

# DATA-DRIVEN SUMMARY OF CONTINUOUS THRUST TRAJECTORIES IN A LOW-FIDELITY MODEL OF CISLUNAR SPACE

Natasha Bosanac\* and Maxwell Joyner†

In cislunar space, continuous thrust can substantially alter the path of a spacecraft. This paper uses clustering to automatically summarize the solution space for a continuously thrusting spacecraft in the Earth-Moon circular restricted three-body problem. First, a segment of the phase space is sampled in a geometry-based manner to generate a representative set of trajectories with fixed thrust directions. These trajectories are grouped via distributed clustering to extract a summary of their distinct types of geometries. This approach is demonstrated for a SmallSat beginning near the Moon with a constant thrust direction in the velocity-normal-conormal axes relative to the Moon.

## INTRODUCTION

Within the chaotic dynamical environment of cislunar space, continuous thrust propulsion systems can substantially influence the path of a spacecraft. Understanding the structure of the solution space when continuous thrust is applied is critical to future cislunar operations. Trajectory designers benefit from such insight to design complex and efficient paths throughout a system. Likewise, analysts must understand the array of possible thrust-enabled motions before predicting future paths of some observed spacecraft. However, the solution space is diverse, high-dimensional, and dependent upon hardware parameters and control laws. As a result, extracting insight into the structure of the solution space when continuous thrust is present is a timely but challenging problem.

Traditional dynamical systems approaches to manually analyzing the solution space have diminished applicability to studying continuous thrust trajectories. Preliminary analysis of the solution space in the Earth-Moon system typically starts with a representative, yet approximate, dynamical model such as the circular restricted three-body problem (CR3BP). In such a model, natural transport mechanisms are often explored using fundamental solutions such as equilibrium points, periodic and quasi-periodic orbits, and their stable or unstable manifolds [1]. In higher-fidelity and nonautonomous dynamical models, comparable types of natural solutions can exist over finite time intervals. However, when a sufficient level of continuous thrust is applied, motion may no longer be sufficiently approximated by natural transport mechanisms. Furthermore, the dimension of the

---

\*Assistant Professor, Colorado Center for Astrodynamics Research, Smead Department of Aerospace Engineering Sciences, University of Colorado Boulder, Boulder, CO 80303.

†Graduate Researcher, Colorado Center for Astrodynamics Research, Smead Department of Aerospace Engineering Sciences, University of Colorado Boulder, Boulder, CO, 80303.

Approved for public release; distribution is unlimited. Public Affairs release approval #AFRL-2024-3705. The views expressed are those of the author and do not reflect the official guidance or position of the United States Government, the Department of Defense, or of the United States Air Force.

problem increases with the specification of the thrust direction and the dependency on hardware parameters. Even techniques such as Poincaré mapping, typically used to simplify visualization and analysis of a wide array of trajectories, can become intractable in this higher-dimensional problem.

The challenges of manually extracting meaningful insight from a complex, higher-dimensional dataset exist across a wide variety of disciplines. In these ‘big data’ problems, clustering techniques have supported exploratory data analysis and knowledge discovery by automatically constructing groups of similar data [2]. As a subset of this problem, trajectory clustering has enabled the discovery of groups of trajectories with similar shapes or itineraries [3]. When used to study the trajectories of animals, humans, and vehicles, trajectory clustering has enabled the extraction of fundamental types of paths, frequently visited locations, and behaviors [3–7]. In astrodynamics, clustering has been used to detect regions of bounded motions near distant retrograde orbits on a Poincaré map, group periodic orbits with similar characteristics, and extract motion primitive sets summarizing families of periodic orbits and their hyperbolic invariant manifolds [8–11].

To address the challenges of analyzing a complex array of spacecraft trajectories, Bosanac as well as Bonasera and Bosanac have developed a data-driven framework for summarizing natural motions in a multi-body system [12–15]. The most recent implementation of this approach, developed by Bosanac [15], relies on two foundational steps: 1) sampling trajectories throughout a specified region of the solution space in a geometry-aware manner and 2) using distributed clustering to extract a global summary of the types of geometries exhibited by these trajectories. To sample the trajectories that are summarized, initial conditions are defined within a region of interest as maxima in the curvature that summarize a localized set of trajectories. Each continuous trajectory is discretized to capture its global shape by using a set of states that are evenly distributed in the total absolute curvature. Then, finite-dimensional feature vectors are used to describe the geometric variations along this trajectory using the velocity unit vector and position vector at each sampled state. Next, the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm is used to independently cluster trajectories in smaller partitions of the dataset to ensure computational feasibility [16]. The local clustering results for all partitions are then aggregated to produce a global set of clusters that summarizes the dominant geometries exhibited by the computed set of trajectories. This summary supplies a summary of the solution space.

This paper applies the data-driven framework developed by Bosanac [15] for summarizing spacecraft trajectories in a multi-body system to a new problem: extracting a summary of continuous thrust trajectories, generated over medium time horizons with a specified thrust direction, in the Earth-Moon CR3BP. To apply this formulation directly, this paper considers one constant thrust magnitude and constant specific impulse propulsion system for a SmallSat. For this proof of concept, the thrust direction is specified using various combinations of constant coefficients in the velocity-normal-conormal axes that are calculated relative to the Moon. Continuous thrust trajectories governed by these fixed thrust directions are then generated from states that occur near the Moon vicinity for at least 21 days; early termination occurs upon impact with the Moon or when reaching the  $L_1$  or  $L_2$  gateways. The clustering-based procedure is then used to extract groups of trajectories with distinct geometries that are summarized via representative members to simplify visualization. Similar trajectories are then associated across the distinct thrust directions to discover global types of continuous thrust trajectories. As a result, this paper aims to supply a proof of concept for the use of clustering to automatically generate a summary of the impact of one type of continuous thrust acceleration on the motions that a spacecraft could follow in a multi-body system with a specified array of thrust vectors.

## BACKGROUND

### Dynamical Model

The motion of a spacecraft operating in cislunar space is approximated using the circular restricted three-body problem (CR3BP). This model assumes that the Earth and Moon are spherically symmetric bodies following circular orbits about their barycenter [1]. The spacecraft is also assumed to have a negligible mass compared to the two primaries. Typically, mass, length, and time quantities are, respectively, nondimensionalized via the following characteristic quantities:  $m^*$  is the sum of the masses of the two primary bodies,  $l^*$  is the distance between the Earth and Moon, and  $t^*$  produces a mean motion of the two primaries that equals unity. This nondimensionalization scheme results in the definition of the mass ratio  $\mu$  as the ratio of the Moon's mass to the total mass of the primary system [1, 17].

The equations of motion for the CR3BP are formulated in a rotating frame to aid visualization and produce an autonomous dynamical system. This rotating frame is defined with the origin at the barycenter of the Earth-Moon system and the following axes:  $\hat{x}$  points from the Earth to the Moon,  $\hat{z}$  is parallel to the orbital angular momentum vector of the Earth and Moon, and  $\hat{y}$  completes the orthogonal right-handed triad [1]. The nondimensional state of a spacecraft within this frame is defined as  $\bar{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$ , where the notation  $(\dot{\cdot})$  represents a time derivative for an observer in the rotating frame. A pseudopotential function  $U^*$  is defined as

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

where  $r_1 = \sqrt{(x + \mu)^2 + y^2 + z^2}$  and  $r_2 = \sqrt{(x - 1 + \mu)^2 + y^2 + z^2}$ . Then, the nondimensional equations of motion for the spacecraft in the CR3BP are written in the rotating frame as

$$\ddot{x} = 2\dot{y} + \frac{\partial U^*}{\partial x}, \quad \ddot{y} = -2\dot{x} + \frac{\partial U^*}{\partial y}, \quad \ddot{z} = \frac{\partial U^*}{\partial z} \quad (2)$$

This dynamical model also possesses a constant of motion, the Jacobi constant, that is equal to

$$C_J = 2U^* - \dot{x}^2 - \dot{y}^2 - \dot{z}^2 \quad (3)$$

This quantity supplies a useful mechanism for categorizing state vectors and regions of allowable natural motion [1].

In this paper, a spacecraft is equipped with a continuous thrust propulsion system. This engine is modeled by a constant thrust of magnitude  $T$  and constant specific impulse  $I_{sp}$ . With the propulsion system continuously activated, the mass  $m$  of the spacecraft decrements at a constant rate equal to

$$\dot{m} = -\frac{Tt^*}{I_{sp}g_0} \quad (4)$$

in units of kg per nondimensional time, where  $g_0 = 9.81m/s^2$  is the gravitational acceleration on the surface of the Earth [18]. A partially nondimensionalized thrust magnitude is also defined to ensure dimensional consistency in the equations of motion such that  $T_{lt} = T (t^*)^2 / (1000l^*)$ . This paper considers a SmallSat with an initial wet mass of  $m_0 = 180$  kg, 40 kg of propellant, a thrust magnitude of 13 mN and a specific impulse of 1390 s [19]; this propulsion system is modeled after a Busek BHT-200 [20].

To model the motion of a spacecraft with a continuous thrust propulsion system, the equations of motion for the CR3BP are augmented by an additional acceleration term. First, the spacecraft state vector is modified to  $\bar{x}_{LT} = [x, y, z, \dot{x}, \dot{y}, \dot{z}, m]^T$  to capture the spacecraft mass. Using this definition, the equations of motion for a continuous thrust-enabled spacecraft in the CR3BP are written as

$$\ddot{x} = 2\dot{y} + \frac{\partial U^*}{\partial x} + \frac{T_{lt}u_x}{m}, \quad \ddot{y} = -2\dot{x} + \frac{\partial U^*}{\partial y} + \frac{T_{lt}u_y}{m}, \quad \ddot{z} = \frac{\partial U^*}{\partial z} + \frac{T_{lt}u_z}{m} \quad (5)$$

where  $u_x, u_y, u_z$  are the instantaneous components of the thrust unit vector  $\hat{u}$  in the axes of the rotating frame. These equations, along with the mass flow rate expression in Eq. 4, are used to generate the path of a spacecraft with the specified propulsion system.

To describe the thrust vector direction, this paper uses the velocity-normal-conormal (VNC) axes relative to the Moon. These axes rely on the position and velocity vectors,  $\bar{R}_{M,s/c}$  and  $\bar{V}_{M,s/c}$ , of the spacecraft relative to the Moon and using an observer fixed in the inertial frame. The first unit vector,  $\hat{V}$ , is parallel to the spacecraft velocity vector measured relative to the Moon and with respect to an inertial observer. The normal unit vector  $\hat{N}$  is aligned with the orbital angular momentum of the spacecraft relative to the Moon, assuming an observer in the inertial frame. The final unit vector,  $\hat{C}$ , completes the right-handed, orthogonal triad. These axes are calculated as

$$\hat{V} = \frac{\bar{V}_{M,s/c}}{V_{M,s/c}}, \quad \hat{N} = \frac{\bar{R}_{M,s/c} \times \bar{V}_{M,s/c}}{\|\bar{R}_{M,s/c} \times \bar{V}_{M,s/c}\|}, \quad \hat{C} = \hat{V} \times \hat{N} \quad (6)$$

Using these basis vectors, the thrust unit vector is defined as

$$\hat{u} = u_V \hat{V} + u_N \hat{N} + u_C \hat{C} \quad (7)$$

where  $u_V, u_N, u_C$  are constant scalar components and  $u_V^2 + u_N^2 + u_C^2 = 1$ . Once this unit vector is expressed in the VNC axes relative to the Moon, it is transformed at each instant of time to produce the components  $u_x, u_y, u_z$  of the thrust direction in the Earth-Moon rotating frame to support evaluation of Eq. 5. Because the VNC axes defined relative to the Moon possess limited, if any, physical meaning as the spacecraft leaves the vicinity of the Moon, motion of the spacecraft is only examined between the  $L_1$  and  $L_2$  gateways.

## Curvature

Curved paths in three-dimensional space can be described in a generalizable manner using concepts from differential geometry. Consider a continuous trajectory  $\bar{r}(t) = [x(t), y(t), z(t)]^T$  generated for  $t \in [t_0, t_f]$ . The arclength of this curve is calculated as [21]

$$s = \int_{t_0}^{t_f} ds = \int_{t_0}^{t_f} \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2} dt \quad (8)$$

At any location along the trajectory, the curvature  $\kappa(t)$  captures the deviation of the local path from a straight line within the osculating plane. The osculating plane is the plane formed by three points along this curve as the limit of the arclength between them approaches zero [22]. At any point along the trajectory, the following vectors are known: the position vector  $\bar{r}(t)$ , velocity vector  $\dot{\bar{r}}(t) = [\dot{x}(t), \dot{y}(t), \dot{z}(t)]^T$ , and acceleration vector  $\ddot{\bar{r}}(t) = [\ddot{x}(t), \ddot{y}(t), \ddot{z}(t)]^T$ . Using these vectors, the curvature is mathematically calculated as

$$\kappa(t) = \frac{\|\dot{\bar{r}}(t) \times \ddot{\bar{r}}(t)\|}{\|\dot{\bar{r}}(t)\|^3} \quad (9)$$

with a singularity when the speed equals zero. This curvature is also equivalent to the angular rate of change of the tangent to the curve with respect to arclength [22]. The total absolute curvature  $\kappa_{tot}(t_0, t_f)$  is defined as the integral of this curvature along the trajectory over the time interval  $t \in [t_0, t_f]$  [22]. Mathematically, the definition of total absolute curvature is

$$\kappa_{tot}(t_0, t_f) = \int_{t_0}^{t_f} \kappa(s) ds = \int_{t_0}^{t_f} \kappa(s) \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2} dt \quad (10)$$

As a result, this quantity represents the total angle swept out by the trajectory within the osculating plane about a moving center of curvature, increasing monotonically by  $2\pi$  with every revolution. These two quantities,  $\kappa(t)$  and  $\kappa_{tot}(t_0, t_f)$ , are used in shape interrogation methods to describe and discretize curves and surfaces in a manner that captures their geometry [22]; these quantities have also been used by Bosanac to describe and discretize spacecraft trajectories [15].

## Clustering

Clustering algorithms discover groupings within a dataset such that similar members are grouped together and dissimilar members are separated [2]. This process typically begins by constructing a finite dimensional feature vector that describes the relevant characteristics of the members of the dataset for a specific application. Then, the similarity or dissimilarity between two members of the dataset is calculated by the distance between their feature vectors, defined using a specified distance measure. Clustering algorithms use these distances between feature vectors of members of the dataset to construct groups. A variety of clustering approaches exist, with algorithms typically categorized as partitioning, grid-based, model-based, density-based, and/or hierarchical methods [23]. Furthermore, hard clustering algorithms uniquely label each member of a dataset whereas soft or fuzzy algorithms assign membership probabilities to more than one cluster.

This paper uses the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm. Developed by Campello, Moulavi, and Sander [16], this algorithm combines the benefits of density-based methods with hierarchical methods by 1) modeling the similarity between members of the dataset via the density of their ‘neighborhoods’ and 2) generating groupings from a cluster hierarchy. As a result, HDBSCAN can discover groupings of data with distinct densities and shapes within a higher-dimensional feature space and without a priori knowledge of the number of clusters. Furthermore, it can designate members of the dataset in insufficiently sampled or sparse regions as noise. In this paper, the HDBSCAN algorithm is accessed using the Python library *hdbscan* created by McInnes, Healy, and Astels [24].

HDBSCAN begins by describing the dissimilarity between members of the dataset via a mutual reachability distance [16]. First, the core distance  $d_{core}(\bar{f}_i)$  of the  $i$ -th member of the dataset is defined as the distance to its  $m_{pts}$ -nearest neighbor, calculated using a specified distance measure. This quantity offers a representation of the density of the local neighborhood of each member of the dataset within the feature vector space. Next, the mutual reachability distance is calculated between the  $i$ -th and  $j$ -th members of the dataset as

$$d_{MRD} = \max(d(\bar{f}_i, \bar{f}_j), d_{core}(\bar{f}_i), d_{core}(\bar{f}_j)) \quad (11)$$

Assessing the dissimilarity of two members of the dataset via this quantity enables data in sparse regions to be further separated, producing a more robust clustering result [16, 25]. This quantity is calculated for all possible pairs of members in the dataset.

Clusters are identified using a hierarchical representation of all possible groupings [16]. First, a weighted graph is constructed with each node as a member of the dataset whereas each edge connects two nodes and is weighted by their mutual reachability distance. A minimum spanning tree is constructed to summarize this graph, with self-loops added to each node. This minimum spanning tree is used to construct a hierarchy of all possible clusters as a function of the mutual reachability distance. Clusters are then extracted from this hierarchy as the most stable groupings that possess at least a minimum size  $m_{clmin}$ . These clusters capture regions where members are sufficiently densely located within the dataset. Malzer and Baum present a modification to the cluster selection process where any members that possess a mutual reachability distance within a threshold  $\epsilon$  are joined in the same cluster [25]. This modification ensures that groups of data are not excessively subdivided to capture locally dense subsets below a desired resolution.

## TECHNICAL APPROACH

This section outlines the key definitions and technical approach used to summarize a segment of the solution space in a data-driven manner. Two general definitions are presented first: 1) the generation and description of a trajectory and 2) the process for clustering a set of trajectories. These definitions are then used in a three-step clustering-based process for sampling and summarizing a desired region of the solution space. This technical approach was recently developed by Bosanac; accordingly, this section presents a brief summary of this prior work, but more details and a demonstration for natural motion in the Earth-Moon CR3BP can be found in Reference [15].

### Generating and Describing a Spacecraft Trajectory

A key element of the data-driven framework used in this paper is generating continuous trajectories and describing them using finite-dimensional feature vectors. First, each trajectory is generated over a duration that supports the emergence of distinct geometries, the application of interest, and consistency in the clustering process. Then, a shape-based summarization scheme is used to sample the trajectory at a discrete set of states that are concentrated at the most geometrically meaningful locations and support consistency in sampling. These sampled states are then used to define feature vectors that support geometric differentiation during the clustering process.

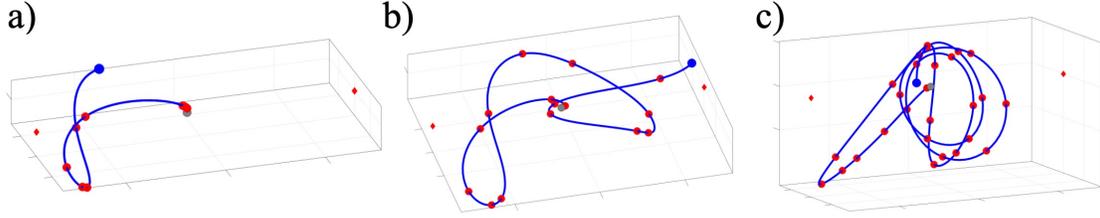
*Specifying Trajectory Duration* Each continuous trajectory is generated from a specified initial condition for a duration that supports a selected application and supplies consistency in the clustering process. First, each trajectory is generated for at least 21 days to both enable distinct geometries to emerge and support cislunar space situational awareness applications. Each trajectory is then propagated further until their total absolute curvature, measured from 0 at the initial condition, equals an integer multiple of  $\pi$ . Physically, this condition is equivalent to propagating a trajectory until it completes an integer number of half revolutions around a moving center of curvature, supporting a more consistent sampling approach and trajectory description in the next steps. Early termination occurs if the spacecraft 1) impacts a spherical approximation of the Moon or 2) departs through the  $L_1$  or  $L_2$  gateways in the Earth-Moon system.

*Sampling Trajectories* Each continuous trajectory is sampled by a discrete set of states in a manner that sufficiently captures its geometry in the rotating frame. Consider a trajectory that completes  $r = \lceil \kappa_{tot}(t_0, t_f) / \pi \rceil$  half-revolutions about a moving center of curvature in the rotating frame. This trajectory is sampled using  $N_s = N_a r + 1$  states:

- one state is sampled at the initial condition,

- $N_a(r - 1)$  states are sampled every  $\pi/N_a$  in the total absolute curvature if  $r > 1$ , and
- $N_a$  states are distributed evenly in the elapsed total absolute curvature along the final half revolution along the trajectory (or less if the trajectory terminates early)

This curvature-based sampling methodology adapts the number of sampled states to the complexity of the trajectory, biases the distribution of states towards more geometrically meaningful locations where the curvature is high, and captures global variations in geometry across the same trajectory as well as between two different trajectories. To balance sampling enough states to capture the overall shape while preventing the dimension of the subsequent trajectory summary from growing too large for longer trajectories,  $N_a = 3$  is used in this paper. Examples for three types of trajectories that are generated with a thrust-vector in the velocity direction and sampled with  $N_a = 3$  via the curvature-based method appear in Figure 1. In each subfigure, the blue continuous trajectory is sampled at the states indicated by circles; most samples are colored red whereas the initial condition is highlighted in blue to supply direction of motion information. In addition, the Moon is located by a gray circle (not to scale) whereas  $L_1$  and  $L_2$  are depicted with red diamonds. In each example, the samples capture the global shape of each path.



**Figure 1. Examples of three trajectories generated with a thrust vector in the velocity direction and sampled at the locations highlighted by circles with  $N_a = 3$  via the curvature-based sampling method.**

*Describing Trajectories* Finite-dimensional feature vectors are defined to summarize the geometry of trajectories and support accurate clustering results. One feature vector describes each trajectory via the velocity unit vector at each sampled state and is used for initial, robust clustering and aggregation to discover groups of trajectories with a similar shape. A second feature vector is defined using the position vector at each sampled state. This position-based feature vector is used in the final refinement of clusters to separate trajectories with a similar shape that encompass distinct regions of the configuration space as well as to remove outliers. Separating these feature vectors and using a two-step clustering process avoids the issue of weighting two distinct types of quantities together in a single feature vector. Of course, other approaches to trajectory description may also produce acceptable clustering results and should be selected based on both the application of interest and the quality of the results.

The shape-based feature vector is composed of the velocity unit vector at each of the  $N_s$  sampled states along the trajectory. For the  $i$ -th trajectory, the shape-based feature vector,  $\bar{f}_s$ , is defined as

$$\bar{f}_{s,i} = [\hat{v}_{1,i}, \hat{v}_{2,i}, \dots, \hat{v}_{N_s-1,i}, \hat{v}_{N_s,i}] \quad (12)$$

where  $\hat{v}_{j,i}$  is the velocity unit vector for the  $j$ -th sampled state along the  $i$ -th trajectory in the rotating frame with an observer also fixed in the rotating frame. This feature vector possesses a dimension of  $3(N_a \lceil \kappa_{\text{tot}}(t_0, t_f) / \pi \rceil + 1)$ , increasing as  $N_a$  or the minimum trajectory duration increases.

The position-based feature vector is designed to capture the location of the trajectory in the configuration space. For the  $i$ -th trajectory, the position feature vector,  $\bar{f}_p$ , is defined as

$$\bar{f}_{p,i} = [\bar{r}_{1,i}, \bar{r}_{2,i}, \dots, \bar{r}_{N_s-1,i}, \bar{r}_{N_s,i}] \quad (13)$$

with each position vector  $\bar{r}_{j,i}$  in the rotating frame. This feature vector possesses a dimension of  $3(N_a \lceil \kappa_{tot}(t_0, t_f) / \pi \rceil + 1)$ , also increasing as  $N_a$  or the minimum trajectory duration increases.

### Clustering a Set of Spacecraft Trajectories

To group a set of trajectories by their geometry, their feature vectors are clustered using HDBSCAN. This step requires specifying the values of the governing parameters, which can influence the clustering results. Following the values selected by Bosanac, these quantities are selected manually to prioritize capturing local variations in trajectory geometry [15]. Accordingly,  $m_{clmin}$  is selected to support the recovery of many small, but geometrically different, groups of trajectories that are generated by a sparse sampling approach, as presented in the next section. Then,  $m_{pts} = m_{clmin} - 1$  is selected to define each neighborhood to be the same size as the smallest possible cluster. The Euclidean distance is also employed to assess the dissimilarity between feature vectors due to its computational speed. Finally, when clustering by the shape-based feature vector,  $\epsilon_{merge} = 2\sqrt{N_s} \sin(\alpha/2)$ , equal to the sum of the Euclidean distance between two sequences of  $N_s$  unit vectors that are separated by an average angle of  $\alpha$ . However, when refining shape-based clusters using the position feature vector,  $\epsilon_{merge} = 0$  due to the challenges of defining a robust threshold for sufficient similarity between trajectories in the configuration space of a chaotic system.

Each clustering step generates groups of geometrically similar trajectories. To facilitate visualization and analysis, the medoid of each cluster is calculated as the member most similar to all other members of a cluster [26]. The medoid trajectory  $\mathcal{T}_{med,k}$  of cluster  $C_k$  is defined mathematically using the feature vectors  $\bar{f}$  of its  $P_k$  members as

$$\mathcal{T}_{med,k} = \operatorname{argmin}_{\mathcal{T}_i \in C_k} \left[ \sum_{j=1, i \neq j}^{P_k} d(\bar{f}_i, \bar{f}_j) \right] \quad (14)$$

where  $\mathcal{T}_i$  is the  $i$ -th trajectory. This medoid supplies a representative member of each cluster that is more straightforward to view than a large set of trajectories.

Along with a set of clusters, some trajectories may be unlabeled and, therefore, are designated as noise points. Consistent with the theory of HDBSCAN, some of the noise points lie at the boundaries of clusters and, as result, may exist in much sparser regions of the dataset. Accordingly, a post-processing step is employed, as suggested by Campello et al. [16]: a noise point is automatically assigned to the nearest cluster if it lies within the  $m_{pts}$ -neighborhood of any member of that cluster. The remaining noise points lie in regions of the high-dimensional feature vector space that are insufficiently populated by the dataset; in this application, those trajectories are not geometrically similar to enough members of the dataset to be assigned to a cluster.

During early steps of the clustering process, a set of trajectories may be described by a distinct number of sampled states and, therefore, feature vectors with a distinct dimension. In this case, the trajectories are divided into subsets with the same feature vector dimension. Each subset is then clustered independently. Given that trajectories with a distinct number of half revolutions are expected to be geometrically distinct and, therefore, separated by a clustering algorithm, this approach produces a smaller dataset at each clustering step.

## Data-Driven Trajectory Summarization Process

This subsection presents the approach used to generate and summarize the geometries of a set of trajectories in the Earth-Moon CR3BP. First, initial conditions are sampled from within a desired region of the configuration space in a geometry-based manner that produces a sparse but representative set of trajectories. Then, the trajectories are grouped using a distributed-clustering approach: 1) trajectories are partitioned into smaller subsets and each partition is independently clustered, 2) clusters of similar trajectories that exist across distinct partitions are aggregated to produce global clusters, and 3) the global clusters are refined. This approach was previously developed by Bosanac and a brief summary is presented here; for more details, refer to Reference [15].

*Step 1: Sample the Solution Space* Position vectors for initial conditions used in trajectory generation are defined using a uniform grid in the vicinity of the Moon between  $L_1$  and  $L_2$ . This grid spans  $x \in [0.835, 1.155]$ ,  $y \in [-0.12, 0.12]$ , and  $z \in [-0.11, 0.11]$ , with each initial condition sampled at an interval of  $\Delta_{pos} = 0.005$ . At each position vector sampled from this grid, the speed  $v$  of any candidate initial conditions is calculated to produce a desired value of the Jacobi constant,  $C_{J,des} = 3.165$ , using the relation  $v = \sqrt{C_{J,des} - 2U^*(x, y, z)}$ . Only the position vectors that possess a real-valued speed are retained as they lie within the zero velocity surfaces at this energy level.

Velocity vectors are then sampled to sparsely capture the dominant geometries of trajectories with initial conditions at each position vector. To achieve this goal, velocity vectors that produce maxima in the curvature are calculated and summarized at each position vector. From a differential geometry perspective, these maxima capture the most geometrically meaningful locations along a trajectory. From the perspective of analyzing trajectories in the CR3BP, maxima in the curvature, calculated in the rotating frame, tend to lie close to apse locations with respect to primaries or equilibrium points. However, unlike apsides, curvature maxima do not require specifying a reference point. Accordingly, sampling the phase space using curvature maxima supplies a geometry-based and robust approach to generating useful initial conditions.

The first step of the geometry-based approach to calculating initial conditions at each position vector involves approximating the one-parameter families of velocity vectors that produce maxima in the curvature. A velocity vector is parameterized using the angles  $\theta_{xy}$  and  $\theta_z$  as

$$\bar{v} = v \cos(\theta_{xy}) \cos(\theta_z) \hat{x} + v \sin(\theta_{xy}) \cos(\theta_z) \hat{y} + v \sin(\theta_z) \hat{z} \quad (15)$$

where  $v$  is calculated to produce the desired Jacobi constant at a specified position vector. At a single value of  $\theta_z$ ,  $\theta_{xy} \in [0, 360^\circ]$  is varied in steps of  $\Delta_{xy} = 1^\circ$  to generate velocity vectors via Eq. 15. These velocity vectors and the position vector are used to calculate  $\dot{\kappa}$  at each state vector. If a pair of adjacent velocity vectors at a single value of  $\theta_z$  produces values of  $\dot{\kappa}$  with opposite signs, an initial guess for a velocity vector that produces a turning point in the curvature is identified. Each initial guess is then used to calculate the values of  $\dot{x}$  and  $\dot{y}$  that satisfy the following conditions:

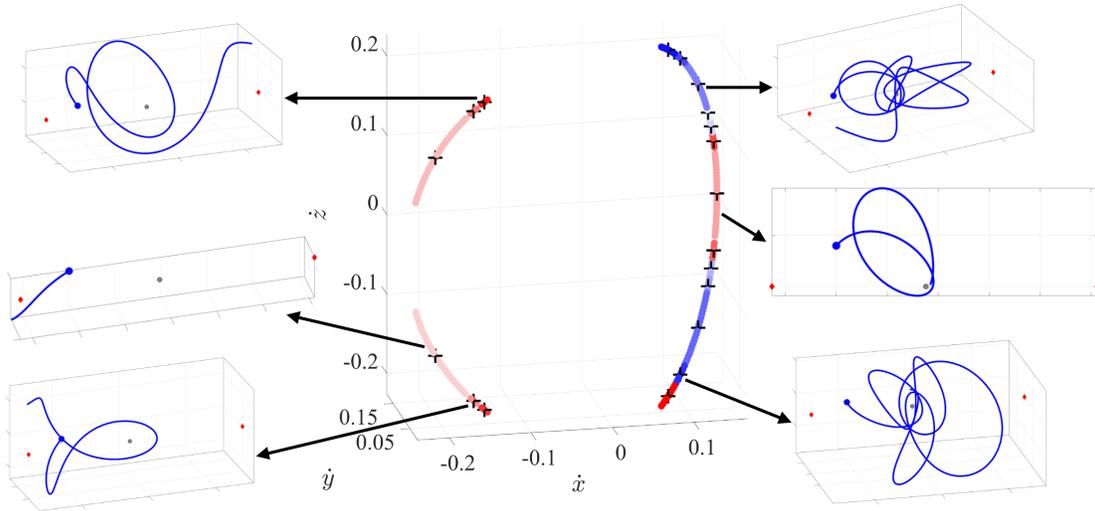
$$\frac{d\kappa(\bar{x})}{dt} = 0, \quad \text{and} \quad \frac{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}{v^2} - 1 = 0 \quad (16)$$

given fixed values of  $x$ ,  $y$ ,  $z$ , and  $\dot{z}$ . The *fsolve* function in MATLAB is employed to solve this root-finding problem for each initial guess for a given  $\theta_z$ . A velocity vector that satisfies these two conditions supplies a turning point in the curvature at the specified Jacobi constant. The second time derivative of the curvature is then calculated and only solutions that produce a maximum are

retained. This process is repeated for various out-of-plane angles  $\theta_z \in [-90^\circ, 90^\circ]$  in steps of  $\Delta_{xy} = 1^\circ$  to produce one or more one-parameter families of velocity vectors that supply maxima in the curvature.

The ranges of velocity vectors that produce curvature maxima are summarized by one or more representative members using a localized application of the clustering process. First, trajectories are generated for each of the initial states identified to produce curvature maxima in 1 degree increments of  $\theta_z$  using the approach outlined at the beginning of this section. These trajectories are locally clustered using the shape-based feature vectors via HDBSCAN with  $m_{clmin} = 3, m_{pts} = 2,$  and  $\epsilon_{merge} = 2 \sin(1^\circ) \sqrt{N_s}$ . The medoids of these clusters then identify a set of representative initial conditions that capture the dominant geometries of trajectories beginning at maxima in the curvature from each position vector; these representative initial conditions are stored to form part of the dataset that is summarized in later steps. This process is repeated for all position vectors in the grid, producing on the order of a million trajectories per thrust direction.

An example of the procedure for sampling initial state vectors in a geometry-based manner is constructed for an initial position vector of  $x = 0.9, y = 0.04, z = 0$  and a desired initial Jacobi constant of 3.1650. The center of Figure 2 displays a set of three curves traced out by velocity vectors that produce a maximum in the curvature for a thrust vector in the velocity direction. Each of the 19 local clusters of velocity vectors that produces a geometrically distinct set of trajectories is colored in shades of red and blue with the medoid velocity vector indicated by a black cross. Representative trajectories for selected clusters appear in the insets. As depicted in Figure 2, each group corresponds to a geometrically distinct trajectory that begins from a maximum in the curvature at a specified position vector and Jacobi constant. Thus, 19 trajectories associated with this one position vector, along with their feature vectors, are stored in the larger dataset for subsequent clustering.



**Figure 2. Examples of local clusters of velocity vectors that produce a maximum in the curvature at an initial position vector of  $x = 0.9, y = 0.04, z = 0$  with an initial Jacobi constant of 3.1650 for a spacecraft thrusting in the velocity direction.**

*Step 2: Cluster Individual Partitions of Trajectories* The large data set of feature vectors and initial conditions is divided into smaller partitions that are independently clustered for ease of processing. Specifically, trajectories with feature vectors of the same dimension and with adjacent

initial position vectors are allocated to evenly-sized partitions of no more than 10,000 members, tending to produce a reasonably-sized subset of the data for fast clustering via HDBSCAN on the order of seconds. The trajectories within each partition are then clustered via HDBSCAN using the shape-based feature vector with the governing parameters selected as  $m_{clmin} = 5$ ,  $m_{pts} = 4$ , and  $\epsilon_{merge} = 2 \sin(5^\circ) \sqrt{N_s}$ . The result is a set of local clusters  $\mathcal{L}_i$  that summarize trajectories by their shape for the  $i$ -th partition. At this step, any noise points that are not assigned as border points of a cluster are no longer moved forward through the clustering process for further analysis.

*Step 3: Aggregate Clusters to Form Global Summary* Trajectories with a particular shape may exist across multiple partitions. Accordingly, local clusters are aggregated across partitions to produce a single global cluster summary of the entire dataset. This aggregation process is performed in two steps using a graph structure: 1) identification of groups of local clusters that could be candidates for merging across partitions, and 2) refinement to identify one or more global clusters from each local cluster group.

The first step of the cluster aggregation process involves identifying groups of local clusters of trajectories with a similar shape. To facilitate this process, an undirected graph is constructed. Each local cluster forms the node of a graph; if the dataset is segmented into  $P$  partitions, each containing  $l_i$  local clusters, then this graph is made up of  $\sum_{i=1}^P l_i$  nodes. Edges indicate candidates for clusters of similar shape that exist across multiple partitions. To identify these edges, the representative trajectory of each local cluster is compared to the representatives of all other local clusters via their shape-based feature vectors. The 4 cluster representatives with the smallest differences in the shape-based feature vectors are then used to coarsely approximate the 4 closest local clusters. Together with the original local cluster, all 5 local clusters are grouped via HDBSCAN using the same governing parameters as Step 2. If any new groups consist of members that originated from distinct local clusters, an edge is added between the nodes of only those local clusters in the graph. Once this process has been completed for all local clusters, the connected components of this graph form groups of local clusters that are analyzed further.

Each group of local clusters is independently clustered to identify one or more global clusters. In this step, trajectories from each local cluster in a single connected component of the graph are used to form a new dataset. However, if this dataset possesses more than 10,000 members, the local clusters with a large number of members are subsampled. This new set of trajectories is first clustered by their shape-based feature vectors via HDBSCAN using the same governing parameters as Step 2. The result is a set of groups of trajectories with a similar shape. However, some groups of trajectories with a similar shape may contain subgroups that exist in distinct parts of the configuration space or exhibit more localized groupings in the path. In some cases, there may even be outliers in the group. Accordingly, each shape-based group is then further refined by clustering only those trajectories via their position-based feature vectors in HDBSCAN with  $m_{clmin} = 5$ ,  $m_{pts} = 4$ , and  $\epsilon_{merge} = 0$ . Each group produced by this step is a global cluster whereas any trajectories designated as noise points are discarded. Repeating this process for each local cluster group results in a single global cluster summary composed of  $g_i$  clusters of geometrically similar trajectories.

## RESULTS

Global cluster summaries are generated for a thrust-enabled SmallSat that begins near the Moon with a Jacobi constant of  $C_J = 3.165$  and maintains a constant thrust direction in the VNC axes defined relative to the Moon. Several thrust vector directions are used as specified in Table 1; this table also includes a legend of the colors used to represent trajectories in subsequent figures. The global

cluster summaries of trajectories that begin between the  $L_1$  and  $L_2$  gateways are generated independently for each thrust direction following the technical approach outlined in the previous section. The number of clusters of geometrically distinct trajectories that are automatically discovered using this approach is listed in Table 1 for each thrust direction.

**Table 1. Thrust vector directions used to define several datasets**

$u_v$	$u_n$	$u_c$	Number of global clusters	Color
1	0	0	1079	Red
$1/\sqrt{2}$	0	$1/\sqrt{2}$	1172	Dark red
0	0	1	1209	Dark purple
$-1/\sqrt{2}$	0	$1/\sqrt{2}$	856	Dark blue
-1	0	0	533	Blue
$-1/\sqrt{2}$	0	$-1/\sqrt{2}$	741	Light blue
0	0	-1	1306	Light purple
$1/\sqrt{2}$	0	$-1/\sqrt{2}$	1494	Light red
0	1	0	1404	Black
0	-1	0	1569	Gray

The global clusters of trajectories are further aggregated to identify the dependence of trajectory geometry on the thrust vector directions. This aggregation step follows the general approach outlined in Step 3 of the Technical Approach, with each input being one of several global cluster summaries that are constructed for a single thrust vector direction and the output being coarse groups of clusters that span one or more thrust vectors. Selected groups of aggregated clusters are displayed in Figures 3-7 and organized by their itinerary; more details and discussion of each figure appears in the following paragraphs. In each figure, only representative members of each global cluster from each thrust direction are displayed using the colors listed in Table 1: shades of red (or blue) indicate a thrust vector with a positive (or negative) component in the velocity direction; darker (or lighter) colors indicate a thrust vector with a positive (or negative) component in the conormal direction; whereas black and gray correspond to a thrust vector out of the orbit plane. The initial condition along each trajectory is indicated using a large circle in the same color as the trajectory. In these figures, the Moon is depicted with a gray circle, not to scale, whereas  $L_1$  and  $L_2$  are displayed as red diamonds. Although these figures contain a substantial amount of information, trends in the colors of representatives with a similar geometry reveal the dependence of trajectory geometry on thrust vector direction.

### Lunar Impact Trajectories

As expected, the geometries of impact trajectories do not generally depend on the thrust vector direction. Figure 3 depicts examples of trajectories with distinct geometries that impact the Moon over distinct time scales. Towards the top, trajectories in each group tend to impact the Moon quickly, within less than 2 revolutions. With shorter durations, many of these trajectories have less time to be affected by a continuous-thrust propulsion system. However, the trajectories in the rightmost column of the second row appear to exist for thrust vectors with positive conormal components, traveling away from the Moon before quickly returning to impact. Trajectories in the rightmost column of the third row exist for thrust vectors out of the  $\hat{V}\hat{C}$  plane as well as those with a negative conormal component, completing a high eccentricity and high inclination revolution around the

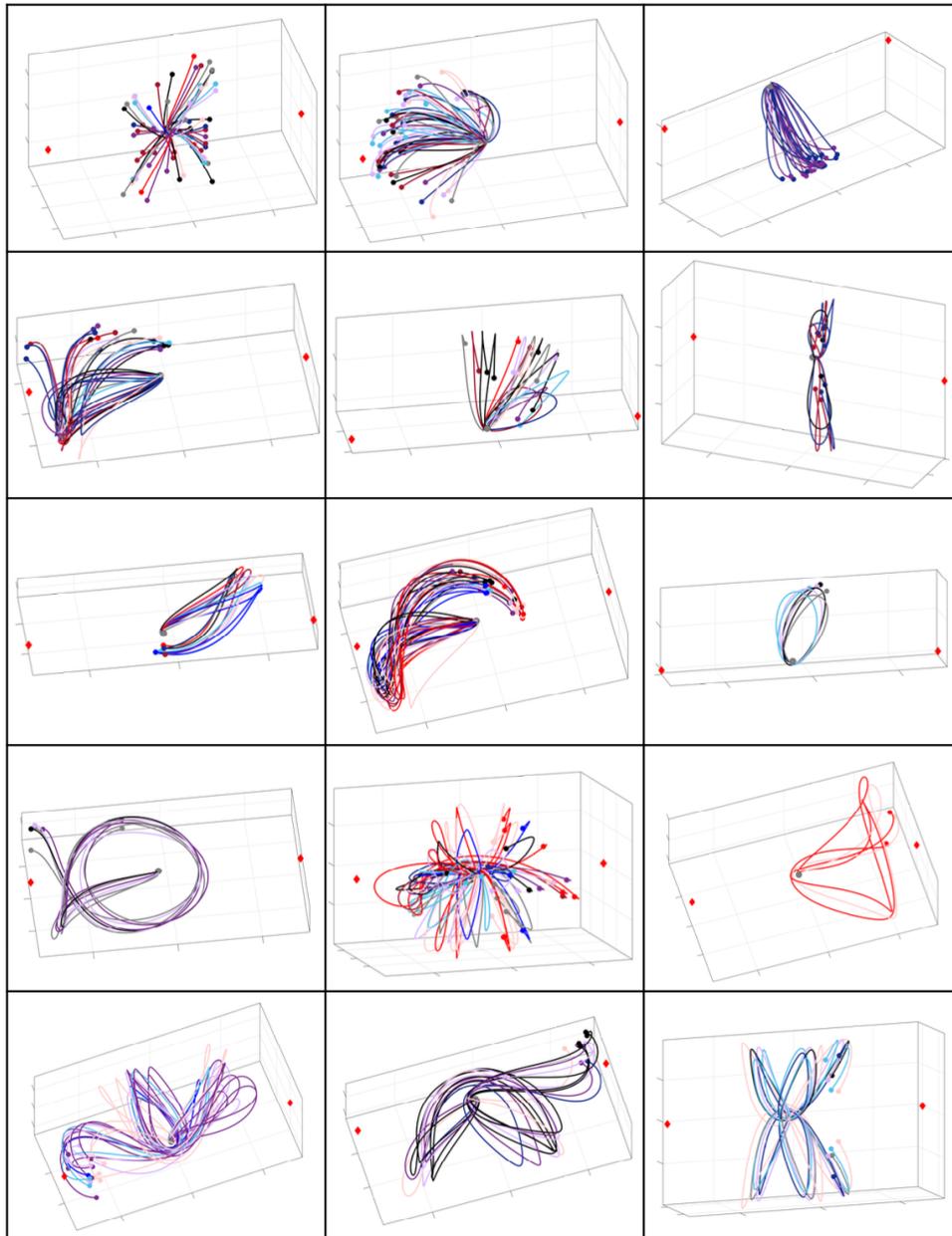
Moon before impact; similar trajectories exist for natural motion. Towards the bottom of Figure 3, trajectories possess a longer duration. Accordingly, those subfigures contain representatives with a limited array of colors that indicate the existence of those trajectory geometries for specific ranges of thrust vectors. For instance, trajectories associated with thrust vectors that possess substantial conormal components tend to be grouped with each other and/or with trajectories generated using thrust vectors directed out of the  $\hat{V}\hat{C}$  plane. In this case, the thrust vectors do not substantially change the spacecraft energy but, rather, the orientation of each revolution around the Moon prior to impact; similar trajectories also exist for natural motion. In the rightmost column of the fourth row, there are also trajectories that impact the Moon with a geometry that is not consistent with natural motion. The red shades of the representatives indicate that these solutions require a substantial thrust vector component in the velocity direction.

### **Departures Through the $L_1$ or $L_2$ Gateways**

The array of geometrically distinct trajectories that reach the  $L_1$  or  $L_2$  gateways captures paths that resemble natural motions as well as those that substantially differ. Figure 4 displays a sample of the groups of trajectories that reach the  $L_1$  gateway whereas Figure 5 includes trajectories leaving through the  $L_2$  gateway within around 21 days. In both figures, there are groups of trajectories composed of representatives that are colored purple, gray, and black. These trajectories, generated by thrusting in the positive or negative conormal or normal directions, tend to resemble natural motions due to the gradual impact of these paths on the orientation of the trajectory. A subset of these trajectories with shorter durations tend to be grouped with paths generated with components in the velocity or anti-velocity directions. However, in those cases, the geometry of trajectories colored in red tends to evolve away from the rest of the group. These figures also include a wide variety of trajectory geometries generated by thrusting predominantly in the velocity direction. As expected, the spacecraft energy is increasing most efficiently, leading to departures through the  $L_1$  and  $L_2$  gateways after revolutions where the spacecraft gradually moves further from the Moon and, in many cases, with substantial out of plane components. Finally, viewing these samples of geometrically distinct trajectories as a whole reveals a diverse array of options for a spacecraft with low-thrust propulsion to leave the lunar vicinity.

### **Trajectories Remaining Within the Lunar Vicinity**

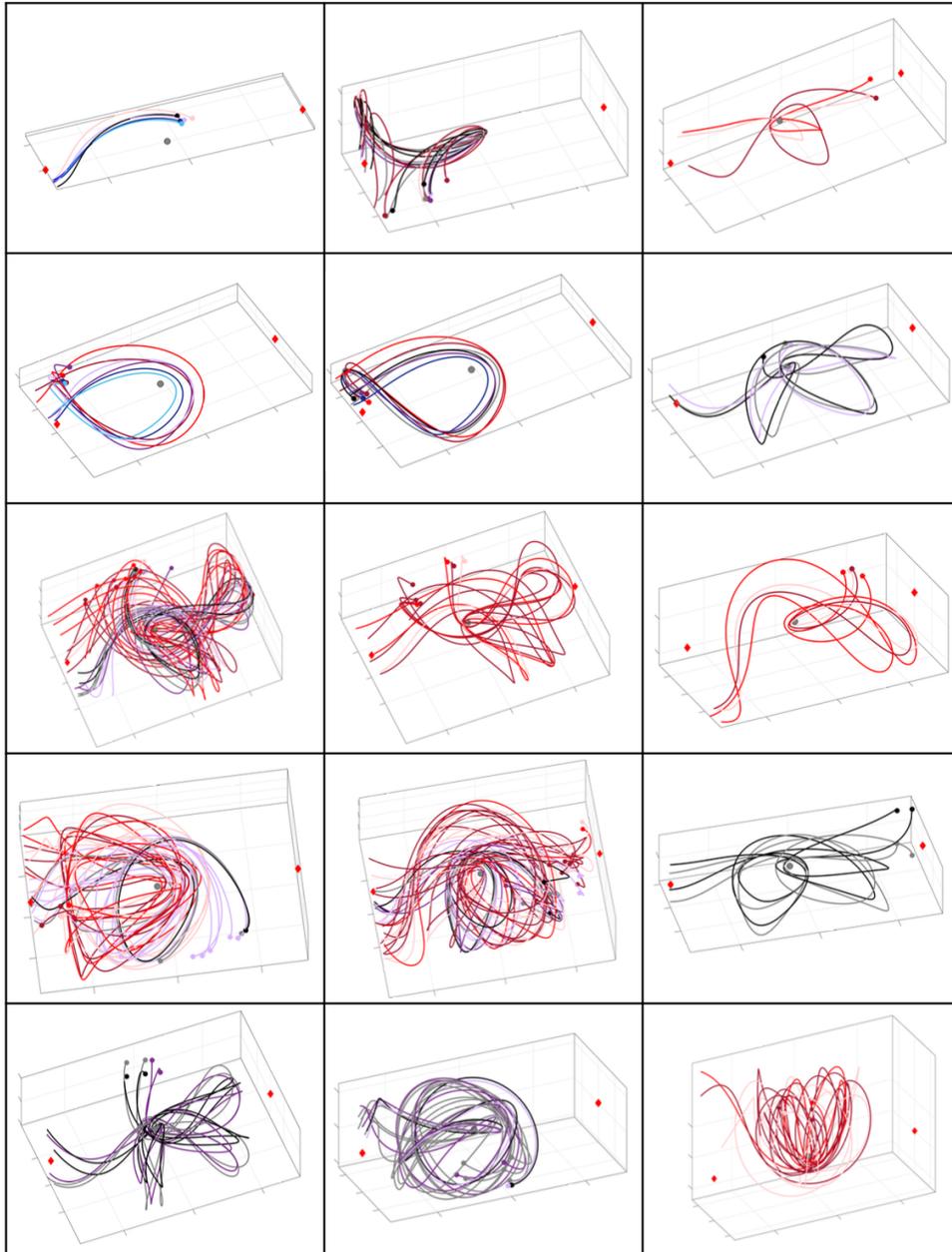
Trajectories that remain within the vicinity of the Moon over the span of a few weeks tend to exist in groups that correspond to a more localized set of thrust vectors. Figure 6 displays a sample of trajectories with a relatively low extension out of the plane of the primaries. At the top of this figure, trajectories in shades of purple and black that are generated with thrust vectors in the positive and negative conormal or normal directions tend to be grouped together and resemble natural motions. For instance, the trajectory in the leftmost column and second row resembles a natural prograde quasi-periodic orbit from the CR3BP. The exact thrust direction does, of course, result in slight variations in the orientation of each revolution around the Moon. There are fewer clusters in shades of red, consistent with thrust components in the velocity direction. Some examples displayed in Figure 6 deviate substantially from natural motions and do not always exhibit an obvious change in perilune or apolune distances as expected from a lower-fidelity model. Towards the bottom of this figure, there are groups of trajectories plotted in shades of blue that correspond to thrust vectors in the anti-velocity direction, exhibiting gradual decreases in the apolune distance and orientation when viewed in the rotating frame, i.e., similar to the well-known spiral-in trajectories. In the leftmost column and first row, consistently thrusting in the positive or negative conormal directions



**Figure 3.** Examples of lunar impact trajectories generated in the thrust-enabled CR3BP from initial conditions at  $C_J = 3.165$  and displayed in the Earth-Moon rotating frame. Within each subfigure, one or two groups of geometrically similar representative trajectories are displayed using the colors listed in Table 1 to indicate the thrust direction.

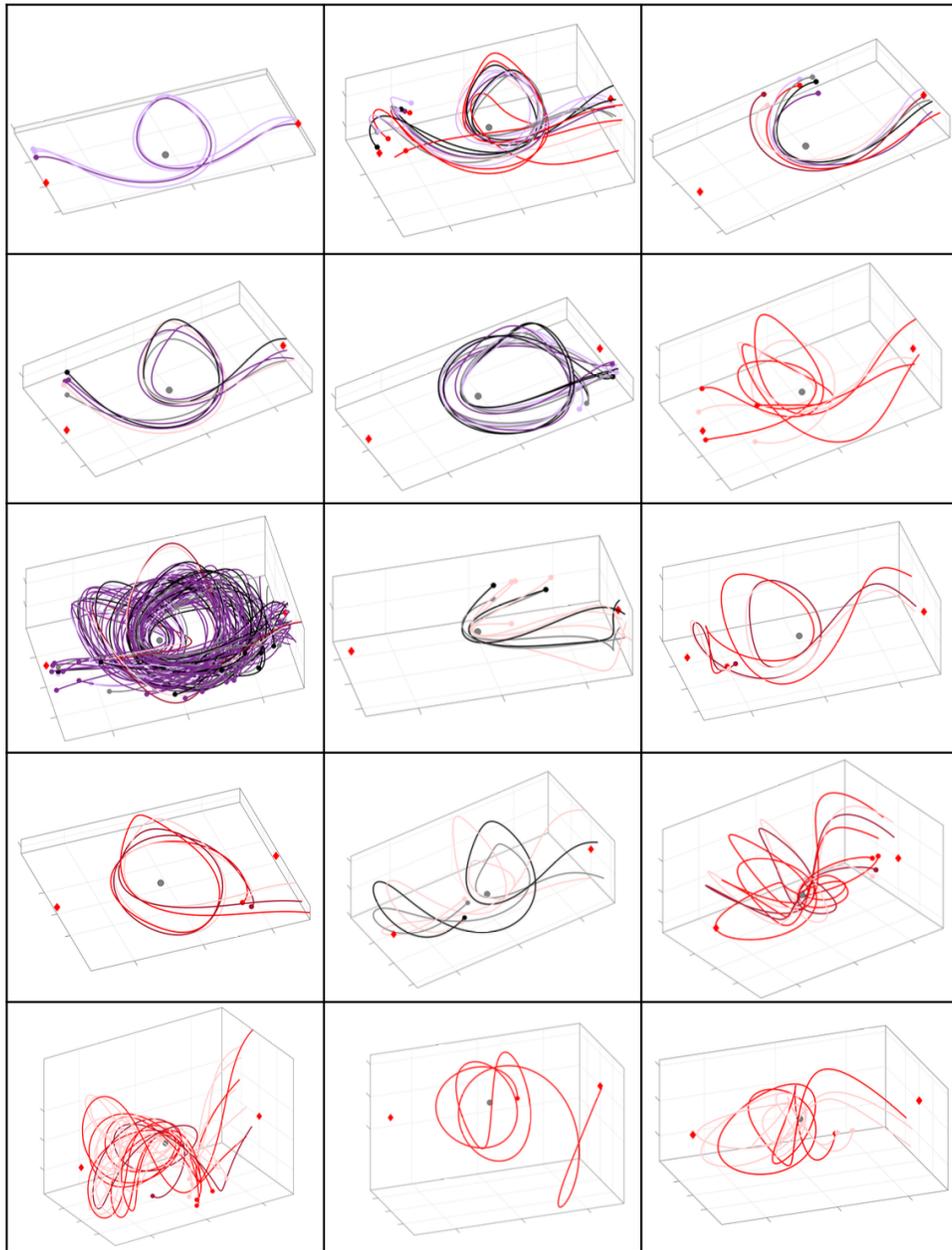
produces bounded motions that resemble natural quasi-periodic orbits near a distant retrograde orbit.

Some clusters of trajectories that remain within the lunar vicinity significantly extend out of the Earth-Moon plane. Trajectories in the top two rows of Figure 7 that are shaded in red are associated with a thrust vector in the velocity direction. These orbits are observed to grow in size with each



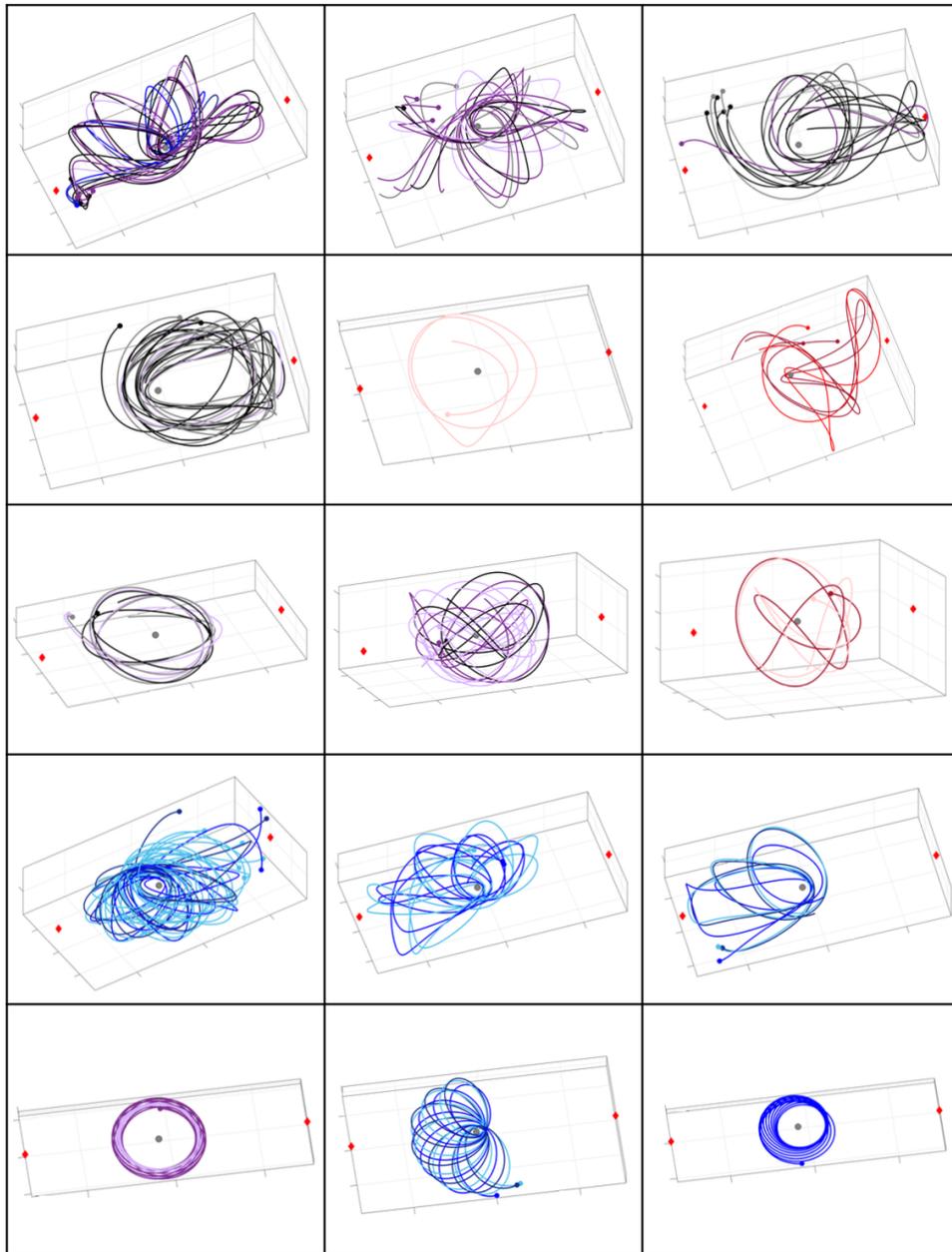
**Figure 4.** Examples of trajectories that depart through the  $L_1$  gateway, generated in the thrust-enabled CR3BP from initial conditions at  $C_J = 3.165$  and displayed in the Earth-Moon rotating frame. Within each subfigure, geometrically similar representative trajectories are displayed using the colors listed in Table 1 to indicate the thrust direction.

revolution as the energy increases. The blue trajectories in the middle three rows associated with thrust in the anti-velocity direction demonstrate the opposite behavior and gradually shrink as the spacecraft energy decreases. Trajectories in black with thrust in the normal direction such as the middle entry in the fourth row also feature similar bounded motion but with more consistent apolune and perilune radii as the orientation of the orbit evolves over time. Thrust directed along the normal



**Figure 5. Examples of trajectories that depart through the  $L_2$  gateway, generated in the thrust-enabled CR3BP from initial conditions at  $C_J = 3.165$  and displayed in the Earth-Moon rotating frame. Within each subfigure, geometrically similar representative trajectories are displayed using the colors listed in Table 1 to indicate the thrust direction.**

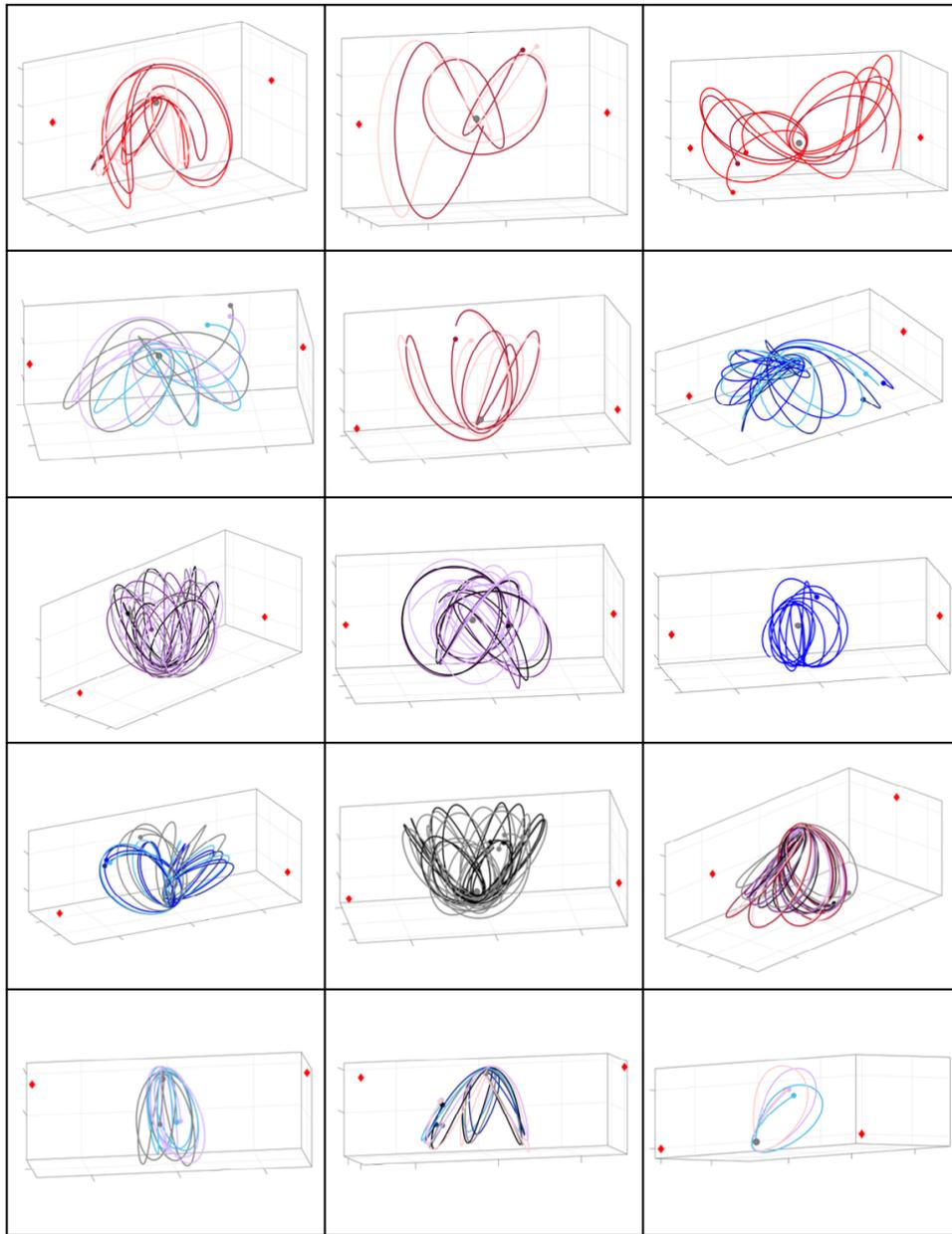
and anti-normal or conormal and anti-conormal directions, colored in shades of black and purple, result in trajectories that are likely to be grouped together as seen in those with lower out-of-plane extensions. These trajectories also resemble natural motions. As evident in the bottom row of the figure, some trajectories revolve around the Moon with a high inclination and high eccentricity.



**Figure 6.** Examples of trajectories that revolve around the Moon with low out-of-plane extension, generated in the thrust-enabled CR3BP from initial conditions at  $C_J = 3.165$  and displayed in the Earth-Moon rotating frame. Within each subfigure, geometrically similar representative trajectories are displayed using the colors listed in Table 1 to indicate the thrust direction.

## CONCLUSIONS

A data-driven framework, developed by Bosanac in prior work [15], was used to automatically extract a summary of the geometries of a set of low-thrust trajectories. These trajectories are generated for a SmallSat with a thrust vector that is fixed in the velocity-normal-conormal axes relative



**Figure 7. Examples of trajectories that revolve around the Moon with high out-of-plane extension, generated in the thrust-enabled CR3BP from initial conditions at  $C_J = 3.165$  and displayed in the Earth-Moon rotating frame. Within each subfigure, geometrically similar representative trajectories are displayed using the colors listed in Table 1 to indicate the thrust direction.**

to the Moon. Each trajectory spans at least 21 days, except if impact with the Moon occurs or the spacecraft passes through the  $L_1$  or  $L_2$  gateways. Each trajectory is sampled to capture its global shape characteristics and then described by finite-dimensional feature vectors. To generate these trajectories, initial conditions are sampled to sparsely capture the dominant geometries of trajectories. Then, the feature vectors of these trajectories are grouped using a distributed clustering approach.

When these clusters are generated for various thrust vector directions and then aggregated, groups of geometrically similar trajectories reveal the influence of the thrust vector direction on the solution space. Selected trajectory geometries are analyzed, revealing both consistency with theoretical expectations and the emergence of geometrically interesting paths within the Moon vicinity. Future work will focus on further leveraging this data-driven framework to automatically summarize the solution space across multiple spacecraft parameters, energy levels, and thrust vector histories.

## ACKNOWLEDGMENT

This work was performed at the University of Colorado Boulder and supported by funding from the Air Force Research Laboratory through a cooperative agreement titled “Multi-Domain Awareness, Decision, and Exploitation”.

## REFERENCES

- [1] V. Szebehely, *Theory of Orbits: The Restricted Problem of Three Bodies*. London: Academic Press, 1967.
- [2] J. Han and M. Kamber, *Data Mining: Concepts and Techniques, 2nd ed.* New York, NY: Proquest Ebook Central: Elsevier Science and Technology, 2006, ch. 7.
- [3] Y. Zheng and X. Zhou, *Computing with Spatial Trajectories*. New York, NY: Springer Science and Business Media, 2011, ch. 1, 5.
- [4] C. Gallego, V. Gómez Comendador, F. Saez Nieto, and M. Martinez, “Discussion of Density-Based Clustering Methods Applied for Automated Identification of Airspace Flows,” in *37th Digital Avionics Systems Conference*, Piscataway, NJ: IEEE Publ., 2018.
- [5] E. Cakmak, M. Plank, D. Calovi, A. Jordan, and D. Keim, “Spatio-Temporal Clustering Benchmark for Collective Animal Behavior,” in *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Animal Movement Ecology and Human Mobility*, 2021.
- [6] A. Palma, V. Bogorny, B. Kuijpers, and L. Alvares, “A Clustering-Based Approach for Discovering Interesting Places in Trajectories,” in *Proceedings of the 2008 ACM Symposium on Applied Computing*, 2008, pp. 863–868.
- [7] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma, “Recommending Friends and Locations Based on Individual Location History,” *ACM Transactions on the Web*, vol. 5, no. 1, 2011.
- [8] A. Hadjighasem, D. Karrasch, H. Teramoto, and G. Haller, “Spectral-Clustering Approach to Lagrangian Vortex Detection,” *Physical Review E*, vol. 93, no. 6, 2016.
- [9] N. Nakhjiri and B. F. Villac, “Automated Stable Region Generation, Detection, and Representation for Applications to Mission Design,” *Celestial Mechanics and Dynamical Astronomy*, vol. 123, no. 1, pp. 63–83, 2015.
- [10] B. Villac, R. Anderson, and A. Pini, “Computer Aided Ballistic Orbit Classification Around Small Bodies,” *Journal of Astronautical Sciences*, vol. 63, no. 3, pp. 175–205, 2016.
- [11] T. Smith and N. Bosanac, “Constructing Motion Primitive Sets to Summarize Periodic Orbit Families and Hyperbolic Invariant Manifolds in a Multi-Body Systems,” *Celestial Mechanics and Dynamical Astronomy*, vol. 134, no. 7, 2022.
- [12] N. Bosanac, “Data-Mining Approach to Poincaré Maps in Multi-Body Trajectory Design,” *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 6, 2020.

- [13] S. Bonasera and N. Bosanac, “Applying Data Mining Techniques to Higher-Dimensional Poincaré Maps in the Circular Restricted Three-Body Problem,” *Celestial Mechanics and Dynamical Astronomy*, vol. 133, no. 51, 2021.
- [14] S. Bonasera and N. Bosanac, “Unsupervised Learning to Aid Visualization of Higher-Dimensional Poincaré Maps in Multi-Body Trajectory Design,” in *AAS/AIAA Astrodynamics Specialist Conference*, Virtual, Aug. 2020.
- [15] N. Bosanac, “Data-Driven Summary of Natural Spacecraft Trajectories in the Earth-Moon System,” 2024, In preparation.
- [16] R. Campello, D. Moulavi, and J. Sander, “Density-Based Clustering Based on Hierarchical Density Estimates,” in *Advances in Knowledge Discovery and Data Mining*, J. Pei, V. Tseng, L. Cao, H. Motoda, and G. Xu, Eds., Heidelberg: Springer, Berlin, 2013.
- [17] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross, *Dynamical Systems, the Three Body Problem and Space Mission Design*. New York: Marsden Books, 2011.
- [18] B. Conway, *Spacecraft Trajectory Optimization*. New York: Cambridge University Press, 2010.
- [19] I. Elliott, C. Sullivan, N. Bosanac, J. Stuart, and F. Alibay, “Designing Low-Thrust Trajectories for a SmallSat Mission to Sun-Earth L5,” *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 10, 2020.
- [20] Busek, *BHT-200 Busek Hall Effect Thruster*, Last accessed: June 28, 2024. [Online]. Available: <https://www.busek.com/bht200>.
- [21] K. Wardle, *Differential Geometry*. Mineola, NY: Dover Publications, Inc., 2008.
- [22] N. Patrikalakis, T. Maekawa, and W. Cho, *Shape Interrogation for Computer Aided Design and Manufacturing*. E-book, 2009.
- [23] C. C. Aggarwal and C. K. Reddy, *Data Clustering: Algorithms and Applications*. Boca Raton, FL: Chapman and Hall CRC Press, 2018, ch. 2.1.2, 3, 9.2.
- [24] L. McInnes, J. Healy, and S. Astels, “hdbscan: Hierarchical Density Based Clustering,” *Journal of Open Source Software*, vol. 2, no. 11, 2017.
- [25] C. Malzer and M. Baum, “A Hybrid Approach To Hierarchical Density-Based Cluster Selection,” in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2020, pp. 223–228.
- [26] P. Cichosz, *Data Mining Algorithms: Explained Using R*. West Sussex, UK: John Wiley and Sons, 2015.