).[4]

C. Overview of Current Visualization Methods and Their Drawbacks

NASA CARA often makes use of the conjunction plane plot for conjunction visualization. This visualization style makes use of the 2D-P_c method assumptions to collapse the three-dimensional space the satellites are traveling in to a two-dimensional plane in which the analysis is performed by marginalizing the distribution in the direction of the relative velocity vector.[1] The primary and secondary object covariance matrices are assumed to represent statistically independent distributions, allowing them to simply be added together; the size of the objects is represented by a combined hard-body radius (HBR), which is the sum of the radii of the spheres entirely circumscribing each object.[1,8] In the event that one of the hard-body radii is uncertain

[*] The interested reader is referred to Hall *et al.* for an in-depth description of the P_c calculation methods, assumptions, and use cases, which lie outside of the scope of this paper but are important in understanding when certain visualizations may be invalid.

(*e.g.*, for a piece of orbital debris with a size characterized by radar observations), an effective HBR can be constructed as described by Hall *et al.* (2023).[4] The primary object is centered at the origin while the secondary is placed along the positive x-axis at a distance equal to the nominal miss distance.[4] The primary is surrounded by a circle with radius equal to the combined hard-body radius; the overlap between this circle and the distribution (represented by an ellipse, a series of ellipses, a Monte Carlo point cloud, or some combination thereof) then illustrates the conjunction.[1,4] Alternate representations (*e.g.*, the one derived by Alfano (2005)) center the secondary at the origin and align the covariance ellipse with the axes; these correspond to a translation and rotation of the axes but are otherwise mathematically equivalent.[3]

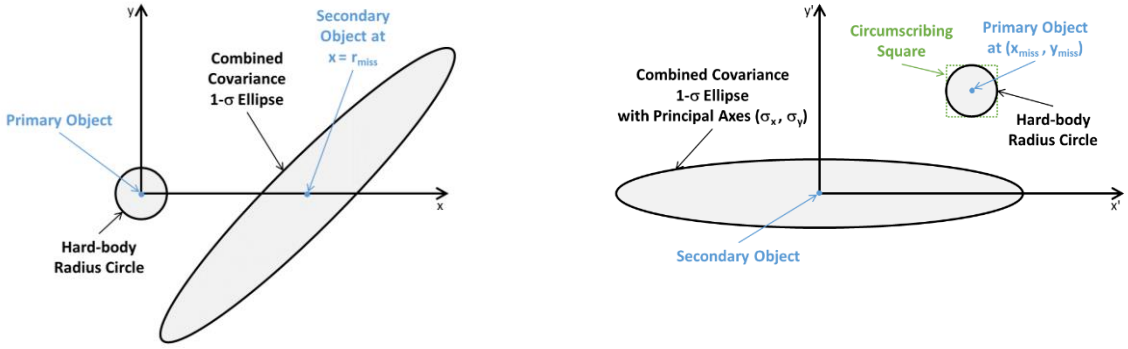

Figure 1. Primary-aligned (left) and distribution-aligned (right) conjunction plane layouts. Figure courtesy of NASA.[1]

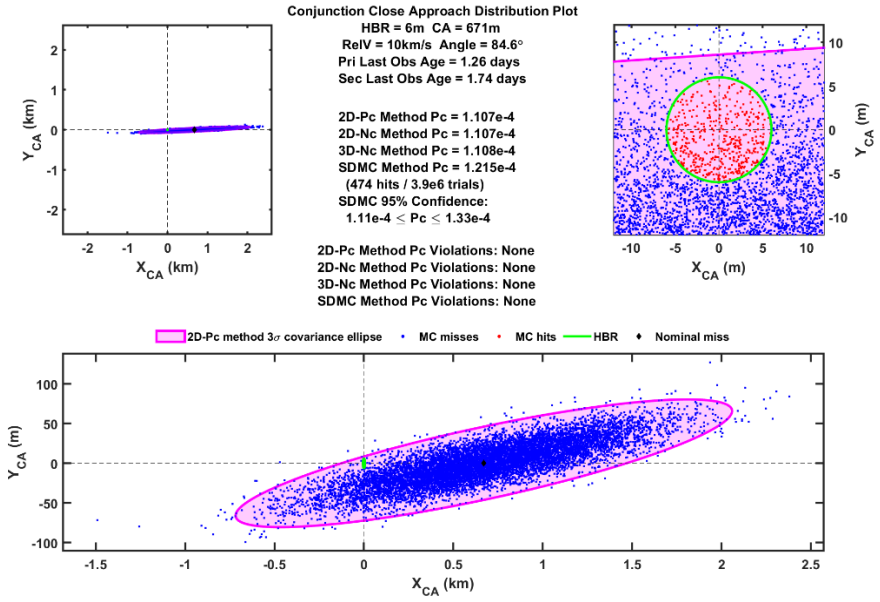

Figure 2. A typical conjunction plane plot of a high-relative-velocity conjunction satisfying the 2D-P_c method assumptions (hereafter "Case A"). Figure courtesy of Hall *et al.*[4]

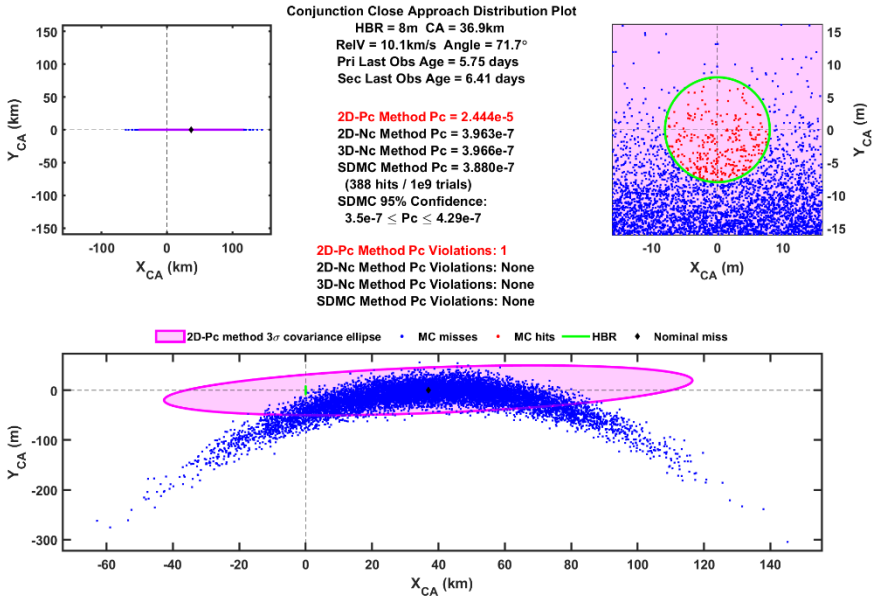

Figure 3. A conjunction plane plot of a high-relative-velocity conjunction violating the 2D-P_c method assumptions (hereafter "Case B"). Figure courtesy of Hall *et al.*[4]

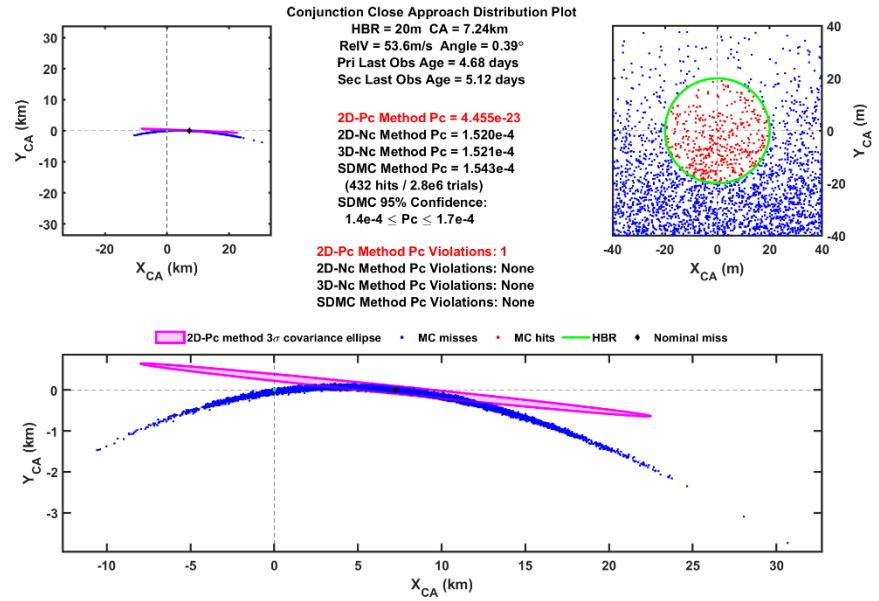

Figure 4. A conjunction plane plot of a low-relative-velocity conjunction violating the 2D-P_c method assumptions (hereafter "Case C"). Figure courtesy of Hall *et al.*[4]

Figure 2 shows a conjunction satisfying the 2D-P_c method assumptions, while Figure 3 and Figure 4 demonstrate violations of the 2D-P_c algorithm. The conjunction shown in Figure 3 has a covariance ellipse that diverges significantly from the actual distribution away from the nominal miss vector (black point), leading to an overestimation of the P_c; Figure 4 illustrates a P_c underestimation for a similar reason.

The conjunctions shown in Figure 2, Figure 3, and Figure 4 will be used consistently throughout this paper as demonstrations of three distinct conjunction geometries to test each visualization model.

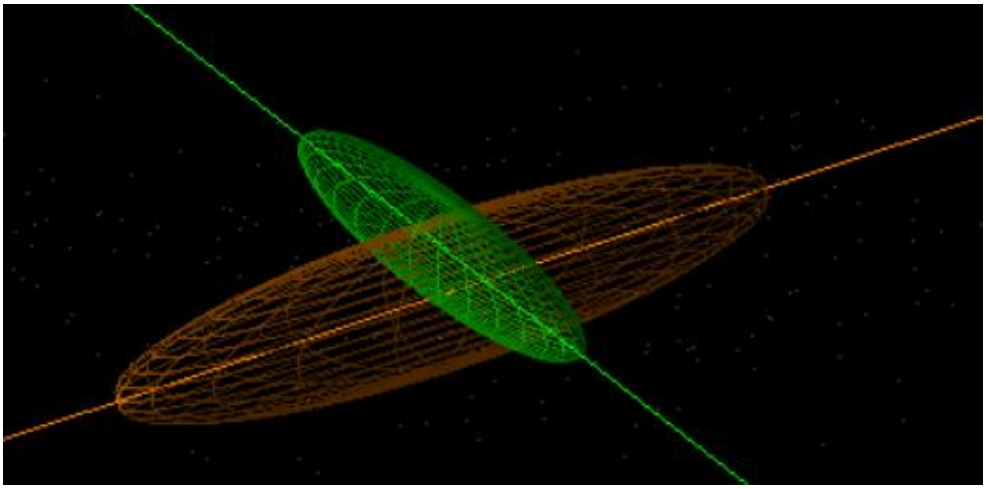

Figure 5. Three-dimensional wireframe covariance ellipsoid visualization method.

Another visualization method uses three-dimensional wireframe representations of the primary and secondary covariance ellipsoids, such as in Figure 5 above. In contrast to the conjunction plane method, in which the covariances are summed into a joint covariance and the hard-body radii summed into a combined hard-body radius, this method preserves the structure of both statistical distributions (that of the primary object and that of the secondary object).

This distinction is critical in analyzing conjunctions visually; while the conjunction plane method better represents the mathematical representation used to calculate P_c values semi-analytically, it is not necessarily an intuitive representation of the three-dimensional interaction. The two-surface representation is often better for seeing the geometry of a conjunction, especially in cases where the surface representing one distribution is entirely contained within the other.

The wireframe representation, while useful intuitively, can be difficult to render and visualize in cases in which a covariance ellipsoid is extremely elongated in one direction; in such situations, it may be difficult to discern regions of interest and importance during the event.

D. Note Regarding Conjunction Visualization Plots

In all subsequent conjunction visualizations, the primary is shown centered in a frame in the upper left, the secondary is shown centered at upper right, the lower right corner shows the estimated statistically expected collision rate over time, and the lower left corner shows information about the conjunction itself, including P_c approximations with various algorithms, combined HBR, and peak rate time relative to TCA (time of closest approach).[6] This layout illustrates both the primary and secondary objects; however, doing so often requires significantly different axis scaling across the plots, particularly in cases where the curvilinear nature of a distribution is especially important. In some cases, this leads to the primary and/or secondary looking rather different between the two views; this is an artifact of the unequal axis scaling, not a flaw with the visualization itself. If desired, these visualizations could be rendered using equal-axis scaling, which is effective for many conjunctions; however, for conjunctions with probability density functions (PDFs) that

are greatly extended in one direction (such as one those shown in Figure 3 and Figure 4), this often results in a visualization that fails to properly illustrate the regions of intersection.

Figure 2, Figure 3, and Figure 4 use unequal axis scaling in the bottom plots, while the upper-left plots use equal-axis scaling; note the difference in the visibility of detail and shape for these plots. Since the approach used in this study considers a two-surface/two-cloud representation, there is no equivalent to the zoomed-in HBR plot shown in the upper right of the CARA conjunction planes; individual zoomed views are used for the same purpose instead. Notably, unequal axis scaling may lead to a somewhat distorted representation of a conjunction by dramatically enhancing curvilinear effects. While this is certainly a drawback of this style of visualization, emphasizing such curvature effects is a focus of this study and so this method will be used throughout.

In all following plots, the yellow surface/cloud represents the PDF of the uncertainty distribution of the primary object, while the magenta surface/cloud represents the PDF of secondary object. The corresponding mean orbits are shown as bold lines of the same colors.

II. METHOD 1: ELLIPSOID REPRESENTATION

The ellipsoid representation, as shown by Alfano and Oltrogge (2018), is both the most computationally straightforward method and the method utilizing the strongest assumptions. The ellipsoid method represents uncertainty distributions as shells (solid or wireframe) corresponding to a desired enclosed percentage of the satellite's PDF (or, equivalently, a surface representing n standard deviations along each axis of the distribution, here referred to as an n-σ surface). In contrast to the wireframes used by Alfano and Oltrogge (2018), this paper focuses on semitransparent ellipsoidal shells, which better illustrate the overlap of the probability distributions and the intersections of the surfaces.

The ellipsoid method provides accurate visualizations for conjunctions in which the 2D-P_c method provides an accurate P_c estimate with no usage violations. Using Cartesian covariance matrices to generate the ellipsoids implicitly assumes that the distributions of the satellites' state vectors are Gaussian in Cartesian space; while this is often nearly true for times sufficiently close to epoch, this approximation becomes invalid with increasing propagation time. This non-Gaussian nature is not negligible for low relative velocity conjunctions and for some high-velocity conjunctions where TCA and the time of peak PDF overlap are sufficiently temporally separated. Sabol *et al*. (2010) further discusses the Gaussian approximation for orbital state PDFs in Cartesian space.[9]

A. Method Derivation

Any determination of a tracked satellite's orbital state will have an associated uncertainty; this can be encapsulated in a mean state vector and covariance matrix, which can then be used to generate a multivariate PDF describing the likelihood of the satellite having a certain state not necessarily equal to the mean state. In Cartesian (rectangular) coordinates, the mean state vector is given by a vector $\bar{\boldsymbol{X}} \in \mathbb{R}^6$, representing the three position components followed by the three velocity components as a combined column vector, while the associated covariance is given by a matrix $\bar{\mathbf{P}} \in \mathbb{M}_6(\mathbb{R})$. The satellite's position in three-space can be represented using a marginalized distribution in $\mathbb{R}^3$; denote by $\bar{\boldsymbol{r}}$ the positional components of $\bar{\boldsymbol{X}}$ and denote by $\bar{\mathbf{A}}$ the position-position components of $\bar{\mathbf{P}}$. The associated three-dimensional normal probability density function, in terms of some arbitrary Cartesian position vector $\boldsymbol{r}$, is[*10]

$$\rho(\boldsymbol{r};\bar{\boldsymbol{r}},\bar{\mathbf{A}}) = \frac{e^{-\frac{(\boldsymbol{r}-\bar{\boldsymbol{r}})^{\mathrm{T}}\bar{\mathbf{A}}^{-1}(\boldsymbol{r}-\bar{\boldsymbol{r}})}{2}}}{\sqrt{|2\pi\bar{\mathbf{A}}|}} = \frac{e^{-\frac{\mathcal{M}(\boldsymbol{r};\bar{\boldsymbol{r}},\bar{\mathbf{A}})}{2}}}{\sqrt{|2\pi\bar{\mathbf{A}}|}} \tag{1}$$

with $\mathcal{M}(\boldsymbol{r};\bar{\boldsymbol{r}},\bar{\mathbf{A}})$ denoting the squared positional Mahalobis distance. This density function can be represented visually as a shell (here called $\partial\Omega$) such that all points in $\partial\Omega$ have a constant squared Mahalanobis

[*] See Notation at the end of this paper for an overview of the mathematical symbols and conventions used throughout.

distance C from the center of the distribution defined by $\bar{\boldsymbol{r}}$ and $\bar{\mathbf{A}}$. Integrating f over Ω (the volume enclosed by $\partial\Omega$) yields a desired fraction p of the total cumulative distribution:

$$\partial\Omega = \{\boldsymbol{r} \in \mathbb{R}^3 \mid \mathcal{M}(\boldsymbol{r}; \bar{\boldsymbol{r}}, \bar{\mathbf{A}}) = C\} \tag{2}$$

$$\iiint_{\Omega} \rho(\boldsymbol{r}; \bar{\boldsymbol{r}}, \bar{\mathbf{A}})\, \mathrm{d}V = p \tag{3}$$

The positive definite covariance matrix $\bar{\mathbf{A}}$ can be decomposed using an eigenvector decomposition into $\bar{\mathbf{A}} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathrm{T}}$, where $\mathbf{Q}$ is the orthogonal matrix of the eigenvectors of $\bar{\mathbf{A}}$ and $\boldsymbol{\Lambda}$ is the diagonal matrix consisting of the (all positive) eigenvalues of $\bar{\mathbf{A}}$.[10] The axes of the ellipsoid are proportional to the inverse squares of the eigenvalues, denoted by $\sqrt{\boldsymbol{\Lambda}^{-1}}$, with some scaling factor along each axis dependent on p (the desired fraction of the PDF enveloped).[10]

To generate the covariance ellipsoid numerically, a gridded ellipsoid centered at the origin with axes aligned with the coordinate frame axes and scaled according to $\sqrt{\boldsymbol{\Lambda}^{-1}}$ is generated as the base shell. This shell is then rotated out of the eigenframe componentwise, with the vector representing each constituent point left-multiplied by $\mathbf{Q}$; since $\mathbf{Q}$ is orthogonal, this multiplication does not introduce any scaling factor ($|\mathbf{Q}| = 1$ or $|\mathbf{Q}| = -1$, and in the latter case $\mathbf{Q}$ can be made special orthogonal by multiplying one of its columns by -1) and the ellipsoid is rotated into the proper orientation.[11] The mean vector $\bar{\boldsymbol{r}}$ is then added, once again componentwise, to the rotated ellipsoid to recenter it on the mean positional state. The final ellipsoid $\partial\Omega$ can then also be written as

$$\partial\Omega = \{\boldsymbol{r} \in \mathbb{R}^3 \mid (\boldsymbol{r} - \bar{\boldsymbol{r}})^{\mathrm{T}} \bar{\mathbf{A}}^{-1} (\boldsymbol{r} - \bar{\boldsymbol{r}}) = n^2\} \tag{4}$$

with $p = \chi_3^2(n^2)$; that is, n^2 is the value of the inverse chi-square distribution with three degrees of freedom evaluated at p.[10] Note also that, in this manner, an n-σ surface can be generated (as opposed to one containing a fraction p of the evaluated PDF) by specifying n directly instead of calculating it from p.

B. Examples and Analysis

The event illustrated in Figure 6 is an example of a typical conjunction analyzed by CARA. With a relative velocity of 10 km/s and a delay between TCA (when covariance is provided) and the time of peak collision rate (when the figure is generated, at the time of minimum Mahalanobis distance between the two distributions) of approximately 0.06 seconds, the relative velocity and both covariances can be assumed to be constant for the brief duration of the encounter.[1,4] In this particular event, the three-sigma surfaces only partially intersect each other, leading to a saddle-shaped contour (visible as a dark purple line).

The conjunction event in Figure 7 shows a strong failure case of the ellipsoid representation. The extreme elongation of the primary covariance in the in-track direction leads to a PDF that does not follow the shape of the orbit; the non-Gaussian nature of the PDF in Cartesian coordinates becomes evident in such cases and thus this representation is not useful for this particular conjunction.[9] The 2D-P_c method significantly overestimates the more accurate 2D-N_c and 3D-N_c methods (by almost two orders of magnitude); this overlap is not evident in the figure since the ellipsoids are propagated along curvilinear trajectories, but would be were they propagated along rectilinear ones (as is done by 2D-P_c). The apparent overlap in the top right panel of this figure is a result of one ellipsoid being entirely in front of the other; this becomes apparent if the viewing angle is changed, but results in the relative angle being harder to distinguish.

Figure 8 shows another failure case of the ellipsoid representation. In this instance, the peak PDF overlap time occurs relatively long (nearly 47 seconds) after TCA; along with the low relative velocity of the encounter, this means that covariance curvilinear effects already become evident in this case and using the Cartesian covariance produces an incorrect final result. The 2D-P_c method underestimates the probability

of collision by nearly 19 orders of magnitude, meaning that if the ellipsoids were propagated along rectilinear trajectories (instead of the curvilinear ones used in figure generation), there would likely be no overlap at all.

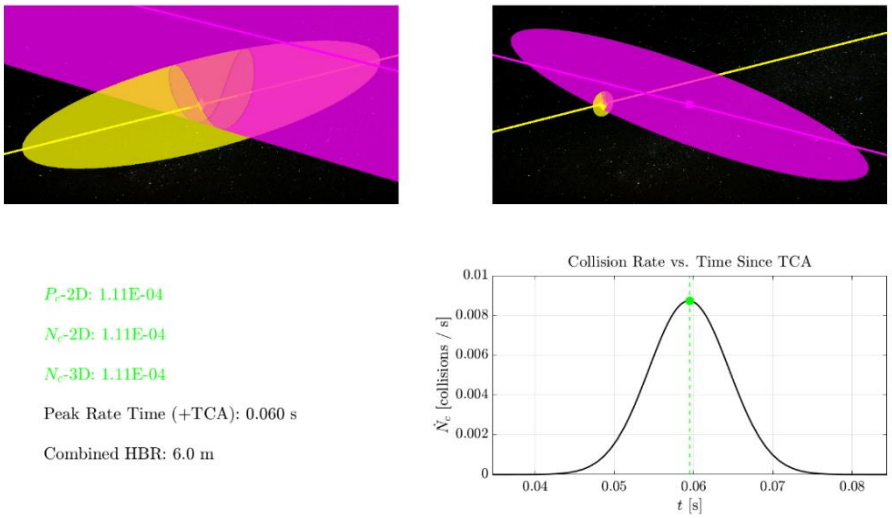

Figure 6. A three-sigma ellipsoid representation of Case A. The mean states of the primary and secondary are represented with a diamond and a square respectively.

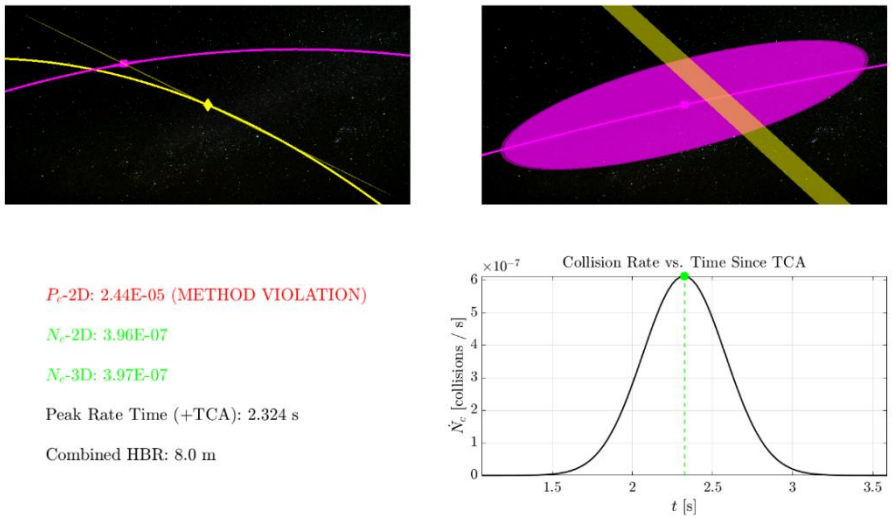

Figure 7. A three-sigma ellipsoid representation of Case B. The mean states of the primary and secondary are represented with a diamond and a square respectively.

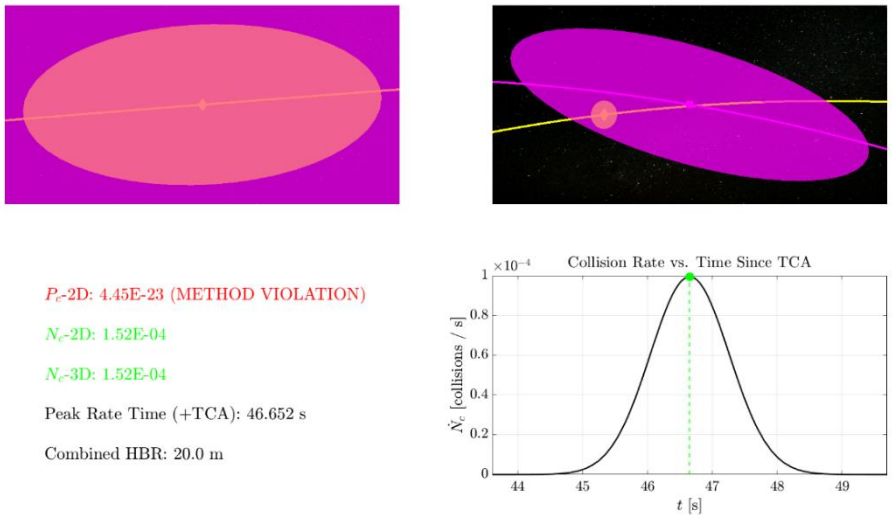

Figure 8. A three-sigma ellipsoid representation of Case C. The mean states of the primary and secondary are represented with a diamond and a square respectively.

C. Advantages and Disadvantages of Ellipsoid Representation

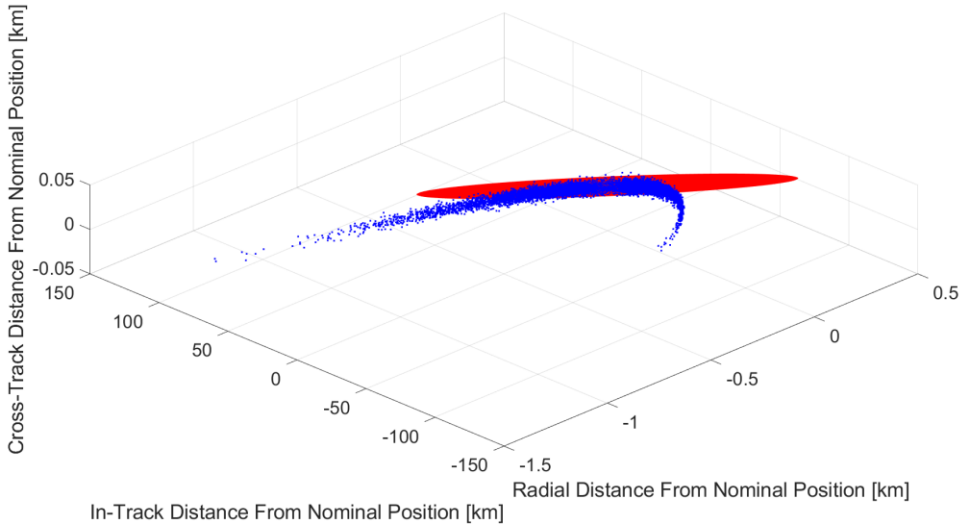

Figure 9. A comparison of a rectilinear covariance ellipsoid (red) and Monte Carlo point cloud (blue) illustrating the effects of PDF curvature. Note the unequal axes scales, particularly in the in-track direction.

The main advantages of the ellipsoid representation are its ease of generation and ease of intuitive understanding. Each ellipsoid can be generated centered at the origin, then transformed into the correct frame with minimal computation. For cases in which a user desires a large number of visualizations or a visualization animation, this computational simplicity can lead to significant speed advantages, especially critical for time-

sensitive decision-making. Additionally, illustrating the overlap of two ellipsoids is relatively simple; Figure 6, Figure 7, and Figure 8 achieve this using transparency effects, but numerical methods such as those shown by McMahon *et al.* (2015) can produce the same effect.[12]

The main disadvantage of the ellipsoid visualization method is its misleading representation of non-rectilinear cases. As illustrated in Figure 7, the axes of a covariance ellipsoid have no curvature; therefore, they are unable to follow the orbit of a target and may lead to a poor representation of the actual overlap of the two uncertainty PDFs. Curvilinear coordinates (such as orbital elements) address this issue, as discussed by Hoots (2001).[13] Figure 9 compares the rectilinear-assumption ellipsoid and a Monte Carlo point cloud sampled from a multivariate normal random distribution in equinoctial space (as discussed later); the ellipsoid diverges significantly from the actual distribution, especially in the in-track direction. In the above example, the three-sigma ellipsoid shell contains only approximately 32% of the Monte Carlo points (as opposed to the 97.07% expected).

The ellipsoid representation also suffers from sampling issues for highly extended shells, for which either a high number of sampling points must be used to prevent low resolution in regions of high curvature or the surface must be sampled dynamically (*e.g.*, according to local curvature), both solutions which remove some of the computational simplicity that make this method attractive in the first place.[14]

III. METHOD 2: BANANOID REPRESENTATION

The "bananoid" representation (so called due to its banana-like shape in extended cases) was recently implemented by the CARA team and based on methods formulated by Hall (2025a).[15] As with the ellipsoid method, the bananoid method represents both the primary and secondary object PDFs as shells or wireframes, illustrating the relationship between both in three-dimensional space as well as the intersection of the two n-σ surfaces. Whereas the ellipsoid method represents the PDFs of the primary and secondary objects using surfaces derived from their Cartesian covariance matrices, the bananoid method instead derives its surfaces from their equinoctial element covariance matrices.

The bananoid representation is most representative of cases that use 2D-N_c or 3D-N_c calculation methods. The method removes the implicit rectilinear assumption present in the ellipsoid by determining the surface in equinoctial space instead of Cartesian space; this uses the equinoctial orbital elements as described by Hernando-Ayuso *et al.* (2023).[16] Poore *et al.* (2016) discusses the effects of covariance shape with particular regards to P_c calculation.[17] The utility of equinoctial coordinates manifests itself in two separate cases.

The first use case for the bananoid representation of an n-σ surface occurs in cases of highly extended PDFs. In these cases, the distribution of the positional components lies along the curvature of the orbit rather than centered on the tangent plane to the orbit; such a distribution is not normal in Cartesian space and is therefore impossible to achieve using ellipsoids alone.[9] Hoots (2001) demonstrates that orbital state distributions preserve Gaussianity provided that they are expressed in a coordinate system in which the equations of motion are linear; while Hoots (2001) uses Keplerian elements with mean motion and mean anomaly, the use of alternate equinoctial elements, as discussed by Hernando-Ayuso *et al.* (2023), is also permissible and results in fewer singularities.[13,16]

The other main use case for the bananoid representation is conjunctions in which the time of peak PDF overlap occurs relatively far away from TCA, therefore requiring covariance propagation. Hall (2025b) discusses the use of equinoctial state transition matrices (STMs) in propagating error while preserving covariance realism; remaining in the equinoctial space, therefore, allows for the creation of realistic surfaces depicting satellite position PDFs.[18]

A. Method Derivation

Recall the previous definition of a satellite's nominal state and covariance matrix in Cartesian space as a vector $\bar{\boldsymbol{X}} \in \mathbb{R}^6$ and matrix $\bar{\mathbf{P}} \in \mathbb{M}_6(\mathbb{R})$. Similarly to the ellipsoid representation, the bananoid representation is based off a user-determined fraction of the PDF to be enveloped by the surface; unlike the ellipsoid representation, however, the associated PDF is not assumed to be normal in Cartesian space.[13]

Based on the user's selected fraction of the PDF to be enveloped, an associated Mahalanobis distance can be determined which is then used to calculate the boundary of the surface. The surface itself is determined numerically as a series of curves of equal Mahalanobis distance spaced along a central curve of minimum Mahalanobis distance which intersects the nominal trajectory at the mean point $\bar{\boldsymbol{r}}$. The surface is defined using two parameters, τ and ϕ, where τ varies along the direction of the orbit (*i.e.*, as a latitude-like parameter) and ϕ is an azimuthal angle (*i.e.*, a longitude-like parameter).[15]

The boundaries of τ are determined using an iterative process (similar to that described by Hall (2021)) and a concept here referred to as a τ-plane.[6] The total number of τ-planes is a user-controlled parameter in the generation of the surface; a higher number of τ-planes leads to a higher-resolution grid. Each τ-plane is perpendicular to the local nominal velocity vector $\widehat{\boldsymbol{V}}$ and is therefore equivalent to the N-B plane in the body-centered VNB (or NTW) frame.[15,19] For each value of τ (measured in seconds relative to the time of interest), the nominal Cartesian position state vector $\bar{\boldsymbol{X}}_\tau$ is calculated by propagating from the mean equinoctial state $\bar{\boldsymbol{E}}$. The vector $\tilde{\boldsymbol{X}} = \bar{\boldsymbol{X}}_\tau$ is selected as the first center of linearization for a Taylor expansion to obtain the point of minimum Mahalanobis distance in the τ-plane. For each iteration, a covariance matrix $\tilde{\mathbf{P}}$ associated with the state $\tilde{\boldsymbol{X}}$ is calculated as[6,15]

$$\tilde{\mathbf{P}} = \tilde{\boldsymbol{\psi}}\bar{\boldsymbol{\mathcal{P}}}\tilde{\boldsymbol{\psi}}^{\mathrm{T}} \tag{6}$$

$$\tilde{\boldsymbol{\psi}} = \frac{\partial \tilde{\boldsymbol{X}}}{\partial \tilde{\boldsymbol{E}}} \tag{7}$$

where $\bar{\boldsymbol{\mathcal{P}}}$ is the equinoctial covariance matrix of the mean state at the time of interest and $\tilde{\boldsymbol{\psi}}$ is the Jacobian matrix encoding the partial derivatives of the Cartesian and equinoctial state vectors at the center of linearization. An effective mean state is then calculated as[6,15]

$$\check{\boldsymbol{X}} = \tilde{\boldsymbol{X}} + \tilde{\boldsymbol{\psi}}\left(\bar{\boldsymbol{E}} - \tilde{\boldsymbol{E}}\right) \tag{8}$$

This effective mean state is then used, along with a predicted velocity from the equinoctial distribution, to generate a new $\tilde{\boldsymbol{X}}$; this process is iterated until the minimum Mahalanobis distance converges.[6,15] If the minimum Mahalanobis distance in a given τ-plane is greater than n, that τ-plane is beyond the bounds of the surface. Otherwise, the point of minimum Mahalanobis distance $\check{\boldsymbol{X}}$ with associated covariance matrix $\tilde{\mathbf{P}}$ is selected as the center of a second Taylor series expansion, this one to determine all points with a squared Mahalanobis distance equal to n^2.[6,15] This yields a quadratic expression in terms of w, the distance on the τ-plane to the n-σ surface measured from the minimum Mahalanobis distance point, which has the form

$$\mathcal{M} = (\boldsymbol{r} - \check{\boldsymbol{r}})^{\mathrm{T}}\tilde{\mathbf{A}}^{-1}(\boldsymbol{r} - \check{\boldsymbol{r}}) = \mathcal{A}w^2 + \mathcal{B}w + \mathcal{C} = n^2 \tag{9}$$

derived by expressing $\boldsymbol{r}$ in terms of w, ϕ, $\widehat{\boldsymbol{N}}$, and $\widehat{\boldsymbol{B}}$. The coefficients $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ then depend on $\check{\boldsymbol{r}}$, the positional component of $\check{\boldsymbol{X}}$, and $\tilde{\mathbf{A}}$, the position-position component of $\tilde{\mathbf{P}}$.[15] Iterating over a sufficient number of ϕ points yields a closed, convex curve of points lying on the surface of bananoid.

B. Examples and Analysis

Figure 10 illustrates a conjunction satisfying the rectilinear assumption; such an event is not one that necessitates the bananoid representation for accurate analysis (note the identical appearance of Figure 6) but nonetheless serves as a qualitative verification of its correctness.

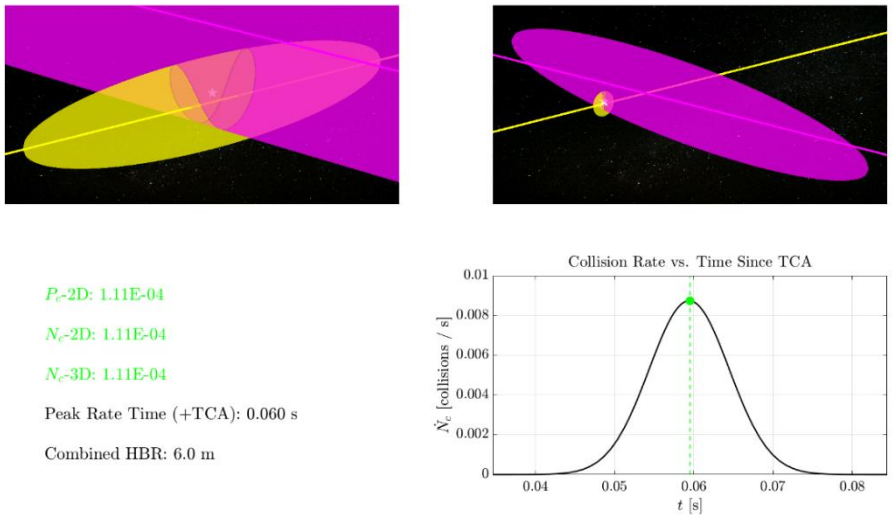

Figure 10. A three-sigma bananoid representation of Case A. The peak overlap point is represented with a white star.

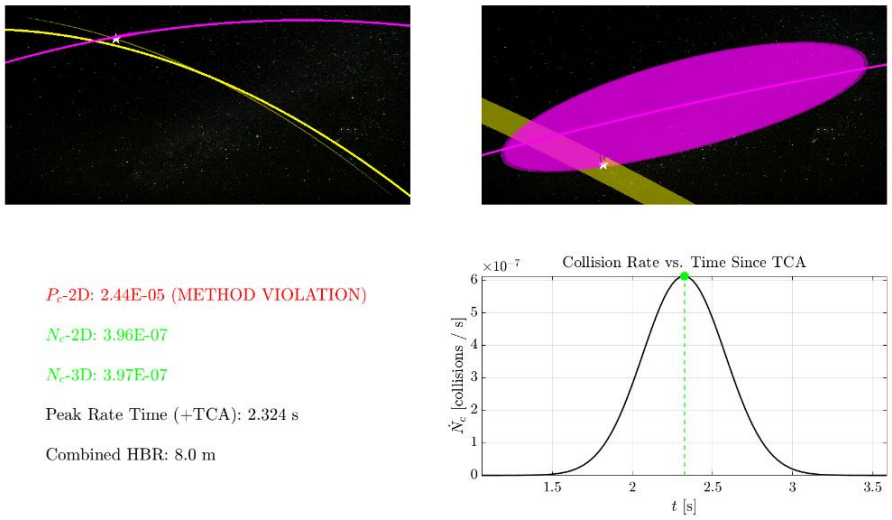

Figure 11. A three-sigma bananoid representation of Case B. The peak overlap point is represented with a white star.

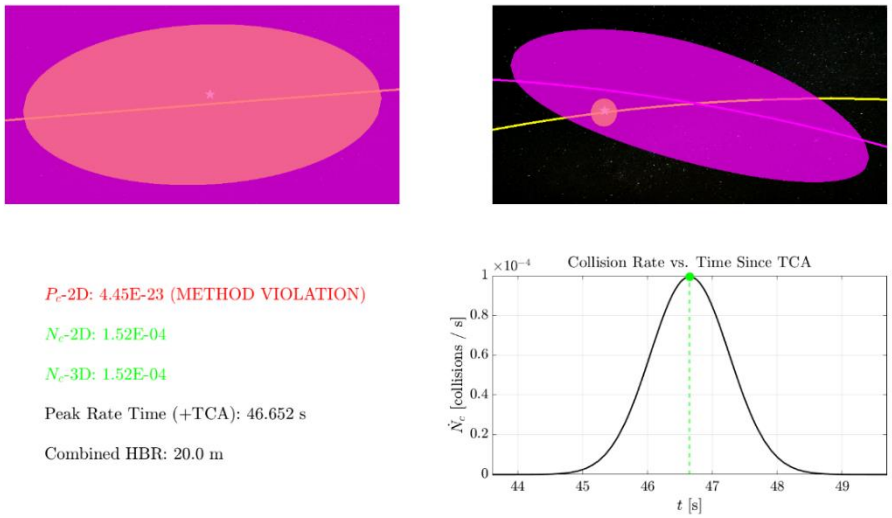

Figure 12. A three-sigma bananoid representation of Case C. The peak overlap point is represented with a white star.

Figure 11 illustrates a particularly important use case for the bananoid covariance representation. Notice how the curvature of the primary surface closely follows the curvature of the primary orbit (solid yellow line), to the point that the two overlap in some areas; this is in contrast to the same conjunction as represented by the ellipsoid method in Figure 7. Being able to visualize curved PDFs, which more closely resemble the reality of satellite state distributions, is the primary advantage of this method over the ellipsoid.[13]

Figure 12 illustrates the use of the bananoid surface in visualizing a conjunction for which the peak overlap time occurs significantly after TCA. Similarly to the contrast between Figure 11 and Figure 7, the bananoid better represents the PDF curvature due to covariance propagation.[18]

C. Advantages and Disadvantages of Bananoid Representation

The greatest benefit of the bananoid representation is its ability to represent a satellite's PDF as an accurate surface or wireframe, allowing for easier visualization of aspects of a conjunction such as intersection regions or fully-overlapping PDFs; the accuracy is lost in the ellipsoid representation and the surface is lost in the point cloud representation. The bananoid is also a good way to represent covariance propagation effects; in cases in which the user opts to generate an animation instead of a still-frame, especially if the animation is over a long time period, the increasing size of the PDF (especially in the in-track direction) can be seen by the corresponding increase in the size of the surface.[13,18]

Unfortunately, the bananoid representation does have several drawbacks. These largely stem from the fact that the bananoid, although familiar as a distribution shape, has only recently been parameterized mathematically by the CARA team; therefore, as the theory is still in development, it has yet to be tested to the extent that ellipsoids and Monte Carlo simulations have been.[13,15] Coppola and Tanygin (2015) derive a method to display a similar surface to the bananoid (referred to as a "bent ellipsoid") using a system called downrange coordinates; however, these do not appear to be used extensively, nor were they a focus of comparison for this study.[20] These coordinates are shown to be effective at modeling non-Gaussian behavior by Sabol *et al*. (2010), but ignore cross-correlational terms between position and velocity by keeping them completely separate (as opposed to combining them as equinoctial elements do); the full effects of this on the shape of the surface are not entirely clear.[9,20]

One flaw of the current theory is that, for highly-extended PDFs, the expected percentage of the distribution enveloped by the surface (as defined by the user) diverges from the actual amount numerically measured

using a comparison of the surface and a Monte Carlo point cloud. It is unclear as of yet how this deviation relates to the derivation of the surface; however, as the surface is based on first-order approximations, it is likely that this is an effect of the surface growing more distant from the center of linearization.[18] Higher-order approximations have not yet been explored.

Another flaw with the bananoid (and the theory behind it) is similar to the issue of improper sampling in the ellipsoid representation, but manifests differently; since the surface itself is determined parametrically in a latitude/longitude fashion, improper sampling can lead to entirely incorrect surfaces, rather than simply low-resolution ones. The current solution uses an algorithm to determine the number of grid lines to allocate based on local curvature of the surface; however, this adds complexity to the representation and thus results in it being slower than the ellipsoid representation.

IV. METHOD 3: POINT CLOUD REPRESENTATION

The point cloud representation of a conjunction is the most representative of the actual probability distributions of both the primary and secondary satellites, with the shapes, relative densities, and extents of both PDFs evident. The point cloud representation uses Monte Carlo sampling in equinoctial coordinates to visualize both PDFs and therefore most closely aligns with the BFMC and SDMC P_c calculation methods; disregarding the two-body motion assumption (which is used when propagating all covariances in this study), the only assumption made by the Monte Carlo point cloud representation is that the distributions of the satellites' orbital states are Gaussian in equinoctial space, an assumption discussed by Sabol *et al.* (2010).[4,7,9]

A. Method Derivation

Consider once again the nominal state vector $\overline{\boldsymbol{X}} \in \mathbb{R}^6$ and covariance matrix $\overline{\mathbf{P}} \in \mathbb{M}_6(\mathbb{R})$. Both of these can be converted into the equinoctial coordinate frame using the equations and Jacobian matrix given by Vallado and Alfano (2015)[21]

$$\overline{\boldsymbol{\mathcal{P}}} = \bar{\mathbf{J}}\,\overline{\mathbf{P}}\,\bar{\mathbf{J}}^{\mathrm{T}} \tag{10}$$

$$\bar{\mathbf{J}} = \frac{\partial \overline{\boldsymbol{E}}}{\partial \overline{\boldsymbol{X}}} \tag{11}$$

where $\overline{\boldsymbol{\mathcal{P}}}$ is the equinoctial covariance matrix, $\overline{\boldsymbol{E}}$ is the equinoctial state, and $\bar{\mathbf{J}}$ is the Jacobian matrix to convert from Cartesian to equinoctial coordinates. Since the distribution of orbital states is assumed to be Gaussian in the equinoctial space, this PDF can then be sampled using Monte Carlo methods; these samples are then converted back into Cartesian space to retrieve the positional components of the state vectors.[7,9]

It is important to note that the Cartesian and equinoctial states are both six-dimensional, yet the final visualization is rendered in three dimensions. The reason for this is that the three velocity dimensions in Cartesian space are marginalized out of the resulting distribution; a full representation of the state distribution would additionally include velocity vectors for each point in the distribution. These are omitted for clarity (and would not be viable in the ellipsoid and bananoid representations, which consider a position distribution only).

Figure 13 illustrates a conjunction in which the Monte Carlo point cloud, despite being accurate, is a poor choice for visualization. Since the clouds differ so greatly in size, the zoomed-in primary view contains few secondary cloud points, while the secondary-centered view fails to illustrate the intersection of the primary and secondary clouds.

Figure 14 demonstrates a better use case for the point cloud visualization. Due to the extended primary PDF, the rectilinear assumption fails and semi-analytical or numerical methods must be used instead; the point cloud method, thanks to its lack of simplifying assumptions, illustrates the true shape of the distribution in Cartesian space (see Figure 9 for a comparison of this conjunction with the ellipsoid method). Both clouds are sampled equally and are clearly visible in both frames, including the region of intersection. Additionally, the relative densities of the clouds at the point of intersection is evident and illustrates the difference in the densities of the PDFs in the intersection region.

B. Examples and Analysis

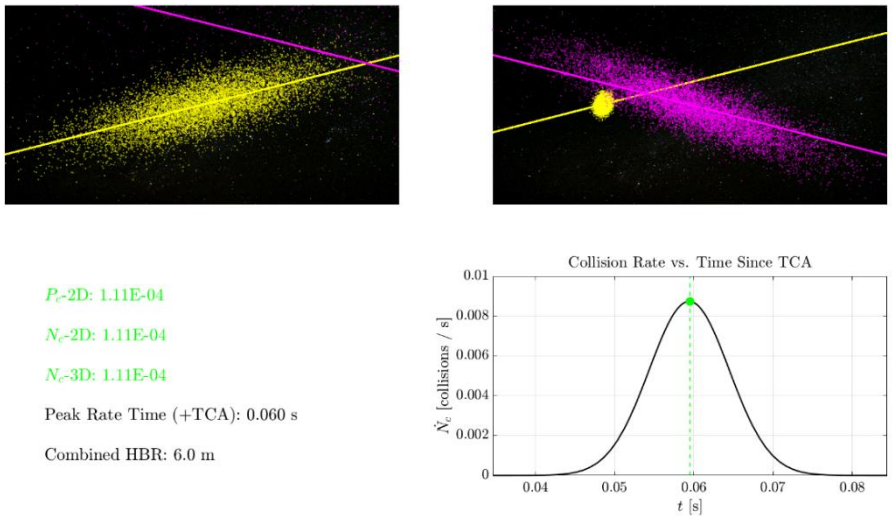

Figure 13. A Monte Carlo point cloud (with 10000 samples per cloud) representation of Case A.

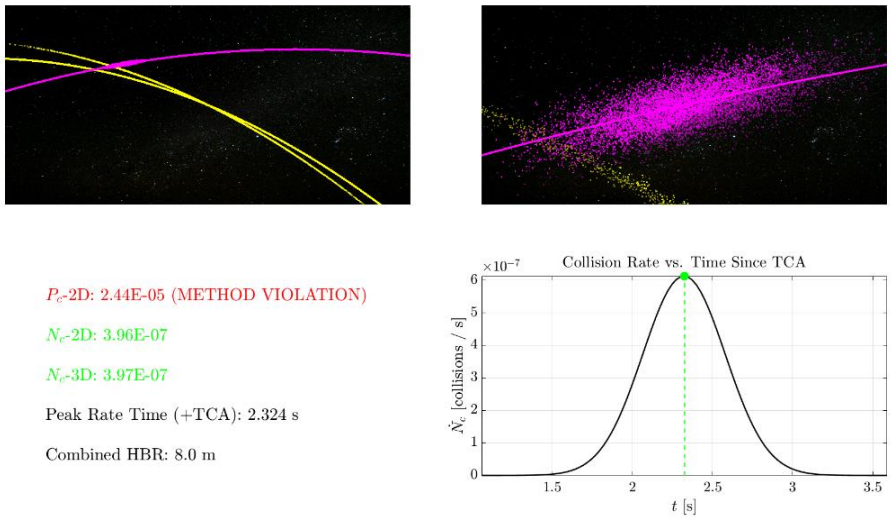

Figure 14. A Monte Carlo point cloud (with 10000 samples per cloud) representation of Case B.

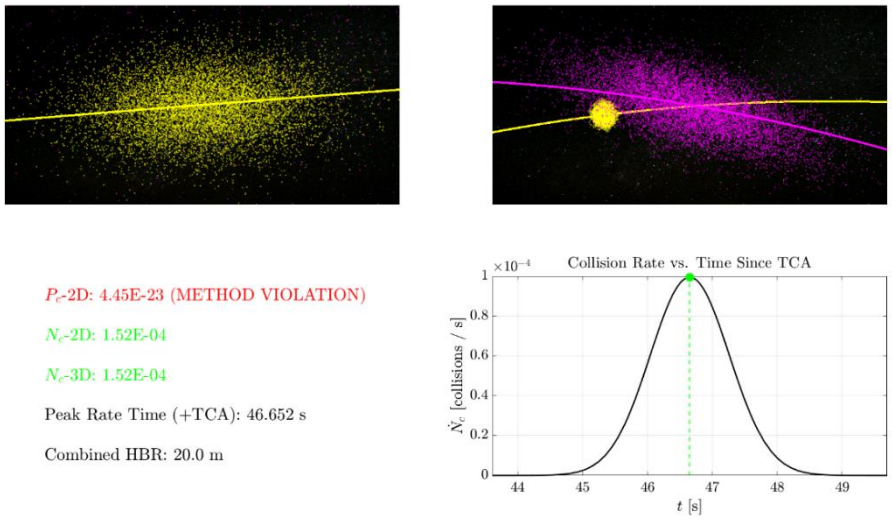

Figure 15. A Monte Carlo point cloud (with 10000 samples per cloud) representation of Case C.

Figure 15 once again shows a suboptimal use case for the point cloud visualization. While it does successfully show the curvature of the secondary PDF in the secondary-centered plot, it once again undersamples the secondary in the primary-centered view. While the number of samples is adjustable, experimenting to produce the correct number can be costly in regards to CPU time, particularly if two clouds barely intersect; the implementation used to generate the figures for this paper allows for different primary and secondary sampling counts, simplifying this process, but it can be costly nonetheless.

C. Advantages and Disadvantages of Point Cloud Representation

The major benefit of the point cloud representation is the lack of simplifying assumptions leading to an accurate visualization of the conjunction environment. Removing the need to approximate an n-σ surface semi-analytically eliminates the somewhat complicated formulation required to generate the bananoid surface, which itself is intended to address the inaccuracies of the ellipsoid surface; this makes the Monte Carlo point cloud representation the most mathematically simple of the three visualizations. This also makes it a benchmark for the other two representations (and any future methods), serving as a sort of "truth" for the appearance of a conjunction event.[9]

Since the point cloud is the most mathematically straightforward of the three methods, it is also easier to verify, experiment with, and modify on a case-by-case basis. The development of more accurate theories and models would not render the point cloud method obsolete, as it has the potential to do to the other methods; instead, a simple alteration of the underlying distribution would once again produce an accurate covariance representation in light of new information.[7]

The point cloud representation does have several major drawbacks that prevent it from being viable for large-scale use. The most prohibitive of these is computational time; while rendering a modest amount of points can be done fairly quickly on modern CPUs, rendering the amount of points necessary to show highly-extended distributions in sufficient detail becomes untenable if a large number of conjunctions need to be processed (or if an animation of an event needs to be generated).[1]

The point cloud representation can also fail to provide a clear visualization of PDF intersections. Although the region of overlap between two clouds may be apparent (depending on the local densities of the clouds),

the exact boundary between two point clouds is rarely clear, especially for cases in which the two local cloud densities differ.

Sample density can also be an issue, as dense sampling results in higher computational costs; this is particularly apparent in extended PDFs, where a zoomed-in view of a given region (*e.g.*, with a focus on a portion of the cloud that may be intersecting another PDF) may have so few points as to not present a comprehensive picture of the situation. Dynamically sampling points based on the region of interest can result in oversampling elsewhere, once again affecting performance.

V. USE CASE ANALYSIS

The ellipsoid representation can be used in a majority of conjunctions due to the large number of events satisfying the rectilinear assumption.[1] In these cases, ellipsoids accurately illustrate both the primary and secondary PDFs; since the typical duration of conjunctions satisfying the rectilinear assumption is short, there is negligible change in the covariances due to propagation and therefore little change in the ellipsoids (assumed to be none when analyzing the conjunction using the 2D-P_c method).[1,4] Since the generation of the ellipsoids is fast and rectilinear conjunctions make up the majority of high-P_c events analyzed by CARA, the ellipsoid method can be used to produce visualizations efficiently (*e.g.*, for O/O's or other analysts) on short notice.[4]

The bananoid representation can likely be used for nearly all remaining conjunctions that fail to satisfy the rectilinear assumption. Since the bananoid surfaces can be modified to envelop a user-specified percentage of the PDF (or, equivalently, a selected n-σ surface), it is easy to generate surfaces of the size required to show intersection by determining the Mahalanobis distance between the center of each distribution and the peak overlap point.[4,6] This is not possible in curvilinear cases using only the ellipsoid representation and can be prohibitively inefficient using the point cloud representation if the peak overlap point is sufficiently distant from the mean states of both objects.

The bananoid surface visualization method requires more research to determine the limitations of its use; however, testing with numerous conjunctions appears to produce accurate surfaces, an encouraging result for more widespread implementation.

The point cloud representation is ideally used for conjunctions in which even the bananoid surface is insufficient or for high-interest events (HIEs) which require further analysis. This includes conjunctions with very extended primary and/or secondary PDFs, as well as potentially multi-encounter conjunctions which violate the 3D-N_c assumptions. The point cloud representation is also ideal if a visualization of variable density in the PDFs is desirable; by removing the need for a surface entirely, an end user can distinguish regions of high and low risk visually prior to a more in-depth analysis.

It is also possible to transform a point cloud into a shell representation similar to that of the bananoid by first trimming the cloud down to those points within a certain Mahalanobis distance in equinoctial space, then converting back to Cartesian space and forming a surface with the points closest to the exterior of the remaining cloud. If inadequate samples are used, however, this produces an irregular (*i.e.*, bumpy) surface, possibly leading to misinterpretation of the PDF shape and scale. This surface generation method also requires significantly more computation than the bananoid method and therefore increased runtime.

VI. FUTURE WORK

The bulk of the research required to build upon this work will be in the process of making improvements to the bananoid representation method, either through further development of the mathematics behind the model or through computational efficiency improvements of its implementation. This research will allow for the bananoid method to be applied much more widely in the two-body-style conjunction visualizations, potentially superseding the less accurate ellipsoid or less computationally efficient point cloud; however, before the bananoid can be deployed reliably, significantly more testing and verification of the theory is necessary.[15]

A different path of research, and one potentially both more illuminating and more important, is that of cislunar conjunction visualization. As described by Holzinger, Chow, and Garretson (2021), describing trajectories in cislunar space is much more difficult than describing orbits assuming only one gravitationally significant body (the Earth), even accounting for perturbations.[22] Among other issues, orbital elements often do not provide an accurate representation of cislunar trajectories, rendering many tools and assumptions currently used for orbital analysis (*e.g.*, two-line element sets [TLEs]) useless.[22] NASA's Multimission Automated Deepspace Conjunction Assessment Process (MADCAP) currently aims to perform conjunction assessment for cislunar space (and beyond); however, a lack of a central catalogue of cislunar objects and a lack of realistic covariances for objects that are screened makes this a challenging effort.[1,23] Visualizing cislunar conjunctions would aid not only in analysis, but also in building an understanding of the cislunar environment and the geometry of deep-space mission interactions. Approaching this problem will require careful consideration of the covariances used for propagation, as equinoctial elements will no longer be viable; realistic covariances in cislunar space and beyond are currently still an open topic of research.[1,22,23]

Another possible avenue of research involves creating a three-dimensional analog of the two-dimensional conjunction plane plots to illustrate relative behavior in a manner more similar to that used currently by NASA CARA.[1] Combining the uncertainty into one surface will yield ellipsoids in rectilinear cases and thus be relatively easy to demonstrate; early experimentation using point clouds, however, suggests that more exotic shapes can arise, including those with self-intersections and large out-of-plane components. More experimentation with these methods is needed before they are viable for production- or analysis-grade visualizations.

VII. CONCLUSION

This study explores three methods for the visualization of conjunctions involving two Earth-orbiting bodies (ellipsoid, bananoid, and point cloud) and reviews potential use cases for all three. Additionally, the study draws parallels between the methods used for visualization and the methods used to calculate probability of collision estimates.

Given the widespread usability, relatively low cost, and visual utility of the bananoid conjunction visualization method, this study recommends the bananoid method for further research and development for individuals and groups interested in conjunction visualization. The ability to visualize a wide range of conjunction events accurately makes it a viable tool for both analysis and intuitive understanding, as well as a stepping-stone to visualizing conjunctions that occur in cislunar space and beyond; despite equinoctial elements not being applicable beyond Earth's orbit, the concepts employed in the derivation of the bananoid method (using expansions in a separate coordinate system and mapping back to Cartesian space to avoid direct Monte Carlo sampling) is applicable to other coordinate systems as well.

Using the bananoid and point cloud methods in unison for analysis of HIEs additionally offers analysts a better understanding of the conjunction environment, both illustrating the probability of a conjunction event through the overlap in covariance n-σ surfaces (something not directly possible using a pure Monte Carlo approach) and visualizing the calculations used in Monte Carlo approaches. Where the bananoid surface fails for some rare conjunction geometries (such as the one in Case C), the three-dimensional point cloud approach can be used not only as an alternative visualization tool, but also as a means by which to understand the shortcomings of analytical and semi-analytical P_c calculation algorithms in such events.

Visualization methods allow for increased understanding of how conjunction geometry and trajectory uncertainty impact P_c. For instance, the above methods can be used to visualize a steady climb in P_c followed by a sharp drop as primary and secondary covariances start large, shrink towards each other while still overlapping, and finally shrink small enough (with measurements closest to TCA) that the overlap rapidly vanishes. Such insights are not available from the P_c metric alone (which does not convey any information about the geometry of a conjunction) and can help analysts develop a better intuition of conjunction geometries. Visualization provides end users not necessarily familiar with the mathematical P_c calculation process with a means to understand better the risks posed by conjunctions and the need (or lack thereof) to take action on such events.

The current lack of a universal tool for conjunction visualization highlights the importance of developing such a method in the near future; providing astrodynamicists with the means to better understand the near-Earth environment in a time when new satellites are being deployed very frequently is critical to ensuring both the longevity of missions and the safety of the space environment.

VIII. ACKNOWLEDGMENTS

The authors would like to thank Doyle Hall and Eliot Toumey for their continual advice, discussions, and pointers in producing this paper, and especially for their insight into the types of visualizations that would be useful for the conjunction assessment community and their generous time spent proofreading and providing suggestions to improve the paper. The authors would also like to thank Doyle Hall for his willingness to share his new theory on representing probability density functions using bananoids and allowing its use and inclusion in this paper, as well as his guidance on and review of its derivation as it is presented in this paper.

NOTATION

$\mathcal{A}$	Bananoid quadratic equation coefficient
$\bar{\mathbf{A}}$	The position-position component of a Cartesian covariance matrix
$\tilde{\mathbf{A}}$	The position-position component of a Cartesian covariance matrix at the center of linearization of a Taylor series
$\mathcal{B}$	Bananoid quadratic equation coefficient
$\hat{\boldsymbol{B}}$	The binormal vector in the velocity-normal-binormal frame
$\mathcal{C}$	Bananoid quadratic equation coefficient
$\bar{\boldsymbol{E}}$	An equinoctial state vector
$\tilde{\boldsymbol{E}}$	An equinoctial state vector at the center of linearization of a Taylor series
$\bar{\mathbf{J}}$	A Jacobian matrix for conversion between Cartesian and equinoctial coordinates
$\mathcal{M}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$	The squared Mahalanobis distance of the vector $\boldsymbol{x}$ from the center of the distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\hat{\boldsymbol{N}}$	The normal vector in the velocity-normal-binormal frame
$\bar{\mathbf{P}}$	A Cartesian covariance matrix
$\tilde{\mathbf{P}}$	A Cartesian covariance matrix at the center of linearization of a Taylor series
$\bar{\boldsymbol{\mathcal{P}}}$	An equinoctial covariance matrix
$\mathbf{Q}$	An orthogonal eigenvector matrix
$\boldsymbol{r}$	An arbitrary Cartesian position vector
$\bar{\boldsymbol{r}}$	The position component of a Cartesian state vector
$\breve{\boldsymbol{r}}$	The position component of a Cartesian effective mean state used in minimum Mahalanobis distance iteration

$\widehat{\boldsymbol{V}}$	The velocity vector in the velocity-normal-binormal frame
$\overline{\boldsymbol{X}}$	A Cartesian state vector
$\widetilde{\boldsymbol{X}}$	A Cartesian state vector at the center of linearization of a Taylor series
$\breve{\boldsymbol{X}}$	A Cartesian effective mean state used in minimum Mahalanobis distance iteration
$\boldsymbol{\Lambda}$	A diagonal eigenvalue matrix
ϕ	A parameter for bananoid generation roughly equivalent to longitude
$\rho(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$	A normal probability density function as a function of $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
τ	A parameter for bananoid generation roughly equivalent to latitude
$\chi^2_m(n)$	The chi-squared distribution with m degrees of freedom evaluated at n
$\widetilde{\boldsymbol{\Psi}}$	A Jacobian matrix encoding the partial derivatives of the Cartesian and equinoctial state vectors at the center of linearization
$\partial\Omega$	The boundary of a set Ω

REFERENCES

[1] National Aeronautics and Space Administration, "NASA Spacecraft Conjunction Assessment and Collision Avoidance Best Practices Handbook," NASA/SP-20230002470 Rev 1, Feb 2023.

[2] J. L. Foster and H. S. Estes, "A Parametric Analysis of Orbital Debris Collision Probability and Maneuver Rate for Space Vehicles," NASA/JSC-25898, Aug. 1992.

[3] S. Alfano, "A Numerical Implementation of Spherical Object Collision Probability." *Journal of the Astronautical Sciences*, Vol. 53, No. 1, pp. 103-109, 2005.

[4] D. T. Hall, L. G. Baars, and S. J Casali, "A Multistep Probability of Collision Computational Algorithm," *AAS/AIAA Astrodynamics Specialist Conference*, Big Sky, MT, Paper 23-398, Jul. 2023.

[5] T. Coppola, "Evaluating the Short Encounter Assumption of the Probability of Collision Formula," *AAS/AIAA Spaceflight Mechanics Meeting*, Charleston, SC, Paper 12-248, Feb. 2012.

[6] D. T. Hall, "Expected Collision Rates for Tracked Satellites," *Journal of Spacecraft and Rockets*, Vol. 58, No. 3, pp. 715-728, May-June, 2021.

[7] D. T. Hall , Casali, S. J., Johnson, L.C., Skrehart, B. B., and Baars, L. G., "High Fidelity Collision Probabilities Estimated Using Brute Force Monte Carlo Simulations," *AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, UT, Paper 18-244, Aug. 2018.

[8] S. Alfano and D. Oltrogge, "Probability of Collision: Valuation, Variability, Visualization, and Validity," *Acta Astronautica*, Vol. 148, pp. 310-316, April 2018.

[9] C. Sabol, T. Sukut, K. Hill, K. T. Alfriend, B. Wright, Y. Li, and P. Schumacher, "Linearized Orbit Covariance Generation and Propagation Analysis via Simple Monte Carlo Simulations," *AAS Guidance, Navigation, and Control Conference*, Toronto, ON, Paper 10-134, Aug. 2010.

[10] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, 6th ed., Pearson Education, Upper Saddle River, NJ, 2007.

[11] P. J. Olver and C. Shakiban, *Applied Linear Algebra*, 2nd ed., Springer International Publishing, Cham, CH, 2018.

[12] J. W. McMahon, N. Baresi, and D. J. Scheeres, "On the Projection of Covariance Ellipsoids on Non-Planar Surfaces," *AAS/AIAA Astrodynamics Specialist Conference*, Vail, CO, Paper 15-667, Aug. 2015.

[13] F. R. Hoots, "Satellite Location Uncertainty Prediction," *AAS/AIAA Astrodynamics Specialist Conference*, Girdwood, AK, Paper 11-579, Aug. 2011.

[14] L. Pagani and P. J. Scott, "Curvature Based Sampling of Curves and Surfaces," *Computer Aided Geometric Design*, Vol. 59, pp. 32-48, 2018.

[15] D. T. Hall, "A Semi-analytical Method of Representing Conjunction Probability Density Functions," manuscript in preparation, June 2025.

[16] J. Hernando-Ayuso, C. Bombardelli, G. Baù, and A. Martínez-Cacho, "Near-Linear Orbit Uncertainty Propagation Using the Generalized Equinoctial Orbital Elements," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 4, pp. 654-665, 2023.

[17] Poore *et al.*, "Covariance and Uncertainty Realism in Space Surveillance and Tracking," Defense Technical Information Center, Technical Report, Accession No. 1020892, Jun. 2016.

[18] D. T. Hall, "Ephemeris-Based Satellite Collision Rates and Probabilities," *Journal of Spacecraft and Rockets*, Published Online, DOI: 10.2514/1.a36227, Jan. 22, 2025.

[19] D. A. Vallado, *Fundamentals of Astrodynamics and Applications*, 2nd ed., Microcosm Press, El Segundo, CA, 2004.

[20] V. T. Coppola and S. Tanygin, "Using Bent Ellipsoids to Represent Large Position Covariance in Orbit Propagation," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 9, pp. 1775-1784, 2015.

[21] D. A. Vallado and S. Alfano, "Updated Analytical Partials for Covariance Transformations and Optimization," *AAS/AIAA Astrodynamics Specialist Conference*, Vail, CO, Paper 15-537, Aug. 2015.

[22] M.J. Holzinger, C. C. Chow, and P. Garretson, "A Primer on Cislunar Space," Air Force Research Laboratory, Technical Report, May 2021.

[23] Z. B. Tarzi, D. S. Berry, Z. Kaufman, J. Ma, and L. K. Newman, "Deep-space Conjunction Assessment: Recent Developments and Future Evolution," *29th International Symposium on Space Flight Dynamics*, Darmstadt, DE, Apr. 2024.