(Preprint) AAS 24-339

PROCESSING ANGLES-ONLY TRACKLETS FOR CISLUNAR MULTI-TARGET TRACKING

Dalton Durant[*], Andrey A. Popov[†], Kyle J. DeMars[‡], and Renato Zanetti[§]

This work addresses the challenge of multi-target tracking in cislunar orbits utilizing angles-only tracklets from ground-based optical telescopes. Traditional orbit determination methods rely on the batch processing of angles-only measurements into a six-dimensional state-space. These methods encounter limitations in handling nonlinearities within cislunar dynamics. To overcome these limitations, this work leverages Markov Chain Monte Carlo methods for processing angles-only tracklets in Cartesian coordinates and performs multi-target tracking with Joint Probabilistic Data Association techniques. Numerical simulations illustrate the superiority of the proposed methods in scenarios with multiple targets and increased orbital congestion. This work contributes to a comprehensive solution for angles-only multi-target tracking in complex cislunar environments.

INTRODUCTION

Optical telescopes detect apparent positions of space objects (or targets) as streaks or discrete sequences on the sensor's sensitive element. These detections are converted into azimuth/elevation angles, essential for orbit determination algorithms to estimate target orbits. Figure 1 provides a visual representation of how an optical telescope can generate tracklets consisting of multiple targets.

While point-wise-in-time unobservable, a sequence of angles theoretically identifies the orbit, yet tracklets still harbor significant range and range-rate uncertainty. In practice, angles-only measurements from a tracklet are typically assimilated by orbit determination algorithms in one of two ways: either pre-processed into an orbit estimate by a batch algorithm, or processed sequentially by a recursive algorithm.

In Earth orbit, pre-processing angles-only tracklets into a six-dimensional state-space of orbital element coordinates proves advantageous, resulting in nearly linear dynamics and Gaussian probability density functions.[1] However, in the cislunar regime, nonlinear dynamics render Earth orbital elements unsuitable, and Cartesian coordinates lead to non-Gaussian distributions.

Our prior work proposed using Markov Chain Monte Carlo (MCMC) to produce accurate, non-Gaussian likelihood functions for the pre-processed algorithms.[2] MCMC techniques, used for

[*]Ph.D. Student, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX.

[†]Postdoctoral Fellow, Oden Institute for Computational Engineering & Sciences, The University of Texas at Austin, Austin, TX.

[‡]Associate Professor, Department of Aerospace Engineering, Texas A&M University, College Station, TX.

[§]Associate Professor, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX.

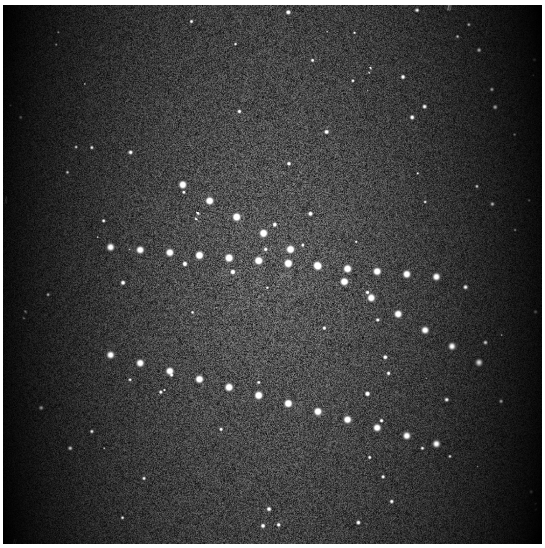

Figure 1: Example image shows 3 targets being tracked with a telescope at a sidereal rate. The stars present themselves as points, whereas the targets discretely streak across the frame generating a tracklet each.

Bayesian inversion, generate samples from a target probability density function (PDF) using relatively straightforward proposal and rejection principles. The most prominent MCMC algorithm is the Metropolis-Hastings (MH) algorithm, known for its effectiveness in tackling complex statistical problems with high dimensions.[3,4] In recent times, MH has found application in initial orbit determination,[5,6] angles-only relative orbit determination,[7] and angles-only orbit determination using tracklets from ground-based optical telescopes.[8]

By constructing a Gaussian Mixture Model (GMM) from MCMC samples, the non-Gaussian measurement probability distribution in the state-space can be accurately approximated in the sample limit.[2] The resulting GMM can be used as processed measurements for a single target nonlinear filter like the Ensemble Gaussian Mixture Filter (EnGMF).[2,9,10]

This work performs Multi-Target Tracking (MTT) using MCMC processed angles-only tracklets in conjunction with Joint Probabilistic Data Association (JPDA) techniques. To ensure accurate orbit determination, each target is estimated using the EnGMF. This work not only enhances the effectiveness of data association but also contributes to a more comprehensive and cost-effective solution for MTT in cislunar orbits by associating entire tracklets with a non-Gaussian likelihood function accurately represented through MCMC samples. The methodology is detailed, encompassing kernel-based ensemble Gaussian mixture filtering, MCMC background, and JPDA methods, with empirical results presented for a cislunar Near Rectilinear Halo Orbit (NRHO) problem using optical telescopes.

KERNEL-BASED ENSEMBLE GAUSSIAN MIXTURE FILTERING

The EnGMF[9] stands out as a powerful particle filter, exhibiting the ability to recover true Bayesian inference with few particles. Recent applications have demonstrated its effectiveness in tracking targets even when confronted with sparse observation data.[2,10–15] This section provides a brief in-

troduction to the EnGMF as a single-target nonlinear filter, with comprehensive algorithmic details available in the References 2 and 10. It utilizes the concept of kernel density estimation (KDE);[16] a non-parametric method that estimates the probability density function of random variables. The EnGMF uses KDE to combine two popular filtering techniques, *particle filters* and *Gaussian sum filters*; producing a smoothed representation of the distribution of particles, which in the limit of ensemble size converges to exact Bayesian inference. This can result in a more accurate representation of the posterior distribution, and can subsequently improve state estimation performance.

This work makes use of the Modified Kernel-Based Ensemble Gaussian Mixture Filtering method[10] which is a sequential estimation filter that approximates the distribution of a system based on sparse data. It is used to perform orbit determination for a target given an associated tracklet. The EnGMF assumes the knowledge of the distribution at the prior time, $p(\mathrm{x}_{k-1})$, and approximates it with N independent and identically distributed (i.i.d.) samples (particles) $x_{k-1}^{(i)}$,

$$p(\mathrm{x}_{k-1}) \approx \sum_{i=1}^{N} \frac{1}{N} \delta(\mathrm{x}_{k-1} - x_{k-1}^{(i)}) \approx \lim_{P_{k-1} \to \mathbf{0}} \sum_{i=1}^{N} \frac{1}{N} \mathcal{N}(\mathrm{x}_{k-1}; x_{k-1}^{(i)}, P_{k-1}), \tag{1}$$

where k is the discrete time step, $\delta(\cdot)$ is the Dirac delta distribution and can be said to be the normal distribution with covariance that tends towards the zero matrix $\mathbf{0}$ in the limit. The EnGMF only makes an empirical measure assumption just before the propagation step, represented by Equation (1). The particles are then propagated to time step k and are converted into Gaussian mixtures using a standard KDE method, where each particle is considered a Gaussian component with non-zero covariance of equal weights:

$$p(\mathrm{x}_k) \approx \sum_{i=1}^{N} \frac{1}{N} \mathcal{N}(\mathrm{x}_k; \bar{x}_k^{(i)}, B_S), \tag{2}$$

where $\bar{x}_k^{(i)}$ is the propagated state of the i^{th} particle of time step k. The covariance (bandwidth) matrix B_S for each component is determined using Silverman's Rule of Thumb[16] to reduce computational costs. The bandwidth matrix B_S is estimated as follows:

$$B_S = \beta_S \bar{P}_k = \left(\frac{4}{n_x + 2} \right)^{\frac{2}{n_x+4}} N^{-\frac{2}{n_x+4}} \bar{P}_k, \tag{3}$$

where β_S is the bandwidth parameter, n_x is the dimension of the state, and $\bar{P}_k$ is the sample covariance matrix computed from the propagated samples. The measurement information is integrated by updating the means, covariance matrices, and weights of all N Gaussian components, following the measurement update procedure of the Gaussian sum filter (GSF) in Reference 17. In this step, each Gaussian component can be updated using the extended Kalman filter (EKF)[18] or unscented Kalman filter (UKF).[19,20] Subsequently, N i.i.d. samples are drawn from the GMM representation of the posterior distribution, serving as the starting point for the next iteration.

MARKOV CHAIN MONTE CARLO

Constructing a GMM from MCMC samples provides an accurate approximation of the non-Gaussian measurement probability distribution within the state-space. This approach is especially effective as the number of samples increases, allowing the GMM to capture the intricacies of the distribution in the sample limit. In this work, this resulting GMM serves a dual purpose: (i) acting

as processed measurements suitable for a nonlinear filter, like the EnGMF, and (ii) facilitating the accurate approximation of the non-Gaussian measurement distribution. Moreover, employing KDE alongside MCMC samples enables the arbitrary approximation of any distribution. The combination of MCMC samples and KDE proves to be a powerful tool, as demonstrated by Reference 2. They show that with a sufficient number of MH samples, it is possible to construct a GMM that accurately approximates the non-Gaussian measurement distribution for processed tracklets.

The following is how this work uses the MH algorithm to generate samples from the measurement distribution and accurately approximates it using KDE. Consider the state dynamics model and measurement model in discrete form:

$$\mathrm{x}_{k+1} \;=\; f(\mathrm{x}_k) \;+\; \epsilon_k, \tag{4}$$

$$\mathrm{y}_k \;=\; h(\mathrm{x}_k) \;+\; \eta_k, \tag{5}$$

where x_k is the state at time step k with propagation function $f(\cdot)$ and white, Gaussian process noise $\epsilon_k \sim \mathcal{N}(\mathbf{0}, Q)$. Additionally, y_k is the measurement at time k with mapping function $h(\cdot)$ and white, Gaussian measurement noise $\eta_k \sim \mathcal{N}(\mathbf{0}, R)$ uncorrelated to $\epsilon_k \; \forall \; k$. Then, a tracklet can be represented by $Y_{k+s|k} = [\mathrm{y}_k, \mathrm{y}_{k+1}, \cdots, \mathrm{y}_{k+s}] \subset \mathbb{Y}$ with individual angles-only measurements y from time steps k to the end of the tracklet $k+s$; $\mathbb{Y}$ is the single-target measurement space.

The non-Gaussian distribution to be known is $p(\mathrm{y}_k)$ which is the measurement PDF at time step k. With MCMC, $p(\mathrm{y}_k)$ is approximated in the state-space with

$$p(\mathrm{y}_k) \;\Rightarrow\; p(\tilde{\mathrm{x}}_k) \;\approx\; \sum_{u=1}^{M} \frac{1}{M} \mathcal{N}(\tilde{\mathrm{x}}_k; \tilde{x}_k^{(u)}, \tilde{P}_k^{(u)}), \tag{6}$$

where $p(\tilde{\mathrm{x}}_k)$ is a GMM built by M MCMC samples $\tilde{x}_k^{(u)}$ with covariance $\tilde{P}_k^{(u)}$. Each $\tilde{x}_k^{(u)}$ is the last state in the Markov chain produced by the MH algorithm below and each $\tilde{P}_k^{(u)} = \beta_S \tilde{P}_k$, where β_S and $\tilde{P}_k$ are the bandwidth parameter and sample covariance, respectively, of the M MCMC samples.

Metropolis-Hastings.[4]

---

1. Choose an initial value for the states: $\tilde{x}_k = \tilde{x}_{k,0}$.
2. Propose a new value for the states $\tilde{x}_k^{'}$ using a proposal distribution $p(\tilde{x}_k^{'} | \tilde{x}_k)$.
3. Compute the acceptance probability $\alpha = \min \left\{ 1, \frac{p(\tilde{x}_k^{'} | Y_{k+s|k}) \cdot p(\tilde{x}_k | \tilde{x}_k^{'})}{p(\tilde{x}_k | Y_{k+s|k}) \cdot p(\tilde{x}_k^{'} | \tilde{x}_k)} \right\}$.
4. Generate a uniform random number u from the interval $[0, 1]$.
5. If $u \leqslant \alpha$, accept and store the proposal and set $\tilde{x}_k = \tilde{x}_k^{'}$, otherwise reject the proposal and keep $\tilde{x}_k$ unchanged.
6. Repeat steps 2-5 until the desired number of samples is obtained.

This work uses the proposal distribution

$$p(\tilde{x}_k^{'}|\tilde{x}_k) = \mathcal{N}(\tilde{x}_k^{'}; \tilde{x}_k, \tilde{P}_{k,0}) = \mathcal{N}(\tilde{x}_k; \tilde{x}_k^{'}, \tilde{P}_{k,0}) = p(\tilde{x}_k|\tilde{x}_k^{'}), \tag{7}$$

where initially $\tilde{x}_k = \tilde{x}_{k,0}$ and $\tilde{x}_k^{'} \sim \mathcal{N}(\tilde{x}_{k,0}, \tilde{P}_{k,0})$ with $\tilde{x}_{k,0}$ and $\tilde{P}_{k,0}$ being anything, but for a good initial guess they can be chosen to be the Gaussian posterior update given tracklet $Y_{k+s|k}$. Note the two proposals are symmetric, inevitably canceling out in step 3 of the MH algorithm (*i.e.* reducing to the Metropolis algorithm).

The target distribution is

$$\begin{aligned} p(\tilde{x}_k|Y_{k+s|k}) &\approx c \cdot p(Y_{k+s|k}|\tilde{x}_k) \\ &= c \prod_{i=k}^{k+s} \exp\left(-\frac{1}{2}[Y_i - h(f(\tilde{x}_k)_i)]^T R^{-1}[Y_i - h(f(\tilde{x}_k)_i)]\right), \end{aligned} \tag{8}$$

where $p(\tilde{x}_k|Y_{k+s|k}) \approx c \cdot p(Y_{k+s|k}|\tilde{x}_k)$ because the prior $p(\tilde{x}_k)$ is assumed to be diffuse. The unknown constant c in Equation (8) will cancel out with itself when computing the acceptance ratio α. Furthermore, Equation (8) represents the likelihood of the tracklet of measurements $Y_{k+s|k}$ given a proposed state $\tilde{x}_k$. In other words, it is the likelihood that the proposed Markov Chain state is a feasible realization (through time) for every angles-only measurement of the tracklet. Equation (8) can similarly be used to compute $p(\tilde{x}_k^{'}|Y_{k+s|k})$ by swapping $\tilde{x}_k$ with $\tilde{x}_k^{'}$. For each tracklet, this work produces a GMM of $M = 100$ samples with a Markov chain length of 10 acceptances.

JOINT PROBABILISTIC DATA ASSOCIATION

Although not particularly attractive, JPDA can be simple to implement and it provides the required framework for labeling/assignment of target-tracklet pairs. Additionally, it enables this work to showcase the benefits of processing tracklets using MCMC and how it can be incorporated into data association methods for MTT.

Similar to the suboptimal JPDA filter[21] and the Fitzgerald *ad hoc* method,[22] this work simplifies the optimal JPDA filter.[23] This simplification assumes the probability of detection for each target is one or very nearly one and there exists good track initiation: all of the targets will be detected. The probability of an event θ_k given the set of all state-space MCMC processed tracklets $\tilde{X}_k$ at time k can be represented by*

$$\mathrm{P}(\theta|\tilde{X}) = \frac{1}{c} \prod_{j=1}^{m} [p(\nu_j)]^{\tau_j}, \tag{9}$$

where $\tau_j = 1$ if tracklet j is assigned any target $t = 1 : T$ and zero otherwise. This leaves the problem of calculating the PDF $p(\nu_j)$ for all possible joint target-to-tracklet assignments. The Fitzgerald *ad hoc* method eliminates the need to compute all possible joint events by replacing the joint events with partial joint events called single events G_{tj}. Fitzgerald's *ad hoc* JPDA or "cheap JPDA" calculates the suboptimal probability of target t being associated only with tracklet j:

$$\mathcal{P}_{tj} = \frac{G_{tj}}{S_t + S_j - G_{tj} + B}, \tag{10}$$

*The subscript k, denoting the time step, is now dropped to avoid confusion with the subscripts denoting the targets and tracklets, t and j, respectively. For the rest of this section and subsequent sections, all variables exist at the same time step k unless otherwise noted.

where

$$\begin{aligned} G_{tj} &= p(\nu_j), \\ S_t &= \sum_{j=1}^{m} G_{tj}, \\ S_j &= \sum_{t=1}^{T} G_{tj}, \\ B &= \text{constant which depends on clutter density.} \end{aligned} \tag{11}$$

This *ad hoc* method exhibits traits similar to the optimal Joint Probabilistic Data Association (JPDA) filter. Specifically, it strongly weights tracklets within only one target gate and lightly weights tracklets situated in regions where multiple target gates intersect and compete. This method prioritizes greater weights for tracklets in close proximity to the predicted targets. Fitzgerald suggests that $B = 0$ gives satisfactory results in all cases except very dense clutter. Finally, if there is only one target that associates with a tracklet and there is only one unique tracklet for every target, then Reference 21 reduces the suboptimal probability to

$$\mathcal{P}_{tj} = \frac{G_{tj}}{\sum_{j=1}^{m} G_{tj}}. \tag{12}$$

In a scenario where targets dominate the environment, Fitzgerald advocated abandoning the optimal Joint Probabilistic Data Association (JPDA) filter in favor of the Nearest Neighbor Joint Probabilistic Data Association (NNJPDA) approach. The NNJPDA approach closely resembles the conventional recursive method of associating targets with tracklets. Typically in each gate, the statistical distance from predicted targets to tracklets is calculated, and a two-dimensional assignment determines which tracklet is chosen to update a target.[21,22,24,25] The simplest approach is the nearest neighbor algorithm, assigning each target the tracklet with the closest position to the predicted target.[21,22,26] However, this method is limited to low-density target and clutter environments as it neglects alternative target associations.

A more effective assignment method for NNJPDA is the greedy nearest neighbor algorithm.[21] This algorithm selects the target-to-tracklet association with the closest score, eliminates all other associations for that target and tracklet, and proceeds to the next closest assignment. While it results in a nonconflicting set of target-tracklet associations, it is not optimal, yet remains valuable and computationally more efficient than the optimal JPDA filter. Additionally, unlike the nearest neighbor algorithm, the greedy nearest neighbor algorithm considers other target-tracklet assignments when making decisions. Furthermore, by eliminating all associations after making an assignment, it prevents assigning duplicate tracklets arising from potential conflicting assignments.

This work uses the NNJPDA with the greedy assignment algorithm. The two-dimensional assignment is

$$\mathcal{P} \in \mathbb{R}^{T \times m}. \tag{13}$$

Furthermore, this work assumes that each target's gate encompasses every tracklet (*i.e.* each target has possible associations with every tracklet). This means that every $\mathcal{P}_{tj}$ has the same denominator and can be removed from the greedy algorithm. Thus, Equation (12) reduces to

$$\mathcal{P}_{tj} = G_{tj}. \tag{14}$$

Therefore, the main assumptions made are: (1) every tracklet is processed into the state-space to the same discrete time step k — (2) Every target is assigned a unique tracklet by the greedy algorithm. These assumptions imply that the gates of each target are large enough to encompass every processed tracklet at time step k (*i.e.* every tracklet has the possibility of being uniquely associated with every target at k).

The rest of this section explores the different methods for performing NNJPDA by computing the single events G_{tj} given processed tracklet data. The methods presented are:

① NNJPDA for Gaussians.

② NNJPDA for Gaussian Mixtures.

NNJPDA for Gaussians

Using Batch Least Squares Processed Tracklets. The batch least squares information filter[27] is a batch processing algorithm used to compute the Maximum *a Posteriori* (MAP) estimate of the state from all available raw measurements in a given tracklet. It condenses a tracklet's raw measurements into a single mean and covariance in the state-space at time step k, serving as "processed" measurements for the assigned target's EnGMF. Then, the components of the EnGMF are individually updated via the EKF. This subsequently updates the target's GMM approximation of the posterior distribution. See Reference 2 for details.

For this to work for MTT, the batch processed tracklets must be assigned to the best possible target. In Figure 2, the single event G_{tj} is computed between the batch output of tracklet j and the weighted EnGMF GMM components of target t at time step k. Let the processed tracklet j be

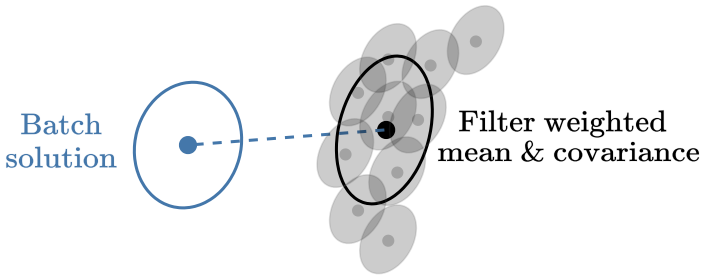

Figure 2: Illustration of computing the single event between a tracklet's measurement distribution (represented by the batch least squares solution) and a target's state distribution (represented by the weighted sum of components of the propagated EnGMF GMM).

represented by a random variable $\tilde{\mathrm{x}}$ distributed by $p(\tilde{\mathrm{x}};\ \tilde{x}_j,\ \tilde{P}_j) = \mathcal{N}(\tilde{x}_j, \tilde{P}_j)$, and let the target t be represented by a different random variable indepent of $\tilde{\mathrm{x}}$ and distributed by $p(\mathrm{x};\ x_t,\ P_t) = \mathcal{N}(x_t, P_t)$. Since the distribution of the processed tracklet j is Gaussian and the distribution of the target t is also Gaussian, then can calculate the negative log of their difference and reassign G_{tj}:

$$\begin{aligned} G_{tj} &= -\log\left(p(\mathrm{x};\ x_t,\ P_t) - p(\tilde{\mathrm{x}};\ \tilde{x}_j,\ \tilde{P}_j)\right) \\ &= -\log\left(p(\mathrm{x}-\tilde{\mathrm{x}}; x_t - \tilde{x}_j, P_t + \tilde{P}_j)\right) \\ &= \frac{1}{2}(x_t - \tilde{x}_j)^T (P_t + \tilde{P}_j)^{-1}(x_t - \tilde{x}_j) + \frac{1}{2}\log|P_t + \tilde{P}_j| + \text{const.}, \end{aligned} \tag{15}$$

where $(x_t, P_t) \in X \subset \mathbb{X}$ are the state and covariance of the target t, and $(\tilde{x}_j, \tilde{P}_j) \in \tilde{X} \subset \tilde{\mathbb{X}}$ are the state and covariance of the processed tracklet j. $\mathbb{X}$ represents the single-target space and $\tilde{\mathbb{X}}$ represents the processed single-tracklet space. The single events are computed for every possible target-tracklet combination.

Using MCMC Processed Tracklets. Instead of processing raw measurements through a batch least squares algorithm, a better approach may be to use the MCMC Metropolis-Hastings method[3,4,28] to sample from the target measurement distribution directly (see Reference 2 for details). Once enough samples are collected, a GMM is constructed using KDE and Silverman's rule of thumb such that each sample is weighted equally and has the same covariance. This means that the weighted Gaussian mixture covariance is equivalent to the sample covariance of all the MCMC samples. Additionally, the weighted Gaussian mixture mean is equivalent to the sample mean of all the MCMC samples. These are leveraged to perform JPDA. In Figure 3, the single event G_{tj} is computed between the weighted GMM components of the MCMC output and the weighted GMM components of the EnGMF. The joint PDF is computed for every possible target-tracklet combination using Equation (15) and stored into the two-dimensional assignment matrix $\mathcal{P}$.

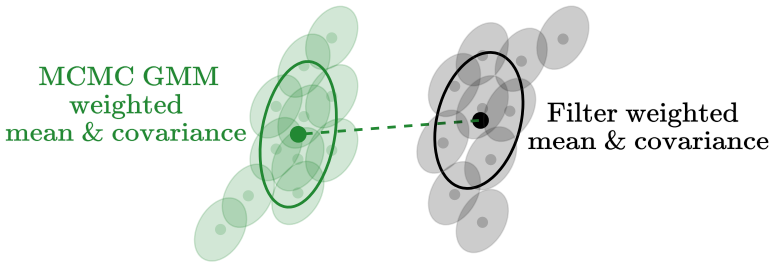

Figure 3: Illustration of computing the single event between a tracklet's measurement distribution (represented by the weighted sum of components of the MCMC GMM) and a target's state distribution (represented by the weighted sum of components of the propagated EnGMF GMM).

NNJPDA for Gaussian Mixtures

Using MCMC Processed Tracklets. The previous two methods are based on the assumption that the probability distributions of both the processed tracklets and the predicted targets have a single mean and covariance each. When these distributions cannot be expressed with simply mean and covariance, Equation (15) may not accurately capture the true statistical characteristics of the state-data relationship. For example, when non-Gaussian information is forced into a Gaussian framework, there may be a loss of important information, leading to poor data association and tracking results.

In cislunar orbit determination, nonlinearities often play a significant role in target motion and measurement processes. GMM's can better capture the effects of these nonlinearities and model uncertainties. In Figure 4, the single event G_{tj} is computed between the MCMC GMM of tracklet j and the propagated EnGMF GMM of target t at time step k.

This work proposes using the single event of Gaussian mixtures such that each component of the propagated target's GMM is compared with each component of the processed tracklet's GMM. This

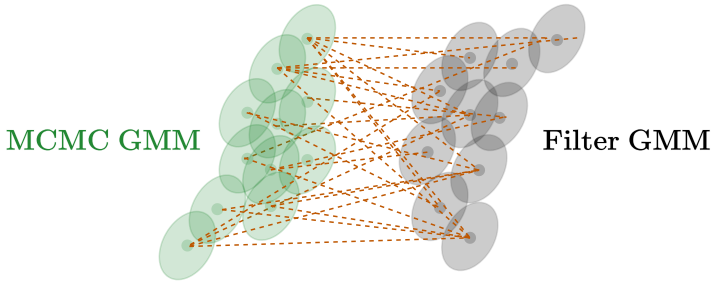

Figure 4: Illustration of computing the single event between a tracklet's measurement distribution (represented by the individual components of the MCMC GMM) and a target's state distribution (represented by the components of the propagated EnGMF GMM).

results in the negative log, single event for GMMs:

$$\begin{aligned} G_{tj} &= -\log\left(p(\mathrm{x};\, x_t,\, P_t) \,-\, p(\tilde{\mathrm{x}};\, \tilde{x}_j,\, \tilde{P}_j)\right) \\ &= -\log\left(\frac{1}{N\cdot M}\sum_{i=1}^{N}\sum_{u=1}^{M}\left(p(\mathrm{x};\, x_t^{(i)},\, P_t^{(i)}) \,-\, p(\tilde{\mathrm{x}};\, \tilde{x}_j^{(u)},\, \tilde{P}_j^{(u)})\right)\right) \\ &= -\log\left(\frac{1}{N\cdot M}\sum_{i=1}^{N}\sum_{u=1}^{M} p(\mathrm{x}-\tilde{\mathrm{x}}; x_t^{(i)}-\tilde{x}_j^{(u)}, P_t^{(i)}+\tilde{P}_j^{(u)})\right), \end{aligned} \tag{16}$$

where $x_t^{(i)}$ and $P_t^{(i)}$ are the state and covariance of the i-th GMM component of target t, and $\tilde{x}_j^{(u)}$ and $\tilde{P}_j^{(u)}$ are the state and covariance of the u-th GMM component of tracklet j. Equation (16) can be computed using a log-sum-exponential trick for numerical stability. The components of the targets GMM all share the same covariance such that $P_t^{(i)} = \beta_{S_t} P_t$ and can be calculated using Silverman's Rule of Thumb from Equation (3). Similarly, the GMM components of the processed tracklet's MCMC samples all share the same covariance such that $\tilde{P}_j^{(u)} = \beta_{S_j}\tilde{P}_j$ and also can be calculated using Silverman's Rule of Thumb from Equation (3). The single event G_{tj} is computed for every possible target-tracklet combination to construct $\mathcal{P}$.

Additional Considerations. In addition to computing single events G_{tj} between just Gaussians or between just Gaussian mixtures, a combination of the two can be made to mix-and-match methods; conceptually, mix-and-matching Figures 2, 3, and 4. Single events can therefore be computed between a single Gaussian and Gaussian mixture resulting in more flexibility and potential computational savings. Single events can be between the processed batch least squares solution and the individual components of the propagated EnGMF GMM; between the weighted sum of components of the MCMC GMM and the individual components of the propagated EnGMF GMM, or between the individual components of the MCMC GMM and the weighted sum of components of the propagated EnGMF GMM.

Assignment

Up until this point, this work has only shown how to compute the single events G_{tj} of the two-dimensional assignment matrix $\mathcal{P}$. Figure 5 shows an example of the two-dimensional matrix containing these single events. To actually perform the assignments, this work uses a greedy nearest

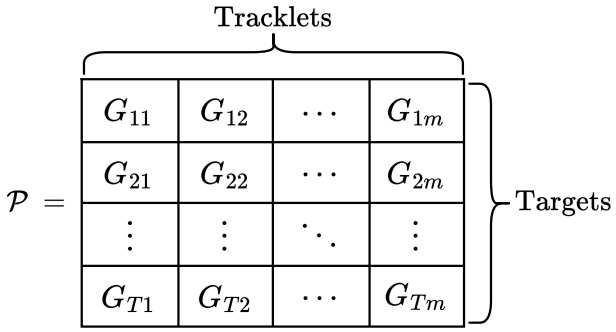

Figure 5: Two-dimensional assignment matrix with columns representing m tracklets and rows representing T targets. The elements of $\mathcal{P}$ are the individual single evenets between targets and tracklets.

neighbor algorithm. The minimum single event of $\mathcal{P}$ is found to assign the most likely target-tracklet combination. It then eliminates the corresponding row and column from future assignments to prevent duplicate assignments. This greedy approach allows for the consideration of all target-tracklet assignments when making a decision.

Algorithm 1 shows this assignment process. It takes input: the number of targets T, the number of tracklets m, the set $X = \{(x_1, P_1), (x_2, P_2), \cdots, (x_T, P_T)\} \subset \mathbb{X}$ of the propagated targets, the set $\tilde{X} = \{(\tilde{x}_1, \tilde{P}_1), (\tilde{x}_2, \tilde{P}_2), \cdots, (\tilde{x}_m, \tilde{P}_m)\} \subset \tilde{\mathbb{X}}$ of the processed tracklets, and the two-dimensional assignment matrix $\mathcal{P}$. It outputs

$$\begin{aligned} \mathbb{A} &\subset \mathbb{X} \times \tilde{\mathbb{X}} \\ &= \{(x_1, P_1, \tilde{x}_1, \tilde{P}_1), (x_2, P_2, \tilde{x}_2, \tilde{P}_2), \cdots, (x_T, P_T, \tilde{x}_T, \tilde{P}_T)\}, \end{aligned} \tag{17}$$

where $\mathbb{A}$ is the assigned target-tracklet subset of the combined single-target space $\mathbb{X}$ and single-tracklet space $\tilde{\mathbb{X}}$. Noticeably, if there are more tracklets than targets such that $m > T$, then $\mathbb{A}$ will contain the assignments of the most likely T combinations throwing away the remaining tracklets.

Algorithm 1 - Greedy Nearest Neighbor Assignment of Tracklets to Targets

Input: T, m, X, $\tilde{X}$, $\mathcal{P}$
Output: $\mathbb{A}$

$\mathcal{P}^* = \mathcal{P}$
$\mathbb{A} = \{\}$
for $t = 1$ to T **do**
  glob_min $= \min(\mathcal{P}^*)$ Get global minimum value in $\mathcal{P}^*$
  [idx, idy] = find($\mathcal{P}$ == glob_min) Find global minimum location in $\mathcal{P}$
  $\mathbb{A}\{t\} = (X\{\text{idx}\}, \tilde{X}\{\text{idy}\})$
  Shrink $\mathcal{P}^*$ by removing row associated with glob_min
  Shrink $\mathcal{P}^*$ by removing column associated with glob_min
end for

NRHO EXAMPLE

Problem Setup

During a lunar month lasting 27.3 days, the Moon undergoes waxing and waning phases across Earth's horizon.[29] Optimal Moon visibility occurs during either the first quarter or last quarter lunar phase, striking a balance between excessive sunlight and complete darkness (*i.e.* midway between Full Moon and New Moon phases). The Moon is visible for about 12 hours daily from moonrise to moonset.[29]

To evaluate the performance of the presented methods, two numerical examples are considered. A sensitivity study on the accuracy and consistency of each method is performed for tracking 3 targets and for tracking T targets. The system dynamic equations are numerically integrated with an embedded Runge-Kutta 8(7) method.[30] Angle measurements are simulated using an Earth-based optical ground telescope mapped to the Barycenter of the system. In this simulation, tracking passes are short and sparse. Raw angle measurements are available every 5 minutes and formed into tracklets that last 8 hours according to lunar moonrise to moonset visibility constraints.[29] A break in each tracklet occurs for 16 hours and then continues for another 8 to simulate the Moon not being visible during portions of the day. The system is propagated a month and the cycle continues: 8 hours sensing at a rate of 5 minutes, 16 hours propagate, 8 hours sensing at a rate of 5 minutes, propagate a month. This cycle is performed 3 times, so, for each target there will be 6 distinct tracklets of data.

Dynamics Model. This work models cislunar Near Rectilinear Halo Orbit (NRHO) dynamics using the Circular Restricted Three Body Problem (CR3BP) for the Earth-Moon system with a 6 dimensional state-space represented by $x = [r_1, r_2, r_3, v_1, v_2, v_3]^{\mathrm{T}}$:

$$\begin{aligned}
\dot{r}_1 &= v_1 \\
\dot{r}_2 &= v_2 \\
\dot{r}_3 &= v_3 \\
\dot{v}_1 &= r_1 + 2v_2 - \frac{(1-\mu)(r_1+\mu)}{r_{\oplus}^3} - \frac{\mu(r_1-1+\mu)}{r_{\leftmoon}^3} \\
\dot{v}_2 &= r_2 - 2v_1 - \frac{(1-\mu)r_2}{r_{\oplus}^3} - \frac{\mu r_2}{r_{\leftmoon}^3} \\
\dot{v}_3 &= -\frac{(1-\mu)r_3}{r_{\oplus}^3} - \frac{\mu r_3}{r_{\leftmoon}^3},
\end{aligned} \tag{18}$$

where r_1, r_2, r_3 and v_1, v_2, v_3 represent the scaled Cartesian positions and velocities of the satellite with-respect-to the Barycenter origin, μ is the scaled Moon geocentric gravitational constant, and $r_{\oplus}$ and $r_{\leftmoon}$ are the distances of the satellite with-respect-to the Earth and Moon in the Barycenter reference frame:

$$\mu = \frac{\mu_{\leftmoon}}{\mu_{\oplus} + \mu_{\leftmoon}}, \tag{19}$$

$$r_{\oplus} = \sqrt{(r_1+\mu)^2 + r_2^2 + r_3^2}, \tag{20}$$

$$r_{\leftmoon} = \sqrt{(r_1-1+\mu)^2 + r_2^2 + r_3^2}. \tag{21}$$

In this work, $\mu_{\oplus} = G \cdot m_{\oplus}$ and $\mu_{\leftmoon} = G \cdot m_{\leftmoon}$. The gravitational constant $G = 6.6743e-11$ [$\frac{\mathrm{m}^3}{\mathrm{s}^2\mathrm{kg}}$], the mass of the Earth $m_{\oplus} = 5.972e24$ [kg], and the mass of the Moon $m_{\leftmoon} = 7.342e22$ [kg]. The units for distance and time were non-dimensionalized by length units LU $= 384400e3$ [m] and time units TU $= \sqrt{\mathrm{LU}^3 / (\mu_{\oplus} + \mu_{\leftmoon})}$ [s].

Measurement Model. The raw measurement vector $\mathrm{y} = [\alpha,\ \delta]^T$ contains right ascension α and declination δ of the observed target as seen by an Earth-based optical ground telescope mapped to the Barycenter origin. The right ascension α and declination δ are

$$\alpha \ = \ \tan^{-1}\left(\frac{r_2 \ - \ r_{S_2}}{r_1 \ - \ r_{S_1}}\right), \tag{22}$$

$$\delta \ = \ \sin^{-1}\left(\frac{r_3 \ - \ r_{S_3}}{|||[r_1, r_2, r_3]^{\mathrm{T}} - \boldsymbol{r_S}||_2}\right), \tag{23}$$

where $||\cdot||_2$ is the 2-norm, and $\boldsymbol{r_S} = [r_{S_1}, r_{S_2}, r_{S_3}]^{\mathrm{T}}$ is the position of the telescope, which, in this work, is at the Barycenter of the system, defined to be the origin: $\boldsymbol{r_S} = [0, 0, 0]^{\mathrm{T}}$. The measurements are corrupted by additive zero-mean Gaussian white noise with standard deviation of 100 arc-seconds for both the right ascension and declination observations. Light travel time delay and measurement biases are not considered.

Initial Conditions. Each target truth and each EnGMF GMM component are initialized by the same distribution centered at the non-dimensional coordinates[31] and covariance[32]

$$x_0 \ = \ [1.0110350588,\ 0,\ -0.1731500000,\ 0,\ -0.0780141199,\ 0]^{\mathrm{T}} \tag{24}$$

$$P_0 \ = \ \mathrm{diag}([2.5e-5,\ 2.5e-5,\ 2.5e-5,\ 1e-6,\ 1e-6,\ 1e-6]^2). \tag{25}$$

Each target is spaced apart by 2.5 hours. This means when the targets are created they are propagated to be 2.5 hours apart along their trajectory. The non-dimensional period of each target is roughly 1.3632096570 from Reference 31. One key assumption this work makes is that the measurement data has already undergone some preprocessing to delineate the available tracklets. Therefore, it does not account for any simulated clutter, births, deaths, nor target spawning of the like. The last key assumption, for the following results, is that there exists the same number of tracklets and targets, although, the methods presented can handle varying numbers of tracklets and targets.

Estimation Accuracy Criterion

The hypothetical scenario above describes a process with sparse-in-time angles-only measurements. This provides a case that is not terribly far fetched from reality for optical telescopes and allows us to test the performance of our data association methods.

It has become common practice to use Optimal Subpattern Assignment (OSPA)[33] to evaluate MTT filter performance. In this work, the OSPA computes the filter error of the state estimates with-respect-to the truth and combines the errors of every target. The general OSPA takes into account different cardinalities between the number of estimated targets and number of true targets; whereas, this work assumes the same cardinality always. That said, given the set of filter estimates for T targets $\hat{X} = \{\hat{x}_1,\ \hat{x}_2,\ \cdots,\ \hat{x}_T\}$ and the set of truths for T targets $X = \{x_1,\ x_2,\ \cdots,\ x_T\}$,

this work reduces the OSPA to

$$\bar{d}_p(X, \hat{X}) \equiv \left(\frac{1}{T} \sum_{t=1}^{T} d(x_t, \hat{x}_t)^p \right)^{1/p}, \tag{26}$$

where $d(a, b)$ can be any metric as defined by Reference.[33] This work chooses $d(a, b) = ||a - b||_2$ which is the 2-norm of the difference between a and b. Additionally, the OSPA parameter $p = 2$ which is common practice. A low OSPA value means good filter and tracking accuracy.

Estimation Consistency Criterion

The filter and tracking consistency is examined using the scaled normalized estimation error squared (SNEES) which is defined as[34]

$$\text{SNEES} = \frac{1}{n_{\text{x}}} \sum_{i=1}^{\mathcal{M}} \frac{1}{\mathcal{M}} \sum_{t=1}^{T} \frac{1}{T} (\text{x}_t^{(i)} - \hat{\text{x}}_t^{(i)})^{\text{T}} (\hat{\text{P}}_t^{(i)})^{-1} (\text{x}_t^{(i)} - \hat{\text{x}}_t^{(i)}), \tag{27}$$

where $\mathcal{M}$ is the number of Monte Carlo iterations, n_{x} is the size of the state-space, T is the number of targets, $\text{x}_k^{(i)}$ are the true states, $\hat{\text{x}}_k^{(i)}$ are the estimated states, and $\hat{\text{P}}_k^{(i)}$ is the estimated covariance of the i^{th} Monte Carlo iteration for target t. The size of the state-space $n_{\text{x}} = 6$ is used to scale the NEES value.[34] A SNEES value of 1 means good filter and tracking consistency. Anything less than 1 tells that the filter is conservative and anything greater than 1 tells that the filter is too confident.

RESULTS

Simulation Results for 3 Targets

This set of results tracks 3 closely spaced targets on the same NRHO trajectory using an Earth ground-based optical telescope. The trajectories of the 3 targets are shown in Figure 6. The observations are collected over 2 months, featuring the following different NNJPDA single event computation methods:

1. **Batch-EnGMF:** single events computed between the processed batch least squares solution and the weighted sum of components of the propagated EnGMF GMM. This is a Gaussian to Gaussian comparison.

2. **Batch-EnGMF(GMM):** single events computed between the processed batch least squares solution and the individual components of the propagated EnGMF GMM. This is a Gaussian to Gaussian mixture comparison.

3. **MCMC-EnGMF:** single events computed between the weighted sum of components of the MCMC GMM and the weighted sum of components of the propagated EnGMF GMM. This is a Gaussian to Gaussian comparison.

4. **MCMC-EnGMF(GMM):** single events computed between the weighted sum of components of the MCMC GMM and the individual components of the propagated EnGMF GMM. This is a Gaussian to Gaussian mixture comparison.

5. **MCMC(GMM)-EnGMF:** single events computed between the individual components of the MCMC GMM and the weighted sum of components of the propagated EnGMF GMM. This is a Gaussian mixture to Gaussian comparison.

6. **MCMC(GMM)-EnGMF(GMM):** single events computed between the individual components of the MCMC GMM and the individual components of the propagated EnGMF GMM. This is a Gaussian mixture to Gaussian mixture comparison.

Figure 7 compares the OSPA results in [km] and [km/s]. Both batch least squares methods, (1.) Batch-EnGMF and (2.) Batch-EnGMF(GMM), are not able to capture the nonlineariteies in the measurement distribution when processing the raw tracklets of data. This leads to poor data association which then subsequently leads to an inaccurate EnGMF filter. The MCMC methods on-the-other-hand perform well and are able to handle the non-Gaussianities. For tracking 3 targets, all MCMC methods have similar accuracy performances and are significantly better and more stable than the batch least squares methods.

Next, to get an idea of overall MTT filter consistencies, Figure 8 compares the SNEES of each method. Figure 8 shows that over the course of the 2 months of sensing, the Gaussian assumption made by method (3.) MCMC-EnGMF is too strong and over time it produces estimates that are too confident. The MCMC methods that account for Gaussian mixtures during data association: (4.) MCMC-EnGMF(GMM), (5.) MCMC(GMM)-EnGMF, (6.) MCMC(GMM)-EnGMF(GMM); provide the best overall filter consistency suggesting that they are able to successfully label and track the correct targets. Furthermore, for this 3 target case, the MCMC GMM methods are the better methods in both accuracy and consistency because they are effectively using most of the information provided by the nonlinear, non-Gaussian system.

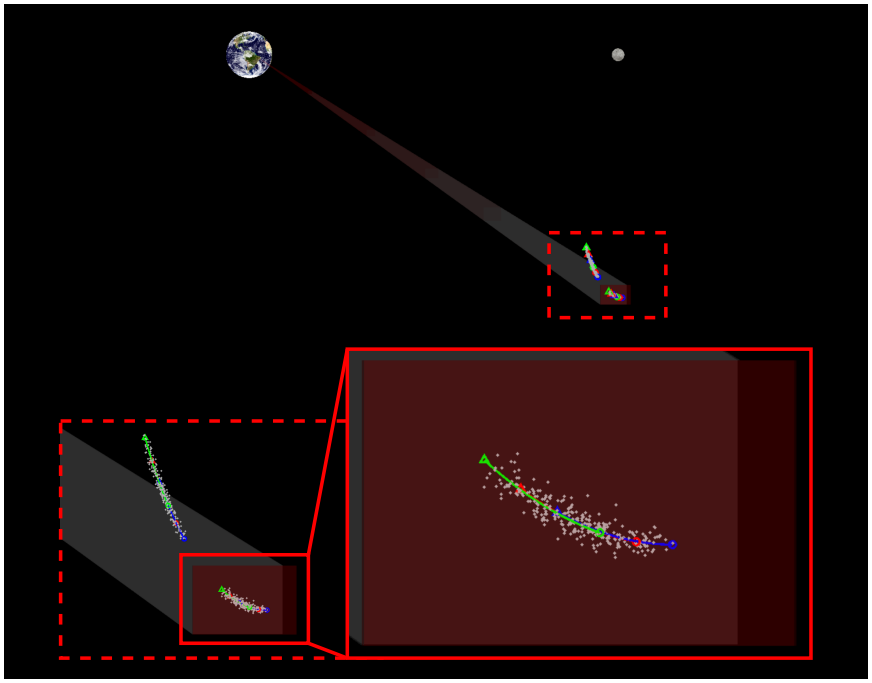

Figure 6: Two sets of 8 hour trajectories for 3 closely spaced targets in the same orbit. The true trajectories are distinguished by the red, green, blue solid lines. The tracklets of raw measurements are shown by the white markers and have not been assigned a target yet. Notice the trajectories of the targets all pass through the same volume of measurements.

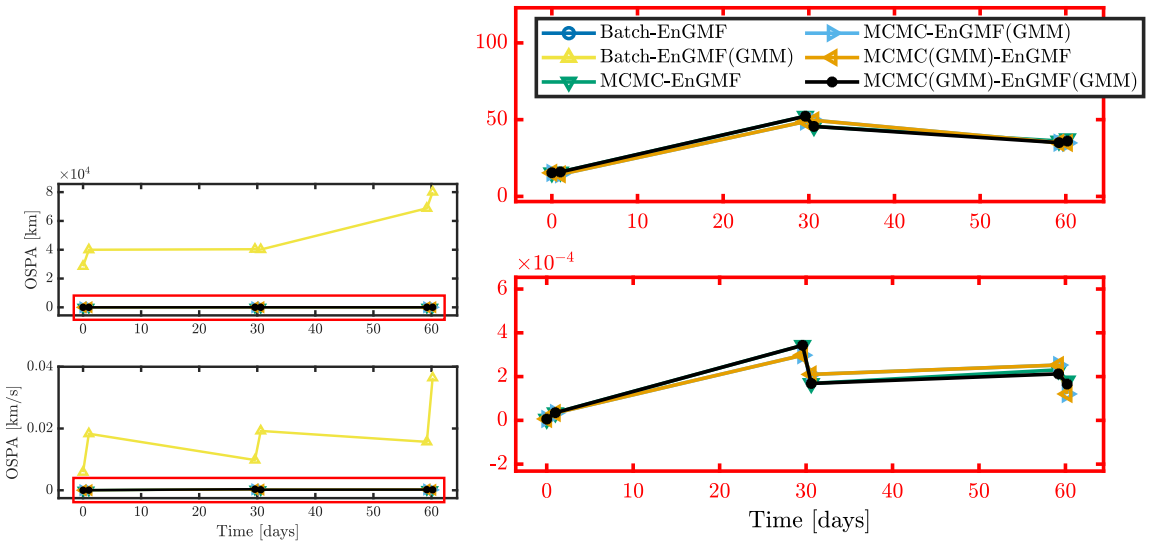

Figure 7: Comparing the OSPA position and velocity metrics for tracking 3 targets closely spaced in the same NRHO trajectory using a Earth ground-based optical telescope collecting angles-only observations over the course of 2 months. Averaged over 100 Monte Carlo iterations. The Batch-EnGMF method fails to produce numerical results and is not within reason to compare. The MCMC methods are highlighted in red and zoomed in on the right.

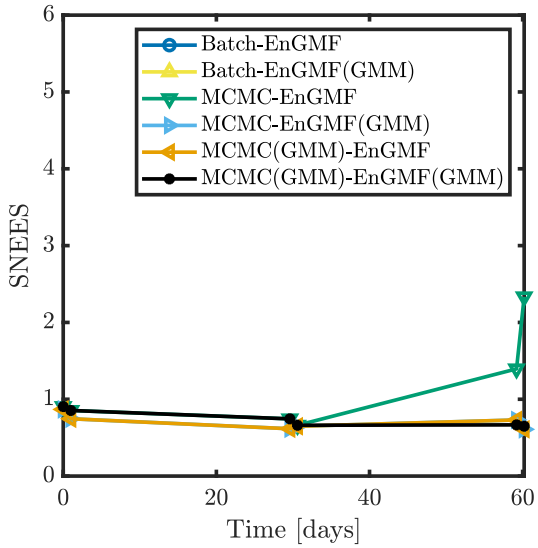

Figure 8: Comparing the SNEES for tracking 3 targets closely spaced in the same NRHO trajectory using a Earth ground-based optical telescope collecting angles-only observations over the course of 2 months. Averaged over 100 Monte Carlo iterations. The Batch-EnGMF and Batch-EnGMF(GMM) NNJPDA methods are plotted out of frame and are not within reason to compare.

Simulation Results for Varying Targets

Using the same problem setup and target spacing as the 3 target case, the following are the time-averaged results ranging from 2 targets to 10 targets. This to show how the algorithms compare as the orbit becomes more congested. Figure 9 shows that as the number of targets increases, the MCMC methods retain similar accuracies. Noticeably, the batch least squares methods have poor performance (again) and do not produce numerical results after trying to track 3 targets. Next, to show the MTT filter consistencies, Figure 10 compares the time-averaged SNEES of each method.

Figure 10 shows that over the course of the 2 months of sensing and introducing more congestion, the MCMC methods are able to provide consistent estimates implying they are able to successfully label and track the correct targets. The most consistent being the MCMC(GMM)-EnGMF(GMM) do to it's full GMM to GMM data assocaition between tracklets and targets. To see the assignment accuracy, Figure 11 shows how accurate the different methods are at assigning the correct corresponding target after data association. The accuracy is computed by counting the number of correct associations and averaging it over time, the number of Monte Carlo iterations, and by the number of true targets.

In Figure 11, the assignment accuracies of the batch least squares methods are poor and they fail to produce numerical results after adding just 3 targets to the constellation. The accuracies of the MCMC methods waiver near 100% with the most accurate being the MCMC(GMM)-EnGMF(GMM) method. These results provide insight that processing raw tracklets with MCMC methods and performing data association on Gaussian mixtures is important for accurate and consistent estimates, and the accurate assignment of targets.

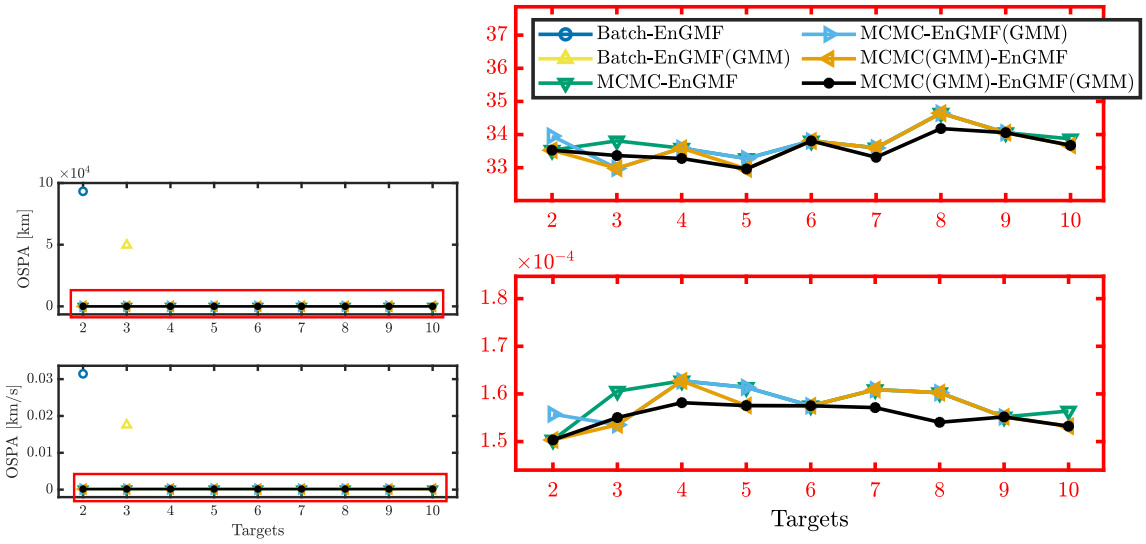

Figure 9: Comparing the OSPA position and velocity metrics for tracking 2-10 targets closely spaced in the same NRHO trajectory using a Earth ground-based optical telescope collecting angles-only observations. Averaged over 2 months and 100 Monte Carlo iterations. The Batch-EnGMF and Batch-EnGMF(GMM) methods both fail to produce numerical results after just 3 targets and are not within reason to compare. The MCMC methods are highlighted in red and zoomed in on the right.

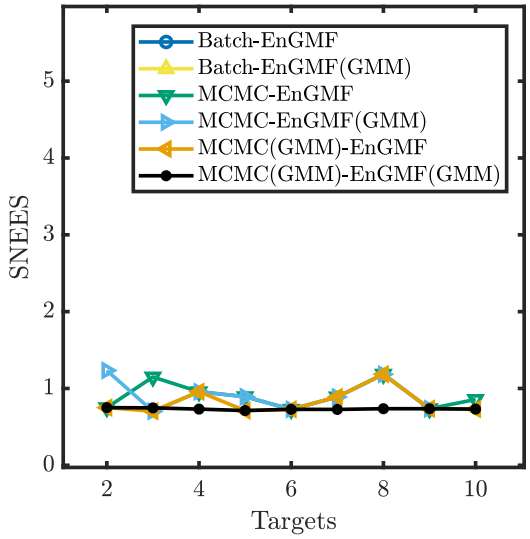

Figure 10: Comparing the SNEES for tracking 2-10 targets closely spaced in the same NRHO trajectory using a Earth ground-based optical telescope collecting angles-only observations. Averaged over 2 months and 100 Monte Carlo iterations. The Batch-EnGMF and Batch-EnGMF(GMM) NNJPDA methods are plotted out of frame and are not within reason to compare.

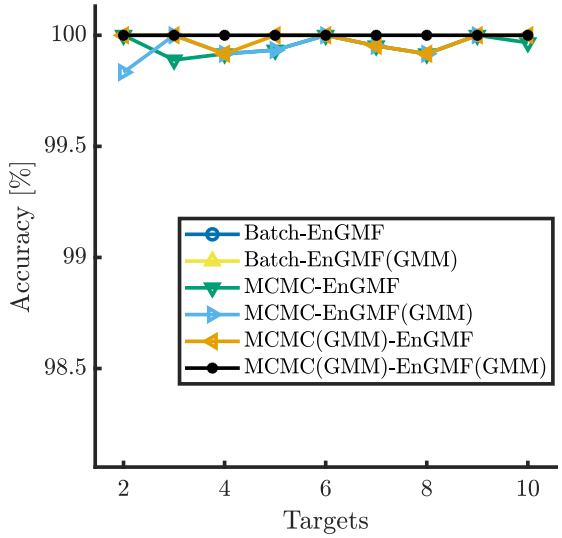

Figure 11: Comparing the assignment accuracy for tracking 2-10 targets closely spaced in the same NRHO trajectory using a Earth ground-based optical telescope collecting angles-only observations. Averaged over time and by 100 Monte Carlo iterations. After adding 3 targets, the batch least squares methods fail to produce numerical results.

CONCLUSION

This work presents an approach to address the intricate challenges of MTT in cislunar orbits utilizing angles-only tracklets from ground-based optical telescopes. By leveraging MCMC methods, this work processes angles-only tracklets for Cartesian coordinates, providing a more suitable alternative to traditional methods in the presence of nonlinearities. In addition, JPDA is performed between the measurement components of the MCMC generated GMM and the filter components of the EnGMF GMM. JPDA in this work uses a nearest neighbor greedy approach, preventing duplicate target-tracklet assignments.

The empirical results, showcased through numerical simulations tracking multiple closely spaced targets on a NRHO trajectory, underscore the superiority of MCMC methods over traditional batch processing approaches. The findings reveal the effectiveness of MCMC-based MTT methods in accurately associating tracklets with targets, even in scenarios with increased orbital congestion. The proposed methods maintain accuracy and consistency over time, thereby offering valuable insights for practical implementations in real-world cislunar tracking scenarios.

This research opens avenues for further exploration and refinement of MCMC-based methodologies in improving the robustness and efficiency of space target tracking. Future work in this area is to expand the research to other data association methods like Random Finite Sets, leveraging the the technique of processing data via MCMC.

ACKNOWLEDGMENT

This material is based on research sponsored by the Air Force Office of Scientific Research (AFOSR) under agreement number FA9550-23-1-0646, *Create the Future Independent Research Effort (CFIRE)*.

REFERENCES

[1] J. T. Horwood, A. B. Poore, and K. T. Alfriend, "Orbit determination and data fusion in GEO," *Proc. of the 2011 AMOS Conference,(Wailea, HI)*, 2011.

[2] D. Durant, A. A. Popov, and R. Zanetti, "MCMC EnGMF for Sparse Data Orbit Determination," *Astrodynamics Specialist Conference, Big Sky, MT*, No. 23(356), AAS/AIAA, 2023.

[3] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The journal of chemical physics*, Vol. 21, No. 6, 1953, pp. 1087–1092.

[4] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, 1970.

[5] N. Moretti, M. Rutten, T. Bessell, and B. Morreale, "Autonomous space object catalogue construction and upkeep using sensor control theory," *Proc. the Advanced Maui Optical and Space Surveillance Technologies Conf.(AMOS)*, 2017.

[6] K. Muinonen, G. Fedorets, H. Pentikäinen, T. Pieniluoma, D. Oszkiewicz, M. Granvik, J. Virtanen, P. Tanga, F. Mignard, J. Berthier, *et al.*, "Asteroid orbits with Gaia using random-walk statistical ranging," *Planetary and Space Science*, Vol. 123, 2016, pp. 95–100.

[7] L. Schlenker, A. J. Sinclair, and R. Linares, "Angles-Only Orbit Determination Using Hamiltonian Monte Carlo," *2018 Space Flight Mechanics Meeting*, 2018, p. 1975.

[8] J. Africano, P. Kervin, D. Hall, P. Sydney, J. Ross, T. Payne, S. Gregory, K. Jorgensen, K. Jarvis, T. Parr-Thumm, *et al.*, "Understanding photometric phase angle corrections," *Proceedings of the 4th European Conference on Space Debris*, Vol. 587, 2005, pp. 141–146.

[9] J. L. Anderson and S. L. Anderson, "A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts," *Monthly weather review*, Vol. 127, No. 12, 1999, pp. 2741–2758.

[10] S. Yun, R. Zanetti, and B. A. Jones, "Kernel-based ensemble gaussian mixture filtering for orbit determination with sparse data," *Advances in Space Research*, Vol. 69, No. 12, 2022, pp. 4179–4197.

[11] A. A. Popov and R. Zanetti, "Ensemble gaussian mixture filtering with particle-localized covariances," *2023 26th International Conference on Information Fusion (FUSION)*, IEEE, 2023, pp. 1–7.

[12] A. A. Popov and R. Zanetti, "An Adaptive Covariance Parameterization Technique for the Ensemble Gaussian Mixture Filter," *arXiv preprint arXiv:2212.10323*, 2022.

[13] A. A. Popov and R. Zanetti, "Ensemble-localized Kernel Density Estimation with Applications to the Ensemble Gaussian Mixture Filter," *arXiv preprint arXiv:2308.14143*, 2023.

[14] B. L. Reifler, S. Yun, B. A. Jones, and R. Zanetti, "Multi-target ensemble Gaussian mixture tracking with sparse observations," *AMOS Conf. Proc*, 2021.

[15] B. L. Reifler, A. A. Popov, B. A. Jones, and R. Zanetti, "Large-Scale Space Object Tracking in a Proliferated LEO Scenario," *2023 26th International Conference on Information Fusion (FUSION)*, IEEE, 2023, pp. 1–8.

[16] B. W. Silverman, *Density estimation for statistics and data analysis*, Vol. 26. CRC press, 1986.

[17] S. Yun and R. Zanetti, "Sequential monte carlo filtering with gaussian mixture sampling," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 9, 2019, pp. 2069–2077.

[18] A. Gelb *et al.*, *Applied optimal estimation*. MIT press, 1974.

[19] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *Signal processing, sensor fusion, and target recognition VI*, Vol. 3068, Spie, 1997, pp. 182–193.

[20] R. v. d. Merwe and E. Wan, *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*. PhD thesis, The faculty of the OGI School of Science & Engineering at Oregon ..., 2004.

[21] J. Roecker and G. Phillis, "Suboptimal joint probabilistic data association," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, No. 2, 1993, pp. 510–517.

[22] Y. Bar-Shalom, *Multitarget-multisensor tracking: Advanced applications*. Norwood, 1990.

[23] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and data fusion*, Vol. 11. YBS publishing Storrs, CT, USA:, 2011.

[24] F. Bourgeois and J.-C. Lassalle, "An extension of the Munkres algorithm for the assignment problem to rectangular matrices," *Communications of the ACM*, Vol. 14, No. 12, 1971, pp. 802–804.

[25] D. Drummond, D. Castanon, and M. Bellovin, "Comparison of 2-D Assignment Algorithms for Sparse," *Rectangular, Floating Point Cost Matrices, self-published*, 1990.

[26] P. D. Konstantinova, A. Udvarev, and T. Semerdjiev, "A study of a target tracking algorithm using global nearest neighbor approach.," *Compsystech*, Vol. 3, 2003, pp. 290–295.

[27] B. T. B. Schutz, G. Born, *Statistical Orbit Determination*. Elsevier Academic Press, 2004.

[28] R. M. Neal *et al.*, "MCMC using Hamiltonian dynamics," *Handbook of markov chain monte carlo*, Vol. 2, No. 11, 2011, p. 2.

[29] G. R. Lussky and United States National Weather Service, Western Region, and United States National Oceanic and Atmospheric Administration, "Sunrise/Sunset and Moonrise/Moonset," Tech. Rep. 51, NOAA, 1986.

[30] J. R. Dormand and P. J. Prince, "A family of embedded Runge-Kutta formulae," *Journal of computational and applied mathematics*, Vol. 6, No. 1, 1980, pp. 19–26.

[31] E. M. Z. Spreen, *Trajectory design and targeting for applications to the exploration program in cislunar space*. PhD thesis, Purdue University, 2021.

[32] S. Boone, O. Boodram, and J. McMahon, "Efficient Nonlinear Filtering Methods for Near Rectilinear Halo Orbit Navigation," *AAS Conference*, 2023.

[33] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE transactions on signal processing*, Vol. 56, No. 8, 2008, pp. 3447–3457.

[34] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2001.