Acta Astronautica 189 (2021) 514–529

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Research paper

Autonomous angles-only multitarget tracking for spacecraft swarms ☆

Justin Kruger *, Simone D'Amico

Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, 94305, USA

ARTICLE INFO

Keywords:
Navigation
Multitarget tracking
Distributed space systems
Angles-only

ABSTRACT

This paper presents a new algorithm for autonomous multitarget tracking of resident space objects using optical angles-only measurements from a spaceborne observer. To enable autonomous angles-only navigation of spacecraft swarms, observers must identify and track multiple known or unknown target space objects in view, without reliance on a-priori relative orbit knowledge. Extremely high tracking precision is necessary despite low measurement frequencies and limited computational resources. The new 'Spacecraft Angles-only MUltitarget tracking System' (SAMUS) algorithm has been developed to meet these objectives and constraints. It combines domain-specific modeling of target kinematics with multi-hypothesis techniques to autonomously track multiple unknown targets using only sequential camera images. A measurement transform ensures that target motion in the observer reference frame follows a consistent parametric model; curve fitting is used to predict track behavior; and kinematic track gating and scoring criteria improve the efficiency and accuracy of the multi-hypothesis approach. Monte Carlo testing with high-fidelity simulations demonstrates close to 100% data association precision and high recall across a range of multi-spacecraft formations, in both near-circular and eccentric orbits. Tracking is maintained in the presence of eclipse periods, significant measurement noise, and partially known swarm maneuvers. A comparison to other tracking algorithms reveals strong advantages in precision, robustness and computation time, crucial for spaceborne angles-only navigation.

1. Introduction

Distributed space systems can offer many advantages over traditional monolithic spacecraft, including improved coverage, costs, scalability, flexibility and robustness [1,2]. However, their navigation presents significant challenges, especially in the context of deep space missions aiming to navigate primarily autonomously using only on-board resources. For spacecraft swarms operating at separations of several kilometers to several thousand kilometers, a favorable solution is angles-only navigation, in which observer spacecraft obtain bearing angle measurements to targets using an on-board vision-based sensor (VBS). Cameras are advantageous as they are robust, low-cost, low-power sensors already present on most spacecraft. They possess high dynamic range capabilities and small form factors conducive to both accurate navigation and swarm miniaturization. Many distributed space system proposals therefore present angles-only navigation as a key aspect, with applications to distributed science [3,4], space situational awareness [5], deep space communications [6], autonomous rendezvous [7,8], and on-orbit servicing [9,10].

A number of studies have explored relative orbit determination (ROD) and state estimation for spacecraft using angles-only measurements. Woffinden et al. [11] and Gaias et al. [12] apply linearized rectilinear relative motion and relative orbital elements (ROE) respectively, and conclude that the linearized angles-only ROD problem is not fully observable due to a lack of explicit range information. Conducting maneuvers is suggested to improve range observability but is non-ideal because navigation and control become coupled. Sullivan et al. [13,14] subsequently develop a maneuver-free procedure for angles-only navigation. Improved state estimation is achieved by leveraging nonlinear perturbed orbit dynamics and orbit curvature effects. Others have explored this problem from the perspective of angles-only initial relative orbit determination (IROD). Gong et al. use the offset between the spacecraft VBS and center of mass to improve range observability [15,16], while Koenig et al. employ a numerical sampling approach [17]. Recently, the Absolute and Relative Trajectory Measurement System (ARTMS) [4,14,18] has combined angles-only ROD and IROD into a complete angles-only swarm navigation architecture. ARTMS leverages multiple observers with VBS to each navigate subsets of swarm targets in a distributed manner. It will be flight-tested on board the 2022 NASA Starling mission [6], which consists of four CubeSats in low Earth orbit (LEO) and aims to be the first in-orbit demonstration of autonomous angles-only swarm navigation.

☆ This research has been supported by the NASA Small Spacecraft Technology Program cooperative agreement number 80NSSC18M0058.
* Corresponding author.
E-mail addresses: jjkruger@stanford.edu (J. Kruger), damicos@stanford.edu (S. D'Amico).

https://doi.org/10.1016/j.actaastro.2021.08.049
Received 29 April 2021; Received in revised form 20 July 2021; Accepted 27 August 2021
Available online 2 September 2021

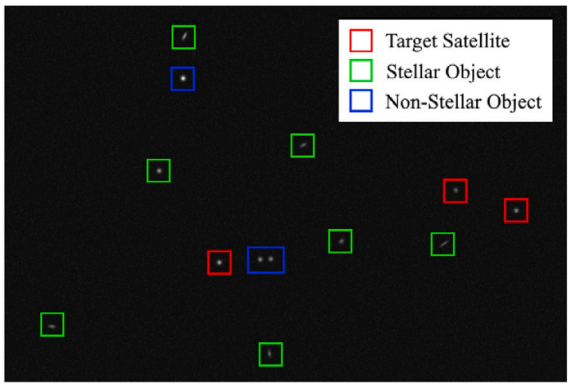

Fig. 1. A synthetic VBS image with point sources labeled.

To successfully conduct angles-only navigation, the observer must form sets of bearing angle measurements over time which correspond to its desired target(s). This underlying task is referred to as data association, and the combination of data association and sequential estimation is known as target tracking. In single target tracking, the observer extracts measurements of one desired target from extraneous measurement clutter, noise, and other false detections. In multitarget tracking (MTT), multiple targets may be simultaneously present and separate measurement tracks must be formed for each [19–21]. Single target tracking is generally less challenging because there is less ambiguity when associating measurements for one target. MTT is more complex due to presence of multiple (often unknown) targets, but is consequently more flexible. Depending on swarm geometry, multiple targets may be present in the VBS field of view (FOV), and architectures such as ARTMS assume that multiple targets can be tracked by each observer. MTT is therefore necessary for angles-only navigation of spacecraft swarms. However, in existing literature, this task is often ignored. Most spaceborne angles-only navigation frameworks consider a single target, and in cases where multiple targets are present [13,22], data association is assumed solved.

Consider identifying multiple targets amongst luminous spots in a VBS image as shown in Fig. 1. Star identification algorithms can remove known stars from consideration, but there may also be stellar objects (SO) not in the on-board star catalog; non-stellar objects (NSO) such as other satellites or debris; or sensor defects such as hotspots. Target state estimates can be used for identification but their existence or quality is not guaranteed, and state initializations often possess significant uncertainty such that multiple bearing angles are target candidates. Furthermore, errors in data association compound errors in the state estimate, and vice versa. In the far range case (≥1 km separation) considered in this paper, it is also impossible to use visual appearance for identification. MTT and data association is therefore necessary to enable angles-only spacecraft swarm navigation, and must be achieved without a-priori target state information if autonomous or self-initializing navigation is desired.

In terrestrial contexts, a variety of MTT algorithms see frequent usage, including global nearest neighbor (GNN); joint probabilistic data association (JPDA); multi-hypothesis tracking (MHT); and random finite set (RFS) methods, commonly in the form of a probability hypothesis density (PHD) filter [19–21,23]. When considered for spaceborne angles-only tracking, each has advantages and disadvantages. GNN is simple but susceptible to poor performance when targets are not well-separated [20]. JPDA, though demonstrably accurate in many scenarios, generally assumes a known number of targets [21]. Both approaches are then non-ideal in that targets may not be well-separated and the number of visible targets may be unknown. MHT is a theoretically optimal approach that performs well for low signal-to-noise ratios. However, it relies on forming increasing numbers of target track hypotheses such that heuristic hypothesis pruning is necessary for reasonable computation [20]. This is particularly challenging for low-powered spacecraft processors. RFS techniques are newer, with many promising varieties seeing continued development [19]; conversely, they are somewhat less proven and approximations are needed for real-time usage. Finally, although machine learning approaches for data association have become increasingly popular, there are difficulties in generating training data representative of the space environment [24].

Spaceflight also places stringent requirements on performance. Angles-only navigation exploits small nonlinearities to estimate spacecraft states [13] and is particularly sensitive to measurement errors. Close to 100% data association precision is necessary for risk-averse on-orbit applications, and onboard resources are limited such that high computational efficiency is needed. Measurement frequencies are low, on the order of minutes between images, and optical measurements imply large data gaps when targets are in eclipse. Relevant hardware-limited examples in terrestrial robotics are presented by Cano et al. [25], who employ a Gaussian mixture PHD filter to track several robots with a single camera, and Farazi et al. [26], who employ a neural-network-based pipeline running in real time on an observer robot. Both display promising accuracy, but rely on more detailed, higher-frequency imagery than is typical for spaceborne cameras [6]. A in-orbit example is given by LeGrand et al. [27] who suggest using a cardinalized PHD filter to track nearby NSO from an inspector spacecraft. However, they apply range information from stereo imagery which may not always be obtainable.

Two prior flight experiments have conducted angles-only navigation with single target tracking. In 2012, the ARGON experiment [7] enabled the rendezvous of two smallsats in LEO from inter-satellite separations of 30 km to 3 km. To identify the target, bearing angles in successive images were linked by finding similarities in the sizes and positions of their associated pixel clusters. The target was assumed to move significantly less than other objects, and its measurement track was chosen as that displaying the largest difference from the average linked motion. In 2016, the AVANTI [8] experiment performed a rendezvous of one smallsat and one picosat from separations of 13 km to 50 m. Unlike ARGON, which utilized ground-in-the-loop elements, AVANTI operated primarily autonomously. The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm was applied for target identification. In sets of successive images, DBSCAN identified the target as a cluster of multiple measurements within a small radius. Target tracks were expected to display curving motion so measurements with outlying residuals after Bezier curve fitting were rejected. While ARGON and AVANTI proved successful, their data association methods assume a single target and cannot be applied to multiple targets without significant modification.

In response to these limitations, this paper develops a new angles-only MTT algorithm suitable for spacecraft swarms. Henceforth the algorithm is referred to as the 'Spacecraft Angles-only MUltitarget tracking System' (SAMUS). SAMUS considers tracking from the perspective of a single observer tracking multiple targets. In a swarm with multiple distributed observers, each observer independently runs SAMUS to track those targets in its FOV. SAMUS is agnostic to orbit eccentricity and requires only (1) coarse absolute orbit knowledge of the observer and (2) knowledge of the magnitudes and execution times of swarm maneuvers (but not which maneuvers correspond to which targets). No knowledge of the number of targets or relative orbits is needed. SAMUS fuses the inspirations of ARGON and AVANTI with an MHT framework to enable multitarget tracking, and leverages kinematic modeling of relative orbits to enhance robustness and reduce computational complexity. In this fashion SAMUS overcomes the limitations of existing tracking algorithms to meet the tight requirements of general autonomous swarm operations in orbit.

Consequently, SAMUS presents the following theoretical contributions to angles-only target tracking in space. First, it is observed that target motion follows a known parametric model in the observer's

reference frame, defined in terms of observer absolute orbit elements (OE) and target relative orbit elements (ROE). Although perturbing forces modify this motion, the relative proximity of targets means they are similarly affected; thus, perturbation effects can be approximately cancelled by differencing the motion of separate targets. Second, this parametric model is fitted to target bearing angle tracks using linear system techniques. Future target measurements can then be accurately predicted. Third, a set of kinematic criteria is developed which target tracks must fulfill to be valid, and target tracks are scored by how well they match kinematic expectations. These criteria enable efficient removal of unlikely data associations and robust selection of likely associations for MHT. Fourth, maneuvers are assigned to tracks by modeling expected changes in kinematic behavior. Combined, these techniques enable angles-only tracking of multiple targets in orbit. SAMUS is validated with rigorous, high-fidelity test suites, comprising of Monte Carlo simulations using both synthetic measurements and hardware-in-the-loop (HIL) imagery. It can be integrated with state estimation frameworks to form a complete autonomous swarm navigation solution and will be flight tested in this form aboard the NASA Starling mission in 2022 [6,18].

Following this introduction, Section 2 discusses the mathematical background of target relative orbit behavior and MHT. The detailed processes and reasoning behind SAMUS are highlighted in Section 3, followed by performance testing and validation in Section 4. Section 5 presents concluding remarks.

2. Background

2.1. Coordinate frames

To obtain measurements, the observer spacecraft processes VBS images to compute time-tagged bearing angles to objects in its FOV. Bearing angles consist of azimuth and elevation $(\alpha, \epsilon)^\top$ and subtend the line-of-sight (LOS) vector $\delta\boldsymbol{r}^{\mathcal{V}} = (\delta r_x^{\mathcal{V}}, \delta r_y^{\mathcal{V}}, \delta r_z^{\mathcal{V}})^\top$ from the observer to its target. Superscript $\mathcal{V}$ indicates that the vector is described in the observer VBS coordinate frame. $\mathcal{V}$ consists of orthogonal basis vectors $\hat{\boldsymbol{x}}^{\mathcal{V}}, \hat{\boldsymbol{y}}^{\mathcal{V}}, \hat{\boldsymbol{z}}^{\mathcal{V}}$ where $\hat{\boldsymbol{z}}^{\mathcal{V}}$ is aligned with the camera boresight and $\hat{\boldsymbol{z}}^{\mathcal{V}} = \hat{\boldsymbol{x}}^{\mathcal{V}} \times \hat{\boldsymbol{y}}^{\mathcal{V}}$. This relates to bearing angles via [13]

$$\begin{bmatrix}\alpha\\ \epsilon\end{bmatrix}^{\mathcal{V}} = \begin{bmatrix}\arcsin\left(\delta r_y^{\mathcal{V}}/\|\delta\boldsymbol{r}^{\mathcal{V}}\|_2\right)\\ \arctan\left(\delta r_x^{\mathcal{V}}/\delta r_z^{\mathcal{V}}\right)\end{bmatrix} \tag{1}$$

In this work, the primary observer reference frame is the observer radial/along-track/cross-track (RTN) frame $\mathcal{R}$. It is centered on and rotates with the observer and consists of orthogonal basis vectors $\hat{\boldsymbol{x}}^{\mathcal{R}}$ (directed along the observer's absolute position vector); $\hat{\boldsymbol{z}}^{\mathcal{R}}$ (directed along the observer's orbital angular momentum vector); and $\hat{\boldsymbol{y}}^{\mathcal{R}} = \hat{\boldsymbol{z}}^{\mathcal{R}} \times \hat{\boldsymbol{x}}^{\mathcal{R}}$ [28]. Similarly, define a frame $\mathcal{W}$ using $\hat{\boldsymbol{y}}^{\mathcal{W}}$ (directed along the observer's velocity vector); $\hat{\boldsymbol{z}}^{\mathcal{W}} = \hat{\boldsymbol{z}}^{\mathcal{R}}$; and $\hat{\boldsymbol{x}}^{\mathcal{W}} = \hat{\boldsymbol{y}}^{\mathcal{W}} \times \hat{\boldsymbol{z}}^{\mathcal{W}}$. $\mathcal{W}$ only differs from $\mathcal{R}$ by a rotation of the observer flight path angle ϕ_f about $\hat{\boldsymbol{z}}^{\mathcal{R}}$, with $\phi_f \approx 0$ in near-circular orbits [28].

Typical angles-only navigation scenarios present targets with large separations in the velocity or anti-velocity directions [4–8]. Thus, when defining the tracking frame $\mathcal{T}$ in which MTT is performed, a natural choice is to align its basis vector $\hat{\boldsymbol{z}}^{\mathcal{T}}$ with the observer's velocity or anti-velocity direction $\pm\hat{\boldsymbol{y}}^{\mathcal{W}}$. Consequently, $\hat{\boldsymbol{y}}^{\mathcal{T}}$ is aligned with the observer's orbital angular momentum vector and $\hat{\boldsymbol{x}}^{\mathcal{T}} = \hat{\boldsymbol{y}}^{\mathcal{T}} \times \hat{\boldsymbol{z}}^{\mathcal{T}}$. $\mathcal{T}$ then differs from $\mathcal{W}$ by a rotation of +90° about $\hat{\boldsymbol{x}}^{\mathcal{W}}$. For convenience, we align the VBS frame $\mathcal{V}$ with $\mathcal{T}$; otherwise, LOS vectors in $\mathcal{V}$ can be rotated into $\mathcal{T}$ using rotation matrices with respect to the Planet-Centered Inertial (PCI) frame $\mathcal{P}$, as per

$$\delta\boldsymbol{r}^{\mathcal{T}} = {}^{\mathcal{W}}\overrightarrow{\boldsymbol{R}}^{\mathcal{T}}\ {}^{\mathcal{P}}\overrightarrow{\boldsymbol{R}}^{\mathcal{W}}\ {}^{\mathcal{V}}\overrightarrow{\boldsymbol{R}}^{\mathcal{P}}\ \delta\boldsymbol{r}^{\mathcal{V}} \tag{2}$$

where ${}^{\mathcal{A}}\overrightarrow{\boldsymbol{R}}^{\mathcal{B}}$ denotes a rotation from frame $\mathcal{A}$ into frame $\mathcal{B}$. ${}^{\mathcal{W}}\overrightarrow{\boldsymbol{R}}^{\mathcal{T}}$ is known from geometry; ${}^{\mathcal{P}}\overrightarrow{\boldsymbol{R}}^{\mathcal{W}}$ is known if the observer's absolute orbit is being estimated; and ${}^{\mathcal{V}}\overrightarrow{\boldsymbol{R}}^{\mathcal{P}}$ is computed by performing star identification and attitude determination with the VBS [7]. Fig. 2 depicts the relationships between frames and measurements.

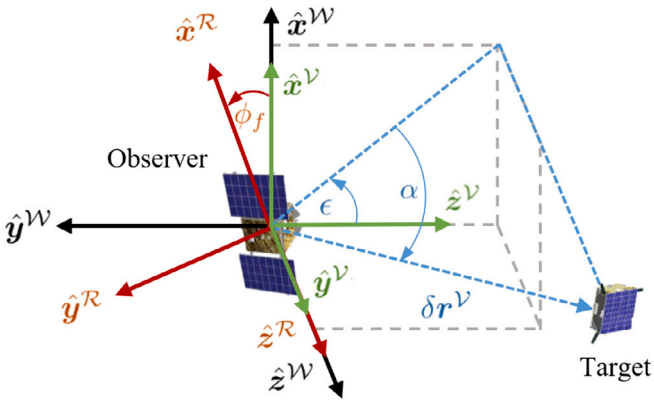

Fig. 2. Observer coordinate frames $\mathcal{W}$ (black), $\mathcal{R}$ (red) and $\mathcal{V}$ (green). Bearing angles $(\alpha, \epsilon)^\top$ and target LOS vector $\delta\boldsymbol{r}^{\mathcal{V}}$ are defined with respect to $\mathcal{V}$. Tracking frame $\mathcal{T}$ is aligned with $\mathcal{V}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To capture orbit curvature with improved accuracy, the relative positions of targets with respect to an observer can be described in curvilinear coordinates $\delta\boldsymbol{r}^{\mathcal{R}}_{\text{curv}} = (\delta r, a\Theta, a\Phi)$. Here, $\delta r, \Theta, \Phi$ are differences in orbit radii, angular in-plane separations and angular out-of-plane separations respectively, for observer semimajor axis a [29]. Hereafter the curvilinear representation is used, which can be mapped back to rectilinear coordinates via

$$\delta\boldsymbol{r}^{\mathcal{R}}_{\text{rect}} = \begin{bmatrix}(a+\delta r)c_\Theta c_\Phi - a\\ (a+\delta r)s_\Theta c_\Phi\\ (a+\delta r)s_\Phi\end{bmatrix} \tag{3}$$

c and s denote cosine and sine of the subscripted argument.

2.2. Relative orbit dynamics

To discuss the dynamic behavior of targets, this paper applies a quasi-nonsingular ROE state representation. The ROE are defined in terms of the OE of the observer and target (denoted by subscripts 'o' and 't' respectively) as [30]

$$\delta\boldsymbol{x}_{\text{roe}} = \begin{bmatrix}\delta a\\ \delta\lambda\\ \delta e_x\\ \delta e_y\\ \delta i_x\\ \delta i_y\end{bmatrix} = \begin{bmatrix}\delta a\\ \delta\lambda\\ |\delta\boldsymbol{e}|\, c_\phi\\ |\delta\boldsymbol{e}|\, s_\phi\\ |\delta\boldsymbol{i}|\, c_\theta\\ |\delta\boldsymbol{i}|\, s_\theta\end{bmatrix} = \begin{bmatrix}(a_t - a_o)/a_o\\ (u_t - u_o) + c_{i_o}(\Omega_t - \Omega_o)\\ e_t c_{\omega_t} - e_o c_{\omega_o}\\ e_t s_{\omega_t} - e_o s_{\omega_o}\\ i_t - i_o\\ s_{i_o}(\Omega_t - \Omega_o)\end{bmatrix} \tag{4}$$

Above, a, e, i, Ω, ω and M are the classical Keplerian OE and $u = M + \omega$ is the mean argument of latitude. The ROE consist of δa, the relative semimajor axis; $\delta\lambda$, the relative mean longitude (analogous to target range); $\delta\boldsymbol{e}$, the relative eccentricity vector with magnitude δe and phase ϕ; and $\delta\boldsymbol{i}$, the relative inclination vector with magnitude δi and phase θ. This representation is singular for equatorial orbits and fully nonsingular ROE have also been developed [31]. For improved application to eccentric orbits, Sullivan et al. [13] present 'eccentric ROE' (EROE). The EROE feature a modified relative mean longitude $\delta\lambda^*$ and modified relative eccentricity vector $\delta\boldsymbol{e}^*$ (with magnitude δe^* and phase ϕ^*), defined by

$$\delta\lambda^* = \xi\delta\lambda + (1-\xi)\left(-\frac{s_{\omega_o}}{e_o}\delta e_x + \frac{c_{\omega_o}}{e_o}\delta e_y + \cot i_o \delta i_y\right) \tag{5}$$

$$\delta e_x^* = \frac{c_{\omega_o}\delta e_x + s_{\omega_o}\delta e_y}{1 - e_o^2} = \delta e^* c_{\phi^*} \tag{6}$$

$$\delta e_y^* = \frac{e_o(-\delta\lambda + \cot i_o \delta i_y) - s_{\omega_o}\delta e_x + c_{\omega_o}\delta e_y}{(1-e_o^2)^{3/2}} = \delta e^* s_{\phi^*} \tag{7}$$

$$\xi = \frac{(1+e_o^2/2)}{(1-e_o^2)^{3/2}} \tag{8}$$

The EROE reduce to the ROE for $e = 0$.

In near-circular orbits, the ROE provide geometric insight because each component corresponds to a feature of the relative motion trajectory in the RTN frame. The terms δa and $\delta\lambda$ capture mean offsets in the radial and along-track directions respectively; magnitudes of δe and δi correspond to magnitudes of in-plane (RT) and out-of-plane (RN) oscillations respectively; and phases of δe and δi dictate the orientation and aspect ratio of the relative motion ellipse in the RN plane. These oscillations possess the same frequency as the observer orbit and are shown in black in Fig. 3. In eccentric orbits, additional offsets and oscillations in the RT and RN planes are produced, defined in δe^* and $\delta\lambda^*$. These oscillations act at twice the orbit frequency and are shown in red in Fig. 3. Sullivan [13] develops a linear map between the EROE and the target's nondimensional relative position in RTN, building upon initial work by D'Amico [30], as per

$$\begin{bmatrix}\delta r_x^{\mathcal{R}} \\ \delta r_y^{\mathcal{R}} \\ \delta r_z^{\mathcal{R}}\end{bmatrix} \approx r_o \begin{bmatrix}\delta a - \frac{e_o}{2}\delta e_x^* - \delta e^*\left(c_{f_o-\phi^*} + \frac{e_o}{2}c_{2f_o-\phi^*}\right) \\ \delta\lambda^* + \delta e^*\left(2s_{f_o-\phi^*} + \frac{e}{2}s_{2f_o-\phi^*}\right) \\ \delta i s_{f_o+\omega_o-\theta}\end{bmatrix} \tag{9}$$

where f_o is the observer true anomaly and r_o is orbit radius. This mapping defines target motion in RTN at small separations and is visually represented by superimposing the black and red components in Fig. 3 [14].

For angles-only tracking, the relevant components of motion are those occurring in the image plane of the VBS. As per the earlier coordinate frame definitions, these are components $(\delta r_x^{\mathcal{R}}, \delta r_z^{\mathcal{R}})$ in Eq. (9) and Fig. 3. The elliptical aspects of this motion can be described using the traditional geometric ellipse parameters of semimajor axis a_e, semiminor axis b_e, center (x_e, y_e) and tilt γ_e via [29]

$$(x_e, y_e) = (\delta a, 0) \tag{10}$$

$$(a_e, b_e) = \left(\frac{\delta e^2 + \delta i^2 \pm \sqrt{\delta e^4 + \delta i^4 - 2\delta e^2\delta i^2 c_{2(\phi-\theta)}}}{2}\right)^{\frac{1}{2}} \tag{11}$$

$$\gamma_e = \frac{1}{2}\arctan\left(\frac{-2\delta e\delta i s_{\phi-\theta}}{\delta e^2 - \delta i^2}\right) \tag{12}$$

Target relative orbits are also affected by disturbing forces such as atmospheric drag, solar radiation pressure (SRP), third-body gravity and spherical harmonic gravity. These cause secular drifts and long- and short-period perturbations [30]. On the timescales of image-to-image tracking (i.e. $\lesssim$ 5 minutes) short-period perturbations are particularly detrimental. In LEO, the most significant perturbation is commonly J_2 Earth oblateness by several orders of magnitude [30]. Its short-period and secular effects on the ROE are [32]

$$\delta\boldsymbol{e}_{\text{sp}} = \begin{bmatrix}\delta e_{x,\text{sp}} \\ \delta e_{y,\text{sp}}\end{bmatrix} = \frac{3J_2R_P^2}{2a^2}\begin{bmatrix}(1-\frac{5}{4}s_i^2)c_u + (\frac{7}{12}s_i^2)c_{3u} \\ (1-\frac{7}{4}s_i^2)s_u + (\frac{7}{12}s_i^2)s_{3u}\end{bmatrix} \tag{13}$$

$$\delta\boldsymbol{i}_{\text{sp}} = \begin{bmatrix}\delta i_{x,\text{sp}} \\ \delta i_{y,\text{sp}}\end{bmatrix} = \frac{3J_2R_P^2}{8a^2}\begin{bmatrix}s_{2i}\,c_{2u} \\ 2c_i\,s_i\,s_{2u}\end{bmatrix} \tag{14}$$

$$\delta\boldsymbol{e}_{\text{sec}} = \begin{bmatrix}\delta e_{x,\text{sec}} \\ \delta e_{y,\text{sec}}\end{bmatrix} = \delta e\begin{bmatrix}\cos(\phi_0 + \frac{3\pi t}{2T}J_2\frac{R_P^2}{a^2}(5c_i^2-1)) \\ \sin(\phi_0 + \frac{3\pi t}{2T}J_2\frac{R_P^2}{a^2}(5c_i^2-1))\end{bmatrix} \tag{15}$$

$$\delta\boldsymbol{i}_{\text{sec}} = \begin{bmatrix}\delta i_{x,\text{sec}} \\ \delta i_{y,\text{sec}}\end{bmatrix} = \delta i\begin{bmatrix}1 \\ 1 - \frac{3\pi t}{T}J_2\frac{R_E^2}{a^2}\delta i s_i^2\end{bmatrix} \tag{16}$$

where t is time, T is the orbit period, R_P is the radius of the central body and ϕ_0 is the phase of δe at initial epoch t_0. These variations in δe and δi must be included in Eq. (9) if their effects are significant in the orbit regime of interest. Similar expressions have been derived for other forces such as drag [31].

Maneuvers by the observer or targets also affect relative motion. Consider a change in the ROE, $\Delta\delta\boldsymbol{x}_{\text{roe}}$, and a maneuver by the observer in RTN, $\delta\boldsymbol{v}_o^{\mathcal{R}} = (\delta v_x^{\mathcal{R}}, \delta v_y^{\mathcal{R}}, \delta v_z^{\mathcal{R}})^\top$. In eccentric orbits, these are related via a control input matrix $\boldsymbol{B}_{\text{roe}}$ [13,33] defined as

$$\Delta\delta\boldsymbol{x}_{\text{roe}} = \boldsymbol{B}_{\text{roe}}\delta\boldsymbol{v}_o^{\mathcal{R}} \tag{17}$$

$$\boldsymbol{B}_{\text{roe}} = -\frac{\eta}{an}\begin{bmatrix}\frac{2e}{\eta^2}s_f & \frac{2k}{\eta^2} & 0 \\ \frac{(\eta-1)kc_f - 2\eta e}{ek} & \frac{(1-\eta)(k+1)s_f}{ek} & 0 \\ s_{f+\omega} & \frac{(k+1)c_{f+\omega}+e_x}{k} & \frac{e_y s_{f+\omega}}{k\tan i} \\ -c_{f+\omega} & \frac{(k+1)s_{f+\omega}+e_y}{k} & -\frac{e_x s_{f+\omega}}{k\tan i} \\ 0 & 0 & \frac{c_{f+\omega}}{k} \\ 0 & 0 & \frac{s_{f+\omega}}{k}\end{bmatrix} \tag{18}$$

$$\eta = \sqrt{1-e^2}, \quad n = \sqrt{\mu/a^3}, \quad k = 1 + ec_f \tag{19}$$

Similarly, the change in ROE from target maneuver $\delta\boldsymbol{v}_t^{\mathcal{R}}$ is obtained via $\Delta\delta\boldsymbol{x}_{\text{roe}} = -\boldsymbol{B}_{\text{roe}}\delta\boldsymbol{v}_t^{\mathcal{R}}$. Change in the EROE, $\Delta\delta\boldsymbol{x}_{\text{eroe}}$, is computed by first computing $\Delta\delta\boldsymbol{x}_{\text{roe}}$ and then mapping to $\Delta\delta\boldsymbol{x}_{\text{eroe}}$ via Eqs. (5)–(7).

2.3. Multi-hypothesis tracking

The objective of data association is to collect sensor data containing one or more potential targets and to partition it into sets of observations – or 'tracks' – produced by the same target over time [34]. Assume that tracks have been formed from previous data and that a new set of measurements – or 'scan' – has become available. Then, a typical MTT system performs five sequential tasks:

1. Sensor data processing: retrieve new measurements.
2. Measurement prediction: use existing tracks to predict new measurements.
3. Measurement gating: assess which new measurements may reasonably be associated with which tracks.
4. Measurement-to-track association: score valid associations and determine the best option(s).
5. Track maintenance: initialize new tracks, confirm likely tracks and delete unlikely tracks.

Difficulties arise when targets are closely spaced and observations may be reasonably associated with multiple tracks. A prominent approach for handling these ambiguities is multi-hypothesis tracking. MHT applies a delayed decision philosophy by propagating and maintaining multiple association hypotheses, since future data can aid in disambiguating past associations. The operational logic is presented in Fig. 4. With each new scan, new measurements are received and are gated with respect to existing tracks. New tracks and hypotheses are formed and evaluated in terms of likelihood. Unlikely hypotheses are deleted and new measurements are predicted for surviving tracks. MHT was initially developed by Reid [35] and has since been expanded into a variety of forms [36–39].

Consider Fig. 5, in which tracks T1 (blue) and T2 (red) lead to predicted observations P1 and P2. Four measurements are received: M1, …, M4. Measurements can be associated with existing tracks if they fall within track gates, or can start a new track. Following the convention of Blackman [34], denote T3 (T1, M1) as Track 3 formed from the association of T1 and M1. Similarly, there exists T4 (T1, M2); T5 (T1, M3); T6 (T2, M2); and T7 (T2, M3). Furthermore, NT1, …, NT4 denote new tracks initiated from M1, …, M4. Tracks are 'compatible' if they have no observations in common, and MHT 'hypotheses' are composed of sets of compatible tracks. In the above example there are 10 feasible hypotheses, including H1: (T1, T2, NT1, …, NT4), H2: (T3, T6, NT3, NT4), H3: (T3, T7, NT2, NT4), and so on. Upon receiving new measurements, existing hypotheses are expanded into sets of new

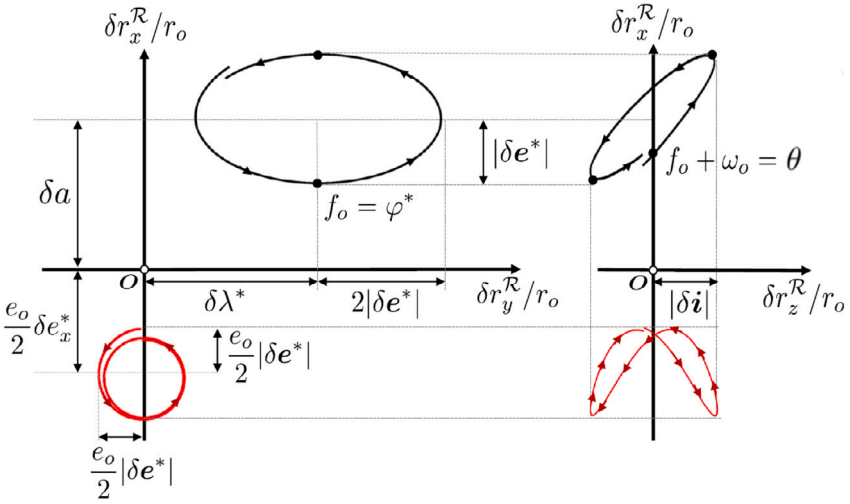

Fig. 3. Target relative motion [14] in the $\hat{x}^R$-$\hat{y}^R$ (RT) and $\hat{x}^R$-$\hat{z}^R$ (RN) planes as defined by the EROE and Eq. (9). Motion consists of components which are first-order in spacecraft separation (black) and components proportional to orbit eccentricity (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

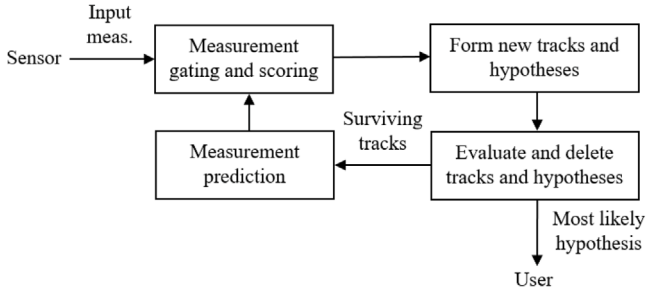

Fig. 4. MHT logic overview [34].

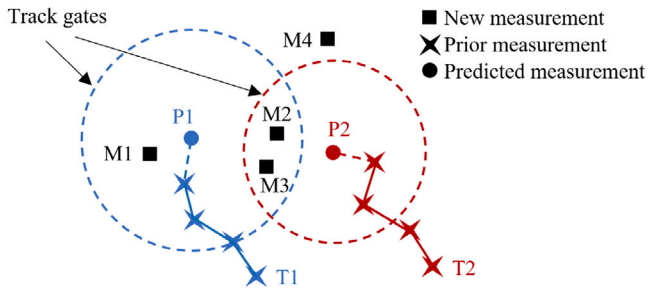

Fig. 5. A notional illustration of tracks T1 (blue) and T2 (red) and their predicted measurements and track gates. New measurements M1, …, M4 are associated to tracks via MHT. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

hypotheses by considering all valid measurement-to-track associations that maintain compatibility.

To evaluate hypotheses, MHT considers physical consistency and the probability of target presence versus false alarms. The likelihood ratio (LR) for associating data into a track is traditionally defined as [40,41]

$$\mathrm{LR} = \frac{p(D|H_1)P_0(H_1)}{p(D|H_0)P_0(H_0)} = \frac{P_T}{P_F} \tag{20}$$

Hypotheses H_1 and H_0 are the true target and false alarm hypotheses with probabilities P_T and P_F respectively. P_T assumes all track observations are of the same target, and P_F assumes all track observations are of the background. D represents data such that $p(D|H_i)$ is the probability density function evaluated with received data D under the assumption that H_1 is correct. $P_0(H_i)$ is the a-priori probability of H_i. In practice, log-likelihood ratio (LLR) is used because it directly relates to true target probability P_T via

$$\mathrm{LLR} = \ln(P_T|P_F) \implies P_T = e^{\mathrm{LLR}}/(1+e^{\mathrm{LLR}}) \tag{21}$$

The LLR is also known as the track score, and the score of a hypothesis is the sum of all constituent track scores. As outputs, MHT may provide the most likely track per target or the mean state estimate and covariance computed from all branch probabilities.

A prominent disadvantage of MHT is the potential combinatorial explosion in the number of generated tracks and hypotheses as new scans arrive [34]. Track pruning and merging must therefore be used to control growth. When describing these operations, tracks can be viewed as branches in a tree: nodes occur when tracks split into multiple hypotheses, and a tree is a set of tracks with a common root node that represents one hypothesized target. 'N-scan pruning' determines which tracks in each tree are part of the best current hypothesis (at step k) and goes back N scans (e.g. $N = 2$) to establish a new root node.

Fig. 6 presents examples of track updates for trees F1 and F2. At step $k-2$, T3 is spawned as the root of a new tree, F2. At step $k-1$, T6 is pruned for being a low-scoring track. At step k, 2-scan pruning is performed: in F1, T2 is part of the best hypothesis and becomes the new root, such that hypotheses on the left-hand branch of F1 are discarded and decisions prior to $k-2$ become final. F2 is not yet old enough to be affected by the update. New global hypotheses are formed by choosing at most one track from each tree: for example, H1 (T2, T8).

Clustering, the m-best method and track-oriented MHT are used to reduce the number of hypotheses for computation [34]. Tracks in a 'cluster' are those which can be linked through shared observations. Clusters can be independently processed because different clusters do not share measurements and by decomposing MHT into a set of smaller problems, fewer computations are required. The m-best algorithm applies Murty's method [42] for finding the m-best solutions to the association problem. It limits the number of new hypotheses formed at scan k to $m(k)$, preventing creation of many low-probability hypotheses. In track-oriented MHT, rather than maintaining and expanding hypotheses from scan-to-scan, existing hypotheses are discarded and new hypotheses are formed from tracks that survived pruning. This improves performance when there are many more hypotheses than tracks.

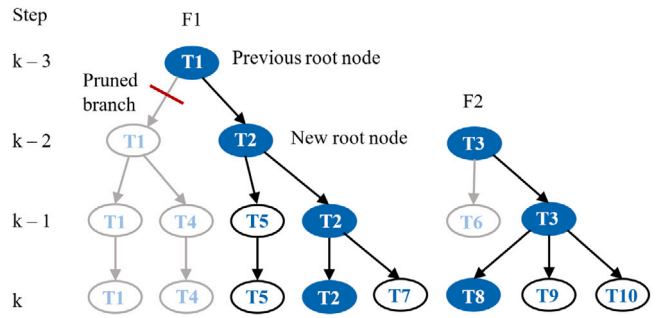

Fig. 6. Trees F1 and F2 with 2-scan pruning [34,36]. Blue denotes the best global hypothesis at step k. New root nodes are selected at $k-2$ and dissimilar branches are pruned. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. SAMUS algorithm

The SAMUS algorithm developed in this paper fuses the single-target kinematic techniques of prior flight projects with a multitarget MHT framework. SAMUS applies the core concept of MHT in that as measurements arrive, multiple tracks and hypotheses are simultaneously initialized, propagated, scored and trimmed, with the intention of robustly converging to the correct hypothesis over time. Novelty arises from the application of domain-specific knowledge to greatly improve the accuracy and efficiency of the approach. MHT is chosen as a basis because it is considered mature and demonstrably accurate [34,36] with the disadvantage of needing to heuristically trim hypotheses for real-time computation. However, the generally consistent behavior of targets in orbit provides particularly effective trimming criteria that support the delayed decision approach of MHT – as more measurements are received, target motion can be judged more conclusively to arrive at the correct associations. MHT is also able to quickly converge to a physical hypothesis, which is not always possible with other MTT methods.

Fig. 7 defines relevant geometric quantities for SAMUS in the bearing angle plane. In the upper figure, $(\alpha,\epsilon)_k$ is a track measurement at epoch k; the vector $\vec{v}_k$ is the track 'step' from $(\alpha,\epsilon)_{k-1}$ to $(\alpha,\epsilon)_k$; and ψ_k is the angle between $\vec{v}_{k-1}$ and $\vec{v}_k$. In the lower figure, d_k is the magnitude of $\vec{v}_k$ while ζ_k is its phase; $(\alpha_p,\epsilon_p)_k$ is the predicted track measurement at epoch k; and $d_{p,k},\zeta_{p,k}$ are defined as the predicted magnitude and phase of $\vec{v}_k$, respectively.

Within SAMUS, target tracks consist of successive 'segments'. Segments are separated by either (1) a hypothesized maneuver, or (2) a measurement gap such as an eclipse period. When tracking within a segment, SAMUS applies knowledge that target motion should be consistent as per Eq. (9). Upon encountering a maneuver or measurement gap, SAMUS applies expected changes in motion from Eqs. (16)–(18) to assign the next measurement. Tracking proceeds as normal during the subsequent segment. Fig. 8 presents a notional illustration of a track with three segments.

To operate, three assumptions are required by SAMUS. First, targets remain sufficiently within the VBS FOV such that consistent measurement arcs of ≥4 successive images are obtained per orbit. Second, the observer's absolute orbit is coarsely known such that the rotations between inertial and observer reference frames can be estimated and expected target visibility can be computed (e.g. expected orbit eclipse periods). Third, maneuvers by the observer and targets during the tracking period are impulsive and their execution times and RTN components are known. However, the correspondence between each specific maneuver and tracked target does not need to be known. This scenario is representative of a swarm mission with active control.

The following sections present SAMUS in detail, echoing the MTT task order from Section 2.3: (1) sensor data processing, (2) measurement prediction, (3) track gating, (4) measurement association, (5) maneuver association and (6) track maintenance.

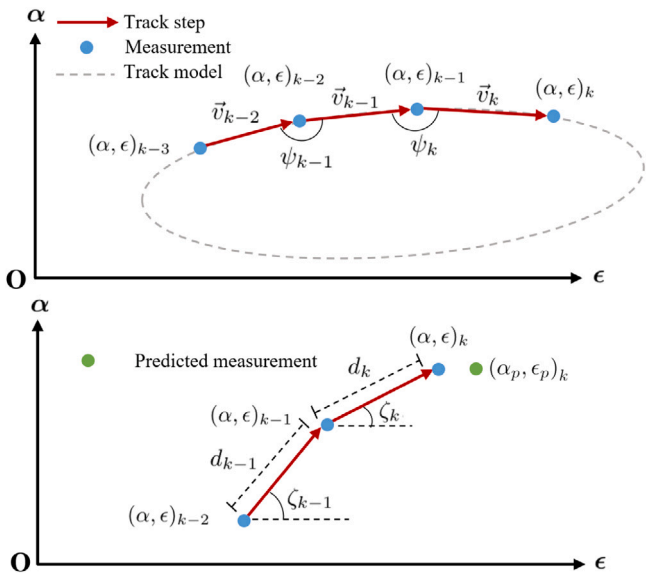

Fig. 7. Geometric target track quantities in the bearing angle plane with elevation on the x-axis and azimuth on the y-axis. $(\alpha,\epsilon)_k$ are bearing angle measurements at epoch k.

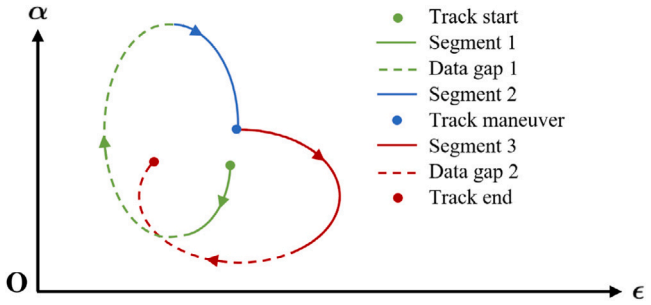

Fig. 8. Notional illustration of a track in bearing angle space with three segments. Segments are separated by data gaps or maneuvers.

3.1. Sensor data processing

A VBS image typically contains many luminous objects, including targets, other NSO, and SO. First, the raw image is simplified into list of pixel cluster centroids. Star identification algorithms such as Pyramid [43] are applied to remove SO from the list of centroids. Non-catalog SO are detected by considering objects with unit vectors in the PCI frame which remain unchanged between images. Camera hotspots are removed by considering objects with constant pixel coordinates between images. These steps are common in star tracker usage [7,8] and are not detailed here, but result in a list of bearing angles to targets and other remaining unidentified objects in the FOV.

Non-Keplerian forces (such as J_2 gravity) affect target motion and distort the parametric form of Eq. (9). To reduce these effects, SAMUS uniquely applies a measurement transform when multiple targets are present. Tracks of a target i are synchronously differenced with respect to tracks of a target j ($i \neq j$), thus using j as the virtual, moving origin of a tracking frame for i, as per

$$(\alpha,\epsilon)_i^{\mathcal{T}_{i/j}} = (\alpha,\epsilon)_i^{\mathcal{T}} - (\alpha,\epsilon)_j^{\mathcal{T}} \tag{22}$$

Frame $\mathcal{T}_{i/j}$ denotes target i viewed with respect to target j. In angles-only scenarios of interest, swarm members are in relative proximity in inertial space and affected similarly by perturbations. Thus, the origin of $\mathcal{T}_{i/j}$ and the target i measurements are affected by similar forces. In $\mathcal{T}_{i/j}$, perturbation effects are approximately cancelled and motion with

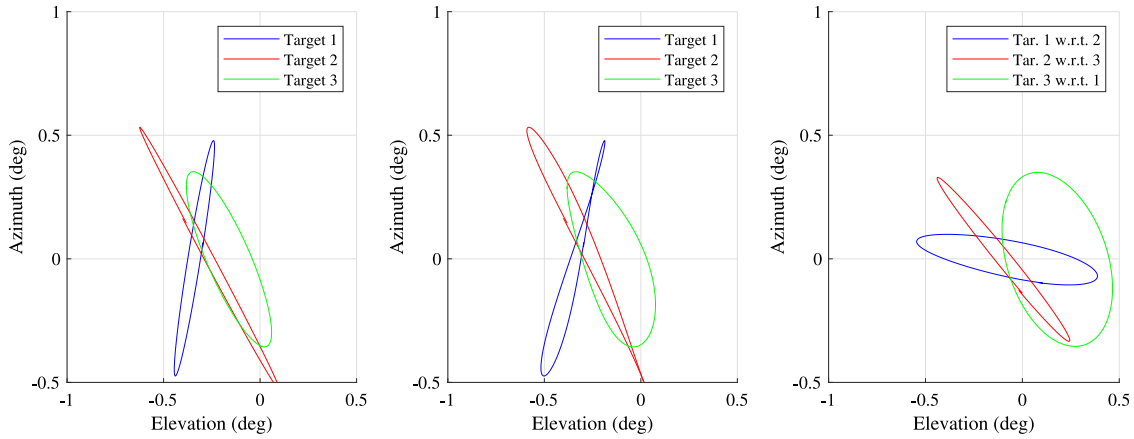

Fig. 9. Target trajectories without J_2 effects (left), with J_2 effects (center), and with J_2 effects and the measurement transform (right).

Table 1
Swarm configuration for Fig. 9.

OE	Obs.	ROE	Tar. 1	Tar. 2	Tar. 3
a (km)	6878	δa (km)	−0.1	−0.05	0
e	0.001	$\delta\lambda$ (km)	−60	−50	−40
i (°)	91	δe_x (km)	−0.05	0.5	0.15
Ω (°)	0	δe_y (km)	−0.1	0.3	0
ω (°)	0	δi_x (km)	−0.5	−0.45	−0.15
M_0 (°)	0	δi_y (km)	0.05	0.1	0.2

the form of Eq. (9) is recovered. Mathematically, consider targets i and j with similar OE such that short-period oscillations $\delta\boldsymbol{e}_{i,\mathrm{sp}} \approx \delta\boldsymbol{e}_{j,\mathrm{sp}}$ and $\delta\boldsymbol{i}_{i,\mathrm{sp}} \approx \delta\boldsymbol{i}_{j,\mathrm{sp}}$. Their $\hat{\boldsymbol{z}}^{\mathcal{R}}$ components of relative position are

$$\delta r^{\mathcal{R}}_{z,i} = (\delta i_{x_i} + \delta i_{x_i,\mathrm{sp}})s_{f_o+\omega_o} - (\delta i_{y_i} + \delta i_{y_i,\mathrm{sp}})c_{f_o+\omega_o} \tag{23}$$

$$\delta r^{\mathcal{R}}_{z,j} = (\delta i_{x_j} + \delta i_{x_j,\mathrm{sp}})s_{f_o+\omega_o} - (\delta i_{y_j} + \delta i_{y_j,\mathrm{sp}})c_{f_o+\omega_o} \tag{24}$$

$$\implies \delta r^{\mathcal{R}_{i/j}}_{z,i} = \delta r^{\mathcal{R}}_{z,i} - \delta r^{\mathcal{R}}_{z,j} \tag{25}$$

$$\approx (\delta i_{x_i} - \delta i_{x_j})s_{f_o+\omega_o} - (\delta i_{y_i} - \delta i_{y_j})c_{f_o+\omega_o} \tag{26}$$

and $\delta r^{\mathcal{R}_{i/j}}_{z,i}$ recovers the form of Eq. (9). Similar results are obtained for the $\hat{\boldsymbol{x}}^{\mathcal{R}}$ and $\hat{\boldsymbol{y}}^{\mathcal{R}}$ components of relative motion. Effects of this transform for the case in Table 1 are shown in Fig. 9.

Typical CubeSat star trackers produce bearing angle measurement noise on the order of 20″ (1σ) [44]. Applying Eq. (13) with $\delta\lambda = 100$ km, short-period distortions of almost 500″ are observed in extreme cases, well above star tracker noise. However, if formations are constrained to ROE magnitude ratios of $\delta\lambda/\delta e \geq 20$ and $\delta\lambda/\delta i \geq 20$ (such that along-track separations are dominant) for $10 \leq \delta\lambda \leq 200$ km, the maximum difference in short-period distortions between targets is less than 10″. Thus, errors after applying the transform are below expected 1σ noise. Data association errors can also become easier to distinguish in the differential frame. If measurements are swapped between Targets i and j, total error in $\mathcal{T}_{i/j}$ is the sum of both association errors because the error affects both the frame origin and the track.

3.2. Measurement prediction

After applying the differential transform, the only quickly-varying term on the right hand side of Eq. (9) is observer true anomaly f_o (or its equivalent, depending on orbit regime). Thus, even if specific ROE are unknown, target motion remains parametric in f_o with known form. Furthermore, the parameter f_k which generated each track measurement $(\alpha,\epsilon)^\top_k$ is obtainable from the observer's absolute orbit estimate, as are other relevant OE. This allows the model of Eq. (9) to be fitted to hypothesized tracks and the resulting fit to predict future track measurements. For bearing angles defined in the RTN frame, azimuth corresponds to $\hat{\boldsymbol{r}}^{\mathcal{R}}_z$ and elevation to $\hat{\boldsymbol{r}}^{\mathcal{R}}_x$. Parametric target motion in bearing angle space can therefore be written as

$$\begin{bmatrix}\epsilon\\ \alpha\end{bmatrix}^{\mathcal{R}} \approx \frac{r}{a}\begin{bmatrix} x_1 - x_2(c_{f-x_3} + \frac{e}{2}c_{2f-x_3}) \\ x_4 + x_5 s_{f+\omega-x_6}\end{bmatrix} \tag{27}$$

$$= \frac{r}{a}\begin{bmatrix} x_1 - x_2 s_{x_3}(s_f + \frac{e}{2}s_{2f}) - x_2 c_{x_3}(c_f + \frac{e}{2}c_{2f}) \\ x_4 + x_5 c_{x_6} s_{f+\omega} - x_5 s_{x_6} c_{f+\omega}\end{bmatrix} \tag{28}$$

where constants $\vec{x} = (x_1,\dots,x_6)^\top$ are scaled bearing angle equivalents of the EROE in Eq. (9). Given a set of $(\alpha,\epsilon)^\top_k$ measurements and their respective f_k, r_k, a_k, ω_k for $k = 1,\dots,n$, $\vec{x}$ can be estimated by solving a pair of separable linear systems in elevation and azimuth, via

$$\begin{bmatrix} \frac{r_1}{a_1}(c_{f_1} + \frac{e_1}{2}c_{2f_1}) & \frac{r_1}{a_1}(s_{f_1} + \frac{e_1}{2}s_{2f_1}) & \frac{r_1}{a_1} \\ \vdots & \vdots & \vdots \\ \frac{r_n}{a_n}(c_{f_n} + \frac{e_n}{2}c_{2f_n}) & \frac{r_n}{a_n}(s_{f_n} + \frac{e_n}{2}s_{2f_n}) & \frac{r_n}{a_n}\end{bmatrix}\begin{bmatrix}y_1\\ y_2\\ y_3\end{bmatrix} = \begin{bmatrix}\epsilon_1\\ \vdots\\ \epsilon_n\end{bmatrix} \tag{29}$$

$$\begin{bmatrix} \frac{r_1}{a_1}c_{f_1+\omega_1} & \frac{r_1}{a_1}s_{f_1+\omega_1} & \frac{r_1}{a_1} \\ \vdots & \vdots & \vdots \\ \frac{r_n}{a_n}c_{f_n+\omega_n} & \frac{r_n}{a_n}s_{f_n+\omega_n} & \frac{r_n}{a_n}\end{bmatrix}\begin{bmatrix}y_4\\ y_5\\ y_6\end{bmatrix} = \begin{bmatrix}\alpha_1\\ \vdots\\ \alpha_n\end{bmatrix} \tag{30}$$

where coefficients $\vec{y} = (y_1,\dots,y_6)^\top$ are related to $\vec{x}$ by

$$x_1 = y_3 \qquad x_4 = y_6$$
$$x_2 = \sqrt{y_1^2 + y_2^2} \qquad x_5 = \sqrt{y_4^2 + y_2^5}$$
$$x_3 = \mathrm{atan2}(-y_2,-y_1) \qquad x_6 = \mathrm{atan2}(-y_4, y_5)$$

Eqs. (29)–(30) are written more compactly as $\boldsymbol{A}_1\vec{y}_1 = \vec{\epsilon}$ and $\boldsymbol{A}_2\vec{y}_2 = \vec{\alpha}$ respectively. Notably, only three measurements are required to define a solution, which is well-suited to slow VBS measurement rates. Typical least squares methods such as QR decomposition can be used to solve each system and recover a target motion model in bearing angle space. Predicted $(\alpha_p,\epsilon_p)^\top_k$ in a new image is then computed using Eq. (27) and the observer absolute orbit estimate at that epoch.

Upon commencing a new track segment, there may be too few measurements to fit the model. If only one prior measurement exists, the next predicted bearing angle is simply the previous angle. If only two prior measurements exist, the predicted bearing angle is computed linearly via

$$(\alpha_p,\epsilon_p)^\top_k \approx (\alpha,\epsilon)^\top_{k-1} + [(\alpha,\epsilon)^\top_{k-1} - (\alpha,\epsilon)^\top_{k-2}] \tag{31}$$

If a separate navigation filter is present and estimating target relative states, these estimates are leveraged to predict upcoming measurements by (1) propagating filter states and covariances into the new image epoch, then (2) performing an unscented transform from the filter state space into bearing angle space [13,14].

3.3. Measurement gating

The formation of new tracks is gated such that tracks must remain physically reasonable, according to the assumption of consistent parametric motion within a track segment. SAMUS applies a set of kinematic rules to each possible measurement-to-track association and valid associations must pass all rules. The kinematic rules are:

1. Track velocities must be below a set maximum.
2. Track velocities must be consistent over time.
3. Tracks should generally not feature acute angles.
4. Tracks must turn in a consistent direction.
5. Assigned measurements must be close to predicted measurements.

Rule 1 stems from the knowledge that target relative velocity in the tracking frame depends on the ROE [30]. Objects in dissimilar orbits have large ROE and proportionally large track velocities, but in angles-only scenarios of interest [4,6–8], targets are in similar orbits to observers. Consequently, a velocity threshold can be placed on tracks depending on the maximum allowed swarm ROE, where for step $\vec{v}_k$ it is required that $d_k < d_{\max}$. For a near-circular orbit in LEO, $d_{\max} \approx 0.005$ rad/min in the bearing angle plane allows $|\delta e/\delta\lambda| \leq 0.05$ and $|\delta i/\delta\lambda| \leq 0.05$. Eccentric orbits will observe larger d_k near periapsis and high-altitude orbits will observe smaller d_k.

Rule 2 states that track velocities should remain relatively consistent between images. Velocities are constant when $e_o = 0$ and target tracks are circular, which occurs if $\delta e = \delta i$ and $\delta \boldsymbol{e} \parallel \delta \boldsymbol{i}$. Otherwise, velocity variations grow with track aspect ratio a_e/b_e (Eq. (12)) and orbit eccentricity. Two tests are applied, given by

$$\frac{1}{r_{\max}} \sum_{i=k-j}^{k-1} \frac{d_i}{j} < d_k < r_{\max} \sum_{i=k-j}^{k-1} \frac{d_i}{j} \tag{32}$$

$$\frac{1}{r_{\max}} < \frac{d_k}{d_{k-1}} < r_{\max} \tag{33}$$

These tests imply that (1) the size of the new track step must be similar to the average step size across the previous j epochs, and that (2) ratios of successive step sizes must fall within an $r_{\max}$ bound, defined by

$$r_{\max} = \left(1 + \frac{a_e}{2b_e} + \frac{10\sigma_{\text{VBS}}}{d_{\text{mean}}}\right)\left(1 + e_o\right) \tag{34}$$

where σ_{VBS} is the bearing angle measurement noise of the VBS (1σ) and d_{mean} is the mean of $d_{1,\ldots,k-1}$. Thus, $r_{\max}$ has a minimum of 1.5 and allows larger velocity variations with larger track aspect ratios and/or eccentricity. The σ_{VBS} term allows for proportionally larger effects of measurement noise when track velocity is near zero. $10\sigma_{\text{VBS}}$ is used as a limit to provide a 5σ 'buffer' against the effects of noise on temporally adjacent observations in a track.

Rule 3 defines a minimum angle ψ between successive steps $\vec{v}$. For $a_e/b_e \approx 1$ and $e_o \approx 0$, ψ is obtuse with $\psi \approx \pi$. Otherwise, ψ is most acute where d_k is small and most obtuse where d_k is large. To match these variations, the rule applies

$$\psi_k > \psi_{\min} \tag{35}$$

$$\psi_{\min} = \min\left(\frac{5\pi}{6}, \frac{5\pi}{6} \frac{d_k}{\max(d_{\text{mean}}, 10\sigma_{\text{VBS}})}\right)(1 - e_o) \tag{36}$$

If d_k is small compared to d_{mean} or $10\sigma_{\text{VBS}}$, or the absolute orbit is eccentric, the minimum allowed angle is more acute. The rule also ensures $\psi_{\min} \leq 150°$, which relates to the maximum expected time interval between images. Larger intervals imply fewer track steps per orbit and thus more acute angles between steps, as per the interior angles of an n-sided regular convex polygon; 150° implies a maximum interval of $\frac{1}{12}$ of the orbit period.

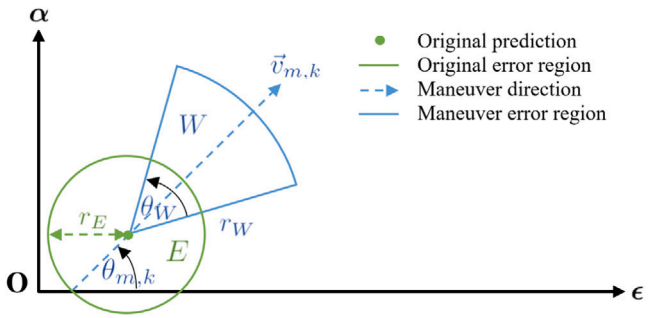

Fig. 10. Notional illustration of an expanded maneuver error region in bearing angle space. The expanded region is the union of original error region E and new region W.

Rule 4 ensures that target tracks possess consistent clockwise (or anticlockwise) behavior, following the expected parametric form. It is defined as

$$\text{sign}(\zeta_k - \zeta_{k-1}) = \text{sign}(\zeta_{k-1} - \zeta_{k-2}) \tag{37}$$

Eccentric orbits can observe direction changes as the first- and second-order components of relative motion constructively or destructively interact throughout the orbit. However, these changes are periodic with $\frac{1}{4}$ of the orbit and on the shorter timescales between images the rule generally holds. Measurement noise can also cause violations when track velocities are small; Rule 4 is therefore only applied if $|\pi - \psi_k| > \frac{\pi}{10}$ and $d_k > 10\sigma_{\text{VBS}}$.

Rule 5 guarantees that assigned measurements lie within some error region around the predicted measurement. If a target state estimate with covariance is available, the error region in the bearing angle plane is obtained via an unscented transform with user-specified σ-bound. If a state estimate is unavailable, the error region E for new measurement $(\alpha, \epsilon)_k^\top$ is a circle centered on predicted measurement $(\alpha_p, \epsilon_p)_k^\top$, with radius

$$r_E = \max(10\sigma_{\text{VBS}}, 2d_{\text{mean}})(1 + e_o) \tag{38}$$

A modified prediction must be applied if SAMUS observes a maneuver or a data gap because the next measurement will be affected by either the maneuver or secular ROE drift. However, the coefficients of track motion models in bearing angle space do not correspond exactly to ROE — as per Eq. (1), bearing angles and the derived coefficients are effectively normalized by target range, and as such, traditional ROE maneuver or drift models (Eqs. (18), (15), (16)) cannot be quantitatively applied. Nevertheless, qualitative effects on the next measurement can be assessed to appropriately modify E.

It is desired for consistency to view all swarm maneuvers as a target maneuvering relative to the observer; thus, if the maneuver is conducted by the observer, its components are negated. RTN maneuver components can then be rotated into the tracking frame to form $\delta\boldsymbol{v}^{\mathcal{T}}$. As per Fig. 2, $\delta v_x^{\mathcal{T}}$ describes change in target relative velocity in the $\hat{\boldsymbol{x}}^{\mathcal{T}}$ or elevation direction, and $\delta v_y^{\mathcal{T}}$ in the $\hat{\boldsymbol{y}}^{\mathcal{T}}$ or azimuth direction. When integrated over the time interval between images, a change in relative velocity causes change in expected relative position and bearing angle in the new image. The post-maneuver predicted measurement $(\alpha_m, \epsilon_m)_k^\top$ is then located in some direction $\vec{v}_{m,k}$ from original prediction $(\alpha_p, \epsilon_p)_k^\top$. The phase in the bearing angle plane that defines $\vec{v}_{m,k}$ is $\theta_{m,k} = \text{atan2}(\delta v_y^{\mathcal{T}}, \delta v_x^{\mathcal{T}})$. Fig. 10 provides an illustration. The modified error region is the union of a 2D wedge W – defined by $\theta_{m,k}$ and $(\alpha_p, \epsilon_p)_k^\top$ – and the original error region. Here, wedge radius r_W is defined as 20% of the FOV with arc angle $\theta_W = \frac{\pi}{4}$. The wedge must be large enough to allow for expected maneuvers but not too large as to allow formation of many false hypotheses. When instead resuming tracking after a data gap, the radius of the error region is doubled but its shape is unchanged because secular ROE changes during a single orbit are generally small.

It is recommended to choose a measurement interval large enough such that $d_{\text{mean}} \gg \sigma_{\text{VBS}}$ and noise does not wrongly validate or invalidate a large number of tracks, and small enough that sufficient information is received for robust stepwise prediction and gating. The authors have found that between 30–100 VBS images per orbit provides a suitable compromise.

3.4. Measurement-to-track association

From the list of valid measurement-to-track associations, SAMUS forms a set of valid global hypotheses. Each hypothesis is scored to assess its potential for selection, propagation or deletion. SAMUS employs an additive track score stemming from the kinematic rules. A hypothesis score is the sum of scores of each constituent track in its possible transformed frames. Consider hypothesis i, which features p tracks (indexed by l) and $p-1$ possible transforms for each track (indexed by m with $l \neq m$). The hypothesis is scored across q epochs (indexed by k). Thus, define $s_{i,j,k,l,m,}$ as scoring criterion j of epoch k, for track l transformed with respect to track m. The total score for criterion j of hypothesis i is

$$s_{i,j} = \sum_{k=1}^{q} \sum_{l=1}^{p} \sum_{m=1}^{p-1} s_{i,j,k,l,m} \tag{39}$$

and the final score of hypothesis i, with its component scores $s_{i,j}$ normalized to lie within $[0,1]$, is

$$s_i = \sum_j \frac{s_{i,j} - \min_i(s_{i,j})}{\max_i(s_{i,j}) - \min_i(s_{i,j})} \tag{40}$$

Criteria $s_{i,j}$ assess how well tracks match kinematic expectations. For epoch t_k, $s_{i,j}$ are defined as

$$s_{i,1} = \|\boldsymbol{A}_1\vec{y}_1 - \vec{\epsilon}\|_2 + \|\boldsymbol{A}_2\vec{y}_2 - \vec{\alpha}\|_2 \tag{41}$$

$$s_{i,2} = \|(\alpha_p, \epsilon_p)_k^\top - (\alpha, \epsilon)_k^\top\|_2 \tag{42}$$

$$s_{i,3} = |d_k - d_{p,k}| \tag{43}$$

$$s_{i,4} = |d_k - d_{\text{mean}}| \tag{44}$$

$$s_{i,5} = |\zeta_k - \zeta_{p,k}| \tag{45}$$

$$s_{i,6} = |\psi_k - \psi_{p,k}| \tag{46}$$

$$s_{i,7} = |\psi_k - \psi_{\text{mean}}| \tag{47}$$

$$s_{i,8} = |f_k - f_o| \tag{48}$$

$$s_{i,9} = 1/|d_k| \tag{49}$$

$$s_{i,10} = 1/|\psi_k| \tag{50}$$

Above, s_1 describes the residuals from track model fitting; s_2 describes the distance between the predicted measurement and new measurement; s_3 and s_4 describe the difference between track step size and the predicted and mean sizes; s_5 describes the difference between track step phase and the predicted phase; s_6 and s_7 describe the difference between track step angle and the predicted and mean angles; s_8 describes the difference between the new measurement parameter f_k (computed from the previously-fitted model) and its parameter from the observer's absolute orbit estimate, f_o; and s_9 and s_{10} bias the track towards smaller steps and larger step angles. The best hypothesis has the smallest score.

Typical scoring methods often apply a single Mahalanobis distance metric between the predicted and assigned measurement for scoring. However, depending on the ROE and measurement frequency, target motion across images may be comparable in magnitude to VBS noise and multiple targets may be in close proximity in the image plane. A single scoring metric is not robust in these cases. By using a set of metrics $s_{1,\ldots,10}$, consensus supports the correct choice over time, even if some criteria temporarily support incorrect hypotheses. SAMUS also avoids the need for probabilistic estimates of false alarm densities or target decay rates as in Eq. (20), which are not easily obtainable for spacecraft.

When no measurements can be assigned to a track, its predicted measurement is used as a placeholder for propagation into future epochs. If a track is not visible for more than 10% of the orbit (and was expected to be visible), it is deleted. SAMUS also gives associations an 'ambiguity' flag in that only measurements which are unambiguously of the target should be passed to a navigation filter, to minimize false positives. Consider best hypothesis h_1 with score s_1 and second-best h_2 with score s_2. Then, if $s_1 < C_1 s_2$, measurements in h_1 which have been members of the target's best track at least C_2 epochs are unambiguous. C_1 and C_2 are user-specified constants. This paper applies $C_1 = 0.5$ to ensure h_1 is superior in at least twice as many criteria, and $C_2 = 3$ as this is the fewest measurements needed for track fitting. All hypotheses h_i with scores $s_i < C_3$ are also propagated, in case they are the true hypothesis. A maximum score of $C_3 = \max(3, 3s_1)$ is chosen to balance robustness and computation cost.

If an angles-only navigation filter is estimating the complete swarm state, Mahalanobis distance [20] is also used to score associations, as per

$$s_{i,11} = \sqrt{(\alpha - \alpha_p, \epsilon - \epsilon_p)_k \Sigma^{-1} (\alpha - \alpha_p, \epsilon - \epsilon_p)_k^\top} \tag{51}$$

where Σ is the predicted measurement covariance. To be considered unambiguous, no other kinematically-valid $(\alpha, \epsilon)_k^\top$ may be contained within the 3Σ region around $(\alpha_p, \epsilon_p)_k^\top$.

3.5. Maneuver-to-track association

If batches of target measurements provided by SAMUS are to be applied for state estimation, presence of maneuvers must be determined. Maneuvers occurring at a specific epoch must be assigned to tracks at that epoch and hypothesis compatibility and scoring must take maneuvers into account. SAMUS assumes knowledge of the execution times, magnitudes, and directions of all planned swarm maneuvers, but no knowledge of which maneuvers correspond to which targets within SAMUS. Instead, the kinematics of tracks over time must be leveraged to determine maneuver correspondence.

In the first epoch after a maneuver, all tracks are split into two branches: one without a maneuver (in case the maneuver is not assigned to that track) and one with a maneuver (commencing a new track segment). If the observer maneuvers or all targets maneuver, all tracks are affected and non-maneuver branches are deleted. Assignment of maneuvers to specific tracks is performed when four new images after the maneuver have been received, to ensure that enough kinematic information is available for a robust assessment. Within each hypothesis, each constituent track is scored via six criteria and the maneuver is assigned to the best-scoring (i.e. lowest-scoring) track.

Define $\vec{x}_{\text{pre}}$ as the track model before the maneuver and $\vec{x}_{\text{post}}$ as the track model after the maneuver. The observed change is $\Delta\vec{x} = \vec{x}_{\text{post}} - \vec{x}_{\text{pre}}$ and a predicted change can be computed with $\Delta\vec{x}_p = \boldsymbol{B}_{\text{eroe}}\delta\boldsymbol{v}^{\mathcal{R}}$. This $\Delta\vec{x}_p$ is not quantitatively accurate because as discussed, $\vec{x}$ represent scaled EROE and $\boldsymbol{B}_{\text{eroe}}\delta\boldsymbol{v}^{\mathcal{R}}$ applies to true EROE. However, qualitative aspects of $\Delta\vec{x}$ are employed to examine consistency between predicted and measured changes. First, $\Delta\vec{x}$ are normalized using

$$\Delta\vec{x}^N = \Delta\vec{x} / \max_j |\Delta\vec{x}_j| \tag{52}$$

where $j = 1, \ldots, 6$ indexes components of $\Delta\vec{x}$. Subsequently, maneuver hypothesis scores s_i^m are computed as

$$s_1^m = \|\Delta\vec{x}_p^N - \Delta\vec{x}^N\|_2 \tag{53}$$

$$s_2^m = |\operatorname{argmax}_j(\Delta x_{p,j}^N) - \operatorname{argmax}_j(\Delta x_j^N)| \tag{54}$$

$$s_3^m = \sum_j [\operatorname{sign}(\Delta x_{p,j}^N) - \operatorname{sign}(\Delta x_j^N)] \tag{55}$$

$$s_4^m = |\theta_{m,k-3} - \angle((\alpha, \epsilon)_{k-3}^\top - (\alpha_p, \epsilon_p)_{k-3}^\top)| \tag{56}$$

$$s_5^m = 1/\|\Delta\vec{x}\|_1 \tag{57}$$

$$s_6^m = \sum_{m=0}^{3} \frac{1}{|(\alpha,\epsilon)_{k-m}^{\top} - (\alpha_p,\epsilon_p)_{\text{pre},k-m}^{\top}|} \tag{58}$$

where k indexes the current image epoch t_k such that the maneuver occurred between t_{k-3} and t_{k-4}. The notation $(\alpha_p,\epsilon_p)_{\text{pre},k-m}^{\top}$ indicates a prediction using model $\vec{x}_{\text{pre}}$. Above, s_1^m examines differences between predicted and measured changes in the motion model; s_2^m examines whether the component of $\Delta\vec{x}$ observing the largest change was as expected; s_3^m examines whether changes in $\Delta\vec{x}$ possess the expected sign; s_4^m examines whether the maneuver caused the expected discrepancy between predicted and assigned measurement; s_5^m biases maneuver assignments towards tracks displaying large changes in motion; and s_6^m biases maneuver assignments towards tracks in which pre- and post-maneuver models are inconsistent. Maneuvers are not assigned to a track if motion does not change after the maneuver epoch, i.e. $\|\vec{x}_{\text{post}} - \vec{x}_{\text{pre}}\|_2 \approx 0$, because maneuver information may not always correspond to a target in view.

Tracking may also be required for completely unknown target maneuvers when observing unidentified, uncooperative, or adversarial spacecraft. If maneuvers are small such that the kinematic rules are not violated, and targets remain well-separated in the bearing angle plane, SAMUS can in principle maintain tracking due to the overall robustness of kinematic scoring. Future work will explore tracking of unknown maneuvers in detail.

3.6. Track maintenance

To initialize new targets, SAMUS applies the DBSCAN algorithm [45] to find clusters of unidentified measurements in the most recent four images. A DBSCAN cluster requires at least n_D points within specified radius ϵ_D. Other points are treated as noise. SAMUS aims to form $\frac{n_D}{4}$ new targets from each cluster and the prior gating and scoring criteria are applied to intra-cluster tracks to compute the best cluster hypothesis. Initialization does not require knowing the number of targets in advance.

A SAMUS target is finalized when SAMUS is cooperating with a navigation filter that estimates the target state, and the filter has converged to steady-state. In this scenario, tracking is greatly simplified because the filter is relied on to provide high-quality measurement prediction, gating and scoring information. Converged state covariances allow finalized targets to collapse into a single hypothesis for typical formation geometries and VBS noise.

To manage hypotheses, merge similar tracks, prune poor tracks, and limit computation costs, SAMUS employs several well-known methods introduced in Section 2.3. Algorithm 1 in the Appendix presents a pseudocode summary of these operations. Algorithm 2 in the Appendix presents pseudocode for the main loop of SAMUS. Fig. 11 presents an overview of relevant operations. When viewed as a whole, a potential weakness of SAMUS is the relatively large number of hyperparameters that affect performance. Although this paper provides suggested values for each, best practices for tuning will be addressed in future implementations.

4. Algorithm validation

SAMUS performance is validated across three test suites: (1) using synthetic bearing angle measurements, (2) using synthetic VBS imagery, and (3) using HIL imagery produced by a star tracker. SAMUS will also be flight-tested in 2022 aboard the NASA Starling mission as part of the ARTMS software payload. ARTMS is a complete, autonomous, angles-only swarm navigation architecture for which SAMUS performs target identification and data association [6,18]. Starling consists of four 6U CubeSats in LEO and its physical parameters form the basis for each simulation. Starling CubeSats will carry Blue Canyon Technologies Nano Star Trackers (NST) for angles-only navigation and NST parameters were applied when generating synthetic test images, listed in Table 2. An NST was also used to collect imagery for HIL tests. Tests were conducted on a PC with an Intel i7-7700HQ CPU and 16GB RAM.

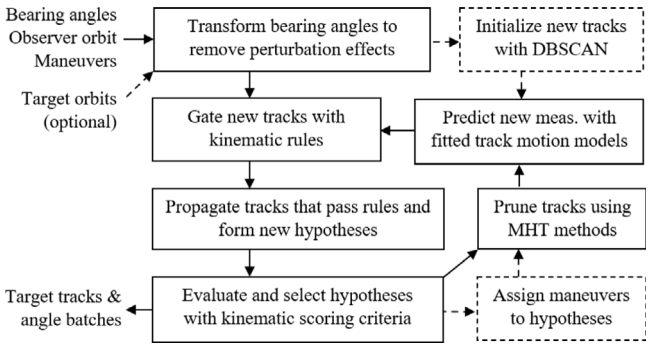

Fig. 11. SAMUS algorithm summary and core sequence of operations. Dashed lines denote steps that only occur at relevant epochs.

Table 2
Intrinsic parameters of the NST.

Intrinsic parameter	Value
Image size (pixels)	1280 × 1024
FOV (°)	12 × 10
Pixel size (μm)	5.3
Focal length (mm)	30.2
Pixel intensity range	0–255

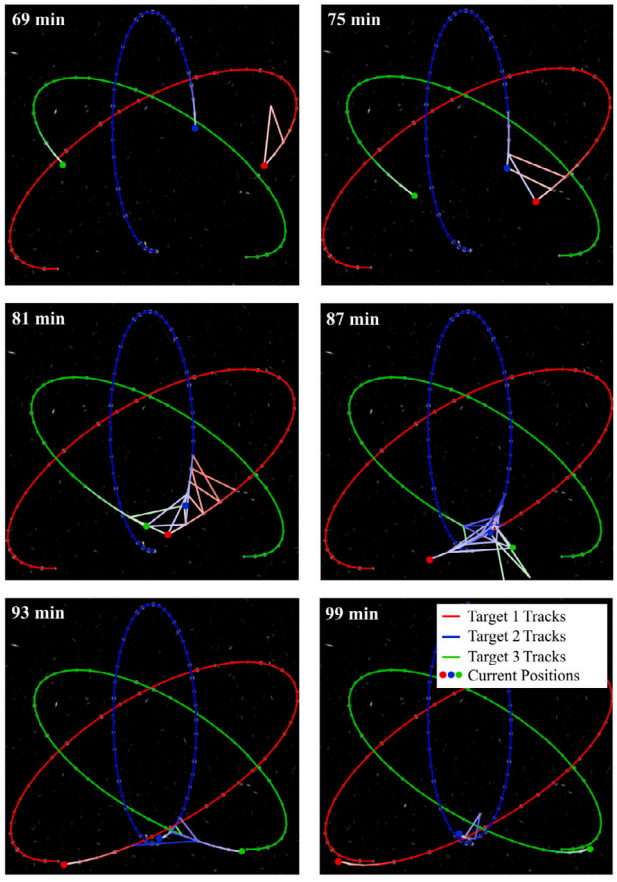

Fig. 12. Illustration of hypothesis evolution. Track hypotheses are overlaid on superimposed VBS images from $t = 0$ up to the specified time. Lighter track segments are newer and considered ambiguous while darker segments are more certain.

4.1. Data generation

To generate test cases, the positions and velocities of a four-spacecraft swarm were numerically integrated using the Stanford

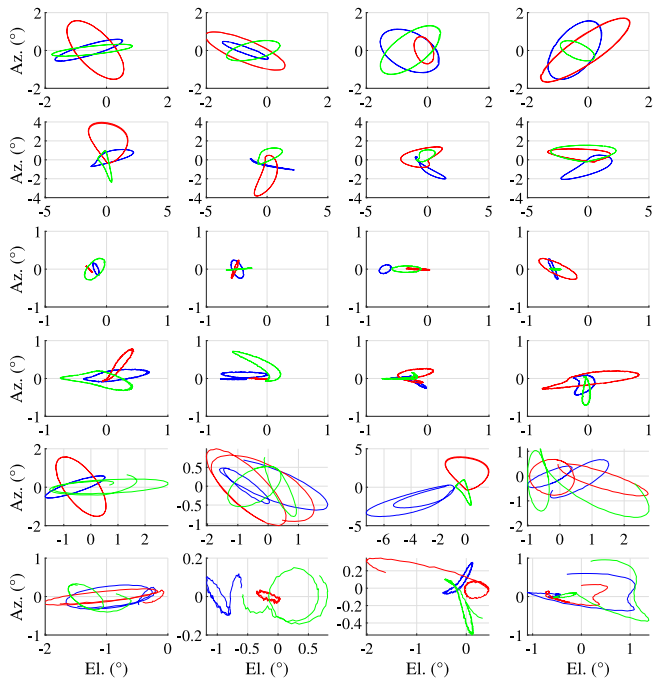

Fig. 13. Examples of target motion in $\mathcal{V}$. Different colors represent distinct targets. Rows 1–6 correspond to datasets NC-EIS, ECC-EIS, NC-IT, ECC-IT, EIS-MAN and IT-MAN respectively.

Space Rendezvous Lab's S^3 software [46]. Included perturbations were a 120 × 120 spherical harmonic gravity model, a cannonball drag model using Harris–Priester atmospheric density, a cannonball solar radiation pressure model with cylindrical Earth shadow, and third-body lunisolar gravity. Spacecraft attitude was fixed such that the VBS boresight points in the $\hat{\boldsymbol{y}}^{\mathcal{W}}$ direction. For tests using synthetic bearing angles, Gaussian white measurement noise was added to ground truth angles with zero mean and $\sigma_{\text{VBS}} = 20''$. Attitude noise was $\sigma_{\text{off-axis}} = 3''$ and $\sigma_{\text{roll}} = 20''$, considered typical for a modern CubeSat star tracker [44,47] and verified in the HIL tests. Between 3–10 extra measurements were added to each bearing angle set to emulate passing satellites, debris and non-catalog stars, with positions pulled from a uniform distribution across the FOV. Observer absolute orbit knowledge was provided via a single ECI position/velocity estimate at the start of the simulation with Gaussian white noise of $\sigma_{\text{pos}} = 10$ m and $\sigma_{\text{vel}} = 0.02$ m/s. SAMUS propagates the estimate into subsequent epochs via numerical integration of the Gauss Variational Equations using a 30-second timestep and 20 × 20 gravity model. No other perturbations are modeled by SAMUS. Maneuvers were executed with 5% (1σ) magnitude error and $60''$ (1σ) direction error [48].

Each simulation consists of one observer tracking three targets for two orbits. Measurements were received every two minutes. Formations were generated from uniform distributions over the ranges of OE and ROE in Table 3. To be consistent with earlier reasoning, limits of $\delta\lambda/\delta e \geq 20$ and $\delta\lambda/\delta i \geq 20$ were applied to ensure targets remain in the FOV without any need for active camera tracking. Simulations are categorized by three aspects: type of absolute orbit, type of relative orbit, and inclusion of maneuvers. Absolute orbits are either near-circular (NC) with $e \in [0.0001, 0.01]$ or eccentric (ECC) with $e \in [0.01, 0.8]$. Relative orbits are either in-train (IT) or E/I-vector separated (EIS). IT formations possess large differences in $\delta\lambda$ with other ROE being approximately zero, presenting a common but constrained case with little relative motion. EIS formations possess differences in all ROE and thus more relative motion. IT formations are defined below by $\delta\lambda/\delta e \geq 200$ and $\delta\lambda/\delta i \geq 200$. Finally, some test cases include impulsive maneuvers (MAN) by the observer or targets. Maneuvers are generated

Table 3
OE and ROE ranges for simulations. r_p is the radius of periapsis of the orbit.

Obs. OE	Range	Tar. ROE	Range
r_p (km)	[6750, 7150]	δa (km)	[−0.2, 0.2]
e	[0.0001, 0.8]	$\delta\lambda$ (km)	[5, 200]
i	$[0, \pi]$	δe_x (km)	[−5, 5]
Ω	$[0, 2\pi]$	δe_y (km)	[−5, 5]
ω	$[0, 2\pi]$	δi_x (km)	[−5, 5]
M_0	$[0, 2\pi]$	δi_y (km)	[−5, 5]

via uniform distributions across spacecraft, execution time, direction, and magnitude, with $0.1 \leq |\delta\boldsymbol{\nu}^{\mathcal{R}}| \leq 2$ m/s. Two maneuvers occur in each simulation, if included. The complete simulation set consists of:

1. 200 NC-EIS and 100 NC-EIS-MAN
2. 200 ECC-EIS and 100 ECC-EIS-MAN
3. 100 NC-IT and 50 NC-IT-MAN
4. 100 ECC-IT and 50 ECC-IT-MAN

Fig. 12 presents an example of hypothesis evolution for three targets in an NC-EIS formation. When targets are in close proximity, many new tracks are created. As more measurements arrive, tracks are scored and pruned to converge to the true hypothesis. Fig. 13 presents examples of simulated target bearing angle tracks.

4.2. Synthetic bearing angle tests

Table 4 presents Monte Carlo results for SAMUS for simulation subsets: near-circular, eccentric, in-train, and E/I-vector separated (without maneuvers); all simulations without maneuvers; and all simulations with maneuvers. Performance metrics of accuracy, precision and recall are computed using 'true positives', or measurements correctly associated with a target; 'true negatives', or measurements correctly not associated with a target; 'false positives', or measurements incorrectly associated with a target; and 'false negatives', or measurements incorrectly not associated with a target. Accuracy assesses overall performance, precision focuses on reliability of associations, and recall focuses on frequency of associations.

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{TN}} \tag{59}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{60}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{61}$$

Precision is considered the most vital metric because angles-only orbit determination filters are very sensitive to measurement errors and a single false positive can cause degradation of the filter state estimate [13]. Here, an association is defined as false positive if the measurement was produced by a different target and is more than $5\sigma_{\text{VBS}}$ from the ground truth measurement.

The 'ALL' row in Table 4 indicates that SAMUS data association precision is 99.7% and false positives are minimized as desired. Despite an emphasis on discarding ambiguous measurements, recall remains high at 96.3% and sufficient data is retained for navigation. The metric of '100% Precision Cases' examines the proportion of tests observing zero false positives. Perfect precision is achieved across the vast majority of formations and mean association error is on the order of measurement noise. In comparison to the near-circular case, eccentric orbits observe slightly diminished performance, because relative motion is more complex and unpredictable at high eccentricities and the reliability of kinematic gating and scoring is negatively impacted. Likewise, In-train formations see worsened performance because targets are in closer proximity in the image plane which increases association ambiguity. The lower track velocities of in-train formations also make measurement noise proportionally more significant. The addition of

Table 4
Monte Carlo results (1σ) for SAMUS for different simulation subsets.

Data	Precision (%)	Recall (%)	Accur. (%)
NC	99.99 ± 0.06	97.12 ± 2.09	98.15 ± 1.34
ECC	99.42 ± 2.59	95.47 ± 8.25	94.85 ± 10.1
IT	99.20 ± 3.05	92.89 ± 9.63	91.99 ± 11.8
EIS	99.95 ± 0.68	97.88 ± 1.64	98.62 ± 1.14
ALL	99.71 ± 1.83	96.31 ± 6.01	96.54 ± 7.31
MAN	99.62 ± 2.14	94.90 ± 5.84	95.45 ± 9.67
Data	100% prec. cases (%)	Mean error (")	Runtime per epoch (ms)
NC	98.0	25 ± 1	17 ± 5
ECC	86.0	30 ± 22	18 ± 6
IT	79.0	31 ± 25	22 ± 8
EIS	98.5	26 ± 8	15 ± 3
ALL	92.0	27 ± 16	18 ± 6
MAN	86.7	42 ± 180	18 ± 6

Table 5
Monte Carlo results (1σ) for different MTT algorithms across the complete simulation set.

Method	Precision (%)	Recall (%)	Accur. (%)
SAMUS	99.68 ± 1.93	95.86 ± 5.98	96.19 ± 8.15
GNN	80.66 ± 15.3	87.83 ± 3.76	89.78 ± 4.99
JPDA	78.79 ± 14.0	71.13 ± 7.54	88.05 ± 3.54
MHT	82.15 ± 10.4	77.21 ± 8.51	88.52 ± 4.45
PHD	89.97 ± 11.1	55.28 ± 18.9	81.98 ± 10.3
Method	100% prec. cases (%)	Mean error (")	Runtime per epoch (ms)
SAMUS	90.2	32 ± 104	18 ± 6
GNN	37.0	434 ± 665	30 ± 42
JPDA	47.5	168 ± 176	90 ± 264
MHT	25.7	439 ± 475	15 ± 3
PHD	58.3	196 ± 232	50 ± 21

maneuvers decreases performance slightly because sudden changes to target trajectories introduce short periods of significant uncertainty. Nevertheless, the majority of maneuvers are tracked successfully and maneuvers are assigned to targets with 95% accuracy. In summary, SAMUS achieves the necessary performance in that precision is above 99% across all datasets and excellent consistency and accuracy are displayed in varying conditions.

Table 5 compares SAMUS with four other common MTT algorithms: GNN, JPDA, traditional MHT, and a PHD filter. Each is implemented within the MATLAB Sensor Fusion and Tracking Toolbox. The R2020a [49] version was used to generate the results in this paper. The MATLAB algorithms were utilized to perform MTT in the bearing angle plane with dynamics models as per Eq. (9), such that target states consist of 2D bearing angle positions and velocities in the observer's tracking frame and measurements are of the target position.

SAMUS demonstrates visibly superior performance. In comparison to the next-best algorithm (the PHD filter), it retains a 10% precision advantage and 30% advantage in the number of 100% precision cases. Other algorithms also display very variable performance, indicating a relative lack of robustness. Degradation is notable, especially during challenging eccentric or in-train scenarios, and significantly higher standard deviations in each metric and large mean association errors are observed. The <90% precision of GNN, MHT, JPDA and the PHD filter in their current form would likely be unusable for angles-only navigation in orbit. SAMUS also demonstrates an advantage in runtime in MATLAB, being an order of magnitude faster than traditional MHT and three times faster than the PHD filter. Although these runtimes are not optimized, it is an indication that SAMUS should not exceed computational costs of other common algorithms. Computational scalability versus swarm size will be quantitatively studied in future when SAMUS is implemented on a CubeSat microprocessor.

Table 6 presents SAMUS performance across the same simulation set when key parameters are modified: measurement noise, measurement availability, and measurement interval. Results largely follow expected trends. Increased noise implies decreased performance because the reliability of the parametric motion model is negatively affected. This proves especially detrimental for in-train formations, for which 40″ of noise can overwhelm target velocity between images, invalidating the kinematic rules and dramatically reducing accuracy. Similar behavior occurs when the measurement interval is halved. The image sample time should thus be chosen such that target velocity is larger than expected noise. Conversely, lower noise improves precision and recall, and more frequent measurements benefits EIS formations in particular because with more data, SAMUS can better predict subsequent behavior. VBS quality and onboard processing power are therefore important considerations for MTT, in addition to formation geometry. Reduced measurement availability impacts performance to a much lesser degree and SAMUS successfully handles long eclipse periods. Slightly lower recall is observed because periods of ambiguous tracking (e.g. when re-initializing tracks after an eclipse) become a greater proportion of the orbit.

Table 6
Monte Carlo results for SAMUS across all datasets when varying simulation parameters. Default parameters are bolded.

Measurement noise std. dev.	Precision (%)	Recall (%)	Accuracy (%)
10"	99.81	97.66	97.05
20"	99.68	95.86	96.19
40"	91.21	83.24	88.73
Measurement gap	Precision (%)	Recall (%)	Accuracy (%)
0%	99.68	95.86	96.19
30%	99.43	87.97	94.44
60%	99.31	74.91	92.56
Measurement interval	Precision (%)	Recall (%)	Accuracy (%)
1 min	92.73	95.34	94.07
2 min	99.68	95.86	96.19
4 min	99.34	92.10	95.89
Abs. orbit uncertainty ($\sigma_{pos}, \sigma_{vel}$)	Precision (%)	Recall (%)	Accuracy (%)
(10 m, 0.02 m/s)	99.68	95.86	96.19
(1 km, 2 m/s)	99.55	94.74	94.77
(10 km, 20 m/s)	99.45	93.63	93.57
Maneuver execution error ($\sigma_{mag}, \sigma_{dir}$)	Precision (%)	Recall (%)	Accuracy (%)
(5%, 60")	99.68	95.86	96.19
(10%, 120")	99.64	95.52	95.81
(15%, 180")	99.51	94.39	95.02
Initial $\delta\mathbf{x}_{roe}$ std. dev. (as % of $\delta\lambda$)	Precision (%)	Recall (%)	Accuracy (%)
(0.1, 5, 0.1, …, 0.1)	99.91	98.93	98.15
(0.2, 10, 0.2, …, 0.2)	99.80	95.25	96.40
(0.4, 20, 0.4, …, 0.4)	95.29	81.71	91.30
Measurement input type	Precision (%)	Recall (%)	Accuracy (%)
Synthetic angles	99.68	95.86	96.19
Synthetic images	97.18	85.33	90.31

Sections 4–5 of Table 6 investigate performance when quality of a-priori data is varied, in the form of the observer's absolute orbit estimate and discrepancy between observer maneuver knowledge and executed maneuvers. SAMUS proves robust to uncertainty in the absolute orbit. The orbit is primarily used to compute rotations between reference frames and to provide data for track model fitting. Position errors on the order of kilometers have minimal effect on fitting accuracy when compared to typical VBS noise, and a poor initial state estimate still provides consistent motion when propagated. Similarly, maneuver execution errors have minor effects on performance and for the three levels of error, maneuvers are assigned with 93%–95% accuracy. SAMUS operates by examining the qualitative changes in track shape from a maneuver, and the simulated errors are generally not large enough to completely modify maneuver outcomes.

Usage of a-priori relative orbit knowledge is also investigated, in the form of cooperation with a navigation filter. In this scenario, SAMUS

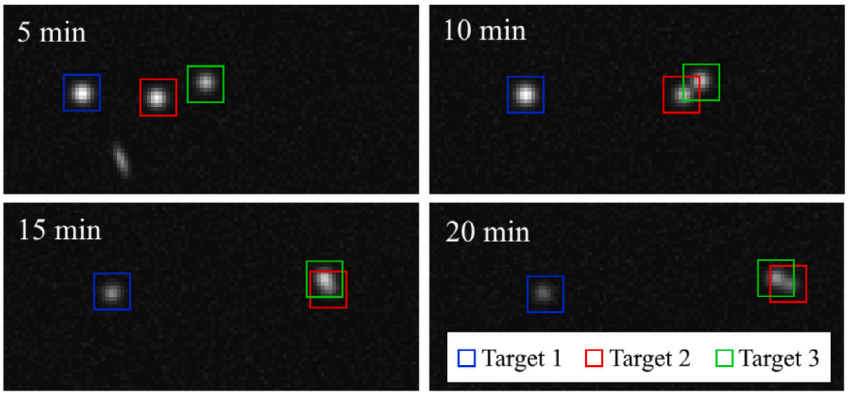

Fig. 14. Example of connected pixel clusters for an in-train formation (three targets).

uses filter relative state estimates to better identify and assign measurements to targets, and the filter subsequently employs SAMUS data associations to update its state. The angles-only unscented Kalman filter developed by Sullivan was applied [13,14], which uses bearing angle measurements to estimate target ROE. Section 6 of Table 6 presents results when varying the quality of the filter state initialization, in the form of an ROE state covariance. Target range, described by $\delta\lambda$, is the most weakly observable ROE and initial $\delta\lambda$ uncertainty is therefore dominant. An initial 5% $\delta\lambda$ uncertainty significantly improves tracking, achieving near-perfect precision and recall. Conversely, an initial 20% $\delta\lambda$ uncertainty negatively impacts tracking, particularly for in-train or intersecting formations in which the large state uncertainty leads to multiple valid data associations. If the filter state estimate possesses $\leq$10% range uncertainty, SAMUS can usefully leverage this information for enhanced performance in challenging scenarios.

4.3. Synthetic image tests

The final row of Table 6 presents results when moving to synthetic input imagery. VBS images are generated using 3D vector graphics in OpenGL [50]. The visual magnitudes, angles, and proper motions of SO are obtained from the Hipparcos star catalog and any objects within the camera FOV are rendered using Gaussian point spread functions (PSF). Background noise is added to every pixel according to a uniform distribution with intensity $I \in [0, 10]$, producing centroiding errors of $\sim$0.1 pixels. SAMUS processes each image to generate input measurements for MTT. The Fast Gaussian Fitting algorithm was used for centroiding [47], the Pyramid algorithm was used for star identification [43], and the Q-method was used for attitude determination [51].

A reduction in performance is observed, especially for in-train formations. Precision is lowered by 6% and recall by 17%. Overall, 29% of IT formations and 8% of EIS formations displayed at least one false positive. This trend stems from overlapping pixel clusters in simulated imagery when the PSF of several objects become connected (see Fig. 14). Traditional centroiding algorithms cannot distinguish this and consider the joined PSF to be one measurement, resulting in one 'missing' measurement and one inaccurate measurement that is the average of the two. When targets are in proximity, high overlap rates affect the ability of SAMUS to distinguish targets, or the joined PSF is inaccurate enough to be classified as an error. It is therefore necessary to test whether the predicted measurements of separate targets are likely too similar to produce distinct pixel clusters and for SAMUS to flag such associations as ambiguous. Alternatively, new centroiding techniques have explored detecting and separating joined PSF [52] and can be considered if in-train tracking is desired.

Table 7
Formation configuration for HIL test.

OE	Obs.	ROE	Tar. 1	Tar. 2	Tar. 3
a (km)	6978	δa (km)	0	0	0
e_x	0.0014	$\delta\lambda$ (km)	65	133	200
e_y	0.0014	δe_x (km)	0	0	0
i (°)	98	δe_y (km)	3	2.6	1.2
Ω (°)	40	δi_x (km)	0	0	0
u (°)	105	δi_y (km)	3	2.6	1.2

4.4. Hardware-in-the-loop tests

For HIL testing, input images are retrieved from a Blue Canyon Technologies NST as stimulated by the Stanford Space Rendezvous Lab's Optical Stimulator (OS). The OS is a variable-magnification testbed consisting of two lenses and a microdisplay. A synthetic space scene is generated in accordance with previous sections (without background noise) and shown on the OS, and by moving the two lenses and display relative to each other, the NST is stimulated. The system is calibrated such that the VBS image is similar in both radiosity and geometry to what would be observed in orbit. Development, calibration and usage of the OS is detailed by Beierle et al. [50] with achievable errors between desired and measured bearing angles of less than 10″. One such test is presented below, based on a proposed optical navigation experiment [18] for the aforementioned Starling mission. Formation OE and ROE are given in Table 7.

Fig. 15 presents the evolution of target tracks across 12 h or approximately 8 orbits. The observed data association errors of up to 80″ between assigned and ground truth measurements are reasonable when considering the expected error sources, which include NST error [44], calibrated OS errors [50], and attitude determination tolerances. Measurements are assigned to targets with 100% precision and 98% recall. Performance is identical to the same test conducted with entirely synthetic imagery. SAMUS is therefore able to operate on representative camera images and flight scenarios.

5. Conclusion

This paper introduces the 'Spacecraft Angles-only MUltitarget tracking System' (SAMUS) algorithm which is able to identify and track multiple target space objects from an observer spacecraft using only sequential images captured by the observer's on-board camera. SAMUS requires coarse observer absolute orbit knowledge but no target relative orbit knowledge, and provides angles-only multitarget tracking capabilities as required for autonomous angles-only navigation of spacecraft

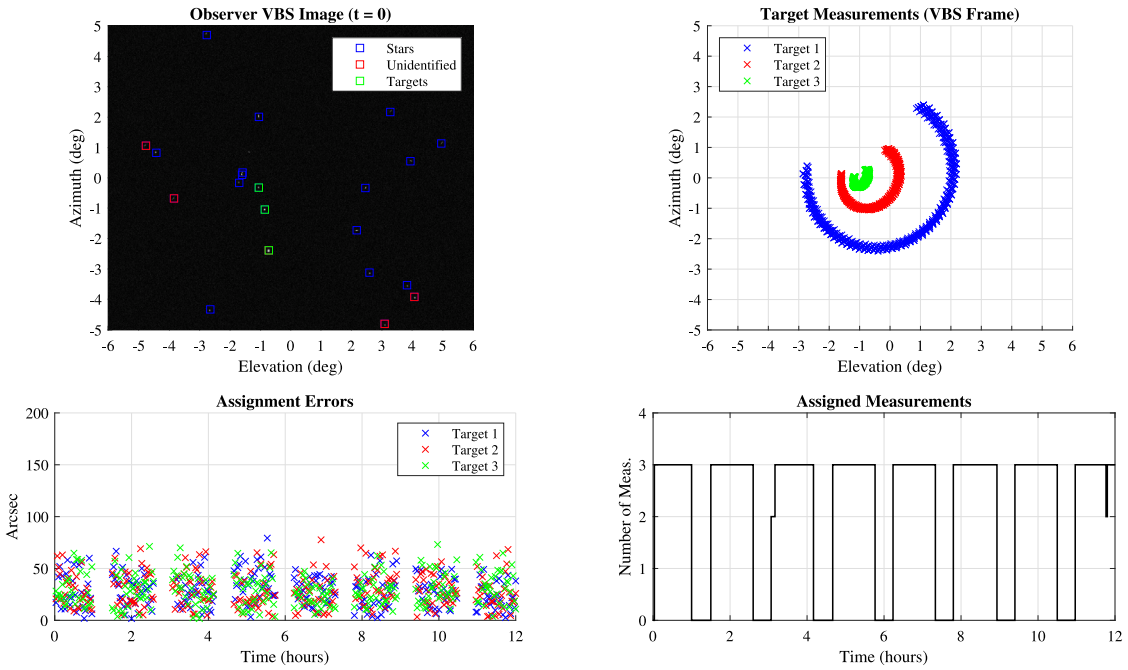

Fig. 15. Example HIL image (top left), target bearing angle measurements (top right), data association errors (bottom left) and frequency of data associations (bottom right).

swarms. Tracking is maintained in near-circular and eccentric orbits and in the presence of partially known swarm maneuvers.

In in-train and E/I-vector separated formations, SAMUS achieves data association precision of 99.20% and 99.95% respectively and >92% recall. Performance is maintained in the presence of expected levels of measurement noise and large measurement gaps, and significant improvements are demonstrated when compared to existing MTT algorithms. If a navigation filter is estimating the swarm state, SAMUS can apply state information to achieve 100% precision in difficult scenarios. Camera-in-the-loop tests verify performance under realistic conditions.

SAMUS can be extended to track spacecraft swarms at larger separations, or swarms in other dynamic environments such as lunar or cislunar space, by developing and implementing new parametric models for target motion that include the relevant dynamics and perturbation effects. Work is also underway to prepare a flight code implementation for the upcoming NASA Starling mission (2022). Starling intends to be the first ever demonstration of autonomous angles-only swarm navigation in orbit, for which SAMUS provides necessary MTT capabilities as part of an on-board navigation software payload.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix. Algorithm pseudocode

Algorithm 1 takes as input the set of all hypotheses which were propagated into the current epoch. It applies various criteria to trim unlikely hypotheses and merge similar hypotheses. The output is a reduced and more efficient set of hypotheses.

Algorithm 2 takes as input the set of existing hypotheses and a new VBS image. It processes the image to obtain new bearing angle measurements; propagates existing tracks using the new measurements; and initializes new tracks. The output is an updated set of hypotheses and a chosen best hypothesis.

Algorithm 1: Track maintenance algorithm.

Data: Propagated hypotheses
Result: Pruned hypotheses
if $n_{\text{targets}} > n_{\text{targets,max}}$ **then**
  keep best $n_{\text{targets, max}}$ targets
get score s_1 of best hypothesis h_1
for all hypotheses h_i **do**
  get s_i
  if $s_i > C_3$ **then**
    delete tracks existing in h_i only
for all targets T_j **do**
  N-scan pruning with $N = 8$
  for all tracks $t_m, t_n \in T_j$ **do**
    if unobserved in $\geq 10\%$ of visible period **then**
      delete track
    if ambiguous for $\geq 50\%$ of visible period **then**
      delete track
    if $t_m = t_n \; \forall$ epochs $k \in [0, N-1]$ **then**
      keep best of $\{t_m, t_n\}$
    if $t_m \neq t_n \; \forall$ epochs $k \in [0, N-1]$ **then**
      keep best of $\{t_m, t_n\}$
  if $n_{\text{tracks}} > n_{\text{tracks,max}}$ **then**
    keep best $n_{\text{tracks, max}}$ tracks
delete existing global hypotheses
cluster remaining tracks
re-form global hypotheses

Algorithm 2: Track propagation algorithm.

Data: Existing hypotheses and new image
Result: Updated best hypothesis
get image from sensor
get absolute orbit estimate from satellite bus
perform image centroiding
perform star identification
perform attitude determination
rotate unidentified angles into tracking frame
if relative orbit estimates exist **then**
  compute predicted angles and covariances
  if relative orbit estimates are new **then**
    initialize new targets
declare new list of propagated tracks t_{all}
for existing tracks t_m **do**
  for valid transforms t_n **do**
    fit motion model to track
    predict new measurement
    for new measurements m_k **do**
      create new track t_{new} from t_m and m_k
      apply kinematic gating rules
      if rules passed **then**
        add t_{new} to t_{all}
form compatible hypotheses h_i from t_{all}
for hypotheses h_i **do**
  compute score s_i using kinematic criteria
get score s_1 of best hypothesis h_1
initialize $n_{\text{hyp}} = 1$
initialize new list of tracks to keep t_{keep}
for hypotheses h_i **do**
  if $s_i < C_3$ **then**
    add $t_{\text{all}} \cap h_i$ to t_{keep}
    n_{hyp} += 1
  if $n_{hyp} > 6$ **then**
    break
if maneuvers occurred at epoch $k-3$ **then**
  for hypotheses h_i **do**
    score maneuver assignments for $t_m \in h_i$
    assign compatible maneuvers to $t_m \in h_i$
for targets T_j **do**
  if measurement assigned **then**
    store propagated tracks $t_{\text{keep}} \cap T_j$
  else
    propagate $t_{\text{all}} \cap T_j$ from epoch $k-1$ using predicted measurement
  update ambiguity flags
run DBSCAN on remaining unidentified angles
if clusters found **then**
  apply kinematic gating rules
  initialize valid new targets
do track maintenance as per Algorithm 1
pass h_1 to output

References

[1] S. D'Amico, M. Pavone, S. Saraf, A. Alhussien, T.S. Mohammed Al-Saud, S. Buchman, R. Byer, C. Farhat, Miniaturized autonomous distributed space system for future science and exploration, in: 8th International Workshop on Satellite Constellations and Formation Flying, Delft, The Netherlands, 2015.

[2] O. Brown, P. Eremenko, Fractionated space architectures: a vision for responsive space, in: Proceedings of the 4th Responsive Space Conference, Los Angeles, California, 2016.

[3] C. Underwood, S. Pellegrino, V.J. Lappas, C.P. Bridges, J. Baker, Using CubeSat/micro-satellite technology to demonstrate the autonomous assembly of a reconfigurable space telescope (AAReST), Acta Astronaut. 114 (2015) 112–122.

[4] J. Kruger, K. Wallace, A.W. Koenig, S. D'Amico, Autonomous angles-only navigation for spacecraft swarms around planetary bodies, in: 2021 IEEE Aerospace Conference, Big Sky, Montana, 2021.

[5] J. Sullivan, T. Lovell, S. D'Amico, Angles-only navigation for autonomous on-orbit space situational awareness applications, in: AAS/AIAA Astrodynamics Specialist Conference, Snowbird, Utah, 2018.

[6] H. Sanchez, D. McIntosh, H. Cannon, C. Pires, J. Sullivan, S. D'Amico, B. O'Connor, Starling1: swarm technology demonstration, in: 32nd Annual AIAA/USU Conference on Small Satellites, Logan, Utah, 2018.

[7] S. D'Amico, J.-S. Ardaens, G. Gaias, H. Benninghoff, B. Schlepp, J.L. Jørgensen, Noncooperative rendezvous using angles-only optical navigation: System design and flight results, J. Guid. Control Dyn. 36 (6) (2013) 1576–1595.

[8] J.-S. Ardaens, G. Gaias, Noncooperative rendezvous using angles-only optical navigation: System design and flight results, Acta Astronaut. 153 (2018) 367–382.

[9] F. Sellmaier, T. Boge, J. Spurmann, S. Gully, T. Rupp, F. Huber, On-orbit servicing missions: challenges and solutions for spacecraft operations, in: SpaceOps 2010 Conference, Huntsville, Alabama, 2010.

[10] B.B. Reed, R.C. Smith, B. Naasz, J. Pellegrino, C. Bacon, The Restore-L servicing mission, in: 2016 AIAA Space Forum, Long Beach, California, 2016.

[11] D. Woffinden, D. Keller, Optimal orbital rendezvous maneuvering for angles-only navigation, J. Guid. Control Dyn. 32 (4) (2009) 1382–1387.

[12] G. Gaias, S. D'Amico, J.-S. Ardaens, Angles-only navigation to a noncooperative satellite using relative orbital elements, J. Guid. Control Dyn. 37 (2) (2014) 439–451.

[13] J. Sullivan, S. D'Amico, Nonlinear Kalman filtering for improved angles-only navigation using relative orbital elements, J. Guid. Control Dyn. 40 (9) (2017) 2183–2200.

[14] J. Sullivan, A.W. Koenig, J. Kruger, S. D'Amico, Generalized angles-only navigation architecture for autonomous distributed space systems, J. Guid. Control Dyn. (2021).

[15] B. Gong, J. Luo, S. Li, W. Li, Observability criterion of angles-only navigation for spacecraft proximity operations, Proc. Inst. Mech. Eng. G 233 (12) (2019) 4302–4315.

[16] B. Gong, S. Li, J. Shi, Y. Yang, Rotation based analytic range-only initial relative orbit solution for natural periodic motion, Acta Astronaut. 178 (2021) 584–594.

[17] A.W. Koenig, S. D'Amico, Observability-aware numerical algorithm for angles-only initial relative orbit determination, in: 2020 AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe, California, 2020.

[18] A.W. Koenig, J. Kruger, S. D'Amico, ARTMS: enabling autonomous distributed angles-only orbit estimation for spacecraft swarms, in: American Control Conference, New Orleans, Louisiana, 2021.

[19] B.-n. Vo, M. Mallick, Y. Bar-shalom, S. Coraluppi, R. Osborne, R. Mahler, B.-t. Vo, Multitarget tracking, in: J.G. Webster (Ed.), Wiley Encyclopedia of Electrical and Electronics Engineering, Springer International Publishing, 2015.

[20] S.S. Blackman, R. Popoli, Design and Analysis of Modern Tracking Systems, Artech House, Boston, Massachusetts, 1999.

[21] Y. Bar-Shalom, P.K. Willett, X. Tian, Tracking and Data Fusion: A Handbook of Algorithms, YBS Publishing, Storrs, Connecticut, 2011.

[22] Y. Hu, I. Sharf, L. Chen, Distributed orbit determination and observability analysis for satellite constellations with angles-only measurements, Automatica 129 (2021) 109626.

[23] R.P.S. Mahler, Multitarget Bayes filtering via first-order multitarget moments, IEEE Trans. Aerosp. Electron. Syst. 39 (4) (2003) 1152–1178.

[24] M. Kisantal, S. Sharma, T.H. Park, D. Izzo, M. Märtens, S. D'Amico, Satellite pose estimation challenge: Dataset, competition design and results, IEEE Trans. Aerosp. Electron. Syst. 56 (5) (2020) 4083–4098.

[25] P. Cano, J.R. del Solar, Robust tracking of soccer robots using random finite sets, IEEE Trans. Intell. Syst. 32 (6) (2017) 22–29.

[26] H. Farazi, S. Behnke, Online visual robot tracking and identification using deep LSTM networks, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vancouver, Canada, 2017.

[27] K. LeGrand, K. DeMars, Relative multiple space object tracking using intensity filters, in: Proceedings of the 18th International Conference on Information Fusion, Washington, DC, 2015.

[28] D.A. Vallado, W.D. McClain, Fundamentals of Astrodynamics and Applications, fourth ed., Microcosm Press, Hawthorne, California, 2013.

[29] J. Sullivan, A. Koenig, S. D'Amico, Improved maneuver-free approach to angles-only navigation for space rendezvous, in: Proceedings of the 26th AAS/AIAA Space Flight Mechanics Conference, Napa, California, 2016.

[30] S. D'Amico, Autonomous formation flying in low earth orbit, (Ph.D. thesis), Delft University, Delft, The Netherlands, 2010.

[31] A. Koenig, T. Guffanti, S. D'Amico, New state transition matrices for relative motion of spacecraft formations in perturbed orbits, J. Guid. Control Dyn. 40 (7) (2017) 1749–1768.

[32] S. D'Amico, O. Montenbruck, Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation, J. Guid. Control Dyn. 29 (3) (2006) 554–563.
[33] M. Chernick, S. D'Amico, New closed-form solutions for optimal impulsive control of spacecraft relative motion, J. Guid. Control Dyn. 41 (2) (2018) 301–319.
[34] S.S. Blackman, Multiple hypothesis tracking for multiple target tracking, IEEE A E Syst. Magaz. 19 (1) (2004) 5–18.
[35] D. Reid, An algorithm for tracking multiple targets, IEEE Trans. Automat. Control 24 (6) (1979) 843–854.
[36] C. Kim, F. Li, A. Ciptadi, J. Rehg, Multiple hypothesis tracking revisited, in: 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 2015.
[37] I. Cox, S. Hongorani, An efficient implementation of Reid's multiple hypotheses tracking algorithm and its evaluation for the purposes of visual tracking, IEEE Trans. Pattern Anal. Mach. Intell. 18 (2) (1996) 138–150.
[38] S. Deb, M. Yeddanapudi, K. Pattipati, Y. Bar-Shalom, A generalized S-D assignment algorithm for multisensor-multitarget state estimation, IEEE Trans. Aerosp. Electron. Syst. 33 (2) (1997) 523–538.
[39] R.L. Streit, T.E. Luginbuhl, Maximum likelihood method for probabilistic multihypothesis tracking, in: Proc. SPIE 2235, Signal and Data Processing of Small Targets, Orlando, Florida, 1994.
[40] S.S. Blackman, Multiple Target Tracking with Radar Applications, Artech House, Dedham, Massachusetts, 1986.
[41] R.W. Sittler, An optimal data association problem in surveillance theory, IEEE Trans. Milit. Electron. 8 (2) (1997) 125–139.
[42] K.G. Murty, An algorithm for ranking all the assignments in order of increasing cost, Oper. Res. 16 (3) (1968) 682–687.
[43] D. Mortari, M. Samaan, C. Bruccoleri, J. Junkins, The Pyramid star identification technique, J. Inst. Navig. 51 (3) (2004) 171–183.
[44] D. Hegel, Small spacecraft subsystem state-of-the-art: Attitude determination and control, in: Proceedings of the 15th International Planetary Probe Workshop, The University of Colorado Boulder, 2018.
[45] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, AAAI Press, 1996, pp. 226–231.
[46] V. Giralo, S. D'Amico, Development of the Stanford GNSS navigation testbed for distributed space systems, in: Proceedings of the 2018 International Technical Meeting of The Institute of Navigation, Reston, Virginia, 2018, pp. 837–856.
[47] X. Wan, G. Wang, X. Wei, J. Li, G. Zhang, Star centroiding based on fast Gaussian fitting for star sensors, Sensors (Basel) 18 (9) (2018).
[48] E.G. Lightsey, T. Stevenson, M. Sorgenfrei, Development and testing of a 3D-printed cold gas thruster for an interplanetary CubeSat, Proc. IEEE 106 (3) (2018) 379–390.
[49] MATLAB Sensor Fusion and Tracking Toolbox R2020a, the MathWorks, Natick, Massachusetts, 2020.
[50] C. Beierle, S. D'Amico, High fidelity validation of vision-based sensors and algorithms for spaceborne navigation, J. Spacecr. Rockets 56 (4) (2019) 1060–1072.
[51] J.R. Wertz, Spacecraft Attitude Determination and Control, first ed., Springer Dordrecht, Dordrecht, The Netherlands, 1978.
[52] W. Ding, D. Gong, Y. Zhang, Y. He, Centroid estimation based on MSER detection and Gaussian mixture model, in: 12th International Conference on Signal Processing, Hangzhou, China, 2014.