AAS 26-751

BEHAVIORAL PREDICTION FOR LOW-THRUST SPACECRAFT IN THE EARTH-MOON SYSTEM

Cole Gillespie* and Natasha Bosanac†

This paper uses behavioral motion primitives for behavioral prediction of low-thrust spacecraft in the Earth-Moon system. A library of behavioral primitives summarizes geometrically distinct types of low-thrust motion for selected spacecraft parameters and control profiles. The arcs resembling each primitive are summarized by a volumetric representation in the phase space. Intersections between an uncertain state estimate and the regions of existence of any primitives supply a short-term trajectory prediction. Given multiple state estimates, sequences of primitives are used to rapidly summarize geometrically distinct types of possible paths and predict ranges of spacecraft parameters and control profiles.

INTRODUCTION

As more missions are expected to operate in the Earth-Moon system, the capability to rapidly predict possible future motions of spacecraft will become increasingly important. In this chaotic gravitational environment, a spacecraft could follow a wide variety of geometrically distinct paths. These geometries become even more diverse for spacecraft equipped with a continuous thrust propulsion system, as the thrust magnitude, specific impulse, spacecraft mass, and control profile impact its path. When observing a spacecraft for prediction purposes, these parameters may be unknown or uncertain. In that case, the possible future paths, spacecraft parameters, and maneuvering objectives are all coupled. Their estimation falls under the problem of behavioral prediction. In the Earth-Moon system, there may be a large and diverse array of possible solutions to this problem.

Existing approaches to orbit determination for thrust-enabled spacecraft in the Earth-Moon system may struggle with the high-dimension of the problem and sensitivity of the solution space. Existing approaches such as batch or sequential filtering algorithms may be used when a sufficient number of measurements and maneuvering information are available.[1] However, in a scenario where only sparse observations are available, this technique may be insufficient. Sampling approaches such as Monte Carlo methods rely on propagating a sufficient number of trajectories from an initial distribution.[2] However, propagation requires specification of the spacecraft parameters and control profile. Repeating propagation while varying those behavioral parameters may produce a large and diverse set of information at the expense of increased computational time. These limitations of existing approaches motivate development of a rapid and digestible behavioral prediction strategy for low-thrust spacecraft operating in the Earth-Moon system.

*Graduate Research Assistant, Ann & H.J. Smead Department of Aerospace Engineering Sciences, Colorado Center for Astrodynamics Research, University of Colorado Boulder, Boulder, CO, 80303.
†Associate Professor, Ann & H.J. Smead Department of Aerospace Engineering Sciences, Colorado Center for Astrodynamics Research, University of Colorado Boulder, Boulder, CO, 80303.

Behavioral prediction is a task that is commonly encountered across the scientific and engineering community: from predicting future robot actions[3] to pedestrian behaviors.[4,5] One approach to solving this problem in a rapid and digestible manner involves the use of motion primitives. These motion primitives are often extracted from real or simulated data to supply a concise set of building blocks of motion, actions, or controls.[6]

In recent years, the concept of a motion primitive has been applied to spacecraft trajectory design in the circular restricted three-body problem (CR3BP). Smith and Bosanac presented an early proof of concept for defining and extracting these motion primitives from periodic orbits and hyperbolic invariant manifolds using clustering.[7] Each motion primitive summarizes a set of arcs with a similar geometry, whereas distinct motion primitives either possess a distinct shape or exist in separated regions of the configuration space. Recently, Gillespie, Miceli, and Bosanac substantially improved on this procedure by using differential geometry and improving the clustering step to generate a more accurate and broader array of motion primitives for natural and thrust-enabled arcs.[8] Smith and Bosanac also constructed a graph-based representation of their sequential composability that supports generating initial guesses for geometrically distinct trajectories.[7] Since then, Miceli and Bosanac substantially improved this graph formulation and search process to improve the accuracy of the sequential composability assessment, produce higher quality initial guesses, and increase the likelihood that a corrected solution resembles the initial guess.[9] Minor updates to reduce computational complexity and support the use of primitives from multiple systems or with thrust have been presented by Bosanac[10,11] as well as Bodin, Bosanac, and Gillespie.[12]

This motion primitive concept has been extended to support trajectory and behavioral prediction for spacecraft operating in the Earth-Moon system. Specifically, Gillespie, Miceli, and Bosanac defined behavioral motion primitives to both 1) summarize a set of arcs with a unique geometry and 2) encode the associated intents, i.e., a description of their future or past itinerary, spacecraft parameters, and control profiles.[8] This definition was used to generate a preliminary library of behavioral motion primitives for natural and low-thrust trajectories in the Earth-Moon system, assuming small spacecraft with maneuvers that maintain a constant thrust magnitude and direction.[8] Bodin, Bosanac, and Gillespie then demonstrated the capability for this library to support long-term trajectory prediction for a spacecraft with impulsive maneuvering capability.[12]

This paper extends our recent proof of concept of behavioral prediction via motion primitives to thrust-enabled spacecraft in the planar Earth-Moon CR3BP. First, an updated library of behavioral motion primitives is presented to compactly summarize planar, low-thrust trajectories, generated with a variety of thrust-to-mass ratios and control profiles. Then, an updated voxel representation of each motion primitive is summarized to capture these additional parameters and the elapsed time. Next, simulated state estimates are projected onto this updated library of behavioral primitives to produce a set of short-term trajectory predictions and possible spacecraft parameters or control profiles. This process is repeated for multiple, infrequent state estimates to produce a sequence of possible and geometrically distinct trajectory predictions. Common information across these predictions are used to estimate ranges of spacecraft parameters and control profiles.

BACKGROUND

Circular Restricted Three-Body Problem

The Earth-Moon circular restricted three-body problem (CR3BP) is used to approximate the dynamics governing the motion of a spacecraft. The CR3BP relies on the assumption that the Earth

(P_1) and Moon (P_2) follow circular orbits around their mutual barycenter.[13] The celestial bodies are also modeled to possess the same gravitational fields as constant point masses whereas the third body, i.e., a spacecraft or small object, possesses a comparatively negligible mass.[13]

Physical quantities are nondimensionalized to produce values of similar orders of magnitude.[13, 14] Specifically, length values are normalized by the characteristic length, $l^* = 384,400$ km, corresponding to the average distance between the Earth and the Moon. To produce a mean motion that is equal to unity, time is normalized by $t^* \approx 3.751903 \times 10^5$ s. Mass values are normalized by the sum of the mass of the Earth and the Moon, i.e., $m^* \approx 6.046804 \times 10^{24}$ kg.[15] The mass ratio, defined as the ratio of the Moon's mass to the total system mass, is equal to $\mu \approx 0.012151$.

Motion in the CR3BP is typically analyzed in the P_1-P_2 rotating frame. This frame is defined with the origin at the barycenter of the system. Then, the three axes ($\hat{x}\hat{y}\hat{z}$) are defined with $\hat{x}$ pointing from the system barycenter towards the center of P_2, $\hat{z}$ aligned with the orbital angular momentum vector of the primaries, and $\hat{y}$ completing the right-handed, orthogonal triad.[13]

The planar equations of motion for a spacecraft in the Earth-Moon CR3BP are typically expressed in nondimensional form in the Earth-Moon rotating frame. The state vector is defined as $\bar{X}(t) = [x, y, \dot{x}, \dot{y}]^T$. Then, the equations of motion for the spacecraft are written as[13]

$$\ddot{x} = 2\dot{y} + \frac{\partial U^*}{\partial x}, \quad \ddot{y} = -2\dot{x} + \frac{\partial U^*}{\partial y} \tag{1}$$

where U^* is the pseudo-potential function that is equal to

$$U^* = \frac{1}{2}(x^2 + y^2) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{2}$$

and $r_1 = \sqrt{(x+\mu)^2 + y^2}$ and $r_2 = \sqrt{(x-1+\mu)^2 + y^2}$. This autonomous dynamical model also possesses a constant of motion, labeled the Jacobi constant,[13] that is calculated as $C_J = 2U^* - \dot{x}^2 - \dot{y}^2$. Fundamental solutions exist in the rotating frame, including five libration points, L_1 to L_5; periodic and quasi-periodic orbits; and any associated stable or unstable manifolds.[13, 14]

Thrust-Enabled Motion

Because the spacecraft is equipped with a continuous-thrust propulsion system, the CR3BP is augmented with an additional acceleration. Consider a spacecraft traveling within the Earth-Moon plane with an initial wet mass of $m_{0,dim}$. This spacecraft is equipped with a continuous-thrust propulsion system described by a constant thrust magnitude T_{dim} and constant specific impulse I_{sp}, applying thrust in a direction that is denoted as $\hat{u} = [u_x, u_y]^T$ via its components in the rotating frame. In this paper, the spacecraft hardware is summarized by the initial, nondimensional thrust-to-mass ratio $T/m_0 = (T_{dim}t^{*2})/(1000 l^* m_{0,dim})$ and I_{sp}. Furthermore, the instantaneous spacecraft mass is normalized by the initial wet mass to produce $m(t)$. Then, the spacecraft state vector is augmented with this nondimensional mass as $\bar{\chi}(t) = [x, y, \dot{x}, \dot{y}, m]^T$. Accordingly, the nondimensional planar equations of motion governing the spacecraft are written in the rotating frame as[13, 16]

$$\ddot{x} = 2\dot{y} + \frac{\partial U^*}{\partial x} + \frac{T}{m_0}\frac{u_x}{m}, \quad \ddot{y} = -2\dot{x} + \frac{\partial U^*}{\partial y} + \frac{T}{m_0}\frac{u_y}{m}, \quad \dot{m} = -\frac{T_{dim}}{m_{0,dim}}\frac{t*}{I_{sp} g_0} \tag{3}$$

where g_0 is the gravitational acceleration on the surface of the Earth.

In this paper, the thrust vector is defined to maintain a constant direction in the velocity-normal-conormal (VNC) axes, relative to the Moon and using an inertial observer. These three axes are defined as follows: $\hat{V}$ points along the velocity vector of the spacecraft relative to the Moon, $\hat{N}$ is aligned with the angular momentum vector of the spacecraft relative to the Moon, and $\hat{C}$ completes the right-handed, orthogonal triad. These axes possess a singularity when the speed equals zero or the position and velocity vectors relative to the Moon are parallel or anti-parallel.

Differential Geometry

Differential geometry is used across a variety of mathematical and technical applications to describe curved paths in two- or three-dimensional space. These concepts are used in this paper to describe and discretize nonlinear, thrust-enabled trajectories. Consider a trajectory that is generated over a time interval $t \in [t_0, t_f]$ from a specified initial state vector, subject to a specified spacecraft hardware configuration and control profile. At any location along the trajectory, the following vectors are defined in the Earth-Moon rotating frame: the position vector $\bar{r} = [x, y, z]^T$, the velocity vector $\bar{v} = [\dot{x}, \dot{y}, \dot{z}]^T$, the acceleration vector $\bar{a} = [\ddot{x}, \ddot{y}, \ddot{z}]^T$, and the jerk vector $\bar{j} = \dot{\bar{a}}$. Although the third component of each of these vectors is zero in this paper, it is included for mathematical consistency in the definitions in this subsection.

The trajectory traverses a distance equal to the arclength. This scalar quantity is defined as[17]

$$s_{pos} = \int_{t_0}^{t_f} \sqrt{\dot{x}^2 + \dot{y}^2} dt \tag{4}$$

and monotonically increases along a trajectory. This paper also uses an analog to arclength that is calculated along the velocity hodograph as[12]

$$s_{vel} = \int_{t_0}^{t_f} \sqrt{\ddot{x}^2 + \ddot{y}^2} dt \tag{5}$$

to characterize how the velocity vector of the spacecraft changes along the trajectory.

At any location along a curve, the curvature describes the extent to which the trajectory deviates from a straight line and, therefore, the derivative of the tangent vector with respect to arclength.[17] This scalar quantity κ is calculated from time-parameterized quantities in the rotating frame as[18]

$$\kappa(t) = \frac{||\bar{v} \times \bar{a}||}{||\bar{v}||^3} \tag{6}$$

Local maxima in the curvature, where $\dot{\kappa} = 0$ and $\ddot{\kappa} < 0$, identify geometrically significant locations along a trajectory where the shape is changing most rapidly.[18] These locations have been used in shape interrogation schemes.[18] Note that the unsigned curvature is used here to support future extension to three-dimensional trajectories.

The tangent and principal normal vector capture the local shape at any point along a trajectory. These vectors are two of the three basis vectors of the Frenet frame: the tangent vector $\hat{T}_F$ is tangent to the curve and pointed in the direction of motion, the principal normal $\hat{N}_F$ points towards the center of curvature within the osculating plane, and the binormal vector $\hat{B}_F$ completes the right-handed, orthogonal triad.[19] These three vectors are calculated as

$$\hat{T}_F = \frac{\bar{v}}{||\bar{v}||} \qquad \hat{N}_F = \hat{B}_F \times \hat{T}_F \qquad \hat{B}_F = \frac{\bar{v} \times \bar{a}}{||\bar{v} \times \bar{a}||} \tag{7}$$

The tangent and principal normal vectors possess a singularity when either the speed is equal to zero or the velocity and acceleration vectors are parallel or anti-parallel.

Density-Based Clustering

Density-based clustering automatically groups members of a dataset if they exist in close proximity and form sufficiently dense regions within a selected feature vector space. This paper uses a well-known algorithm, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), developed by Ester, Kriegel, Sander, and Xu.[20] DBSCAN identifies each point within the data as either a core point, a border point, or noise.[20] A core point possesses at least m_{pts} neighbors that are located within a distance of ϵ in the selected feature vector space.[20] Border points lie within the ϵ-neighborhood of a core point but do not qualify as core points.[20] Noise points do not meet either of these two criteria.[20] Connected core points and their associated border points form a cluster.[20] DBSCAN is governed by the parameters m_{pts} and ϵ, which must be specified as constant values before clustering a dataset. In this paper, DBSCAN is accessed in MATLAB.[21]

Volume Representations

A wide variety of approaches have been developed to approximate volumetric objects. For instance, mesh-based approaches leverage polyhedra to describe the surface of a volume[22] whereas point clouds sample the volume directly.[23] In this paper, however, a voxel approach is used to summarize a multi-dimensional volume. Voxels, or volume elements, are cubes that indicate the presence of data, analogous to a pixel in two-dimensions.[24] Voxels can also contain metadata consisting of quantities that further describe the volume, such as color or material properties.[24] Voxels have been applied to a variety of technical disciplines, from medicine[25] to computer graphics.[26]

GENERATING THRUST-ENABLED BEHAVIORAL MOTION PRIMITIVES

A library of behavioral motion primitives is constructed to summarize thrust-enabled arcs in the Earth-Moon planar CR3BP, subject to a variety of prespecified control profiles and hardware parameters. A behavioral motion primitive supplies two pieces of information: 1) a summary of the common geometry of a set of arcs that exist in a local region of the phase space, labeled its region of existence; and 2) labels capturing the possible control profiles and thrust-to-mass ratios of the spacecraft. This section summarizes the procedure used to generate this library, modeled after prior works by Gillespie, Miceli, and Bosanac;[8] Bodin, Bosanac, and Gillespie;[12] and Bosanac.[27]

Step 1: Generating the Training Dataset

Trajectories are generated in the planar thrust-enabled CR3BP to span a desired region of the phase space as well as discretely sampled spacecraft parameters and control profiles. This set of trajectories forms the training dataset used to extract a library of behavioral motion primitives. In this paper, initial conditions for these trajectories are generated across a region of the phase space defined by $x \in [0.8, 1.2]$, $y \in [-0.2, 0.2]$, and $C_J = [3.10, 3.20]$. Conceptually, these initial conditions exist between the L_1 and L_2 gateways over a wide range of energy levels. The spacecraft parameters are defined via the initial thrust-to-mass ratio T/m_0 and the specific impulse I_{sp}. The values of these quantities are selected based on current technology, using the characteristics of existing Hall effect thrusters and previously flown spacecraft.[28] In this paper, the specific impulse is set equal to a constant value of 1400s due to its limited influence over the resulting trajectories via a consistently small mass flow rate. However, the initial thrust-to-mass ratio is set equal to one of five discretely sampled values: $T/m_0 \in [3\times10^{-5}, 6\times10^{-5}, 9\times10^{-5}, 1.2\times10^{-4}, 1.5\times10^{-4}]$. Finally, the control profile associated with a trajectory is defined using a constant thrust magnitude and constant direction in the VNC axes, measured relative to the Moon and using an inertial observer.

In this paper, the thrust vector is oriented by an in-plane angle θ, measured relative to the $\hat{V}$ axis and in the $\hat{V}\hat{C}$-plane, that is set equal to one of 12 equally-spaced values $\theta \in [0°, 30°, ..., 330°]$.

The initial position vectors and Jacobi constants of possible initial conditions are randomly sampled using a grid structure. First, a uniform grid is constructed in the configuration space using steps of $\Delta_{pos} = 0.004$ in each of the x and y variables. Then, the C_J range is discretized into equally-sized intervals with a length of 0.002. These values are selected empirically, using previous clustering-based summaries by Bosanac[27] as a reference, to balance producing a sufficient grid resolution with reducing the computational burden of generating and summarizing a large set of trajectories. Within each grid element in the configuration space and interval in the Jacobi constant, values of x, y, and C_J are randomly generated from a uniform distribution. If a combination of x, y, and C_J produces a real-valued, nonzero speed in the natural CR3BP, it is retained for use in generating the initial conditions of trajectories in the training dataset.

Initial conditions of the sampled trajectories are defined as curvature maxima. Following an approach developed by Bosanac,[27] these maxima in the curvature supply a geometrically meaningful location along a trajectory. Consider a specified position vector defined by x_d and y_d, desired Jacobi constant $C_{J,d}$ which determines the desired speed v_d, initial thrust-to-mass ratio $T/m_0|_d$, and thrust direction θ_d. For this combination of variables, the velocity vector $\bar{v} = [\dot{x}, \dot{y}]^T$ that produces a stationary point in the curvature is calculated by solving

$$\frac{d\kappa}{dt} = 0 \qquad \text{and} \qquad ||\bar{v}||/v_d - 1 = 0 \tag{8}$$

where the curvature κ depends on $T/m_0|_d$ and θ_d due to their influence on the acceleration. These two equations are solved using MATLAB's fsolve.[21] Once a solution is produced, the second time derivative of curvature $\ddot{\kappa}$ is calculated. If $\ddot{\kappa} < 0$, a maximum in the curvature has been calculated, and the solution is retained to produce a full state vector $\bar{\chi} = [x_d, y_d, \dot{x}, \dot{y}, 1]^T$. This state vector is stored with $T/m_0|_d$ and θ_d.

A trajectory is generated from each initial condition by propagating forward in time for a duration that is defined geometrically. Specifically, each trajectory is propagated until passing through four subsequent curvature maxima. As presented by Gillespie, Miceli, and Bosanac,[8] this definition accommodates trajectories passing through distinct regions of the phase space with distinct speeds. In many cases, the trajectory completes approximately two revolutions in the rotating frame. For more geometrically complex trajectories, the trajectory may trace out a smaller path with additional local maxima in the curvature occurring as the trajectory transitions between concave and convex motion. Propagation terminates early, however, if any of the following conditions are satisfied:

1. Exceeding a distance of one nondimensional unit from the Moon.
2. Impacting the surface of the Earth or Moon, modeled as spheres with radii of 6,378.137 km and 1,738 km, respectively.[15]
3. The VNC axes are close to becoming undefined. This condition occurs when either 1) the position and inertial velocity vectors, measured relative to the Moon, are parallel or antiparallel, to within 5 degrees; or 2) the speed passes below 10^{-6} nondimensional units.
4. The Jacobi constant either exceeds 3.3 or falls below 3.0; although C_J is not constant for thrust-enabled motion, this condition bounds the region of the phase space.
5. The spacecraft mass falls below 75% of the initial wet mass, i.e., $m \leq 0.75$.

For any trajectory that terminates early, its associated termination condition is stored.

Trajectories are propagated numerically in the thrust-enabled Earth-Moon planar CR3BP. Numerical integration is implemented in C++ using the Runge-Kutta Prince-Dormand (8, 9) method in the GNU Scientific Library.[29] During propagation, each event is expressed in the form of a scalar function $e(\bar{\chi}) = 0$ and the state along the trajectory that occurs at this event is identified via root-finding. If the sign of $e(\bar{\chi})$ changes between the beginning and end of each time step by the integrator, Brent's method[30] is used to identify the event, if one exists, within that interval. These events are computed if $e(\bar{\chi}) = 0$ to within a relative tolerance of 10^{-8}, calculated using the values of $e(\bar{\chi})$ at the beginning and end of the bounding time interval.

Each continuous trajectory is segmented geometrically to support constructing a finite-dimensional description.[8] First, each trajectory is sampled by $2 \leq N_K \leq 5$ coarsely-generated states: all local maxima in curvature, including the initial state, and the final state if early termination occurs. Then, between each pair of subsequent coarsely-generated states, $N_s = 2$ finely-generated samples are distributed evenly in the elapsed arclength. Accordingly, each trajectory is sampled using $N_a = (N_k - 1)(N_s + 1) + 1$ states that are geometrically distributed to sufficiently capture its shape. Each pair of sampled states also defines an interval along the trajectory, labeled as a segment;[8] this information is used to describe flow along the trajectory and support subsequent prediction tasks. Thus, a trajectory that is sampled using N_a states is also discretized into $N_a - 1$ segments.

A finite-dimensional description of a trajectory is defined using feature vectors that are calculated at each sampled state to capture its shape and location within the configuration space.[27] At the ith sampled state along a trajectory, two feature vectors describe the local shape: a tangent-based feature vector $\bar{f}_T$ and a feature vector $\bar{f}_N$ that uses the principal normal. These two-dimensional feature vectors are defined for the ith sampled state as

$$\bar{f}_{T,i} = [\hat{T}_{x,i}, \hat{T}_{y,i}] \qquad \bar{f}_{N,i} = [\hat{N}_{x,i}, \hat{N}_{y,i}] \tag{9}$$

where $\hat{T}_{j,i}$ and $\hat{N}_{j,i}$ are the components of $\hat{T}$ and $\hat{N}$ along the jth axis of the Earth-Moon rotating frame. Then, a position-based feature vector is defined at the ith sampled state as

$$\bar{f}_{p,i} = [x_i, y_i] \tag{10}$$

These three feature vectors are separated, rather than combined, to support clustering in each type of distinct quantity without additional scalar weights.

Each entry into the training dataset consists of the information needed to recreate the trajectory, its finite-dimensional description, and its behavioral label. First, the initial state vector and its duration are stored. The behavioral label includes the spacecraft parameters; thrust direction, i.e., maneuvering objective; and termination condition. Finally, the feature vectors and number of sampled segments along the trajectory are stored to support clustering. Following the procedure outlined in this section, the training dataset constructed in this paper consists of 40,055,363 entries.

Step 2: Clustering Trajectories in the Training Dataset

Clustering techniques are used to group the trajectories contained in the training dataset by their geometry. This paper leverages a procedure that has been developed in recent years by Bosanac,[27] relying on distributed clustering strategies for computational efficiency, density-based clustering methods, and convoy detection schemes from trajectory clustering. First, the entries into the training dataset are separated into smaller partitions. Then, the members of each partition are grouped to produce local clusters. These local clusters are then aggregated across partitions to produce a set of global clusters that summarize a complex training dataset.

The training data are partitioned to create smaller clustering problems and simplify the clustering logic. Specifically, subsets of no more than 10,000 trajectories are formed to possess the same values of $T/m_0|_d$ and θ_d, i.e., the same spacecraft hardware and maneuvering approach. In this paper, 5,005 partitions are generated and typically each consist of 6,666-9,997 trajectories.

The trajectories within each partition are clustered in a procedure that relies on convoy detection schemes from trajectory clustering. In convoy detection schemes, moving objects are grouped together in a 'convoy' if they consistently follow similar paths throughout their duration.[31,32] This assessment is generated by discretizing the path of each object into a sequence of samples or segments.[32] The ith sample or segment along all paths are described by a feature vector. If multiple moving objects possess connected neighborhoods in this feature vector space, they are grouped. Often this assessment is made using a density-based clustering algorithm such as DBSCAN to also enforce that a sufficient number of moving objects must exist with these neighborhoods.[31] This assessment is performed for all samples or segments along the paths of the moving objects. If any objects are consistently grouped together for their entire duration, they form a 'convoy'.[32] This general procedure has been adapted by Bosanac,[27] in a two-step procedure that accommodates the complexity of trajectories in a multi-body system.

Coarse, shape-based clusters are first generated by grouping trajectories with similarly shaped paths. At the ith sampled state, the two feature vectors $\bar{f}_{T,i}$ and $\bar{f}_{N,i}$ describe the local shape. The constant radius of the neighborhood of each member is defined as $\epsilon_T = 2\sin(10°)$ in $\bar{f}_{T,i}$ and $\epsilon_N = 2\sin(10°)$ in $\bar{f}_{N,i}$. Furthermore, density is assessed using the definition from DBSCAN, with modifications to accommodate the use of two feature vectors, as presented by Birant and Kut in the adapted ST-DBSCAN algorithm.[33] Specifically, a member of a dataset is sufficiently dense when at least $m_{pts} = 4$ additional members exist within each of its ϵ_T- and ϵ_N-neighborhoods in $\bar{f}_{T,i}$ and $\bar{f}_{N,i}$, respectively. This calculation is implemented by clustering in each of $\bar{f}_{T,i}$ and $\bar{f}_{N,i}$ using DBSCAN to produce two clustering results, each consisting of the following information: the cluster labels and an adjacency matrix encoding which members exist in each others' neighborhoods. Repeating this process for a set of trajectories that is described by N_a samples produces a total of $2N_a$ clustering results. Any trajectories that are grouped together and remain within each others' neighborhoods in all $2N_a$ clustering results form a candidate cluster. Trajectories that are clustered together are also constrained to 1) terminate for the same reason, i.e., satisfying the same termination condition or passing through the selected number of curvature maxima; and 2) be sampled by the same number of states N_a. If a candidate cluster consists of at least $m_{min,clust} = 5$ members, it is retained as a coarse, shaped-based cluster. Any trajectories designated as noise are discarded.

Each coarse, shape-based cluster is then refined to produce zero or more clusters of trajectories that follow neighboring paths throughout the configuration space. At the ith sampled state, all three feature vector descriptions are employed, i.e., $\bar{f}_{T,i}$, $\bar{f}_{N,i}$, and $\bar{f}_{p,i}$. The radius of the neighborhood of each member of the dataset in $\bar{f}_{T,i}$ and $\bar{f}_{N,i}$ is defined using the same constant values of ϵ_T in $\bar{f}_{T,i}$ and ϵ_N in $\bar{f}_{N,i}$ as in the previous step. However, in $\bar{f}_{p,i}$, neighborhood radii ϵ_p are nonconstant across sampled states and/or between trajectories, consistent with the complex and nonlinear solution space. The procedure for heuristically estimating ϵ_p for each sampled state is described in the next paragraph. The ith samples along two trajectories, j and k, lie in each others' neighborhood if the distance between them satisfies the following conditions in each feature vector space:

$$d_{i,j,k,T} \leq \epsilon_T \qquad \text{in } \bar{f}_{T,i} \tag{11}$$
$$d_{i,j,k,N} \leq \epsilon_N \qquad \text{in } \bar{f}_{N,i} \tag{12}$$
$$d_{i,j,k,p} \leq \max(\min(\epsilon_{pos,i,j}, \epsilon_{pos,i,k}), \epsilon_{p,thresh}) \qquad \text{in } \bar{f}_{p,i} \tag{13}$$

In these conditions, $d_{i,j,k,T}, d_{i,j,k,N}, d_{i,j,k,p}$ are the distances between the ith samples along trajectories j and k in $\bar{f}_{T,i}$, $\bar{f}_{N,i}$, and $\bar{f}_{p,i}$, respectively, whereas $\epsilon_{p,thresh} = 0.008$ is a threshold value. Using these definitions, samples are clustered together if they lie in connected neighborhoods; density-based criteria are no longer applied at this step and, therefore, DBSCAN is not employed. This procedure generates three independent clustering results. After repeating this process for all N_a sampled states along a trajectory, $3N_a$ clustering results are generated. Any sets of at least $m_{min,clust} = 5$ trajectories that are consistently grouped together and remain in each others' neighborhoods form a refined cluster. However, trajectories designated as noise are discarded. Through this approach, the ℓth partition is summarized by $N_{c,\ell}$ clusters $\{\mathcal{C}_1^\ell, ..., \mathcal{C}_{N_{c,\ell}}^\ell\}$.

To estimate the neighborhood radius in the position-based feature vector space for each sample, heuristics are employed. First, for the ith sampled state along the jth trajectory, nearby trajectories are loosely defined as those with an initial position vector within a tolerance of $\delta_{pos} = 0.016$ and an initial Jacobi constant within a tolerance of $\delta_{CJ} = 0.008$; both values are integer multiples of the sample resolution for generating initial conditions. The m_{pts}th largest distance between the ith sampled state along the jth trajectory and the ith sampled state along nearby trajectories is equal to half the estimated neighborhood radius, $\epsilon_{pos,i,j}$. This heuristic estimates the neighborhood radius in a data-driven manner using only trajectories that begin from a similar region of the phase space and already possess a similar shape, while adding some margin. Unlike a constant value of ϵ_p, this estimate evolves with the trajectory.

Each local cluster is summarized by a representative member. Following the approach used by Bosanac[34] as well as Smith and Bosanac,[35] this representative is selected as the medoid of the cluster, i.e., the member of a cluster that is most similar to all other members,[36] in the position-based feature vector space calculated along the entire trajectory. Mathematically, this medoid $\mathcal{T}_{j,m}$ is calculated for the jth cluster of trajectories, each labeled $\mathcal{T}$, as

$$\mathcal{T}_{j,m} = \arg \min_{\mathcal{T}_i \in \mathcal{L}_j} \sum_{k=1}^{|\mathcal{L}_j|} ||\bar{f}_p(\mathcal{T}_i) - \bar{f}_p(\mathcal{T}_k)|| \tag{14}$$

where $\bar{f}_p = [\bar{f}_{p,1}, ..., \bar{f}_{p,N_a}]$ is composed of the position vectors of all N_a sampled states.

The local clusters are then merged across partitions, if they possess a similar geometry, to produce a global cluster summary. This procedure is implemented following the approach presented by Bosanac.[27] Each local cluster is described by the feature vectors of its representative member, i.e., $\bar{f}_p = [\bar{f}_{p,1}, ..., \bar{f}_{p,N_a}]$ and $\bar{f}_T = [\bar{f}_{T,1}, ..., \bar{f}_{T,N_a}]$. Then, the feature vectors of the representative members of the local clusters that exist in other partitions with the same values of $T/m_0|_d$ and θ_d are isolated. Then, up to k representatives with the nearest vectors $\bar{f}_p$ and $\bar{f}_T$ to the local cluster of interest are identified as candidates for merging. This step produces up to $2k \sum_{i=1}^{P_{total}} N_{c,i}$ candidates across the local clusters of all P_{total} partitions. Each pair of candidate local clusters is then compared by repeating the cluster refinement procedure. If any members of the two clusters are grouped together, the local clusters are merged. Following these pairwise merging decisions, a global cluster is formed by a connected set of local clusters.

The aggregation method produces a total of N_g global clusters $\{\mathcal{G}_1, ..., \mathcal{G}_{N_g}\}$ that summarize the entire training dataset. In this paper, between 21 and 111 local clusters are identified per partition, grouping between 89.85% and 98.70% of the associated trajectories. Thus, a total of 329,686 local clusters are generated across the $P_{total} = 5,005$ partitions. Following aggregation, 47,947 global clusters are generated to summarize the training data, with each global cluster corresponding to one combination of the initial thrust-to-mass ratio and control profile.

Step 3: Extracting the Behavioral Motion Primitive and its Region of Existence

Each global cluster from the previous step is summarized by a behavioral motion primitive. This trajectory is defined as the representative member of the entire global cluster, calculated as the medoid, using the same approach applied to local clusters. The behavioral labels of this motion primitive are extracted from the representative member and possess the same description as all other trajectories in the global cluster, conveying the following information: any early termination condition that is satisfied; the possible spacecraft hardware via T/m_0; and the possible maneuvering objectives via θ_0 that defines the constant thrust vector in the VNC axes.

Each behavioral motion primitive in the library is accompanied by an approximation of its region of existence. This term describes a volume in the phase space spanned by the other trajectories in the global cluster with a similar geometry, as originally defined by Smith and Bosanac.[35] This paper leverages an improved approach for representing this volumetric region by using the concept of a voxel approximation, as presented by Bodin, Bosanac, and Gillespie.[12] This procedure is similar in concept to the "cells" described by Howell et al.[37] for approximating manifold surfaces, but constructed with distinct geometric properties and encoding additional metadata.

Voxel approximations of the region of existence in the four-dimensional phase space are formed by a hierarchical set of variable-sized voxels that intersect the trajectories in a global cluster. First, a set of two-dimensional position voxels are defined in the configuration space coordinates x and y. The ith position voxel possesses a side length or size $s_{pos,i}$ and is centered on $\bar{c}_{pos,i}$. Each of these position voxels is then hierarchically associated with a set of two-dimensional velocity voxels defined in the coordinates $\dot{x}$ and $\dot{y}$. Similarly, the jth velocity voxel possesses a size $s_{vel,j}$ and is centered on $\bar{c}_{vel,j}$. When a pair of position and velocity voxels are combined during the prediction process, the pair produces a hypervoxel that is four-dimensional. Each set of voxels along the same section is also associated with metadata that describes other quantities that may vary along the global cluster and be useful during trajectory and behavioral prediction. In this paper, these metadata include the elapsed time and the acceleration, i.e., $(T/m_0)(1/m)$, discretized in increments of 1×10^{-8}, of any states that intersect the voxel. Note that the word 'voxel' is used here, rather than pixel, to support ongoing work extending this implementation to spatial, low-thrust trajectories where each voxel contains three-dimensional information.

Only position and velocity voxels that intersect with states along any trajectories are stored. To identify these intersections, a point cloud is constructed to represent the position and velocity vectors along each section of the trajectories in a global cluster. Specifically, states are sampled along each section at equal intervals in the traditional arclength in the configuration space, s_{pos}, and the arclength of the velocity hodograph, s_{vel}. In the configuration space, states are sampled at arclength intervals of $d^s_{pos} = 0.05 l^{min}_{pos}$, where l^{min}_{pos} is a constant, minimum voxel size. This definition bounds the distance that a trajectory can traverse through the smallest possible voxel without registering a sample.[12] When sampling the velocity hodograph along each section, the intervals are defined as $d^s_{vel} = 0.05 l^{des}_{vel}$ where l^{des}_{vel} is a desired voxel size, defined in a subsequent paragraph.

The center and size of the position and velocity voxels representing a section of a cluster of trajectories are calculated in a data-driven manner. These values vary as needed to accommodate the sparsity of the training data and nonuniform sensitivity of the solution space, limit gaps in the voxel sets, and limit the computational memory burden. Furthermore, these values are calculated for each section along the trajectories in a global cluster. Thus, position voxel sizes may be larger along sections where the trajectory traverses large distances in the exterior region but smaller along sections that pass closer to the Moon. Ongoing work includes more strategically increasing the size of voxels only if they are interior to the region of existence.

Position voxel centers and sizes are selected along each section to conform to a predefined grid structure in x and y. The voxel size is calculated as one of the following integer multiples of the minimum size $l_{pos}^{min} = 0.01$ non-dimensional units: 1, 3, or 9. This voxel sizing scheme preserves the grid-alignment of the voxel centers at values of x and y that equal integer multiples of l_{pos}^{min}, allowing for simpler evaluations of voxel intersections. Limiting the growth factor to $9l_{pos}^{min}$ ensures that the resolution remains useful in accurately describing the volume of the region. When either of the following two conditions are satisfied, the multiplicative factor is increased from the default value of one: 1) if the third largest distance between samples along neighboring trajectories at a section boundary exceeds the current voxel size, or 2) if the longest arclength of a section along a trajectory exceeds a user-defined threshold. Thus, position voxel sizes within a section increase as the data is more sparse or the trajectory traverses a larger distance along one section.

Velocity voxel centers and sizes are selected based on the variation in the velocity vector along a section and not constrained to follow a consistent grid structure. All velocity voxel sizes are required to meet or exceed a minimum size of $l_{vel}^{min} = 0.01$ non-dimensional units. The size is increased as needed to ensure that approximately ten velocity voxels span the width of a section, balancing resolution and storage space. The centers of these voxels are then positioned at integer multiples of the selected size.

To demonstrate the computation of a voxel approximation of a region of existence, consider a set of trajectories in a global cluster with $T/m_0 = 3 \times 10^{-5}$, an I_{sp} of 1400 seconds, and $\theta = 0°$. Figure 1 displays a) the position voxels that envelope these trajectories and b) the associated velocity voxels. In these subfigures, the voxels in section $\mathcal{S}_i$ are highlighted in blue with a position voxel in red corresponding to red velocity voxels. Then, the thrust vector and thrust-to-mass ratio associated with these trajectories is plotted in Figure 1c). In this example, these trajectories are generated for a spacecraft with the lowest capacity propulsion system that is thrusting in the velocity direction.

To represent the flow of trajectories across a voxel-based approximation of a region of existence, a set of subrepresentative arcs is also extracted. One subrepresentative is calculated for every three position voxels spanning the section at the widest section boundary. Subrepresentative arcs are identified by applying the well-known k-medoids clustering algorithm[38] to the position-based feature vector space evaluated at the sample at the beginning of the section; this clustering technique can produce approximately evenly-sized clusters. Each subrepresentative inherits the behaviors of its neighbors that produce similar motion. Voxels within the section are also assigned one or more integer identifiers locating any intersecting subrepresentatives. For the example in Figure 1, the evolution of position and velocity along the subrepresentatives is plotted using black curves. In these subfigures, the jth subrepresentative along the ith section is labeled by the notation $\mathcal{P}_{i,j}$.

Between sections, the connectivity of subrepresentatives is described by a representative flow graph. Each layer of the graph corresponds to a section within the region and each node within the

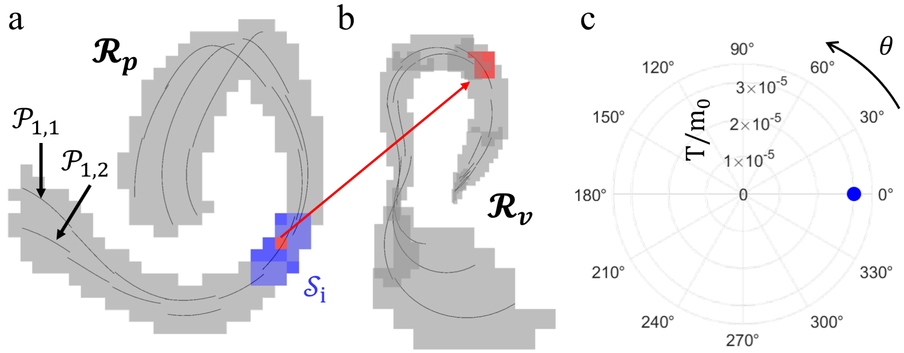

Figure 1. Volumetric approximation of a region of existence for a global cluster: a) position voxels, b) velocity voxels, c) thrust parameters

layer is a subrepresentative of that section. If the final position of a subrepresentative is within l_{pos}^{min} of the initial position of a subrepresentative in the subsequent section when projected into the plane normal to the tangent vector of the current subrepresentative, then the two nodes are connected by a directed edge. If a subrepresentative has no outgoing or incoming connections, it is connected to the nearest neighboring subrepresentative in the adjoining sections.

To demonstrate the process for representing the flow through a region of existence, consider the voxel approximation from Figure 1. The associated region of existence is displayed in the configuration space in Figure 2a), with the voxels colored by the section of the intersecting trajectories: the initial conditions occur within some of the cyan voxels. The subrepresentatives within each section are plotted as small black arcs, labeled by a unique identifier. The associated reachability graph is depicted in Figure 2b) with each node labeled by the associated subrepresentative. This region of existence and the associated reachability graph concisely summarizes the cluster geometry.

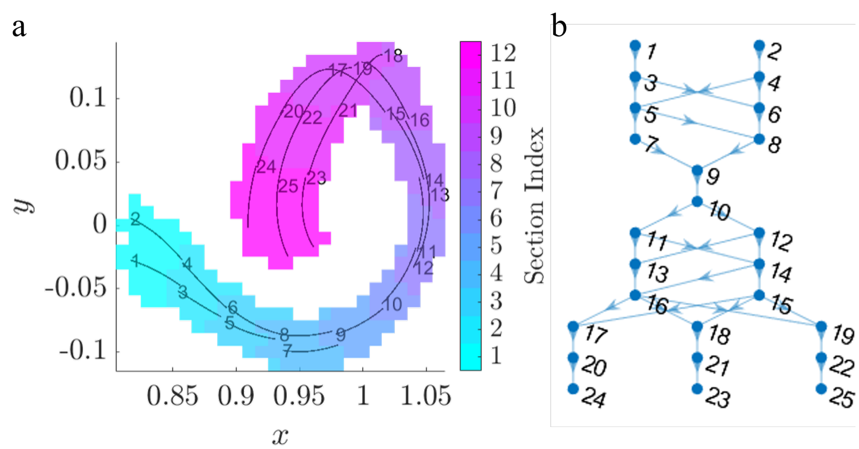

Figure 2. Sample region of existence described by a) a set of subrepresentative arcs and b) a reachability graph that summarizes their connectivity.

Behavior Extraction from a Sequence of Uncertain State Estimates

The library of motion primitives and their regions of existence summarize possible motions and behaviors in the system that can be compared with state estimates.[12] Uncertain state estimates are represented as six-dimensional hyperellipsoids centered at an estimated state, $\bar{X}$, and encompassing 3σ uncertainty in the combined position and velocity space. Then, each hyperellipsoid is compared with each voxel approximation of a region of existence in the library to identify an intersection. To determine if a region intersects with a hyperellipsoid, a set of hypervoxels is generated as a four-dimensional discrete representation of the state-space formed by combining each position voxel with each of its velocity voxels. For a region to intersect with a hyperellipsoid, it is sufficient for a single hypervoxel belonging to a region to intersect.

Hypervoxel-hyperellipsoid intersections are checked in several steps, following the process outlined by Bodin, Bosanac, and Gillespie.[12] First, a coarse center-to-center check is performed. If the distance between the two centers is greater than the sum of the 3σ-scaled semi-major axis of the hyperellipsoid and the center-to-corner distance of the hypervoxel, then there is no possibility of intersection. If this condition is not satisfied, then further investigation is required. If the center of the hypervoxel lies within the hyperellipsoid bounds or vice versa, then an intersection is recorded. If the previous checks fail, then a gradient-descent algorithm is used to identify a state that lies along the surface of the hyperellipsoid and within the bounds of the hypervoxel. If a solution is recovered within 100 iterations, then an intersection is identified. Otherwise, the hypervoxel and hyperellipsoid are deemed non-intersecting.

The consolidated set of regions that intersect with a 3σ uncertainty hyperellipsoid provide behavioral information, bounding the possibilities for the observed spacecraft. For example, if all the regions associated with an uncertain state estimate contain thrust vectors closely aligned with the $+\hat{V}$ vector, the prediction process may indicate that the spacecraft is thrusting along the velocity vector. Alternatively, if a wide range of thrust angles and/or initial thrust-to-mass ratios are extracted from the intersecting regions, the current estimate may correspond to either 1) one or more predicted paths that are not sensitive to the thrust and spacecraft parameters or 2) a wide variety of possible motions that cannot be adequately predicted with only one estimate. In these cases, a sequence of infrequent estimates may be used to narrow down possible trajectory geometries and/or governing parameters, or even detect behavioral changes.

The regions of existence associated with subsequent estimates are connected to generate trajectory and behavioral predictions. Consider a sequence of multiple estimates. Each estimate is associated with a set of intersecting regions of existence. Then, the potential for two or more estimates to be described by a sequence of one or more regions of existence. For two regions to be sequenceable between estimates, they must meet five criteria:

1. There must be an intersection of the position voxels of the two regions
2. The velocity voxels must be close for at least one of the intersecting position voxels
3. Under the assumption of constant thrust magnitude, the acceleration of the spacecraft must remain consistent between region transitions
4. The time traversing the region or regions between estimates must match the known time between the estimates
5. To prevent motion transverse to the flow within a region, the subrepresentatives between estimates must be composable

Criteria 1 and 2 are assessed at the voxel level and criteria 3-5 are assessed between sections.

Position voxel intersections are checked using two approaches, depending on the size of the voxels. Each pair of sections along two regions of existence are compared. If position voxels in the pair of sections are the same size, their center positions are compared. Then, if two voxels share the same center, they intersect. If the position voxels of the two sections possess different sizes, then their boundaries are checked for overlap. For all intersecting position voxels, their associated velocities are compared. If each projection of the distance between two velocity voxel centers onto each of the principal axes is less than or equal to half the side length of the larger velocity voxel, then the two voxels are considered to be sufficiently close in velocity.

Each section of a region supplies a set of time ranges, $[t_0, t_f]$, that describe the possible elapsed time from the start of the region to the current section. The set of time ranges across all sections that intersect with estimate i is represented by $\mathcal{E}_i$, with $\mathcal{E}_i^-$ denoting the lower-bounds of all the ranges and $\mathcal{E}_i^+$ denoting the upper-bounds. The Minkowski difference of the time ranges between the incoming and outgoing estimates provides both a lower bound, $\mathcal{L} = \mathcal{E}_2^- \ominus \mathcal{E}_1^+$, and an upper bound, $\mathcal{U} = \mathcal{E}_2^+ \ominus \mathcal{E}_1^-$, for the time spent within the region. For subsequent estimates within the same region, it must be true that $\exists\ i \ \mid\ \mathcal{L}_i \leq \Delta t_{obs} \leq \mathcal{U}_i :\ i = 1 \ldots n$ where Δt_{obs} is the known time between estimates and n is the number of elements in $\mathcal{L}$ and $\mathcal{U}$. For subsequent estimates that are associated with separate intersecting regions, the possible times of intersection between the regions must be considered. $\mathcal{L}$ and $\mathcal{U}$ are modified such that $\mathcal{L}' = (\mathcal{E}_\alpha^- \ominus \mathcal{E}_1^+) \oplus (\mathcal{E}_2^- \ominus \mathcal{E}_\beta^+)$ and $\mathcal{U}' = (\mathcal{E}_\alpha^+ \ominus \mathcal{E}_1^-) \oplus (\mathcal{E}_2^+ \ominus \mathcal{E}_\beta^-)$ where $\mathcal{E}_\alpha$ are the time ranges of intersection after the estimate in the first region and $\mathcal{E}_\beta$ are the time ranges of intersection in the second region before the intersection. Then, $\exists\ i \ \mid\ \mathcal{L}_i' \leq \Delta t_{obs} \leq \mathcal{U}_i' :\ i = 1 \ldots m$ where m is the number of elements in $\mathcal{L}'$ and $\mathcal{U}'$.

The spacecraft initial thrust-to-mass ratio is assumed to remain constant along a trajectory and the spacecraft mass is assumed to decrease continuously. These two assumptions imply that the acceleration of the spacecraft must remain consistent between any intersecting sections. Each region contains a set of possible initial thrust-to-mass ratios within the cluster. This set is defined as $\mathcal{A} = \frac{T_i}{m_0} :\ i = 1, ..., n$ where n is the number of initial thrust-to-mass ratios in the region. Within each section, ranges of nondimensional masses are stored. Then, the nondimensional mass ranges for all sections of intersection are represented by $\mathcal{M}$ with $\mathcal{M}^-$ indicating the lower bounds of the ranges and $\mathcal{M}^+$ indicating the upper bounds. The lower (or upper) bound possibilities of acceleration within sections of intersection for a region, $\mathcal{A}^-$ (or $\mathcal{A}^+$), is formed from dividing all possible combinations of the elements in $\mathcal{A}$ by the elements in $\mathcal{M}^+$ (or $\mathcal{M}^-$). Similar to the time criterion, lower and upper bounds of possible acceleration range differences across two subsequent regions, 1 and 2, in their sections of intersection are created: $\mathcal{L} = \mathcal{A}_2^- \ominus \mathcal{A}_1^+, \mathcal{U} = \mathcal{A}_2^+ \ominus \mathcal{A}_1^-$ For the regions to be sequenceable, it must be true that $\exists\ i \ \mid\ \mathcal{L}_i \leq 0 \leq \mathcal{U}_i,\ i = 1 \ldots m$ where m is the number of elements in $\mathcal{L}$ and $\mathcal{U}$. A zero value between at least one lower and upper bound of the acceleration differences indicates a possibility of a continuous acceleration between the regions.

Maintaining subrepresentative continuity between subsequent estimates prevents motion that is transverse to the flow within a region of existence. Each voxel is associated with at least one subrepresentative within its respective section. The voxels that intersect with an estimate produce a set of associated subrepresentatives. Between estimates, at least one subrepresentative associated with the second estimate must be reachable in a graph search from a subrepresentative associated with the first estimate. If the estimates lie in different, intersecting regions, two graph searches are performed. Each intersecting voxel produces two sets of subrepresentatives, $\mathcal{P}_\alpha$ from the first region and $\mathcal{P}_\beta$ from the second. In this case, there must exist an intersecting voxel such that a member of $\mathcal{P}_\alpha$ is reachable from a subrepresentative associated with the first estimate and a subrepresentative associated with the second estimate is reachable from a member of $\mathcal{P}_\beta$.

After all valid connections between regions associated with sequential estimates are identified, a graph is constructed to represent all possible paths through the estimates via intersecting regions of existence. Regions corresponding to each estimate are organized into layers and represented by nodes. Edges between regions of subsequent layers indicate a valid connection between them as described in the steps above. Paths through the graph are then identified by searching the graph from the first to the last layer. Each path represents a sequence of one or more regions of existence that summarize the possible motion of the spacecraft.

RESULTS

The presented technical approach is applied to a sequence of eight simulated state estimates. These mean state estimates are sampled from a trajectory generated with $T/m_0 = 3 \times 10^{-5}$ and $\theta_0 = 0°$. Table 1 lists the components of the nondimensional mean state estimate along with the nondimensional epoch after the start of the trajectory. Six-dimensional hyperellipsoid covariances are constructed around each mean estimate with a maximum principal axis length that is several orders of magnitude smaller than either l_{min}^{pos} or l_{min}^{vel}, corresponding to a confident estimate of the state. These estimates are generated via a Monte Carlo simulation of trajectories from the truth value of the previous estimate with an initial 1σ position uncertainty of 100 m and a 1σ velocity uncertainty of 1 m/s.[39] Figure 3 displays the truth trajectory along with the mean state estimates at the eight epochs marked in magenta. This trajectory features a close pass by the Moon and a high apolune. These uncertain state estimates are projected onto the behavioral motion primitive library to generate predictions of the path and possible ranges of the spacecraft parameters and control profile. This section explores the results generated in this example through three cases using: 1) a single estimate, 2) a subset of three sequential estimates, and 3) all eight estimates. Through these distinct cases, the evolution of the trajectory and behavioral predictions is examined.

Case 1: Possible Behaviors from a Single Estimate

Consider a scenario where only the first estimate is available. By applying the primitive-based prediction approach, the first estimate is associated with 837 different clusters with 56 unique path

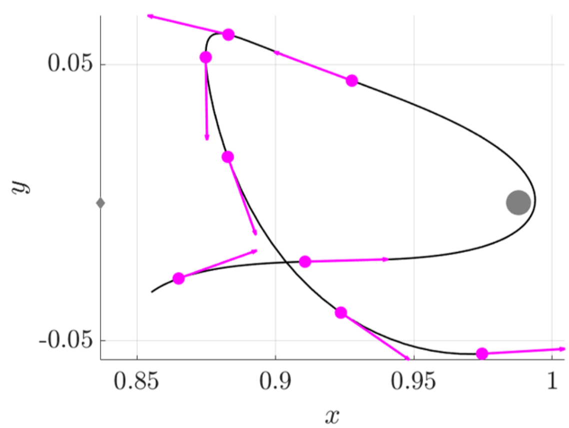

Figure 3. Eight mean state estimates along a planar trajectory generated with $T/m_0 = 3 \times 10^{-5}$ and $\theta_0 = 0°$

Table 1. Mean state estimates and epochs

	x	y	$\dot{x}$	$\dot{y}$	t_{obs}
$\bar{X}_1$	0.8650	-0.0274	0.0789	0.0284	0.2303
$\bar{X}_2$	0.9107	-0.0213	0.2970	0.0084	0.5103
$\bar{X}_3$	0.9276	0.0442	-0.3054	0.1137	0.7903
$\bar{X}_4$	0.8830	0.0608	-0.0701	0.0167	1.0703
$\bar{X}_5$	0.8748	0.0527	0.0012	-0.0779	1.3503
$\bar{X}_6$	0.8828	0.0166	0.0639	-0.1777	1.6303
$\bar{X}_7$	0.9237	-0.0398	0.2700	-0.1866	1.9103
$\bar{X}_8$	0.9747	-0.0546	0.4546	0.0265	2.0503

geometries. These unique path geometries are identified by aggregating clusters across neighboring spacecraft parameters and control profiles. As an example, six of these path predictions are shown in Figure 4. In these figures, the pink dot indicates the location of the estimate with the pink arrow aligned with the velocity vector. Blue trajectories are selected members of the cluster of an intersecting region of existence, with blue dots indicating their initial conditions. The Moon is located by a gray circle, drawn to scale, and the equilibrium points are depicted using gray diamonds. In this case, a single estimate could lead to a geometrically diverse array of possible paths that impact the Moon, perform one or more revolutions around the Moon, or pass through the L_1 gateway.

Multiple possible thrust parameters are predicted from this estimate. Figure 5 displays a heatmap of these possible parameters in terms of the angle describing the thrust vector and the initial thrust-to-mass ratio. The number inside each cell represents the number of clusters with that combination of parameters. For this estimate, the possibilities span the parameter space.

Case 2: Three-Estimate Sequence

If a constant behavior is assumed, each sequence of regions of existence that are associated with multiple estimates must have the same θ_0 and T/m_0. In cases such as this example where the esti-

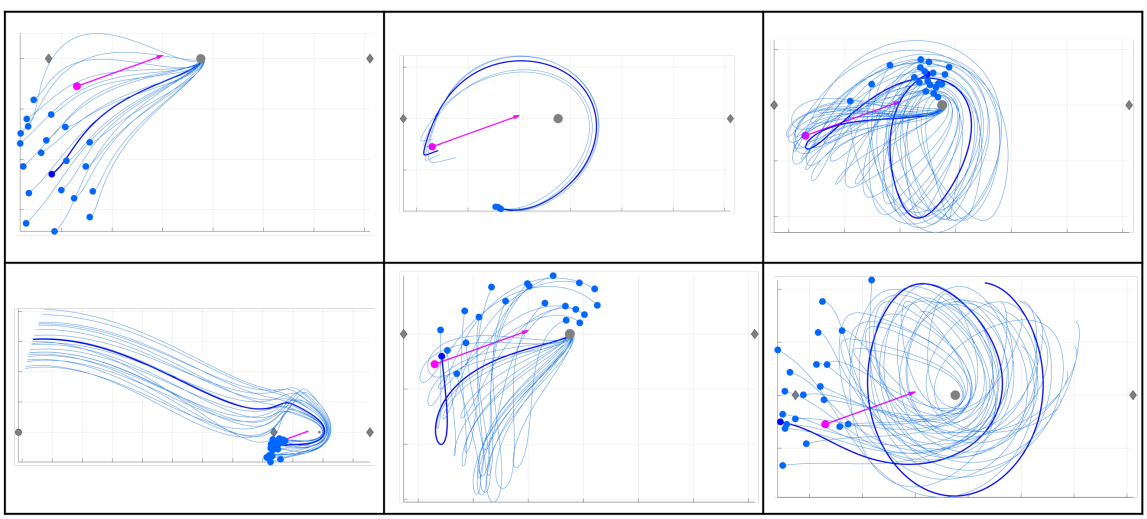

Figure 4. A subset of unique path predictions associated with the first estimate

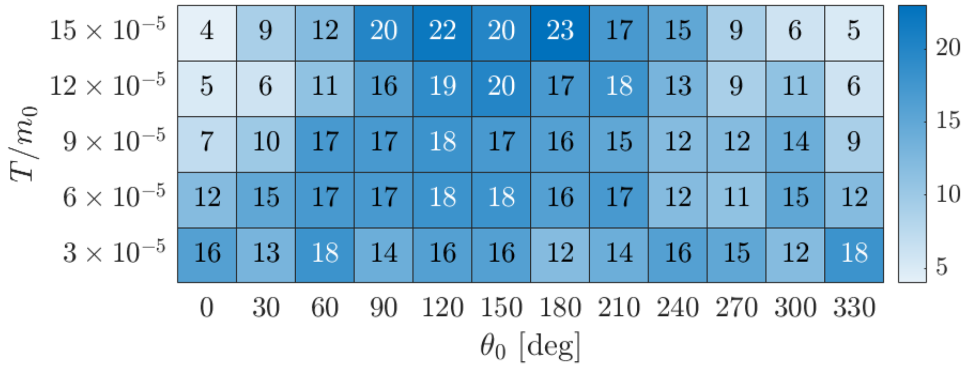

Figure 5. Spacecraft and thrust parameters associated with regions of existence that intersect the first estimate

mates remain within the span of one region of existence, constraints are applied to the graph: edges are only added between the same region across subsequent layers. Using the first three uncertain state estimates produces 304 possible regions that exhibit 19 geometrically distinct path predictions. Trajectories within selected regions of existence are plotted in Figure 6. Compared to Figure 4, these trajectories possess a similar geometry as the spacecraft approaches perilune, highlighting the effect of additional estimates on reducing possible path predictions. The differences in geometry occur before or after the sections that intersect these three estimates. Furthermore, the discrete representation of possible trajectories via the behavioral motion primitives produces an efficient summary of these estimates. Additionally, Figure 7 demonstrates the reduction in the number of options for each combination of thrust parameters.

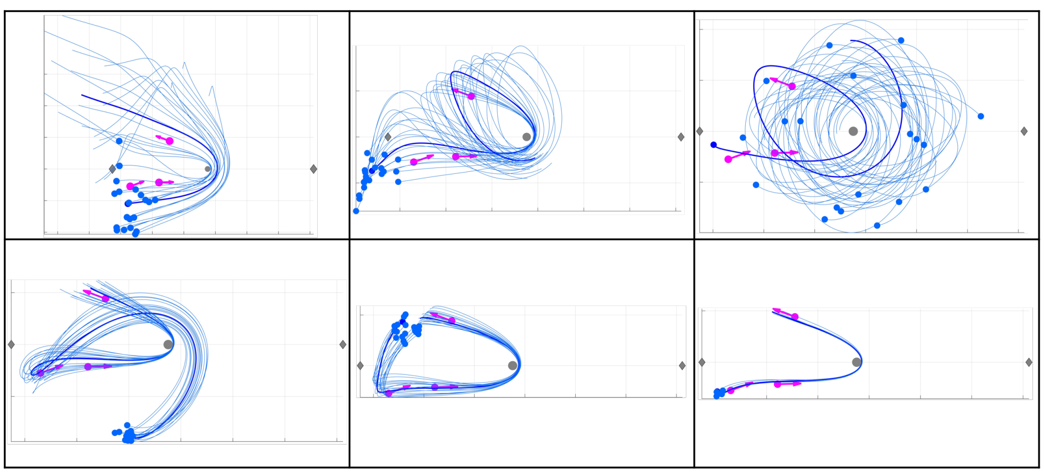

Figure 6. A subset of unique path predictions associated with the first three estimates

Case 3: Eight-Estimate Sequence

Using all eight estimates provides a more concise set of predicted paths and behaviors. Specifically, 96 regions of existence intersect all eight estimates and correspond to four geometrically distinct path predictions as plotted in Figure 8. With this set of estimates, the predicted path can be

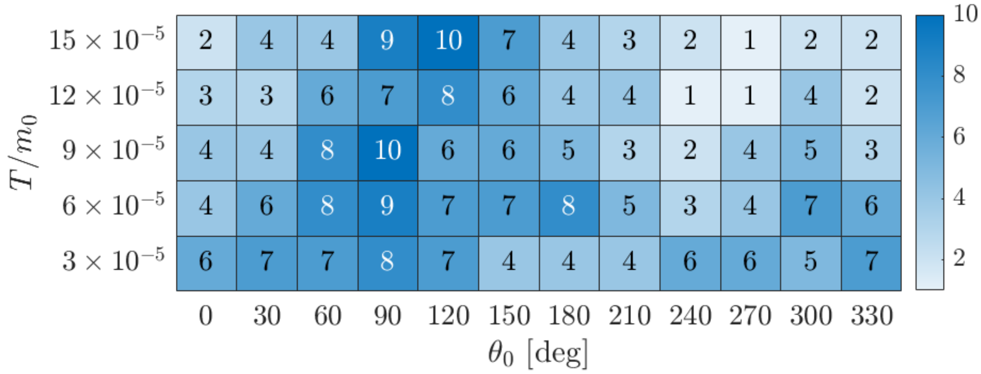

Figure 7. Spacecraft and thrust parameters associated with regions of existence that intersect the first three estimates

more accurately characterized, closely resembling the simulated trajectory. In fact, the simulated trajectory was formed from a member of the third cluster in Figure 8.

Figure 7 displays the reduction in thrust parameter options. When thrusting in the velocity direction, the selected geometries can only be recovered with a low capacity propulsion system. However, as the thrust vector approaches $-\hat{V}$, any of the modeled propulsion systems could produce that motion. With the concise set of trajectory and behavioral prediction, this approach provides a starting point to generate more precise estimates of the path followed by the spacecraft and/or more strategically target future observations of the spacecraft.

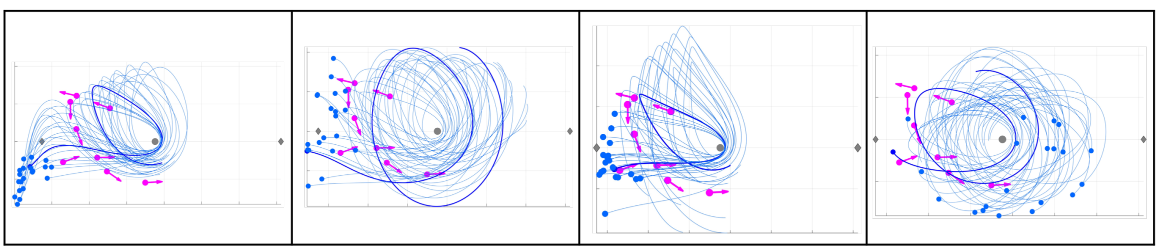

Figure 8. A subset of unique path predictions associated with the all eight estimates

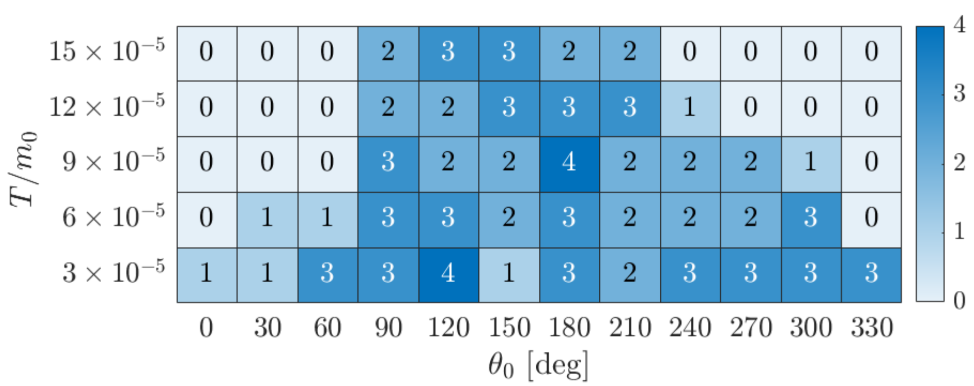

Figure 9. Spacecraft and thrust parameters associated with the eight estimates

CONCLUSIONS

This paper focuses on using behavioral motion primitives to support rapid trajectory and behavioral prediction for spacecraft that continuously thrust. First, the solution space is summarized via a library of behavioral motion primitives and voxel approximations of their regions of existence. Uncertain state estimates of spacecraft are then projected onto this library to identify intersecting regions of existence that could describe possible trajectories of the spacecraft. Composable regions, generated from sequential estimates, are used to improve these estimates of the possible paths. The common behaviors associated with these regions are used to predict the array of combinations of spacecraft parameters and control profiles that could produce these possible motions.

ACKNOWLEDGMENT

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-23-1-0235. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force.

REFERENCES

[1] B. D. Tapley, B. E. Schutz, and G. H. Born, *Statistical Orbit Determination*, pp. 194–199. New York: Elsevier, 2004.

[2] Y. z. Luo and Z. Yang, "A review of uncertainty propagation in orbital mechanics," *Progress in Aerospace Sciences*, Vol. 89, 2017, pp. 23–39, https://doi.org/10.1016/j.paerosci.2016.12.002.

[3] O. Dermy, A. Paraschos, M. Ewerton, J. Peters, F. Charpillet, and S. Ivaldi, "Prediction of Intention during Interaction with iCub with Probabilistic Movement Primitives," *Frontiers in Robotics and AI*, Vol. 4, 2017, 10.3389/frobt.2017.00045.

[4] S. Ferguson, B. Luders, R. C. Grande, and J. P. How, "Real-Time Predictive Modeling and Robust Avoidance of Pedestrians with Uncertain, Changing Intentions," *CoRR*, Vol. abs/1405.5581, 2014.

[5] G. Habibi, N. Jaipuria, and J. P. How, "SILA: An Incremental Learning Approach for Pedestrian Trajectory Prediction," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 4411–4421, 10.1109/CVPRW50498.2020.00520.

[6] A. Wolek and C. A. Woolsey, *Model-Based Path Planning*. Cham: Springer International Publishing, 2017, 10.1007/978-3-319-55372-6_9.

[7] T. R. Smith and N. Bosanac, "Motion Primitive Approach to Spacecraft Trajectory Design in a Multibody System," *The Journal of the Astronautical Sciences*, Vol. 70, No. 34, 2023, 10.1007/s40295-023-00395-7.

[8] C. Gillespie, G. E. Miceli, and N. Bosanac, "Summarizing Natural and Controlled Motion in Cislunar Space With Behavioral Motion Primitives," *AAS/AIAA Space Flight Mechanics Meeting*, 2025.

[9] G. E. Miceli and N. Bosanac, "Generating Planar Trajectories for Neptunian System Exploration Using Motion Primitives," *The Journal of the Astronautical Sciences*, Vol. 73, Feb 2026, p. 11, 10.1007/s40295-025-00545-z.

[10] N. Bosanac, "Using Motion Primitives to Generate Initial Guesses for Trajectories to Sun-Earth L2," *AAS/AIAA Astrodynamics Specialist Conference*, August 2025.

[11] N. Bosanac, "Using Motion Primitives to Design Thrust-Enabled Trajectories in the Earth-Moon System," *76th International Astronautical Congress*, September-October 2025.

[12] A. Bodin, N. Bosanac, and C. Gillespie, "Long-Term Spacecraft Trajectory Prediction Using Behavioral Motion Primitives," *AAS/AIAA Astrodynamics Specialist Conference*, 2025.

[13] V. Szebehely, *Theory of Orbits: The Restricted Problem of Three Bodies*. London: Academic Press, 1967.

[14] W. Koon, M. Lo, J. Marsden, and S. Ross, *Dynamical Systems, The Three-Body Problem, and Space Mission Design*. New York, USA: Springer, 2011.

[15] R. Park, W. Folkner, J. Williams, and D. Boggs, "The JPL Planetary and Lunar Ephemerides DE440 and DE441," *The Astronomical Journal*, Vol. 161, No. 105, 2021, 10.3847/1538-3881/abd414.

[16] B. A. Conway, *Spacecraft Trajectory Optimization*. Cambridge Aerospace Series, Cambridge University Press, 2010.

[17] K. Wardle, *Differential Geometry*. Mineola, NY: Dover Publications, Inc., 2008.

[18] N. Patrikalakis, T. Maekawa, and W. Cho, *Shape Interrogation for Computer Aided Design and Manufacturing*. Springer, Berlin, Heidelberg, 2009. E-book.

[19] T. Banchoff and S. Lovett, *Differential Geometry of Curves and Surfaces, 2nd Edition*. Boca Raton, FL.: CRC Press, 2016.

[20] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, AAAI Press, 1996, p. 226–231.

[21] The MathWorks Inc., "MATLAB version: 9.14.0 (R2023a)," 2023.

[22] J. Gomes, L. Velho, and M. Sousa, *Computer Graphics: Theory and Practice*. Boca Raton, FL: CRC Press, 2012. Ch. 8.

[23] H. Laga, Y. Guo, H. Tabia, R. Fisher, and M. Bennamoun, *3D Shape Analysis: Fundamentals, Theory, and Applications*. Hoboken, N.J.: Wiley, 2019. Ch. 2.

[24] A. Kaufman, D. Cohen, and R. Yagel, "Volume graphics," *Computer*, Vol. 26, No. 7, 1993, pp. 51–64, 10.1109/MC.1993.274942.

[25] M. Caon, "Voxel-based computational models of real human anatomy: a review," *Radiation and Environmental Biophysics*, Vol. 42, Feb. 2004, p. 229–235, 10.1007/s00411-003-0221-8.

[26] J. D. Foley, A. v. Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*. Reading, Massachusetts: Addison-Wesley, 2nd ed., 1995.

[27] N. Bosanac, "Clustering Natural Trajectories in the Earth-Moon Circular Restricted Three-Body Problem," *The Journal of the Astronautical Sciences*, Vol. 73, Jan 2026, 10.1007/s40295-025-00539-x.

[28] A. Hsu, H. Dragnea, J. Schilling, and A. Doumitt, "2023 Small Satellite Propulsion Technologies Compendium The Aerospace Corporation, DISTRO A Version 1.41," 2023.

[29] M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, P. Alken, M. Booth, and F. Rossi, *GNU Scientific Library Reference Manual - Third Edition*. Godalming, Surrey: Network Theory Limited, 2009.

[30] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes: The Art of Scientific Computing, 3rd Edition*. Cambridge University Press, 2007.

[31] Y. Zheng and X. Zhou, *Computing with Spatial Trajectories*. New York, NY: Springer Science and Business Media, 2011. DOI: 10.1007/978-1-4614-1629-6.

[32] H. Jeung, M. L. Yiu, X. Zhou, C. S. Jensen, and H. T. Shen, "Discovery of Convoys in Trajectory Databases," *Proceedings of the VLDB Endowment* (P. Buneman, B. Ooi, K. Ross, and G. Weber, eds.), VLDB Endowment, 2008. DOI: 10.14778/1453856.1453971.

[33] D. Birant and A. Kut, "ST-DBSCAN: An Algorithm for Clustering Sspatial–Temporal Data," *Data Knowl. Eng.*, Vol. 60, 2007, pp. 208–221. DOI: 10.1016/j.datak.2006.01.013.

[34] N. Bosanac, "Data-Mining Approach to Poincaré Maps in Multi-Body Trajectory Design," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 6, 2020, pp. 1190–1200, 10.2514/1.G004857.

[35] T. R. Smith and N. Bosanac, "Constructing motion primitive sets to summarize periodic orbit families and hyperbolic invariant manifolds in a multi-body system," *Celestial Mechanics and Dynamical Astronomy*, Vol. 134, No. 7, 2022, 10.1007/s10569-022-10063-x.

[36] P. Cichosz, *Data Mining Algorithms: Explained Using R*. West Sussex, United Kingdom: John Wiley and Sons, 2015.

[37] K. C. Howell, M. Beckman, C. Patterson, and D. Folta, "Representations of Invariant Manifolds for Applications in Three-Body Systems," *The Journal of the Astronautical Sciences*, Vol. 54, Mar 2006, pp. 69–93, 10.1007/BF03256477.

[38] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction To Cluster Analysis*. John Wiley Sons, Inc, 1990, 10.2307/2532178.

[39] C. Frueh, K. Howell, K. J. DeMars, and S. Bhadauria, "Cislunar Space Situational Awareness," *AAS/AIAA Space Flight Mechanics Meeting*, February 2021.