AAS 21-580

USING MOTION PRIMITIVES TO DESIGN LIBRATION POINT ORBIT TRANSFERS IN THE EARTH-MOON SYSTEM

Thomas R. Smith* and Natasha Bosanac†

Motion primitives offer a summary of the diverse and complex solution space within a multi-body system. In this paper, sets of motion primitives, as well as their corresponding regions of existence, are leveraged to facilitate rapid identification of suitable trajectory segments for initial guess construction. To develop a continuous solution from an initial set of primitives, a constrained optimization approach that leverages a direct collocation scheme is formulated to compute libration point orbit transfers in the Earth-Moon circular restricted three-body problem. This primitive-based initial guess construction strategy limits the analytical burden on a human designer and supports rapid trajectory design.

1 INTRODUCTION

Constructing an initial guess for a trajectory in a multi-body system is critical in enabling and assessing the feasibility of a specific itinerary. The initial guess often serves as the starting point for developing a solution that satisfies a variety of mission constraints and/or a desired optimality criterion. However, both robotic and human spacecraft are more frequently operating in the chaotic regimes of multi-body systems where conic solutions cannot be pieced together with either impulsive or low-thrust maneuvers to develop an initial guess.[1,2] Leveraging the dynamics of a multi-body system enables the design of more complex trajectories, but suffers from the absence of general analytical expressions to describe the solution space. Motivated by this need, data mining techniques are used in this paper to construct motion primitives that are leveraged as the fundamental building blocks for initial guess construction.

Data mining involves extracting knowledge from a large set of data.[3] In the context of a multi-body environment, the dataset is composed of trajectories in the nonlinear dynamical system. In a simplified model of a multi-body system, fundamental solutions such as libration points, families of periodic orbits, families of quasi-periodic orbits, and hyperbolic invariant manifolds govern motion throughout the system. Recently, Smith and Bosanac have used clustering algorithms to construct sets of motion primitives that summarize families of periodic orbits and hyperbolic invariant manifolds in the Earth-Moon circular restricted three-body problem (CR3BP).[4,5] A motion primitive is defined as the most representative solution in a cluster of trajectories. Periodic orbits are summarized based on geometry, stability, and energy while hyperbolic invariant manifolds are summarized

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

based on geometry. The resulting motion primitives are valuable for initial guess construction to reduce the burden on a human designer and enable rapid trajectory design.

The focus of this paper is to develop a strategy to employ motion primitives in initial guess construction to further advance rapid trajectory design strategies.[1,6,7] Visualizing a large set of trajectories may be overwhelming and difficult for a human to process in a high-dimensional space.[8] Therefore, a summarizing set of motion primitives supplies a more manageable representation of the available solutions from a family of trajectories. However, it is also vital to retain information about the corresponding cluster associated with each motion primitive, i.e. its region of existence, for use in constructing an initial guess. Selecting a motion primitive for use in the initial guess construction process identifies the general type of desired trajectory in a specific region of the system, but a slightly different trajectory may be desired within its region of existence.

Effectively storing the region of existence associated with a motion primitive enables the use of motion primitives for rapid trajectory design. Previously, the concepts of curve boxplots and trajectory funnels have been developed to summarize trajectories in the vicinity of a reference path in a dynamical system; curve boxplots have been leveraged to summarize and analyze paths in neuroimaging, hurricane forecasting, and fluid dynamics applications while trajectory funnels have been leveraged for robust robotic motion planning through a complex environment.[8,9] Fundamentally, both of these concepts leverage a discrete approximation of the reachable sets throughout an environment around a reference path for analysis and/or motion planning. Inspired by these approaches and similar to the clustering approach used to construct a set of distinct motion primitives from a large set of trajectories, k-means clustering is employed to develop an efficient and effective discrete approximation of the region of existence associated with a motion primitive as a set of representative trajectories and parameter bounds. The description strategy is explored for both periodic orbits as well as general nonperiodic trajectories along a hyperbolic invariant manifold to capture irregular cluster boundaries. With this description strategy, the simplified summarization of the solution space in a multi-body system is leveraged for initial guess construction.

To construct an initial guess for a trajectory, a qualitative itinerary is first established including the desired type of initial orbit, the general transfer geometry, and the type of final orbit. A qualitative itinerary supplies the foundation for focusing the initial exploration of the solution space on specific regions of the dynamical system. Based on the desired itinerary, various motion primitives that resemble the desired qualitative properties of the trajectory are selected to construct a rough initial guess. Design parameters of interest often include geometry, stability, and energy. The visualization of each motion primitive and its region of existence reduces the burden on the human analyst when selecting initial trajectory segments and facilitates a rapid exploration of the desired regions of the diverse solution space.

Based on the selection of initial primitives, an end-to-end initial guess may contain significant discontinuities but a similar type of continuous and/or smooth trajectory may be desired. To refine the initial guess and produce a continuous solution, this paper explores a constrained optimization approach that leverages a direct collocation scheme due to its robustness with respect to the quality of the initial guess.[10] Previously, Bosanac formulated a constrained optimization problem using discrete variational mechanics to generate periodic orbit solutions that closely resemble the geometry and desired behavior of the initial guess.[11] Furthermore, collocation schemes have been employed to robustly and effectively correct end-to-end trajectories in multi-body systems for a variety of applications.[12–14] This paper outlines a collocation targeting approach using Legendre-Gauss-Lobatto nodes, explores the development of an objective function to effectively quantify the resemblance between two trajectories, and investigates using constrained optimization to produce a continuous initial guess that resembles an initial set of selected trajectory segments. To demonstrate the proposed initial guess construction strategy, multiple planar transfers are constructed between an L_1 and L_2 Lyapunov orbit as well as a spatial transfer between an L_1 and L_2 northern halo orbit in the Earth-Moon CR3BP. The primitive-based transfer design approach presented in this paper leverages motion primitives to facilitate rapid trajectory design and reduces the analytical burden on a human designer when exploring the solution space for initial guess construction to construct a trajectory with a desired geometry.

2 BACKGROUND: DYNAMICAL MODEL

The CR3BP is commonly used to approximate the motion of a spacecraft under the gravitational influence of two celestial bodies.[15,16] In this paper, the Earth-Moon CR3BP is used to simulate, design, and analyze the motion of a spacecraft in the vicinity of the Earth and Moon. In this model, the Earth and Moon are modeled as constant point masses following circular orbits about their mutual barycenter.[15] The mass of the Earth is denoted as M_1, the mass of the Moon is denoted as M_2, and the spacecraft is assumed to possess a negligible mass compared to both primary bodies. Then, a rotating reference frame, $R : \{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$, is defined; the origin of R is at the barycenter of the system, $\hat{\boldsymbol{x}}$ is directed from the Earth to the Moon, $\hat{\boldsymbol{z}}$ is aligned with the orbital angular momentum of the system, and $\hat{\boldsymbol{y}}$ completes the right-handed orthogonal triad.[15] In addition, the length, mass, and time quantities of the system are nondimensionalized using the characteristic parameters l^*, m^*, and t^*, respectively.[16] The characteristic parameter l^* is set equal to the assumed constant distance between the Earth and the Moon, m^* is calculated as the total mass of the system, and t^* is defined such that the mean motion of the primary system is equal to unity. The nondimensional state of the spacecraft is then defined in the rotating frame as $\bar{\boldsymbol{x}} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$ relative to the origin of R. Given the preceding assumptions and definitions, the nondimensional equations of motion for a spacecraft in the CR3BP are written in the rotating frame as

$$\bar{\boldsymbol{f}}(\bar{\boldsymbol{x}}) = [\dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}]^T \tag{1}$$

where

$$\begin{aligned} \ddot{x} &= 2\dot{y} + x - \frac{(1-\mu)(x+\mu)}{r_1^3} - \frac{\mu(x-1+\mu)}{r_2^3} \\ \ddot{y} &= -2\dot{x} + y - \frac{(1-\mu)y}{r_1^3} - \frac{\mu y}{r_2^3} \\ \ddot{z} &= -\frac{(1-\mu)z}{r_1^3} - \frac{\mu z}{r_2^3} \end{aligned} \tag{2}$$

and $\mu = M_2/(M_1 + M_2)$ is the mass ratio of the system, while $r_1 = \sqrt{(x+\mu)^2 + y^2 + z^2}$ and $r_2 = \sqrt{(x-1+\mu)^2 + y^2 + z^2}$ are the distances between the spacecraft and the Earth and Moon, respectively. This autonomous dynamical system also admits an integral of motion known as the Jacobi constant that is conserved along natural trajectories and is defined as

$$C_J = (x^2 + y^2) + \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} - \dot{x}^2 - \dot{y}^2 - \dot{z}^2 \tag{3}$$

to produce an energy-like quantity that supplies insight into the allowable regions of motion.[16]

3 BACKGROUND: COMPUTING AND VISUALIZING FUNDAMENTAL SOLUTIONS

Motion in the CR3BP is governed by a variety of fundamental solutions from which more complex trajectories may be constructed. Fundamental solutions in the CR3BP include libration points, periodic orbits, quasi-periodic orbits, and hyperbolic invariant manifolds which are computed numerically.[11,16] This section supplies a general overview of the numerical techniques used to compute and analyze periodic orbits and hyperbolic invariant manifolds prior to transfer design.

3.1 Periodic Orbits

A periodic orbit in the CR3BP is a trajectory that repeats in the phase space in the rotating frame after a minimal time that is labeled the orbital period.[15] To compute a periodic orbit in the CR3BP, an initial guess is first constructed. An initial guess may be constructed using a variety of analytical and numerical methods.[11,15–17] Given an initial guess, a continuous and periodic solution is computed in this paper using a multiple shooting corrections scheme and multivariate Newton's method.[11,16] The initial guess is discretized into a series of arcs, each defined by an initial state and a common propagation time across all arcs. The initial state of each arc and the propagation time shared by the arcs are considered free variables. Using Newton's method, the free variables are iteratively updated until a continuous and periodic solution is computed within a numerical tolerance, where the continuity and periodic constraints are used to form a constraint vector. Then, pseudo-arclength continuation is used to compute additional members along the continuous family.[11,18]

Periodic orbits may serve as departure, staging, or target orbits in the trajectory design process. The local stability of a periodic orbit indicates the behavior of motion in the vicinity of the orbit and is often described using stability indices labeled s_1 and s_2.[19] These stability indices are calculated as the sum of nontrivial pairs of eigenvalues of the monodromy matrix for the associated periodic orbit. For planar periodic orbits, s_1 corresponds to the local in-plane stability and s_2 corresponds to the local out-of-plane stability. However, s_1 and s_2 do not have this clear distinction for spatial periodic orbits. A stability index with a magnitude greater than 2 indicates the existence of stable and unstable invariant manifolds, while a stability index with a magnitude less than or equal to 2 indicates the existence of bounded oscillatory motion in the vicinity of the periodic orbit.

3.2 Hyperbolic Invariant Manifolds

The hyperbolic invariant manifolds of periodic orbits are often used to construct transfers between periodic orbits.[16,20,21] Trajectories that lie along the unstable manifold asymptotically depart the periodic orbit while trajectories along the stable manifold asymptotically approach the periodic orbit.[16] To compute a stable or unstable manifold, an unstable periodic orbit is first discretized into a set of fixed points.[11,16] Then, a small perturbation is applied to each state in the direction of either the stable or unstable eigenspace associated with the fixed point. Finally, each perturbed state that approximately lies in the desired local manifold is propagated backwards or forwards in time to generate a portion of the global stable or unstable manifold, respectively. These trajectories that approximately lie along hyperbolic invariant manifold structures serve as powerful mechanisms for constructing transfers between periodic orbits in a multi-body system.

3.3 Poincaré Mapping

Poincaré mapping is a technique from dynamical systems theory that is often used in the trajectory design process.[22,23] To compute a Poincaré map, a desired hyperplane, Σ, is first defined by the trajectory designer. Then, the desired set of trajectories, such as the initial states along a hyperbolic invariant manifold, are numerically propagated for a desired time, number of returns to the map, or until a set of termination criteria are satisfied. Each crossing of the hyperplane along each trajectory is recorded and the resulting Poincaré map is a lower-dimensional representation of these crossings. Poincaré mapping has been leveraged extensively to analyze general motion in the CR3BP as well as to find potential connections between manifolds with relatively small discontinuities for initial guess construction during the transfer design process.

4 BACKGROUND: NUMERICALLY CORRECTING TRAJECTORIES

A differential corrections algorithm enables recovery of a continuous trajectory from a discontinuous initial guess. In this paper, a direct collocation scheme is used in conjunction with constrained optimization for corrections. The direct collocation scheme closely follows previous formulations of a generalized collocation approach with mesh refinement used by Grebow and Pavlak as well as Pritchett to solve trajectory design optimization problems in multi-body systems.[13,14]

4.1 Collocation

Collocation is a numerical method used to implicitly integrate the differential equations of a dynamical system.[10,24,25] Using collocation, a solution to a dynamical system is recovered by approximating the solution as sets of piecewise polynomials that satisfy the system dynamics at collocation points. Given an initial guess for a trajectory composed of multiple segments, the first step of collocation is to define a discrete mesh of nodes along the trajectory. The i^{th} segment is discretized into m_i arcs and, consequently, a total of $m_i + 1$ nodes at their boundaries, equally spaced in time. The nodes generated in this discretization process are referred to as boundary nodes where each node is described by its state and integration time. For a trajectory composed of N segments, this discretization produces a total of m arcs, where the time associated with the boundary node at the end of segment i is set equal to the time associated with the boundary node at the beginning of segment $i + 1$. After defining the initial mesh of boundary nodes along each segment of the trajectory, collocation nodes are placed along each arc using a desired implicit integration method and node spacing strategy. The implicit integration method is selected first because it determines the number of nodes placed along each arc. Higher-order n^{th} degree polynomials are often used for implicit integration in nonlinear dynamical systems due to the complexity of the dynamics.[10,25,26] Lower-order polynomials may be used; however, to achieve a similar level of accuracy, a fine mesh with many small arcs is needed when using lower-order polynomials compared to a coarser mesh with less arcs when using higher-order polynomials. Based on previous applications of collocation for trajectory design in multi-body systems, 7^{th} order polynomials are used in this paper.[12–14] Therefore, 7 collocation nodes are placed along each arc within each segment of a trajectory and the placement of the nodes is determined by the selected node spacing strategy.

To facilitate a clear discussion of the node spacing strategy, it is important to establish a set of definitions and notation for the properties of each arc along a trajectory as well as the parameterization of the polynomials. The state and time associated with a given node is defined as $\bar{\boldsymbol{x}}^i_{j,k}$ and $t^i_{j,k}$, respectively, where i refers to the segment index in the overall itinerary, j refers to the arc index in the i^{th} segment, and k refers to the node index along the j^{th} arc. Following this notation, the time along a given arc is calculated as $\Delta t^i_j = t^i_{j,n} - t^i_{j,1}$. Furthermore, each state variable along each arc of a trajectory is approximated with a distinct 7^{th} order polynomial parameterized by a normalized time quantity, τ, spanning from -1 to 1. The conversion from time to normalized time along each arc is defined as

$$\tau = 2\left(\frac{t - t^i_{j,1}}{\Delta t^i_j}\right) - 1 \tag{4}$$

where t is the time along the given arc between $t^i_{j,1}$ and $t^i_{j,n}$. Therefore, a state at τ_k is approximated by the polynomials for the j^{th} arc and labeled as $\bar{\boldsymbol{p}}^i_j(\tau_k)$. Then, the normalized time derivative of the state $\bar{\boldsymbol{x}}^i_{j,k}$ is defined as

$$\dot{\bar{\boldsymbol{x}}}^i_{j,k} = \frac{\Delta t^i_j}{2}\bar{\boldsymbol{f}}(\bar{\boldsymbol{x}}^i_{j,k}) \tag{5}$$

whereas the normalized approximation of the state derivative via the polynomials is $\dot{\bar{\boldsymbol{p}}}^i_j(\tau_k)$.

In collocation, a node spacing strategy is used to determine the location of the collocation nodes along each arc. A common node spacing strategy in trajectory design is Legendre-Gauss-Lobatto (LGL) node spacing.[10,13,27] In this method, collocation nodes are placed at the boundary nodes of each arc as well as at times τ equal to the roots of the derivative of the $n-1^{th}$ order Legendre polynomial, ranging from -1 to 1. Additionally, a LGL weighting term, w, is computed for each node. Leveraging LGL node spacing is particularly advantageous because it simplifies the design problem by placing collocation nodes at the boundary nodes of each arc.

A free variable and constraint vector formulation is used to transform the trajectory design problem into a parameter design problem.[10,24] First, the free variable vector is defined using parameters of the discretized initial guess. Along each arc, the odd-numbered collocation nodes are classified as free nodes and the even-numbered collocation nodes are classified as defect nodes. The free nodes are used to construct the approximating polynomials along each arc, while the defect nodes are defined by the constructed polynomials and are used to evaluate how well the system dynamics are approximated by the polynomials. Figure 1 depicts a conceptual example with each arc containing a set of 7 nodes (4 free nodes and 3 defect nodes) as determined by the 7^{th} order implicit integration method and the nodes are spaced along each arc using LGL node spacing. Free nodes along each arc are denoted in blue, defect nodes are denoted in red, and boundary nodes are outlined in black. When using LGL node spacing, the boundary nodes of each arc are also considered collocation nodes and are further classified as free nodes. As depicted in Figure 1, consecutive arcs within a segment share a common free boundary node. However, the final free boundary node along segment i is distinct from the initial free boundary node along segment $i+1$. Following this structure, the state at each free node along each arc of the trajectory is included in the free variable vector. The change in time along each arc is also included in the free variable vector so that the total time of flight may vary. The resulting free variable vector for the i^{th} segment is defined as

$$\bar{\boldsymbol{V}}_i = \begin{bmatrix} \begin{bmatrix} \bar{\boldsymbol{x}}^i_{1,1} \\ \bar{\boldsymbol{x}}^i_{1,3} \\ \vdots \\ \bar{\boldsymbol{x}}^i_{1,n-2} \end{bmatrix}^T \begin{bmatrix} \bar{\boldsymbol{x}}^i_{2,1} \\ \bar{\boldsymbol{x}}^i_{2,3} \\ \vdots \\ \bar{\boldsymbol{x}}^i_{2,n-2} \end{bmatrix}^T \dots \begin{bmatrix} \bar{\boldsymbol{x}}^i_{m_i-1,1} \\ \bar{\boldsymbol{x}}^i_{m_i-1,3} \\ \vdots \\ \bar{\boldsymbol{x}}^i_{m_i-1,n-2} \end{bmatrix}^T \begin{bmatrix} \bar{\boldsymbol{x}}^i_{m_i,1} \\ \bar{\boldsymbol{x}}^i_{m_i,3} \\ \vdots \\ \bar{\boldsymbol{x}}^i_{m_i,n} \end{bmatrix}^T \begin{bmatrix} \Delta t^i_1 \\ \Delta t^i_2 \\ \vdots \\ \Delta t^i_{m_i} \end{bmatrix}^T \end{bmatrix} \tag{6}$$

where $n = 7$. The resulting free variable vector for the entire trajectory is defined as

$$\bar{\boldsymbol{V}} = \begin{bmatrix} \bar{\boldsymbol{V}}_1 & \bar{\boldsymbol{V}}_2 & \dots & \bar{\boldsymbol{V}}_N \end{bmatrix}^T \tag{7}$$

to produce an $((3n-2)m + 6N)$-dimensional vector for N segments.

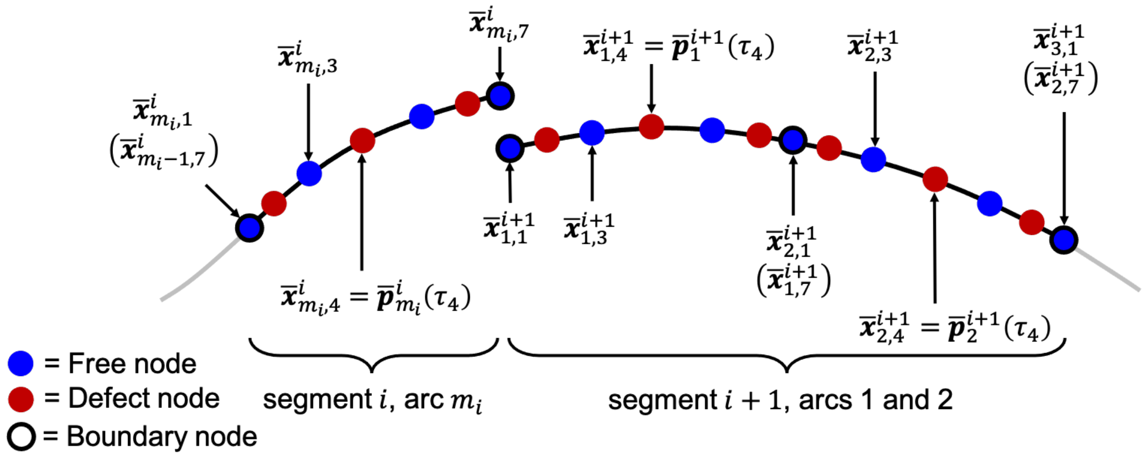

Figure 1. Conceptual example of collocation nodes placed along multiple arcs of segment i and $i+1$ using 7^{th} order LGL node spacing.

To compute an end-to-end continuous trajectory, a set of continuity and defect constraints are defined as functions of the free variables. Continuity is automatically enforced between arcs within a segment due to the use of LGL nodes because each pair of consecutive arcs shares a common free boundary node.[13] However, continuity is not automatically enforced between consecutive segments, as depicted conceptually in Figure 1 between nodes $\bar{\boldsymbol{x}}^i_{m_i,7}$ and $\bar{\boldsymbol{x}}^{i+1}_{1,1}$. Therefore, the continuity constraint is defined between each pair of consecutive segments as $\bar{\boldsymbol{F}}_{c_i} = \bar{\boldsymbol{x}}^{i+1}_{1,1} - \bar{\boldsymbol{x}}^i_{m_i,7}$ and the resulting continuity constraint vector for the entire trajectory is defined as $\bar{\boldsymbol{F}}_c = \begin{bmatrix} \bar{\boldsymbol{F}}^T_{c_1} & \bar{\boldsymbol{F}}^T_{c_2} & \dots & \bar{\boldsymbol{F}}^T_{c_{N-1}} \end{bmatrix}$ to enforce full state continuity between each consecutive segment. Impulsive maneuvers may be incorporated at the beginning of a segment by enforcing only position continuity between consecutive segments. In addition to continuity constraints, defect constraints are needed along each arc of the full trajectory to enforce the system dynamics at each defect node. The state of each defect node is computed directly from the constructed polynomials along the corresponding arc. Then, each defect constraint evaluates the difference between the approximated dynamics computed using the normalized time derivatives of the polynomials and the actual dynamics computed at each defect node using the equations of motion. These constraints ensure that the dynamics are well-approximated by the sets of polynomials constructed along each arc. The defect constraint vector for the j^{th} arc in the i^{th} segment is defined as

$$\bar{\boldsymbol{F}}^i_{d_j} = \begin{bmatrix} \bar{\boldsymbol{\Delta}}^i_{j,2} \\ \bar{\boldsymbol{\Delta}}^i_{j,4} \\ \vdots \\ \bar{\boldsymbol{\Delta}}^i_{j,n-1} \end{bmatrix} = \begin{bmatrix} (\dot{\bar{\boldsymbol{p}}}^i_j(\tau_2) - \dot{\bar{\boldsymbol{x}}}^i_{j,2})w_2 \\ (\dot{\bar{\boldsymbol{p}}}^i_j(\tau_4) - \dot{\bar{\boldsymbol{x}}}^i_{j,4})w_4 \\ \vdots \\ (\dot{\bar{\boldsymbol{p}}}^i_j(\tau_{n-1}) - \dot{\bar{\boldsymbol{x}}}^i_{j,n-1})w_{n-1} \end{bmatrix} \tag{8}$$

where $n = 7$ and each w_k term is the LGL weight associated with the k^{th} collocation node. Then, the defect constraint vector for the i^{th} segment is defined as $\bar{\boldsymbol{F}}_{d_i} = \begin{bmatrix} \bar{\boldsymbol{F}}^{i\ T}_{d_1} & \bar{\boldsymbol{F}}^{i\ T}_{d_2} & \dots & \bar{\boldsymbol{F}}^{i\ T}_{d_{m_i}} \end{bmatrix}$ while the constraint vector for the entire trajectory is defined as

$$\bar{\boldsymbol{F}}(\bar{\boldsymbol{V}}) = \begin{bmatrix} \bar{\boldsymbol{F}}_c & \bar{\boldsymbol{F}}_{d_1} & \bar{\boldsymbol{F}}_{d_2} & \dots & \bar{\boldsymbol{F}}_{d_N} \end{bmatrix}^T \tag{9}$$

to produce an $((3n-3)m + 6(N-1))$-dimensional vector.

4.2 Optimization to Enforce Geometry

Direct collocation is often used to transcribe a trajectory optimization problem into a parameter optimization problem that may be solved using nonlinear programming.[10,24] The free variable and constraint vector formulation outlined in Section 4.1 describes the direct collocation, or transcription, method used in this paper. The goal of an optimization algorithm is then to compute a solution that minimizes, or maximizes, a specified objective function such that all the constraints are satisfied to within a numerical tolerance. In this paper, which focuses on the initial guess construction process, the goal is to find a solution that resembles the geometry of the discontinuous initial guess. Therefore, the initial guess serves as a reference geometry and the goal is to minimize the dissimilarities between the final continuous solution and the discontinuous initial guess. Using optimization to recover a trajectory with a similar geometry as a reference path has previously been explored in both robotic path planning and periodic orbit computation in multi-body systems.[11,28]

The objective function for optimization is formulated to quantify the difference in geometry between two trajectory solutions. This objective function is defined as

$$J(\bar{\boldsymbol{V}}_{Pos}) = (\bar{\boldsymbol{V}}_{Pos} - \bar{\boldsymbol{V}}_{IG_{Pos}})^T(\bar{\boldsymbol{V}}_{Pos} - \bar{\boldsymbol{V}}_{IG_{Pos}}) \tag{10}$$

where $\bar{\boldsymbol{V}}_{Pos}$ is the portion of $\bar{\boldsymbol{V}}$ that only includes the position components of each free node along the current solution and $\bar{\boldsymbol{V}}_{IG_{Pos}}$ is a vector comprised of the position components of each free node along the reference initial guess. The iterative sequential quadratic programming (SQP) algorithm implemented in MATLAB®'s $fmincon$ function is leveraged to minimize the objective function defined in Equation 10 subject to the continuity and defect equality constraints defined in Equation 9 using the free variables defined in Equation 7. When the norm of the constraint vector is below a specified tolerance of 10^{-12} and the first-order optimality tolerance is below 10^{-5}, the resulting solution computed using $fmincon$ is considered an end-to-end continuous trajectory that geometrically resembles the original discontinuous initial guess.

4.3 Mesh Refinement

The accuracy of the initial optimal solution computed using collocation and the optimization algorithm depends on the initial mesh. Each arc in the mesh is equivalent to a single integration step and, despite the defect constraints being satisfied, the solution may not accurately approximate the system dynamics between collocation points, particularly in sensitive regions of the dynamical system.[14] Therefore, a mesh refinement procedure is coupled with collocation to improve the accuracy of the final solution. In this paper, a hybrid mesh refinement algorithm is employed that closely follows the procedure presented by Grebow and Pavlak, using both analytical and numerical analysis to control the dynamical error along the solution.[13,14]

Once the initial optimal solution is computed, Carl de Boor's method is employed to iteratively distribute error equally between arcs along the solution.[13,29,30] During a single iteration of de Boor's method, an analytical approximation of the error along each arc as well as the total error along the entire trajectory are computed. Then, the state and time of the boundary nodes of each arc are updated, using the polynomials of the previously converged mesh, to equally distribute the total error between the arcs in the mesh. The free LGL nodes for each arc are then recomputed using numerical propagation between the updated boundary nodes. During this error distribution process, both the number of arcs in the mesh and the total flight time for the trajectory is constant. Finally, using the updated mesh as the new reference initial guess, the trajectory is input to the optimization scheme to produce a continuous path with a geometry that resembles the updated initial guess. If the maximum approximated error difference between any two arcs in the mesh is greater than a specified tolerance of 10^{-5}, the iterative procedure continues; however, the error distribution step is terminated once the maximum error difference along the solution is less than the tolerance.

When the error distribution step is terminated, Control with Explicit Propagation (CEP) is used to iteratively merge arcs along the mesh to reduce the size of the sparse optimization problem.[13,14] This step of mesh refinement numerically computes the error at the end of each pair of consecutive arcs in the mesh. For example, the state at the initial boundary node of the first arc is propagated until the time associated with the final boundary node of the second arc. Then, the error is computed between the final propagated state and the state associated with the final boundary node of the second arc. If the magnitude of the error vector is below a tolerance of 10^{-12}, then the two arcs are merged into a single arc. The initial boundary node of the first arc and the final boundary node of the second arc serve as the initial and final boundary nodes, respectively, of the merged arc. Then, the free LGL nodes are recomputed between the updated boundary nodes using numerical propagation. If the error is above the specified tolerance, then the two arcs are not merged and the next two consecutive arcs in the mesh are evaluated. If any arcs are merged along the entire trajectory, then the reference initial guess is updated using the new mesh and input to the optimization scheme to produce a continuous path with a geometry that resembles the updated initial guess. The merging process is repeated until no arcs are merged along the solution.

CEP is also used to iteratively split arcs along the mesh after finishing the merging process by numerically computing the error at the end of each arc. For example, the state at the initial boundary node of the first arc is propagated until the time associated with the final boundary node of the arc. Then, the error is computed between the final propagated state and the state at the final boundary node of the arc. If the magnitude of the error vector is above a tolerance of 10^{-12}, then the arc is split into two separate arcs at its midpoint in terms of time. The polynomials from the previously converged mesh are used to compute the state and time at the midpoint of the arc and then the free LGL nodes are recomputed for each arc as previously described. If any arcs are split along the entire trajectory, then the solution is re-computed using the updated mesh as the new reference initial guess and the process is repeated until no arcs are split along the solution. Through this process, the numerical error of the implicit integration method is controlled and a desired level of accuracy is achieved along the final continuous solution that geometrically resembles the initial guess.

5 BACKGROUND: MOTION PRIMITIVE CONSTRUCTION

Motion primitives are used to summarize the solution space of a dynamical system and, subsequently, construct complex paths throughout the system. The concept of a motion primitive has been explored extensively in robotic path planning, human body gesture analysis, and autonomous vehicle transportation.[31–33] In each of these fields and depending on the application, motion primitives are constructed and defined in a variety of manners using analytical and numerical techniques. Recently, Smith and Bosanac formally defined a motion primitive in the context of trajectories in a multi-body system and leveraged clustering, a data mining and machine learning technique, to construct sets of motion primitives.[3–5] This approach has been employed to summarize families of periodic orbits and hyperbolic invariant manifolds in the Earth-Moon system based on geometry, stability, and energy. For example, clustering is used to decompose a family of periodic orbits into multiple clusters in which the members of a given cluster each exhibit a similar geometry, values of the stability indices, and the Jacobi constant. A motion primitive is then extracted from each cluster as its medoid, which is the member of the cluster that is most similar to all other members.[5] Using this process, continuous sets of trajectories that exhibit a finite number of distinct characteristics may be summarized via a finite set of motion primitives. More recently, Smith and Bosanac have refined the motion primitive construction process in multi-body systems to increase its robustness relative to the clustering approach and limit the inputs required from the trajectory designer. The updated motion primitive construction process leverages k-means and agglomerative clustering in conjunction with Weighted Evidence Accumulation Clustering (WEAC) and an automated input parameter selection process to compute sets of motion primitives.[3,34,35] The output of this process for a given set of trajectories, is a set of motion primitives and their corresponding clusters.

6 SUMMARIZING THE REGION OF EXISTENCE FOR A MOTION PRIMITIVE

Trajectories that resemble a motion primitive exist within a region of the phase space labeled its region of existence. The region of existence associated with a motion primitive may (i) provide a trajectory designer with additional dynamical insight when selecting suitable primitives for initial guess construction and (ii) be leveraged to significantly improve the quality of a discontinuous initial guess originally constructed solely from primitives. The data mining approach leveraged by Smith and Bosanac that numerically constructs a motion primitive from a cluster of trajectories in a multi-body system directly supplies an intuitive, discrete approximation of its region of existence.[4,5] The associated cluster of the motion primitive captures the solution variations in the vicinity of the primitive. However, this description of the region of existence does not efficiently scale for large clusters in terms of data storage. To mitigate this challenge, a data driven approach is leveraged to efficiently represent the region of existence associated with a motion primitive in the CR3BP.

Similar to the procedure used to extract motion primitives, k-means clustering is employed to develop a discrete approximation of the region of existence associated with a motion primitive. Given a cluster of trajectories, C, and a corresponding motion primitive in the CR3BP, C is first partitioned into k subclusters using the k-means algorithm.[3] In k-means, the value of k is a user-defined input parameter that, in this application, is selected based on the desired degree of granularity for the region of existence approximation. Furthermore, this common clustering algorithm is computationally efficient and tends to produce evenly-sized clusters.[3] Consequently, a representative trajectory from each subcluster is computed as the medoid of the corresponding subcluster.[5] The resulting k trajectories, the motion primitive, and a set of boundary trajectories are each labeled as a representative trajectory, $\bar{\boldsymbol{x}}_R(t)$. The boundary trajectories are determined based on extrema in Jacobi constant within C. However, if all of the trajectories in C exist at the same Jacobi constant, then the boundaries of C are determined based on extrema in total propagation time, t_{Int}. Furthermore, if C contains k members or less, then all of the trajectories in C are labeled as representative trajectories. The region of existence for a motion primitive is then explicitly defined as the set $R_E = \{\bar{\boldsymbol{x}}_R(t) \in C\}$ and the properties of the representative trajectories are stored.

The region of existence of a motion primitive summarizes the solutions in the phase space that exist in its vicinity. Figure 2 depicts this region of existence description in the Earth-Moon CR3BP for both an L_1 Lyapunov motion primitive and a primitive along the unstable manifold of an L_1 Lyapunov orbit, each constructed using $k = 10$. In Figure 2a, the surface spanned in configuration space by the representative trajectories is displayed in gray and the motion primitive is denoted in blue. Additionally, the boundaries of the region of existence in terms of C_J are denoted with a dashed black line (minimum) and a solid black line (maximum). Figure 2b depicts the region of existence for a general nonperiodic trajectory primitive in the Earth-Moon system along the unstable

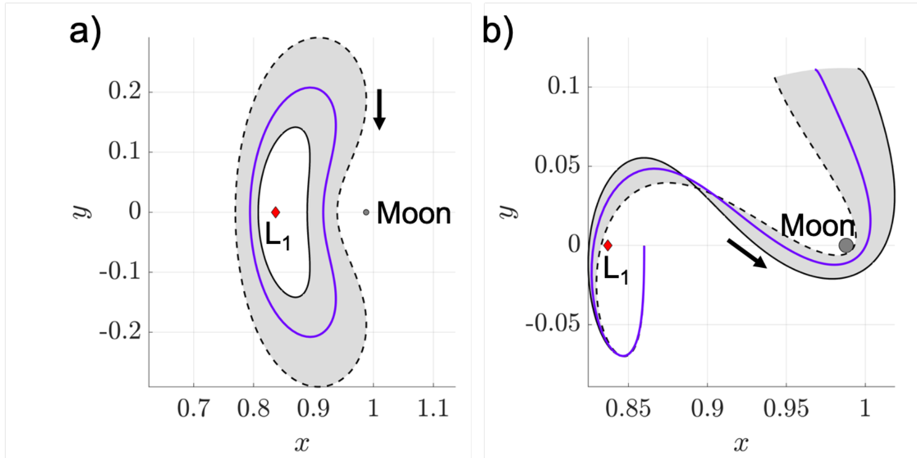

Figure 2. Regions of existence for a primitive in a) the L_1 Lyapunov orbit family and b) an unstable manifold of an L_1 Lyapunov orbit in the Earth-Moon system.

manifold associated with an L_1 Lyapunov orbit. This visual representation of the region of existence associated with a motion primitive along with selected trajectory characteristics, such as C_J, t_{Int}, s_1, and/or s_2, facilitates rapid exploration and analysis of the solution space in a multi-body system. It provides a trajectory designer with a direct summary of the solutions that exist in the phase space in the vicinity of a motion primitive to leverage in the initial guess construction process.

7 USING MOTION PRIMITIVES FOR TRANSFER DESIGN

7.1 Transfer Design Procedure

In this paper, sets of motion primitives and their corresponding regions of existence are leveraged in a transfer design procedure that is demonstrated by computing transfers between libration point orbits in the Earth-Moon CR3BP. The transfer design procedure is split into two main components. First, an initial guess is constructed using sets of motion primitives to enable rapid exploration of the solution space. Then, the initial guess seeds a trajectory corrections and optimization approach that leverages the concepts and techniques presented in Section 4. Figure 3 conceptually depicts the corrections algorithm used to compute a continuous transfer with a similar geometry as the discontinuous initial guess. The result is a continuous trajectory that resembles the initial guess constructed by a trajectory designer when a solution exists. This section demonstrates the transfer design procedure for an L_1 to L_2 Lyapunov orbit transfer example in the Earth-Moon system.

7.1.1 Initial Guess Construction In this example, the first step in the transfer design procedure is to select the initial and target orbits from a set of motion primitives that summarize periodic orbit families of interest. Figure 4 depicts a set of motion primitives constructed from the L_1 Lyapunov family in the rotating frame of the Earth-Moon CR3BP. In Figure 4a, the motion primitives are colored based on C_J, and in Figure 4b, the primitives are colored based on the magnitude of the planar stability index, s_1. The motion primitives displayed in Figure 4 provide a trajectory designer with a summary of the distinct geometric, energetic, and planar stability properties of the representative members of the L_1 Lyapunov family in the Earth-Moon CR3BP. If desired, the region of existence associated with each primitive may be explored in further detail using the representation outlined in Section 6. From these motion primitives, a suitable initial orbit for the transfer is selected. For

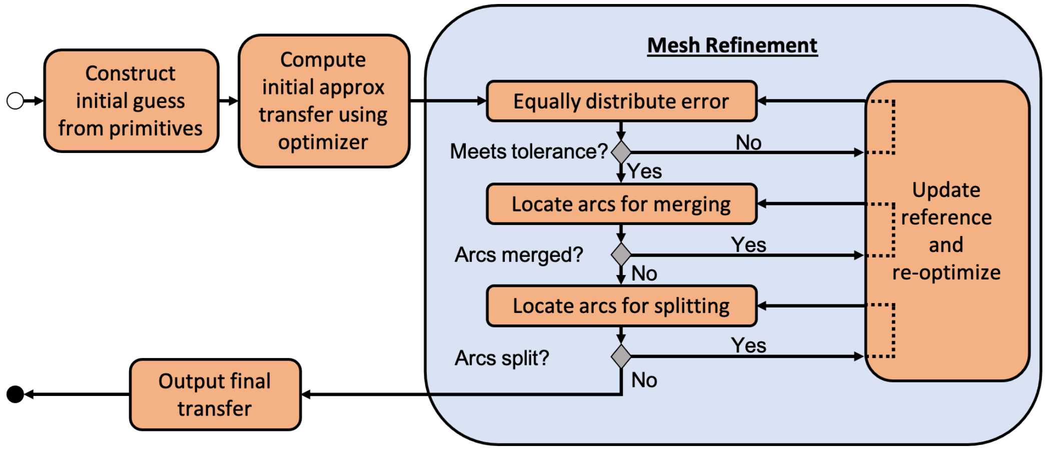

Figure 3. Conceptual overview of the corrections algorithm employed to correct a discontinuous initial guess into a continuous end-to-end trajectory.

transfer design, unstable periodic orbits are specifically desired because they admit stable and unstable hyperbolic manifolds that can be leveraged for transport throughout the multi-body system.[16] As displayed in Figure 4b, all of the primitives constructed from the L_1 Lyapunov orbit family possess stable and unstable planar modes. Therefore, any of the orbits may be suitable in designing a transfer between L_1 and L_2. Similarly, a suitable target orbit is selected from the L_2 Lyapunov family in the Earth-Moon CR3BP. The selected L_1 Lyapunov primitive is at C_J = 3.1670, the selected L_2 Lyapunov primitive is at C_J = 3.1666, and both orbits possess stable and unstable hyperbolic invariant manifolds. The desired qualitative transfer itinerary for the example therefore departs the initial L_1 Lyapunov orbit along its unstable manifold and approaches the target L_2 Lyapunov orbit along its stable manifold. Consequently, a set of motion primitives is constructed to summarize trajectories along the unstable manifold towards the Moon associated with the selected L_1 Lyapunov orbit primitive. Furthermore, a set of motion primitives is constructed from trajectories along the stable manifold towards the Moon associated with the selected L_2 Lyapunov orbit primitive.

Given an initial orbit, a target orbit, and the sets of motion primitives for the associated unstable and stable manifolds, the solution space is explored via motion primitives. First, the initial orbit is discretized into a set of 100 states equally spaced in arclength. For each state along the initial orbit, the c nearest primitives are computed from the set of motion primitives that summarize the unstable manifold. The distance between a state along the initial orbit and a primitive is computed as the magnitude of the full state difference between the given orbit state and the initial state of the primitive. Consequently, the trajectory designer is required to select a value of c to specify the number of nearest primitives that are computed relative to each state. Then, duplicates are filtered out of the final set of nearest primitives for the initial orbit. This process is repeated for the target orbit using the primitive set constructed from its associated stable manifold and the same value of c. However, the final state of each primitive is used instead of the initial state to compute the distance between a primitive and each state along the target orbit. Figure 5 depicts the results of this process for the L_1 to L_2 Lyapunov transfer example using c = 3. In Figure 5a, the final set of nearest primitives computed relative to the initial L_1 Lyapunov orbit along its associated unstable manifold are depicted as different shades of red where each open circle indicates the starting point of the corresponding primitive and each filled circle indicates the terminal point of the corresponding primitive. Similarly, Figure 5b displays the final set of nearest primitives relative to the target L_2

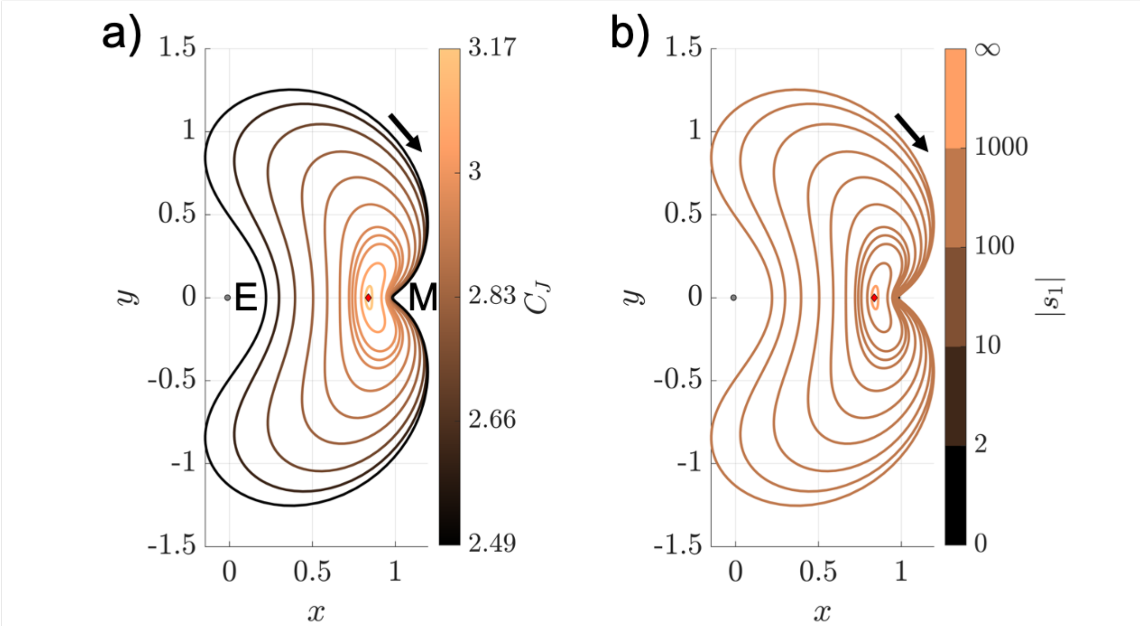

Figure 4. Set of motion primitives constructed from the L_1 Lyapunov family in the Earth-Moon system displayed as a function of a) C_J and b) $|s_1|$.

Lyapunov orbit along its associated stable manifold as different shades of blue.

Visualizing the computed sets of nearest primitives relative to the initial and target orbit, respectively, provides insight into the available departure and arrival geometries for transfer design. In the L_1 to L_2 Lyapunov orbit transfer example, a single revolution around the Moon may be desired. To simplify the solution space further, the motion primitives depicted in Figure 5 help guide the designer to select a candidate departure point along the initial orbit and a candidate arrival point along the final orbit. When a candidate departure point or arrival point is selected, Figures 5a and 5b are updated to show only the c nearest primitives relative to the selected candidate departure or arrival point, respectively. Figures 6a and 6b display the closest primitives relative to a selected departure point and a selected arrival point, respectively. The selected points are denoted as red circles with a black outline while the formatting of the primitives is consistent with Figure 5. Interactively exploring the solution space in this manner using motion primitives limits the amount of information that needs to be processed and provides a trajectory designer with fundamental geometric insights for selecting candidate segments to use in an initial guess for a transfer.

Primitives are selected from each set of candidates within each segment of the transfer to construct an initial guess. To facilitate the selection process, a Poincaré map is used to visualize the potential primitives and their associated regions of existence. Figure 6c displays a perilune map in the $x - y$ plane computed from the primitives in Figures 6a and 6b and their corresponding regions of existence for up to 2 returns. The perilune crossings on the map are colored according to their associated primitive trajectory, where the perilune crossings for each primitive are additionally outlined in black. Furthermore, the planar velocity components of the state in the rotating frame at each perilune are represented by the direction and magnitude of the vectors protruding from each perilune crossing.[23] The length of each vector is normalized by the perilune crossing with the largest velocity magnitude to provide a relative comparison between the velocities at each perilune on the map. The horizontal component of each vector represents $\dot{x}$ and the vertical component represents $\dot{y}$. Finally, a gray vector is associated with each region of existence perilune crossing while a colored vector is associated with each primitive trajectory as denoted in Figures 6a and 6b.

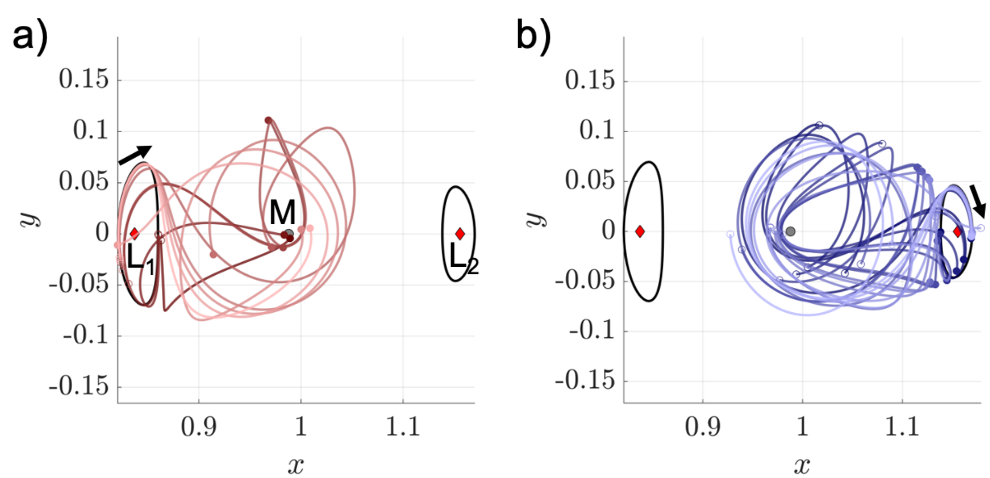

Figure 5. a) Nearest primitives in the unstable manifold primitive set relative to the initial L_1 Lyapunov orbit and b) nearest primitives in the stable manifold primitive set relative to the target L_2 Lyapunov orbit.

Analyzing the generated map displayed in Figure 6c for the L_1 to L_2 Lyapunov transfer example, there appears to be a potential connection between the unstable and stable manifold primitives in the region denoted by the black box. Correlating the data depicted in Figures 6a, 6b, and 6c, this connection corresponds to a geometry that exhibits a single revolution around the Moon when transferring from L_1 to L_2. Figure 6 specifically provides insight into the regions in which trajectories with a specific geometry exist in the configuration space that is invaluable for finding and constructing a good initial guess with a desired geometry. The resulting initial guess is generated by selecting the primitives within the region denoted by the black box in Figure 6c. Then, this initial guess is improved by further trimming the selected primitives to remove any overlap of arcs.

The final step of initial guess construction is to morph the selected primitives within their corresponding regions of existence to improve the initial guess for the desired transfer. The initial guess consists of an ordered set of motion primitives that each possess a region of existence, which is described by a small set of discrete trajectories using the representation method formulated in Section 6. To morph the selected primitives within their regions of existence, a brute force search is first used to compute the average full state discontinuity along every possible transfer that may be constructed from the ordered set of motion primitives and their corresponding regions of existence. Then, the ordered combination of segments with the minimum average full state discontinuity is selected as the initial guess. Figure 7 shows the result of morphing the trimmed initial guess within the corresponding regions of existence associated with the primitives. Each primitive and its region of existence are denoted with a different color. Furthermore, the original initial guess is denoted with dashed lines while the morphed initial guess is denoted with solid lines. As displayed in Figure 7, the morphed initial guess has a smaller average state discontinuity along the transfer than the original initial guess. Finally, two full revolutions of each of the L_1 Lyapunov and L_2 Lyapunov orbits are added to the itinerary to support effective corrections.

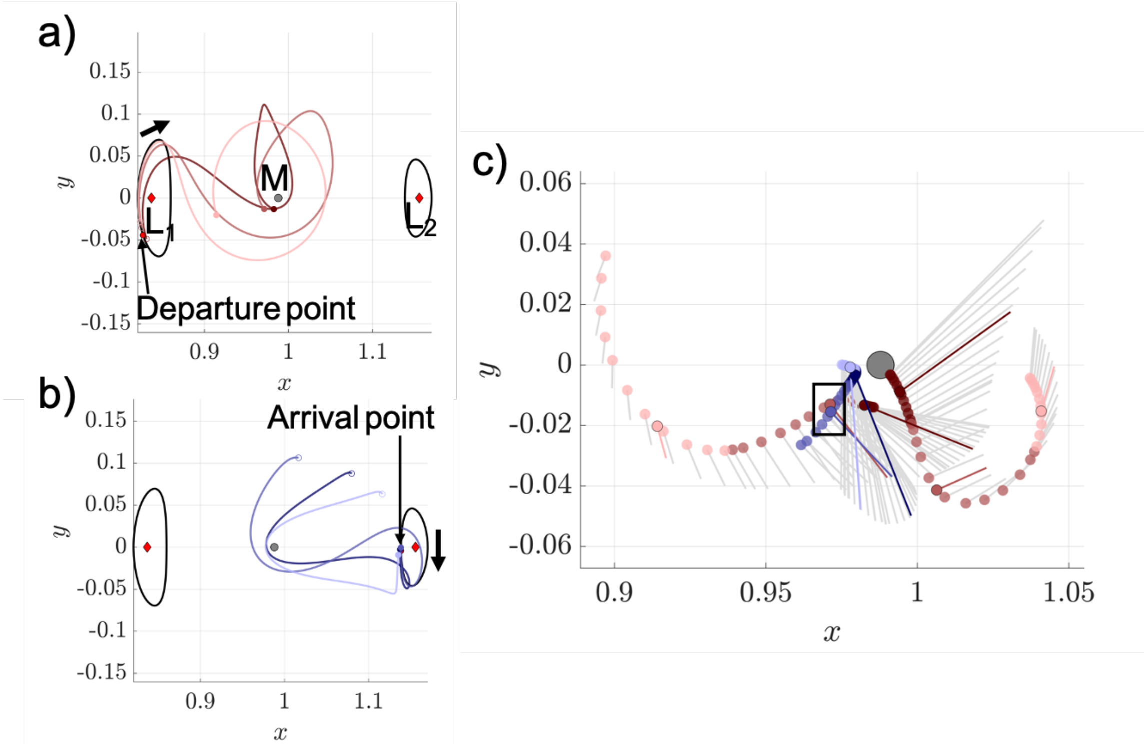

Figure 6. a) Nearest primitives in the unstable manifold primitive set relative to the selected departure point along the initial L_1 Lyapunov orbit, b) nearest primitives in the stable manifold primitive set relative to the selected arrival point along the target L_2 Lyapunov orbit, and c) the resulting perilune map with up to 2 returns.

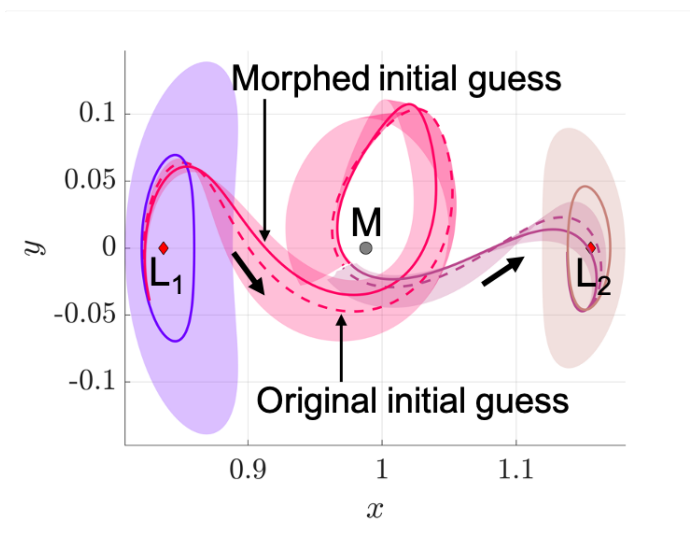

Figure 7. Morphed initial guess for an L_1 to L_2 Lyapunov orbit transfer with a single revolution around the Moon displayed with respect to the original initial guess of primitives and their corresponding regions of existence.

7.1.2 Trajectory Corrections and Optimization Direct collocation and a constrained optimization approach are used to correct the initial guess in Figure 7 to produce a continuous trajectory. Using the collocation scheme outlined in Section 4, each periodic orbit segment is discretized into 10 nodes and the manifold segments are discretized into 20 nodes, equally spaced in time. Additional nodes are included along the manifold primitives because they exist in more sensitive regions of the dynamical system near the Moon. MATLAB®'s $fmincon$ is then used with mesh refinement to compute a continuous transfer and ensure the final solution meets a desired level of accuracy.

7.2 L_1 to L_2 Lyapunov Transfers in the Earth-Moon System

Following the procedure demonstrated in Section 7.1, transfers with two different geometries between an L_1 and L_2 Lyapunov orbit are constructed. The first transfer uses a discontinuous initial guess constructed in Section 7.1 and displayed in Figure 7. A natural transfer is constructed from the morphed initial guess and displayed in Figure 8a. The final solution is denoted in blue and the original morphed initial guess is denoted in gray with dashed lines. The resulting continuous natural transfer closely resembles the constructed initial guess, exists at $C_J = 3.1669$, and has an approximate transfer time of 24.1 days. An additional transfer with the single revolution geometry is computed by incorporating two impulsive maneuvers. The first maneuver, $\Delta\bar{v}_1$, is used to depart the initial orbit and the second maneuver, $\Delta\bar{v}_2$, is used to insert into the target orbit. The resulting impulsive transfer is displayed in Figure 8b with respect to the original morphed initial guess. The segment of the transfer along the manifold primitives are nearly identical to the natural transfer in Figure 8a with a transfer time of approximately 23.5 days. However, the impulsive maneuvers result in a closer resemblance to the original initial and final orbits. The C_J at the initial state of the transfer is equal to 3.1670 while the C_J at the final state of the transfer is equal to 3.1666. Furthermore, the magnitude of $\Delta\bar{v}_1$ is 4.8 m/s and the magnitude of $\Delta\bar{v}_2$ is 5.2 m/s.

In addition to the single revolution transfers, the primitive-based transfer design procedure is also employed to compute transfers with two revolutions around the Moon between the same two Lyapunov orbits. Figure 9a displays the original and morphed initial guesses constructed for a transfer with two revolutions around the Moon while Figure 9b shows the perilune map with up to 2 returns for the selected untrimmed manifold primitives, where the color of the points corresponds

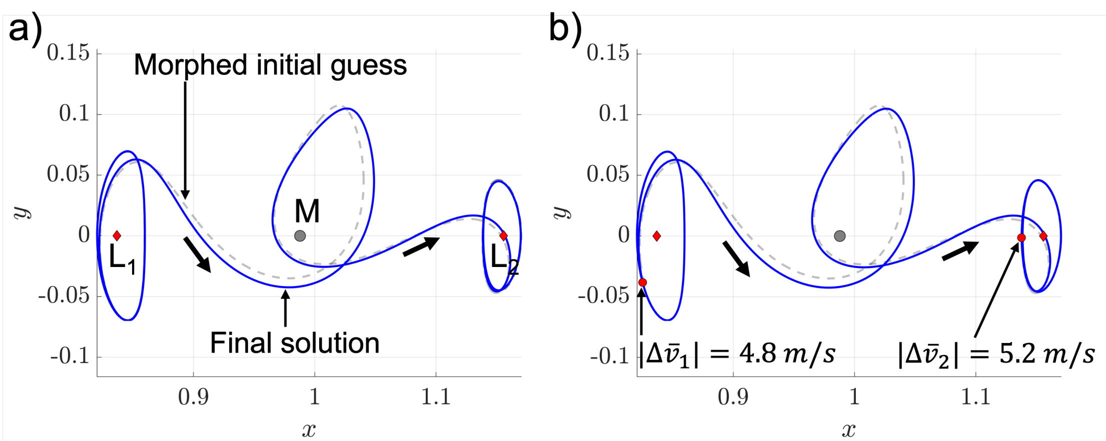

Figure 8. a) Natural and b) maneuver-enabled transfers between an L_1 and L_2 Lyapunov orbit with a single revolution around the Moon.

to the associated manifold primitive. The original initial guess exhibits a significant discontinuity between the manifold primitives; however, this example demonstrates the utility of the morphing process that leverages the regions of existence to significantly improve a poor initial guess before using the corrections algorithm. A natural transfer computed from the initial guess is displayed in Figure 10a while a transfer with a single impulsive maneuver is displayed in Figure 10b. The corrections algorithm successfully recovers a continuous natural solution with a similar geometry as the discontinuous initial guess at $C_J = 3.1668$ with an approximate transfer time of 32.3 days. By introducing an impulsive maneuver between the two manifold primitives, the impulsive transfer more closely retains the geometry of the discontinuous initial guess. The impulsive transfer has an approximate transfer time of 32.9 days, incorporates a $|\Delta\bar{v}|$ of 76.1 m/s between the manifold primitives, starts at $C_J = 3.1669$, and terminates at $C_J = 3.1666$. These L_1 to L_2 Lyapunov orbit transfers demonstrate the capability of a primitive-based transfer design procedure to rapidly construct initial guesses that produce continuous trajectories with a desired geometry.

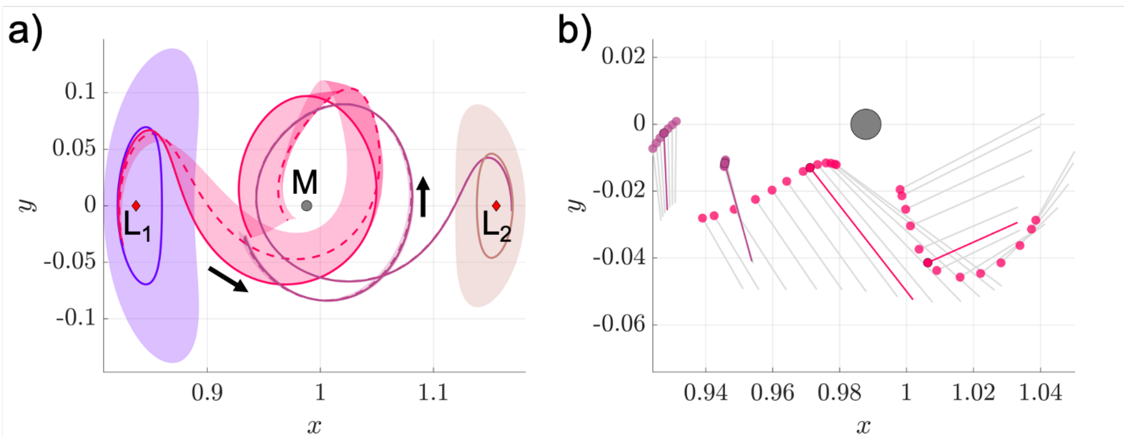

Figure 9. a) Morphed initial guess for an L_1 to L_2 Lyapunov orbit transfer with two revolutions around the Moon displayed with respect to the original initial guess of primitives and their corresponding regions of existence and b) the resulting perilune map with up to 2 returns for the selected untrimmed manifold primitives.

7.3 L_1 to L_2 Northern Halo Transfer in the Earth-Moon System

Following the transfer design procedure outlined in Section 7.1, a maneuver-enabled transfer for a single geometry between an L_1 and L_2 northern halo orbit is constructed. In this application, instead of using a perilune map, a hyperplane is defined at $x = 1-\mu$ and up to 2 crossings of the hyperplane are recorded.[23] This map introduces an additional degree of complexity compared to the planar case due to the increased dimensionality of the problem. In such a scenario, designing spatial transfers using traditional approaches may be cumbersome and challenging. However, utilizing motion primitives reduces the complexity of analysis by both summarizing and providing an effective means of exploring the solution space. Using the initial guess construction process, the original and morphed initial guesses are constructed and displayed in Figure 11a. Then, using the corrections procedure, a maneuver-enabled transfer is constructed from the morphed initial guess and displayed in Figure 11b. The transfer has an approximate transfer time of 31.1 days, incorporates a $|\Delta\bar{v}|$ of 90.6 m/s between the manifold primitives, starts at $C_J = 3.0681$, and terminates at $C_J = 3.0671$. As depicted in Figure 11b, the geometry of the discontinuous initial guess is retained.

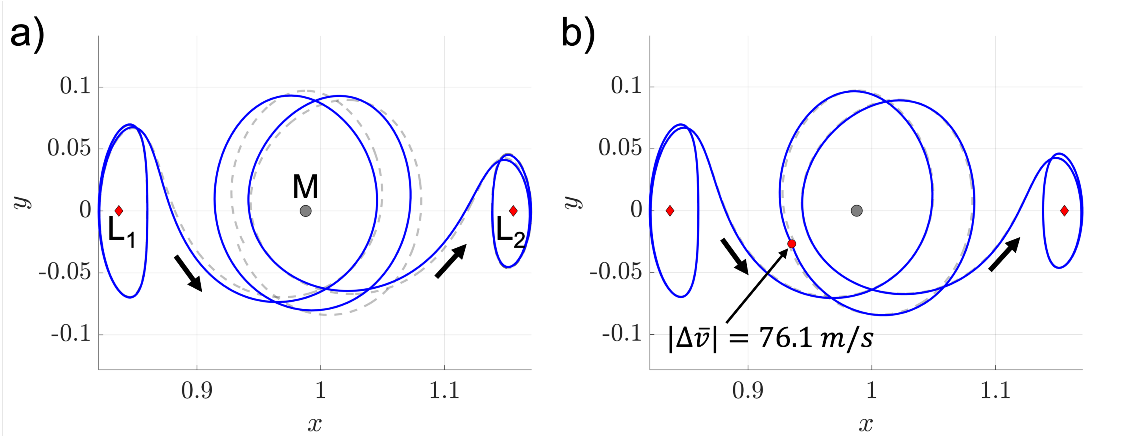

Figure 10. a) Natural and b) maneuver-enabled transfers between an L_1 and L_2 Lyapunov orbit with two revolutions around the Moon.

The transfer exhibits bounded motion in the vicinity of the initial orbit before departing as well as bounded motion in the vicinity of the target orbit after it arrives. This result suggests that a transfer may exist between bounded quasi-periodic orbits near each of the initial and final orbits. Finally, the constructed transfer demonstrates the success of a primitive-based transfer design procedure in developing a spatial transfer with a desired geometry in a multi-body system.

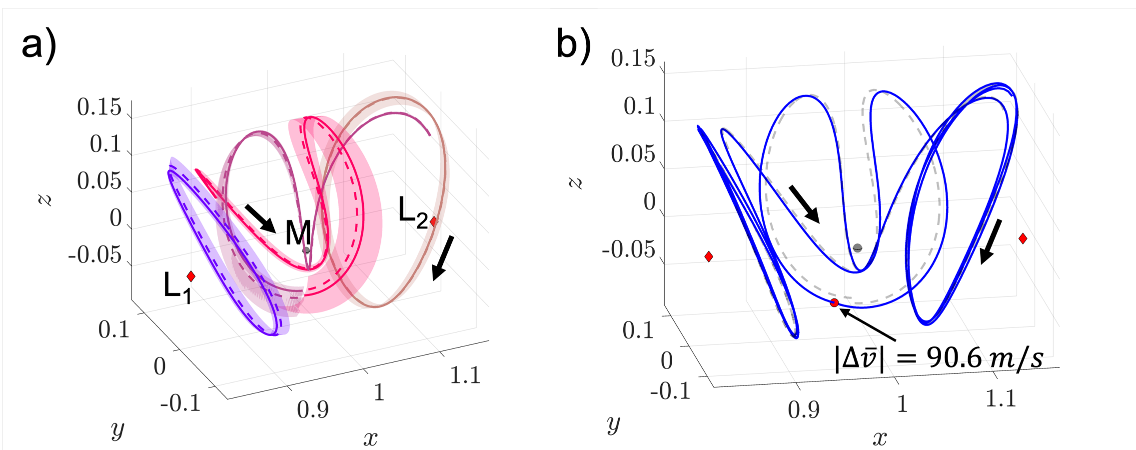

Figure 11. a) Morphed initial guess for an L_1 to L_2 northern halo orbit transfer with two close approaches of the Moon displayed with respect to the original initial guess of primitives and their corresponding regions of existence and b) the resulting maneuver-enabled transfer between an L_1 and L_2 northern halo orbit.

8 CONCLUSION

In this paper, motion primitives and their corresponding regions of existence are employed to rapidly explore the solution space of the Earth-Moon system and construct transfers between libration point orbits near the Moon. Using k-means clustering, an efficient description strategy is presented to effectively store the region of existence associated with a motion primitive via a set of representative trajectories and bounding trajectory parameters. An example of the description strategy is presented for both a periodic orbit and a general nonperiodic trajectory in the Earth-Moon CR3BP. Then, using sets of motion primitives and their regions of existence, a trajectory design procedure is presented to rapidly construct transfers between periodic orbits in a multi-body system. Initial and target orbits are first selected from desired periodic orbit families using sets of motion primitives constructed to summarize each family. Poincaré mapping is then leveraged to explore motion primitives within their regions of existence along the unstable manifold associated with the initial orbit and the stable manifold associated with the target orbit to construct an initial guess with a desired geometry. Finally, a direct collocation scheme and constrained optimization approach is used to compute a continuous end-to-end solution that geometrically resembles the discontinuous initial guess. In this investigation, both natural and maneuver-enabled transfers are constructed between an L_1 and L_2 Lyapunov orbit and a maneuver-enabled transfer is constructed between a L_1 and L_2 northern halo orbit in the Earth-Moon system. The transfers constructed in this paper demonstrate the utility of the presented primitive-based transfer design procedure and its effectiveness in leveraging a summary of the solution space to rapidly design transfers in a multi-body system in a manner that reduces the analytical burden on a human designer. Future work will focus on how to expand this primitive-based approach to construct more complex planar and spatial transfers that piece together multiple primitives between the initial and final orbit, optimize the placement and magnitude of impulsive maneuvers, incorporate additional constraints, and apply the process to other types of transfers in multi-body systems.

9 ACKNOWLEDGEMENTS

This work is being completed at the University of Colorado Boulder. It is funded by the NASA Space Technology Research Fellowship from the National Aeronautics and Space Administration. The authors thank Dr. Christopher Spreen and Dr. Robert Pritchett for their support and feedback on this work.

REFERENCES

[1] N. Bosanac, C. M. Webster, K. C. Howell, and D. C. Folta, "Trajectory Design for the Wide Field Infrared Survey Telescope Mission," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 9, 2019, pp. 1899–1911.

[2] G. L. Condon, C. C. Esty, C. F. Berry, S. P. Downs, C. Ocampo, B. Mahajan, and L. M. Burke, "Mission and Trajectory Design Considerations for a Human Lunar Mission Originating from a Near Rectilinear Halo Orbit," *30th AIAA/AAS Space Flight Mechanics Meeting*, Orlando, FL, Jan 2020.

[3] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*. Waltham, MA: Morgan Kaufmann Publishers, 3 ed., 2012.

[4] T. R. Smith and N. Bosanac, "Constructing a Set of Motion Primitives in the Circular Restricted Three-Body Problem via Clustering," *AAS/AIAA Astrodynamics Specialist Conference*, ME, Aug 2019.

[5] T. R. Smith and N. Bosanac, "Motion Primitives Summarizing Periodic Orbits and Natural Transport Mechanisms in the Earth-Moon System," *AAS/AIAA Astrodynamics Specialist Conference*, Aug 2020.

[6] A. Haapala, K. C. Howell, and D. C. Folta, "Incorporating the Evolution of Multi-Body Orbits Into the Trajectory Trade Space and Design Process," *Acta Astronautica*, Vol. 112, 2015, pp. 1–18.

[7] D. Guzzetti, N. Bosanac, A. Haapala, K. C. Howell, and D. C. Folta, "Rapid Trajectory Design in the Earth-Moon Ephemeris System via an Interactive Catalog of Periodic and Quasi-Periodic Orbits," *Acta Astronautica*, Vol. 126, 2016, pp. 439–455.

[8] M. Mirzargar, R. T. Whitaker, and R. M. Kirby, "Curve Boxplot: Generalization of Boxplot for Ensembles of Curves," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 20, No. 12, 2014, pp. 2654–2663.

[9] A. Majumdar and R. Tedrake, "Funnel Libraries for Real-Time Robust Feedback Motion Planning," *The International Journal of Robotics Research*, Vol. 36, No. 8, 2017, pp. 947–982.

[10] B. A. Conway, *Spacecraft Trajectory Optimization*. New York, USA: Cambridge University Press, 2010.

[11] N. Bosanac, *Leveraging Natural Dynamical Structures to Explore Multi-Body Systems*. PhD Dissertation, Purdue University, West Lafayette, IN, 2016.

[12] M. Ozimek, D. J. Grebow, and K. C. Howell, "A Collocation Approach for Computing Solar Sail Lunar Pole-Sitter Orbits," *The Open Aerospace Engineering Journal*, Vol. 3, 2010, pp. 65–75.

[13] D. J. Grebow and T. A. Pavlak, "MCOLL: MONTE Collocation Trajectory Design Tool," *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, WA, Aug 2017.

[14] R. E. Pritchett, *Strategies for Low-Thrust Transfer Design based on Direct Collocation Techniques*. PhD Dissertation, Purdue University, West Lafayette, IN, 2020.

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

[16] 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, USA: Springer, 2011.

[17] J. D. Hadjidemetriou, "Resonant Motion in the Restricted Three-Body Problem," *Celestial Mechanics and Dynamical Astronomy*, Vol. 56, 1993, pp. 201–219.

[18] H. B. Keller, "Numerical Solution of Bifurcation and Nonlinear Eigenvalue Problems," *Applications of Bifurcation Theory* (P. H. Rabinowitz, ed.), pp. 359–384, New York, USA: Academic Press, 1977.

[19] K. C. Howell, "Three-Dimensional, Periodic, 'Halo' Orbits," *Celestial Mechanics and Dynamical Astronomy*, Vol. 32, No. 1, 1984, pp. 53–71.

[20] M. W. Lo, "The InterPlanetary Superhighway and the Origins Program," *IEEE Aerospace Conference Proceedings*, Big Sky, MT, Mar 2002.

[21] G. Gómez, W. S. Koon, M. W. Lo, J. E. Marsden, J. Masdemont, and S. D. Ross, "Connecting Orbits and Invariant Manifolds in the Spatial Restricted Three-Body Problem," *Nonlinearity*, Vol. 17, 2004, pp. 1571–1606.

[22] L. Perko, *Differential Equations and Dynamical Systems*. New York, USA: Springer, 3 ed., 2001.

[23] A. Haapala, *Trajectory Design in the Spatial Circular Restricted Three-Body Problem Exploiting Higher-Dimensional Poincaré Maps*. PhD Dissertation, Purdue University, West Lafayette, IN, 2014.

[24] J. T. Betts, "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[25] F. Topputo and C. Zhang, "Survey of Direct Transcription for Low-Thrust Space Trajectory Optimization with Applications," *Abstract and Applied Analysis*, Vol. 2014, 2014.

[26] M. Kelly, "An Introduction to Trajectory Optimization: How to Do Your Own Direct Collocation," *SIAM Review*, Vol. 59, No. 4, 2017, pp. 849–904.

[27] P. Williams, "Hermite-Legendre-Gauss-Lobatto Direct Transcription in Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 4, 2009, pp. 1392–1395.

[28] R. M. Holladay and S. S. Srinivasa, "Distance Metrics and Algorithms for Task Space Path Optimization," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Daejeon, Korea, Oct 2016.

[29] C. De Boor, "Good Approximation by Splines with Variable Knots. II," *Conference on the Numerical Solution of Differential Equations*, Dundee, Scotland, 1973.

[30] R. D. Russell and J. Christiansen, "Adaptive Mesh Selection Strategies for Solving Boundary Value Problems," *SIAM Journal on Numerical Analysis*, Vol. 15, No. 1, 1978, pp. 59–80.

[31] E. Frazzoli, *Robust Hybrid Control for Autonomous Vehicle Motion Planning*. PhD dissertation, Massachusetts Institute of Technology, Cambridge, MA, 2001.

[32] O. C. Jenkins and M. J. Mataric, "Deriving Action and Behavior Primitives from Human Motion Data," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Switzerland, Oct 2002.

[33] B. Wang, J. Gong, R. Zhang, and H. Chen, "Learning to Segment and Represent Motion Primitives from Driving Data for Motion Planning Applications," *21st International Conference on Intelligent Transportation Systems*, Maui, HI, Nov 2018, pp. 1408–1414.

[34] A. L. N. Fred and A. K. Jain, "Combining Multiple Clusterings Using Evidence Accumulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 27, No. 6, 2005, pp. 835–850.

[35] D. Huang, J. H. Lai, and C. D. Wang, "Combining Multiple Clusterings via Crowd Agreement Estimation and Multi-Granularity Link Analysis," *Neurocomputing*, Vol. 170, 2015, pp. 240–250.