AAS 25-269

TREE-BASED INITIAL GUESS CONSTRUCTION FOR SPACECRAFT TRAJECTORIES IN MULTI-BODY SYSTEMS

Renee L. Spear* **and Natasha Bosanac†**

Inspired by robotic motion planning, this paper presents a hierarchical tree-based approach to automatically generating spacecraft trajectories in the Earth-Moon circular restricted three-body problem. First, a forest of trees is constructed to explore the configuration space. Then, sequences of trees are located to estimate connected tree sets that are traversed to create paths. The sequences are searched to identify candidate nodes and edges that may be used to form a single path between two boundary conditions. This approach is used to generate initial guesses for planar transfers between two periodic orbits.

INTRODUCTION

To support an expanded presence in cislunar space, agile spacecraft will need to maneuver and replan their paths in response to potential hazards. However, rapidly designing a complex trajectory between two boundary conditions in a chaotic multi-body gravitational environment can be a challenging, computationally intensive problem. Typically, initial guesses for the path and maneuvers are constructed and then corrected to potentially produce a continuous and feasible trajectory. This process often relies on a trajectory designer for both low-level analysis and high-level decision-making. Automating this process to support, for example, onboard decision-making will rely on developing a new approach to rapid trajectory design in cislunar space.

In the field of motion planning, tree-based planners are a popular sampling-based approach for automatically and rapidly constructing a path within an environment. Sampling-based planners sample the configuration space of an environment and then connect those samples to obtain solutions to a motion query.[1] An example of a traditional tree-based planner is a rapidly-exploring random tree (RRT). This approach consists of growing a tree, represented by a set of nodes and edges, from a desired initial state to a target state via random sampling of the configuration space. A path is recovered once the tree reaches the target, to within a specified tolerance.

Variations on the RRT algorithm have supported path planning in complex motion-planning problems. Bidirectional RRTs grow trees from each of the boundary conditions;[1–3] expansive-spaces trees rely on probabilistic sampling to ensure sufficient coverage in regions of low density;[1] and stable-sparse RRTs reduce computational time by retaining only a sparse set of nodes and edges.[4] Multipartite[5] and time-based RRTs[6] have been used for robot path planning in dynamic systems. Multipartite RRTs leverage tree branches from previous motion queries and bias in the sampling

*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.

distribution to support planning in complex environments whereas time-based RRTs track the time at each node, in addition to the state. Furthermore, multi-RRT algorithms employ multiple trees to recover paths and have been used in robotics to visit target locations while minimizing travel cost,[7] exploit local structure for exploring narrow passageways,[8] and utilize exploration and exploitation to rapidly find a feasible path.[9] Multi-RRT algorithms have also been used in uncrewed aerial vehicle path planning for application to indoor rescue scenarios.[10]

Recently, tree-based planners have been applied to spacecraft motion planning. For example, Danielson and Kloeppel used a tree-based search algorithm combined with an invariant-set motion planner for safe, rapid planning of spacecraft attitudes.[11] Tree-based planners have also been used in safe, relative spacecraft motion planning by Starek et. al.[12] and Deka and McMahon.[13] Together, these works have demonstrated the value of tree-based planners in spacecraft motion planning.

In this paper, we use a hierarchical tree-based planning approach to construct initial guesses for spacecraft trajectories in the Earth-Moon circular restricted three-body problem (CR3BP). This approach consists of two phases: 1) growing a forest of trees that captures dynamical flow throughout the environment and 2) using the forest to guide generation of an initial guess for a path connecting two boundary conditions. This approach is demonstrated by generating planar transfers in the Earth-Moon CR3BP between Lyapunov and distant prograde orbits at a single energy level.

The contribution of this paper is a proof-of-concept for an automated tree-based planning approach for rapid initial guess construction of spacecraft trajectories in a multi-body gravitational environment. By stepping forward or backward in time with randomized samples without solving boundary value problems, tree-based approaches support rapidly exploring a high-dimensional solution space. Furthermore, using a forest of trees supports constructing a hierarchical approach for efficient extraction of various geometrically-distinct initial guesses. Through this approach, initial guesses can be coarsely constructed and later refined to increase the likelihood of successfully recovering a continuous path with impulsive maneuvers between two boundary conditions. This capability may eventually be useful for reducing the burden on an operator or supporting onboard redesign of spacecraft trajectories.

DYNAMICAL MODEL

The circular restricted three-body problem approximates the dynamics governing a third body under the influence of two massive primary bodies. These bodies may be, for example, the Earth and Moon with masses M_1 and M_2, respectively. This model assumes that the primaries follow circular orbits about their barycenter, uninfluenced by the third body that possesses a comparatively negligible mass. In addition, all bodies are modeled as point masses with constant mass.[14]

To overcome challenges with differing orders of magnitude for time, mass, and length quantities in the CR3BP, these values are nondimensionalized. This normalization relies on the following characteristic quantities: t^* is defined such that the mean motion of the primaries is equal to unity, m^* is equal to the total system mass, and l^* is the distance between the primaries. In the CR3BP, these quantities are selected as constant values.

An orthogonal, rotating coordinate frame is defined with a rotation rate equal to the rotation of the primaries, which removes time dependence from the system. The origin of the frame is placed at the Earth-Moon barycenter and the axes are denoted as $\hat{x}, \hat{y}, \hat{z}$. Specifically, $\hat{x}$ is directed from the Earth to Moon, $\hat{z}$ is aligned with the primary system's orbital angular momentum vector, and $\hat{y}$ completes the right-handed triad.[14]

Using these definitions, the equations of motion describe the dynamical environment governing the spacecraft. The nondimensionalized state of a spacecraft in the rotating frame is defined as $\bar{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$. Then, the nondimensionalized equations of motion for a spacecraft are expressed in the rotating frame as

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

where the nondimensional distances of the spacecraft with respect to each of the primaries are equal to $r_1 = \sqrt{(x+\mu)^2 + y^2 + z^2}$ and $r_2 = \sqrt{(x-1+\mu)^2 + y^2 + z^2}$ and the mass ratio of the system is defined as $\mu = M_2/(M_1 + M_2)$.

In the rotating frame, an integral of motion, commonly referred to as the Jacobi constant, exists. This quantity is equal to

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

This constant is inversely proportional to a spacecraft's energy: an increase in energy is indicated by a decrease in the Jacobi constant.[15,16] This quantity also supplies insight into the accessible regions of the system, which are separated from forbidden regions by zero velocity surfaces, at a given energy level.

At a single value of the Jacobi constant, the CR3BP admits a variety of fundamental solutions including equilibrium points and periodic orbits.[16] There are five equilibrium points in the CR3BP that lie in the plane of the primaries: L_1, L_2, and L_3 are collinear equilibrium points while L_4 and L_5 are equilateral equilibrium points. Continuous families of periodic orbits also exist throughout the system, including families near or centered on the equilibrium points.[16]

MOTION PLANNING

Motion planning, or path planning, is a concept used to identify a collision-free path for an object moving in an environment. This concept has widely been used in the field of robotics, but has also been applied to a variety of other path planning problems including interactive virtual environment and game design,[17] camera movement,[18] and uncrewed aerial vehicles.[19,20]

A variety of approaches have been developed to solve path planning problems. These approaches include navigation planners such as potential functions; exact and approximate representations of the environment with cell decomposition; and sampling-based algorithms.[1] Many of these approaches were originally developed for path planning queries in linear dynamical systems. However, recent improvements have supported path planning in more complex environments and with uncertainty or dynamic and geometric constraints.[1,21] For example, previous researchers have adapted sampling-based planners to answer motion queries in kinodynamic environments where kinematic and dynamic constraints must be considered.[22–24]

Sampling-based planners employ sampling schemes to explore an environment, then connect those samples to obtain collision-free solutions to a motion query.[1] Two traditional categories of

approaches are single-query and multi-query methods. Single-query methods construct solutions to new path queries with no knowledge of previous solutions whereas multi-query methods leverage information from previous path queries to construct new solutions.[25] Popular single-query sampling-based planners include tree-based algorithms such as rapidly-exploring random trees while probabilistic roadmaps are a popular option for multi-query planners. These methods are also examples of probabilistically complete planners: the probability that the planner will fail to find a path (if one exists) approaches zero as additional samples are drawn.[1,26]

RRTs were originally developed to overcome challenges with other path planning techniques in high-dimensional environments for nonholonomic systems.[26] For example, potential fields rely on selecting a good heuristic potential function and probabilistic roadmaps require connecting pairs of nearby configurations for graph construction. These tasks can be challenging or computationally intensive when the environment is high dimensional or highly constrained. RRTs overcome these challenges by iteratively expanding toward randomly-selected points, therefore not requiring point-to-point connections, solving many boundary value problems, or picking a heuristic function.[26]

Path recovery using an RRT consists of growing a tree from a desired initial state via random sampling of the configuration space. A conceptual example of using an RRT to construct a path between two desired configurations, or states, is shown in Figure 1 where the initial and target configurations are denoted with red and blue circles, respectively, and obstacles are represented with dark grey shapes. First, a random sample is drawn within its configuration space, denoted by the gold circle in Figure 1a). To extend the tree, the nearest node in the tree to the random sample is located, the direction of the velocity vector is aligned with the direction to the random sample, and the state is propagated. This step adds a new branch and node to the tree, shown by a red line and purple circle in Figure 1a). The nearest node is often identified in the configuration space, but could also be found in the velocity or full phase space. The process is repeated to explore the environment, as portrayed in Figure 1b), until a termination condition, such as reaching a goal. By traversing the tree, a path is recovered, as displayed in Figure 1c) in purple.

GRAPH SEARCH TECHNIQUES

A graph is a collection of nodes, representing objects, and edges, representing the relationships between objects.[1,27] These structures may be characterized by their directionality and edge weight. If two nodes are mutually accessible from one another, the graph is undirected; otherwise, it is directed.[1,27] Furthermore, a graph is weighted if numerical values are associated with its edges to reflect the cost of traversing those edges whereas an unweighted graph has no cost associated with moving between nodes.[1]

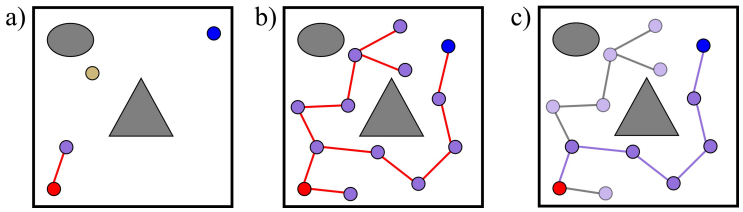

Figure 1. Conceptual RRT generation in a linear environment: a) randomly sample the environment and extend the tree rooted at the initial state towards the random sample; b) repeatedly sample the environment and extend the tree from the nearest node until c) a path is recovered.

A tree is an example of a directed graph and may be described using a parent-child analogy. Traditionally, the root of a tree is a special node that possesses no parents. A parent has nodes accessible to it called children, and a leaf is a node that has no children.[1] The number of edges that are incident into a node from its parents defines its in-degree while the number of edges that are incident from a node to its children defines its out-degree.[27] Characterizing nodes by in- and out-degrees provides insight into the traversability of a directed graph.

Graph search algorithms may be applied to a graph to recover a sequence of nodes connected by edges; in path-planning, this sequence may produce a path through the environment.[1] In a specified application, selection of a search algorithm depends on the desired search metrics, available computational effort, and types of solutions sought. Classical breadth-first searches guarantee that the first solution uses the smallest number of nodes and edges within the graph whereas depth-first searches find longer paths earlier in the search.[28] It may be desirable to optimize path length or other search metrics, including safety, time, number of nodes to explore, or computational efficiency. Heuristic-informed graph search algorithms, such as A* and Dijkstra, may return optimal paths with respect to the search metric. However, selection of a suitable heuristic can be a challenging task in some applications and a poorly-defined heuristic may result in longer search times and suboptimal paths.[1]

In this work, a breadth-first search is used to identify paths between start and target nodes within an unweighted, directed graph. In a traditional tree, a breadth-first search begins searching the graph at the root, then visits all children of that root. Next, the algorithm looks at the grandchildren, great-grandchildren, and so forth until it reaches the leaves of the graph.

TECHNICAL APPROACH

This section presents a hierarchical tree-based planning approach for constructing a path between two boundary conditions. Each step of the approach is demonstrated for a planar transfer between L_1 and L_2 Lyapunov orbits at a Jacobi constant of 3.1556 in the Earth-Moon CR3BP. Additionally, computational times are provided for each step using a computer with an AMD Ryzen 5 5600G with Radeon Graphics 4.20 GHz processor.

Phase 1: Grow a Forest of Trees

The first phase of our tree-based planning approach for trajectory generation involves building a forest of trees to explore the environment. The forest supports the construction of global paths that may revolve around the primaries before reaching a target state. As a result, the use of a forest of trees, as opposed to one or two RRTs, mitigates the risk of recovering short but maneuver-inefficient paths between two boundary conditions. Furthermore, this approach supports constructing complex paths without the use of predefined waypoints. Building the forest of trees consists of two steps: i) sampling the tree roots and ii) growing the trees with branches that satisfy the CR3BP dynamics.

Locating the tree roots relies on sampling position vectors using a uniform grid in the environment. Inspired by the Kinodynamic Motion Planning by Interior-Exterior Cell Exploration[29] planner, cell decomposition, which relies on partitioning the configuration space into cells,[30] is used to represent the environment with a square-cell grid with a specified cell height h_{cell}. Then, one sample is placed at the intersection of four cell vertices to uniformly distribute samples within the configuration space of the environment, excluding its boundary and the regions within the zero velocity curves. Samples are also placed along each periodic orbit at intervals of equal arclength, starting from each boundary condition, to promote dynamical flow departing and arriving onto the

orbits via trees. During state propagation in the rotating frame over the time interval $t \in [t_0, t_f]$, arclength is calculated as[31]

$$s = \int_{t_0}^{t_f} \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}\, dt \tag{3}$$

to capture the traversed distance along the trajectory.

Roots of the trees are located at states that span the full phase space. For each of the sampled position vectors that do not lie along a periodic orbit, m velocity vectors are selected. Uniformly distributed in $[0°, 360°)$, these unit vectors are multiplied by the speed associated with a desired value of the Jacobi constant. One velocity vector pointed in the direction of natural motion is associated with each sample along a periodic orbit to minimize abrupt departures or arrivals. Then, each unique combination of position and velocity vectors defines the state of a tree root.

To demonstrate this sampling approach, consider the construction of an initial guess for a planar transfer between a 12.24-day L_1 Lyapunov and a 14.79-day L_2 Lyapunov orbit at a Jacobi constant equal to 3.1556 in the Earth-Moon CR3BP. Figure 2 presents an environment decomposed into cells with $h_{cell} = 0.025$ nondimensional units. Then, $m = 8$ velocity vectors are defined at each position vector. In addition, eight roots are evenly distributed in arclength along each periodic orbit. These values of the governing parameters produce a total of 1,033 roots distributed throughout the environment. Sampling these roots takes on the order of 10^{-1} seconds on the specified computer. In Figure 2, the position of each root is shown with a gray node whereas the velocity vector directions are denoted with blue and red arrows. The initial state, denoted by an orange circle, is located along the L_1 Lyapunov orbit at $\bar{x}_{init} = [0.81737, -0.0028472, 0, -0.0027187, 0.19160, 0]^T$ nondimensional units and the target state, denoted by an orange star, is found along the L_2 Lyapunov orbit at

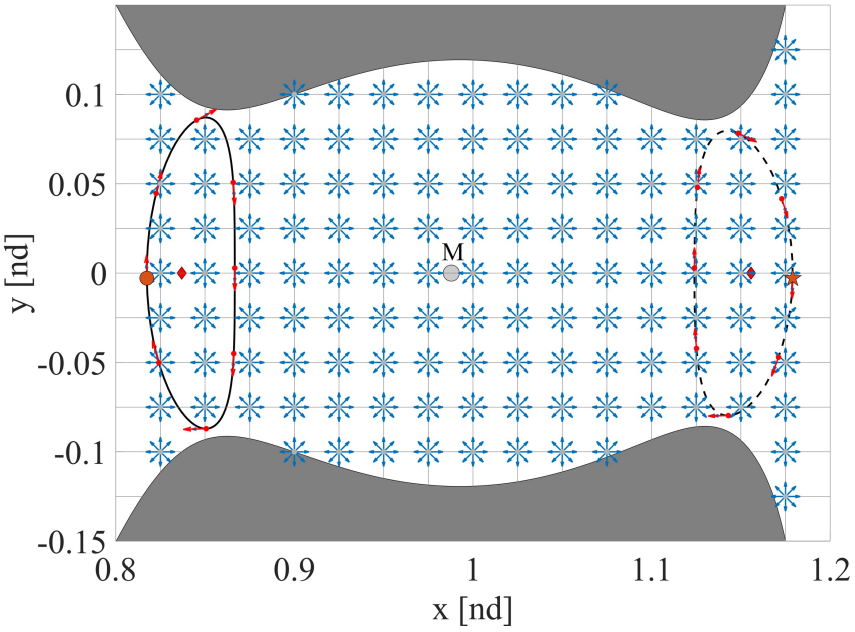

Figure 2. Tree roots for building a forest to support construction of a path between L_1 and L_2 Lyapunov orbits at $C_J = 3.1556$. Light gray circles denote the position samples while red and blue arrows denote the direction of the velocity vectors at each position.

$\bar{x}_{targ} = [1.17899, -0.0029054, 0, -0.0027421, -0.14227, 0]^T$ nondimensional units. The Moon appears as a gray circle scaled to possess its radius; red diamonds locate L_1 and L_2; and the gray regions denote the zero velocity curves at $C_J = 3.1556$.

Grown from a root, each tree randomly explores the environment while satisfying the dynamics of the CR3BP using a variation of the traditional RRT algorithm. In this paper, a tree is extended towards a random sample in configuration space by identifying an existing node within the tree. The velocity vector at this node is modified to produce a new state that is propagated to generate a small arc in the dynamical environment. This propagation produces an edge, or branch, in the tree that connects the selected node to a new node that possesses the state at the end of the arc.

Random sampling is used to select the existing node in the tree to expand. First, a random position vector is sampled in the environment. Then, the direction from each existing node in the tree to the random sample is calculated. The node with the smallest angle between its current velocity vector and this direction to the random sample is selected as a candidate for growth. If the angular difference associated with the selected node exceeds a velocity cone with angular radius $\gamma = \frac{360^\circ}{2m}$, then a new random sample is drawn until this criterion is satisfied. Then, the velocity vector of the selected node is updated: the speed is selected to produce the desired Jacobi constant whereas the velocity is set equal to the direction to the random sample. This growth approach enables random exploration of the environment while limiting any abrupt changes in velocity direction. These characteristics support identifying high-quality initial guesses for maneuver-efficient paths using the forest.

The selected node is used to generate an arc that produces a new edge and child node. This node is propagated forward in time in the dynamical environment for a specified arclength l. However, propagation terminates early if the branch impacts a primary or crosses the boundary of the environment. The branch length is selected to be sufficiently long to capture natural dynamics yet short enough to reasonably allow connections between branches.

Directional branches are initially added via forward propagation until the tree obtains a predefined number of nodes, n. To promote dynamical flow and exploration, additional branches are then added by propagating nodes backward in time that possess at least one out-degree. If the parent node of the new branch lies within a neighborhood, c_{neigh}, of the existing parents of the back propagated node, the branch is considered redundant and not added to the tree.

Roots may be removed from the environment during tree construction. For roots with position vectors that are sufficiently close to the Moon with a velocity vector in the direction of the Moon, the state may attempt to propagate through the Moon as a branch is grown. After n failed propagation attempts, the root is removed and no tree is grown.

To demonstrate the tree generation process, consider the same example used throughout this section. Figure 3 displays four trees in black, red, blue, and purple constructed with $n = 50$ nodes, $l = 2h_{cell}$ nondimensional units, and $c_{neigh} = 5 \times 10^{-3}$ nondimensional units. The roots of each tree are located at the centers of the bold circles and arrows denote the direction of motion. Finally, the initial and target orbits are shown with solid and dashed thick black lines, respectively. In the selected scenario, two roots are removed; therefore, the final forest consists of 1,031 trees. On the specified computer, constructing the forest for this scenario typically requires about three minutes. This approach to tree growth enables generation of a diverse set of trees that explore the full phase space yet limit abrupt changes in velocity direction within each tree.

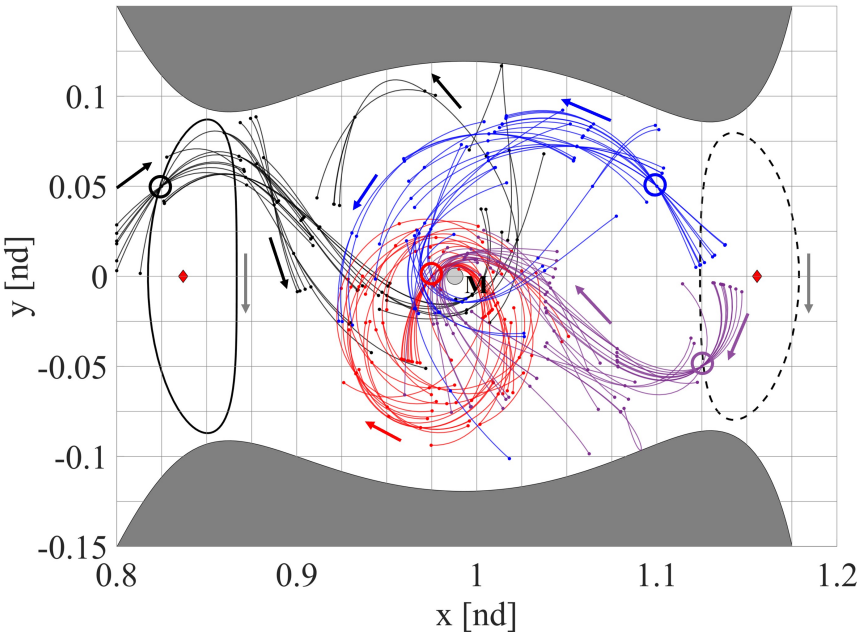

Figure 3. A subset of four trees from the forest, each depicted with a different color, that satisfy the CR3BP dynamics.

Phase 2: Forest-Guided Initial Guess Construction

In the second phase of our hierarchical tree-based planning approach to trajectory generation, a path between the boundary conditions is constructed using the forest. This phase focuses on exploiting the exploration conducted in the first phase to construct an initial guess by following three steps: i) locating connections between trees in the forest; ii) identifying connected tree sequences; and iii) searching the tree sequences for initial guesses that connect a set of boundary conditions. This approach is designed to reduce the computational burden of discovering paths of various geometries.

To support searching the forest for a path, connections between trees are identified. In this paper, pairs of nodes are connected if they lie within a predefined allowable discontinuity, c_{tol}, of one another in the configuration space. This parameter is selected to support identifying nodes that are sufficiently close in position space to promote a successful corrections process while simultaneously providing flexibility for recovering a path in the forest. Furthermore, the angular difference in the velocity directions at the connection points must be less than or equal to the angular radius of the velocity cone. In the selected example, $\gamma = 22.5^{\circ}$. This criterion is selected to minimize abrupt changes in velocity direction in the resulting paths. However, not all connected nodes are used to search for a path. Connected nodes are used only if the second node in the pair possesses at least one out-degree. Otherwise, the connection does not support the goal of traversing the forest. Locating the connections in the forest for the example used throughout this section typically requires about 17 minutes on the specified computer.

A single tree may be connected to multiple trees at one or more locations. Figure 4a) presents an example of two trees with nine connections, denoted with red nodes, where $c_{tol} = 5 \times 10^{-3}$ nondimensional units. The connections are magnified in Figure 4b) where the connecting branches are displayed with thicker lines than branches that do not connect the trees.

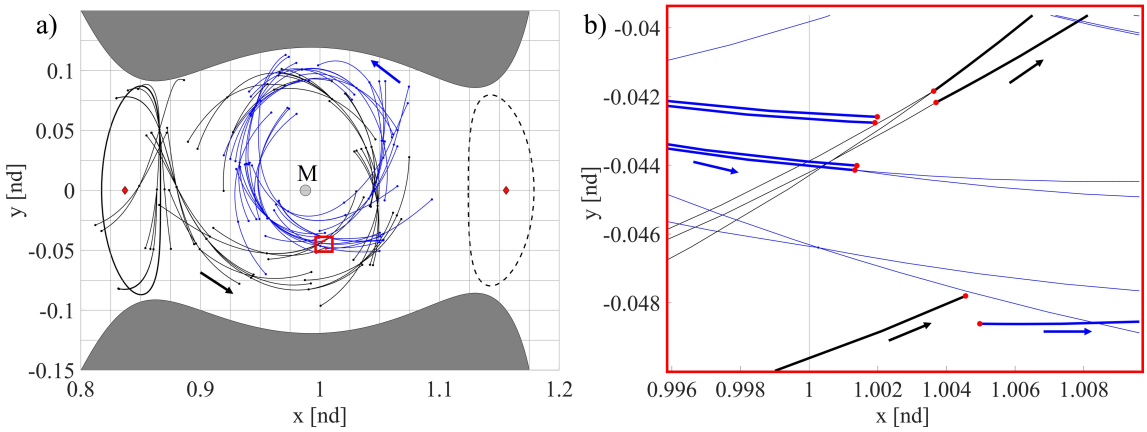

Figure 4. a) Two trees connected at the red nodes. b) The nine connections between the black and blue trees.

To reduce the computational burden of searching for paths through a graph of the connected trees, a hierarchical approach is employed. First, a tree sequence graph is constructed to generate a set of tree sequences. Then, the tree sequences are used to construct a subgraph that is searched to generate one or more paths.

The high-level tree sequence graph is an unweighted, directed graph. Specifically, the nodes represent trees and the edges represent connections between trees. A breadth-first search is applied to the graph via MATLAB's *allpaths* function, which returns the tree sequences in lexicographical order.[32] These sequences estimate connected tree sets that may be traversed to create paths between the boundary conditions. Figure 5 provides a conceptual depiction of a tree sequence graph where τ_i and τ_f are the trees rooted at the initial and target states, respectively. The trees along the initial and final orbits are connected, shown with solid arrows, and may have connections to other trees in the forest, shown with dashed arrows. For the selected example, the graph is searched for sequences up to 15 trees in length. On the specified computer, 124,905 sequences are returned in 0.25 seconds.

Prior to searching the sequences, they are ranked to promote faster recovery of geometrically diverse paths with a low expected cost. Paths connecting the selected boundary conditions that resemble known, low-cost geometries may leverage a partial or full revolution about the initial

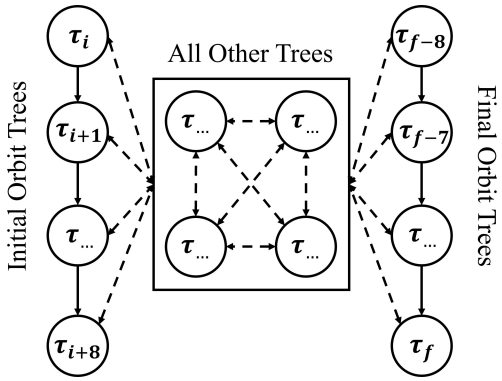

Figure 5. Conceptual depiction of a tree sequence graph.

or target orbits. Therefore, tree sequences that utilize combinations of trees along these orbits are ranked higher than trees that immediately depart for or arrive from the Moon's vicinity. For example, using the tree sequence graph in Figure 5, the sequence $[\tau_i, \tau_{i+1}, \tau_{i+2}, ..., \tau_{f-1}, \tau_f]$ would be ranked higher than the sequence $[\tau_i, ..., \tau_f]$. The higher the rank, the earlier the sequence will be searched for a path. Furthermore, varying the departure or arrival locations from the vicinity of the orbits supports a quicker recovery of geometrically-diverse paths. An example of a high-ranked sequence is shown in Figure 6 and is composed of six trees. Ranking the sequences typically requires about two seconds on the specified computer.

The final step of the tree-based planning approach is to recover an initial guess. For each tree sequence, beginning with the highest ranked, an unweighted, directed graph is constructed from the nodes and branches of each tree in the sequence. A breadth-first search returns paths between the boundary conditions in each subgraph, if at least one path exists. This path supplies an initial guess for a trajectory. Searching a single subgraph on the specified computer takes on the order of 10^{-2} seconds. Future work will focus on correcting this path along with impulsive maneuvers to generate continuous transfers.

Because tree sequences may return a large number of similar paths that vary by a low number of nodes, the paths are sorted by their estimated cost and diversity. The angular cost of each path, $\angle_{tot}$, is estimated by summing the angular difference between velocity vectors, $\bar{v}_{i,a}$ and $\bar{v}_{i,b}$, at locations of velocity discontinuity via

$$\angle_{tot} = \sum_{i=1}^{p} \frac{\bar{v}_{i,a} \cdot \bar{v}_{i,b}}{||\bar{v}_{i,a}||||\bar{v}_{i,b}||} \tag{4}$$

where p is the number of locations with velocity discontinuity, found at nodes along the path. Sim-

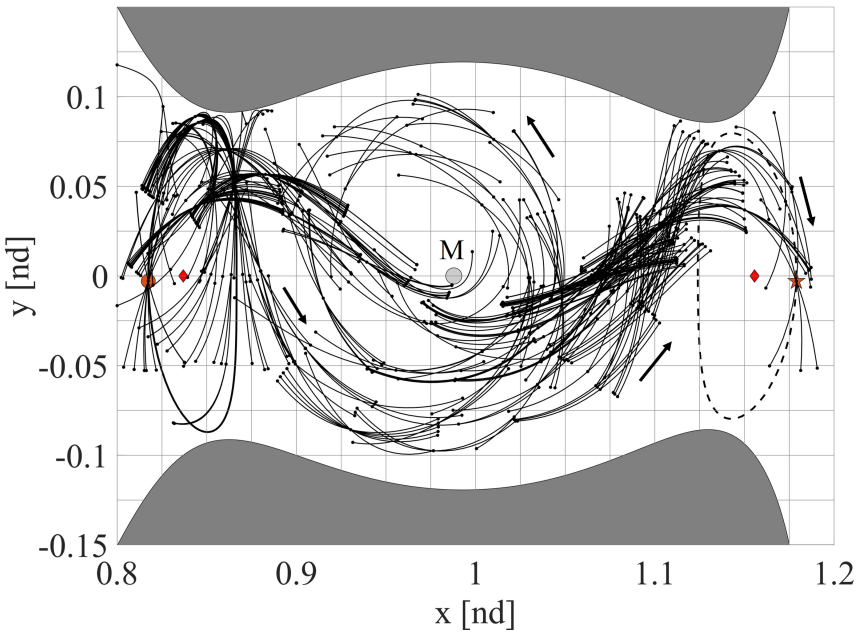

Figure 6. An example of a tree sequence, composed of six trees, that is returned from searching a tree sequence graph.

ilarity is assessed by calculating the Euclidean distance between pairs of nodes along equal length paths. If two paths have at least 50% of their nodes within a distance of $5c_{tol}$, they are considered to be similar. For all similar paths, the path with the lowest relative angular cost is included in the final set of coarse initial guesses along with the paths that are sufficiently different.

Three parameters are used to analyze the final set of coarse initial guesses: time of flight, cumulative velocity discontinuity, and total curvature. Time of flight is calculated by summing the elapsed time along each branch in the initial guess, given in days. The cumulative velocity discontinuity is estimated by summing the velocity discontinuity (Δv), in meters per second (m/s), at each node along the path. Total curvature is calculated to estimate the number of revolutions present along the initial guess without the need to perform a visual inspection. The total absolute curvature is mathematically calculated as

$$\kappa_{tot}(t_0, t_f) = \int_{t_0}^{t_f} \kappa(s)\, ds = \int_{t_0}^{t_f} \kappa(s) \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}\, dt \tag{5}$$

and reflects the cumulative angle traced out by the tangent to the path within the evolving osculating plane.[31] These quantities may vary substantially across the array of generated paths.

To demonstrate the process of generating an initial guess through the hierarchical graph, consider the goal in this paper of connecting boundary conditions along L_1 and L_2 Lyapunov orbits at $C_J = 3.1556$ in the Earth-Moon CR3BP. One initial guess is displayed in Figure 7. This path was recovered from the tree sequence depicted in Figure 6 and is characterized by a cumulative velocity discontinuity of 298.44 m/s and time of flight equal to 8.62 days. Locations of high velocity discontinuity, relative to the path, are marked with light red circles. The path, along with these notable locations, is analyzed in the next section.

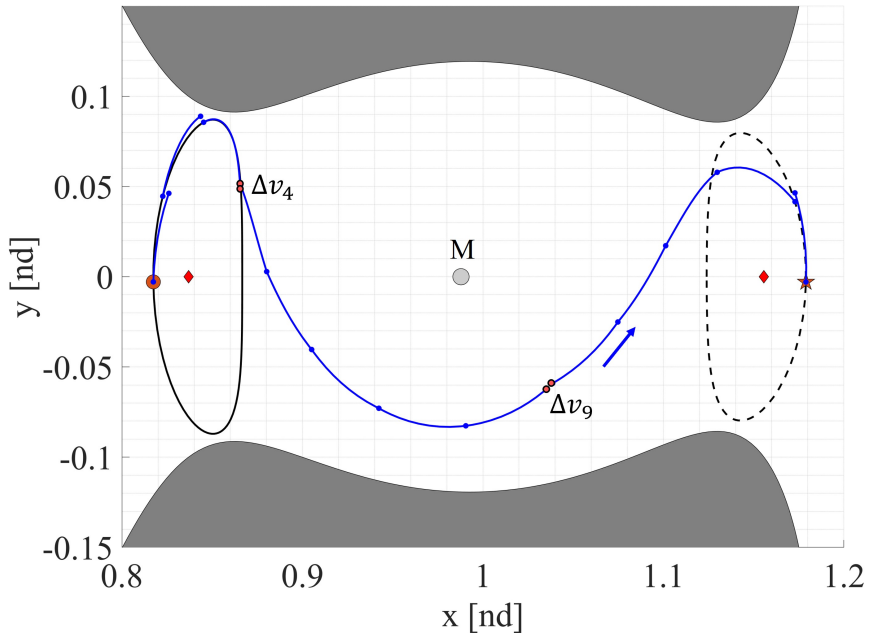

Figure 7. A coarse, zero-revolution initial guess connecting boundary conditions on L_1 and L_2 Lyapunov orbits at $C_J = 3.1556$ with a cumulative velocity discontinuity of 298.44 m/s and time of flight of 8.62 days.

RESULTS

Using this hierarchical tree-based planning approach, initial guesses for planar transfers are constructed between periodic orbits at a single energy level in the Earth-Moon CR3BP. This approach is demonstrated using multiple combinations of initial and final periodic orbits to capture flow in distinct directions through the environment. These results demonstrate the capability for this approach to not only automatically recover one useful initial guess, but also generate additional initial guesses that are geometrically distinct.

Initial Guesses for Transfers Between L_1 and L_2 Lyapunov Orbits at $C_J = 3.1556$

Three initial guesses are constructed between the 12.24-day L_1 and 14.79-day L_2 Lyapunov orbits used throughout the previous section to demonstrate the planning approach. Each initial guess is generated from a unique six-tree sequence with start and end segments that do not abruptly depart from or arrive at the Lyapunov orbits. Although the sequences only differ in the fifth tree, each guess also uses a distinct sequence of nodes and branches from the fourth tree. These differences result in initial guesses with distinct geometries in the vicinity of the Moon.

The first recovered initial guess is a zero-revolution path. The relative angles and velocity discontinuities are provided in Tables 1 and 2, respectively. Out of the 14 velocity discontinuities along this initial guess, the fourth and ninth maneuvers possess the largest magnitudes at 53.93 m/s and 67.61 m/s. A position discontinuity is present at these locations, indicating a change between trees used to construct the initial guess. However, position discontinuities do not guarantee a large discontinuity in velocity. For example, position discontinuities are also present at locations two, three, and thirteen along the path, which correspond to smaller discontinuities in velocity. The velocity discontinuity is a function of the speed, region sensitivity, and relative angle between the velocity vectors at that location. Although the fourth and ninth discontinuities are estimated to be the most expensive, the fourth and twelfth discontinuities possess the largest relative angles. In general, larger velocity discontinuities can be expected near the Moon and zero velocity curves where the instantaneous speed is higher. Despite the large cumulative velocity discontinuity of 298.44 m/s, the geometry and low relative velocity angles indicate a zero-cost transfer may be recovered once correction and optimization processes are applied. Furthermore, this transfer resembles a known heteroclinic connection at this energy level.

The second initial guess includes one large revolution about the Moon. This path, shown in Figure 8, has an estimated cost of 1,280.30 m/s and time of flight of 13.44 days. From the relative velocity

Table 1. Relative velocity angles in degrees for the initial guess shown in Figure 7

$\angle_1$	2.80	$\angle_2$	8.15	$\angle_3$	1.10	$\angle_4$	18.24	$\angle_5$	6.48	$\angle_6$	3.15
$\angle_7$	3.76	$\angle_8$	0.29	$\angle_9$	11.16	$\angle_{10}$	4.86	$\angle_{11}$	2.77	$\angle_{12}$	13.04
$\angle_{13}$	0.033	$\angle_{14}$	1.50	$\boldsymbol{\angle_{tot}}$**: 77.35°**							

Table 2. Velocity discontinuities in m/s for the initial guess shown in Figure 7

Δv_1	9.60	Δv_2	24.73	Δv_3	11.03	Δv_4	53.93	Δv_5	28.09	Δv_6	16.03
Δv_7	19.55	Δv_8	1.58	Δv_9	67.61	Δv_{10}	25.45	Δv_{11}	10.34	Δv_{12}	23.93
Δv_{13}	2.71	Δv_{14}	3.83	**Estimated Cost: 298.44 m/s**							

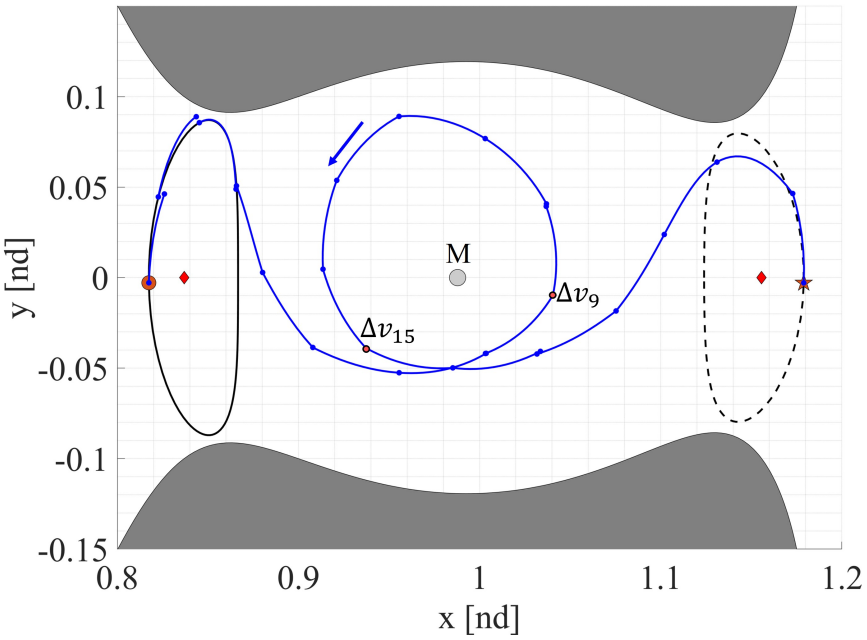

Figure 8. A coarse initial guess with one large Moon revolution connecting boundary conditions on L_1 and L_2 Lyapunov orbits at $C_J = 3.1556$ with an estimated cumulative velocity discontinuity of 1280.30 m/s and time of flight of 13.44 days.

angles provided in Table 3, the two highest changes in velocity direction occur at $\angle_{13}$ and $\angle_{16}$. However, from Table 4, the two largest velocity discontinuities are $\Delta v_9 = 173.08$ m/s and $\Delta v_{15} = 140.26$ m/s. Regardless of these large discontinuities in velocity, the small relative angles along this initial guess suggest that following corrections and optimization, the total impulsive maneuver cost may be much lower than the velocity discontinuities along the initial guess.

The third initial guess selected for analysis also performs a single revolution about the Moon, but with a distinctly different geometry. The initial guess is presented in Figure 9 whereas the relative

Table 3. Relative velocity angles in degrees for the initial guess shown in Figure 8

$\angle_1$	2.80	$\angle_2$	8.15	$\angle_3$	1.10	$\angle_4$	18.24	$\angle_5$	6.48	$\angle_6$	3.15
$\angle_7$	2.07	$\angle_8$	17.75	$\angle_9$	2.42	$\angle_{10}$	5.35	$\angle_{11}$	7.47	$\angle_{12}$	17.01
$\angle_{13}$	19.92	$\angle_{14}$	6.89	$\angle_{15}$	13.54	$\angle_{16}$	18.26	$\angle_{17}$	0.041	$\angle_{18}$	10.88
$\angle_{19}$	4.52	$\angle_{20}$	2.39	$\angle_{21}$	7.78	$\angle_{22}$	9.54	**$\angle_{tot}$: 218.05°**			

Table 4. Velocity discontinuities in m/s for the initial guess shown in Figure 8

Δv_1	9.60	Δv_2	24.73	Δv_3	11.03	Δv_4	53.93	Δv_5	45.65
Δv_6	109.75	Δv_7	0.72	Δv_8	102.69	Δv_9	173.08	Δv_{10}	66.55
Δv_{11}	0	Δv_{12}	93.40	Δv_{13}	46.79	Δv_{14}	84.76	Δv_{15}	140.26
Δv_{16}	117.47	Δv_{17}	81.27	Δv_{18}	72.08	Δv_{19}	0.49	Δv_{20}	0.62
Δv_{21}	41.55	Δv_{22}	3.83	**Estimated Cost: 1280.30 m/s**					

angles and velocity discontinuities are listed in Tables 5 and 6. Note that the high speed near the Moon results in a large estimated magnitude for the fifteenth velocity discontinuity at 132.70 m/s despite a low relative angle of 6.68°. Although the cumulative velocity discontinuity for this initial guess is lower than the other one-revolution path at 880.55 m/s, neither initial guess has been corrected or optimized. The low relative angles and geometry resembling a known heteroclinic connection suggest that this final initial guess may also reveal a zero or low-cost transfer after correction and optimization. This initial guess also possesses lower time of flight at 11.87 days, compared to the previous single-revolution path.

Initial Guesses for Transfers Between L_2 and L_1 Lyapunov Orbits at $C_J = 3.1458$

The forest is constructed to explore the environment and capture flow in a variety of directions. To demonstrate that other transfers may be constructed using a hierarchical tree-based approach, coarse initial guesses are generated to connect an initial state along a 14.89-day L_2 Lyapunov orbit to a target state along a 12.43-day L_1 Lyapunov orbit at $C_J = 3.1458$ in the Earth-Moon CR3BP. In this application, the initial state is located at $\bar{x}_{init} = [1.16384, -0.077597, 0, -0.070145, -0.11163, 0]^T$ nondimensional units while the target state is given as $\bar{x}_{targ} = \big[0.87125, -0.067637, 0, -0.014720, -0.16688, 0\big]^T$ nondimensional units. Initial guesses are successfully recovered using the tree-based planning approach with the same parameter selections as the previous scenario. Although the same values were used to generate the forest, this application has 1,191 trees due to the higher energy level and, therefore, L_1 and L_2 gateways with a wider opening where additional trees are rooted.

Similar to the L_1 to L_2 Lyapunov transfer scenario, one of the lowest-cost initial guesses mimics the geometry of a heteroclinic connection between the selected orbits at this energy level. This

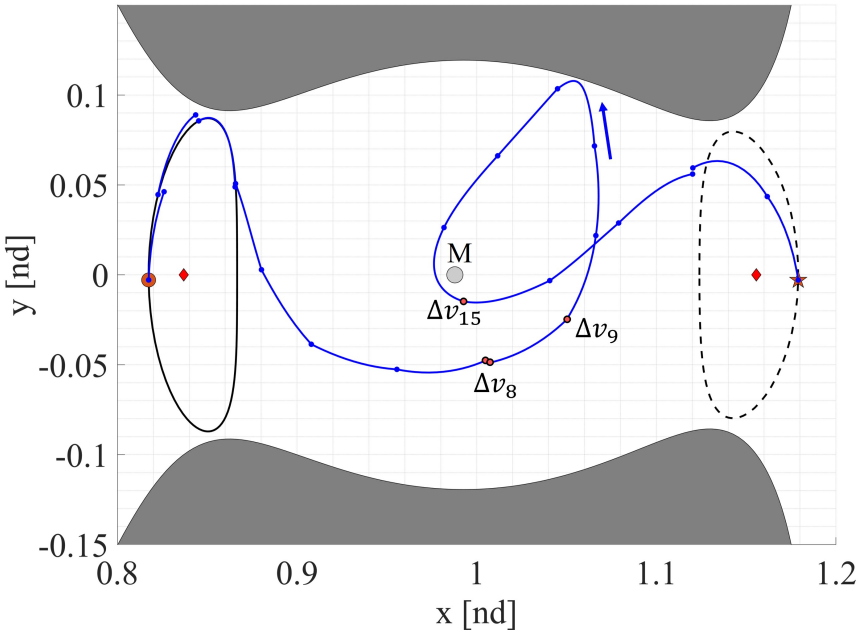

Figure 9. A coarse, one-revolution initial guess with a close approach to the Moon that connects boundary conditions on L_1 and L_2 Lyapunov orbits at $C_J = 3.1556$ with an estimated cumulative velocity discontinuity of 880.55 m/s and time of flight of 11.87 days.

Table 5. Relative velocity angles in degrees for the initial guess shown in Figure 9

$\angle_1$	2.80	$\angle_2$	8.15	$\angle_3$	1.10	$\angle_4$	18.24	$\angle_5$	10.54	$\angle_6$	20.63
$\angle_7$	7.72	$\angle_8$	11.43	$\angle_9$	13.97	$\angle_{10}$	0.56	$\angle_{11}$	5.18	$\angle_{12}$	8.99
$\angle_{13}$	1.16	$\angle_{14}$	0.10	$\angle_{15}$	6.68	$\angle_{16}$	9.08	$\angle_{17}$	3.02	$\angle_{18}$	18.80
$\angle_{19}$	5.48	$\angle_{20}$	9.54	$\boldsymbol{\angle_{tot}}$**: 77.35°**							

Table 6. Velocity discontinuities in m/s for the initial guess shown in Figure 9

Δv_1	9.60	Δv_2	24.73	Δv_3	11.03	Δv_4	53.93	Δv_5	45.66
Δv_6	109.75	Δv_7	60.61	Δv_8	106.99	Δv_9	102.74	Δv_{10}	3.33
Δv_{11}	19.44	Δv_{12}	18.09	Δv_{13}	7.87	Δv_{14}	1.48	Δv_{15}	132.70
Δv_{16}	83.20	Δv_{17}	14.65	Δv_{18}	39.20	Δv_{19}	11.33	Δv_{20}	24.22
Estimated Cost: 880.55 m/s									

coarse initial guess is plotted in Figure 10 with a time of flight of 6.95 days and a cumulative velocity discontinuity equivalent to 309.39 m/s. This initial guess, generated from a four-tree sequence, briefly follows the L_2 Lyapunov orbit before departing, traveling above the Moon, and approaching the boundary condition on the L_1 Lyapunov orbit. All relative angles and velocity discontinuities are provided in Tables 7 and 8. The two largest velocity discontinuities do not correspond to the largest relative angles between velocity vectors at these locations. The position discontinuity and proximity to the Moon, particularly when compared to the placement of Δv_2 for example, result in a reasonable location for a larger discontinuity in velocity. A unique component of the estimated

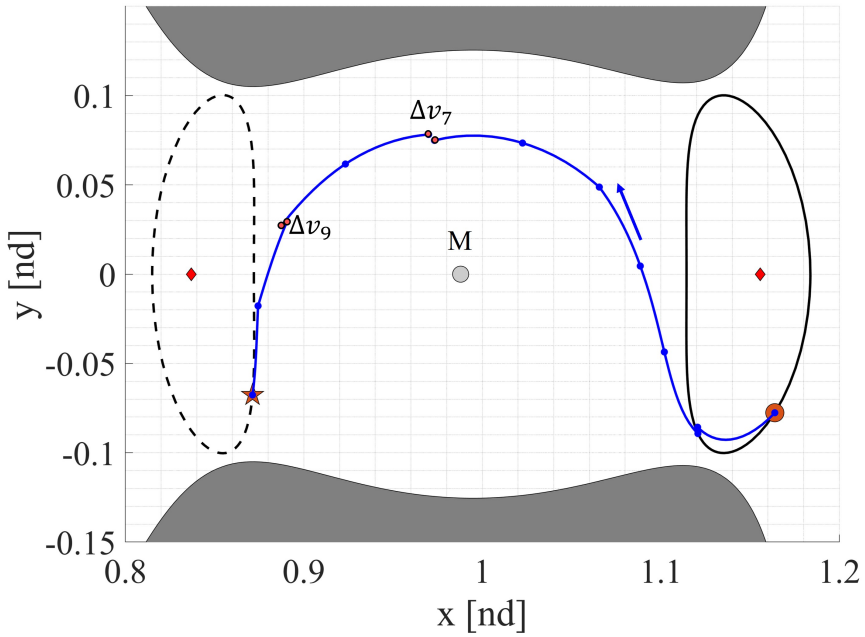

Figure 10. A coarse, zero-revolution initial guess connecting boundary conditions on L_2 and L_1 Lyapunov orbits at $C_J = 3.1458$ with a cumulative velocity discontinuity of 309.39 m/s and time of flight of 6.95 days.

Table 7. Relative velocity angles in degrees for the initial guess shown in Figure 10

$\angle_1$	10.78	$\angle_2$	2.16	$\angle_3$	0.021	$\angle_4$	0.60	$\angle_5$	11.29	$\angle_6$	0.80
$\angle_7$	9.64	$\angle_8$	0	$\angle_9$	15.01	$\angle_{10}$	14.38	$\angle_{11}$	2.92	$\angle_{tot}$: 67.61°	

Table 8. Velocity discontinuities in m/s for the initial guess shown in Figure 10

Δv_1	25.42	Δv_2	9.65	Δv_3	0.078	Δv_4	2.92	Δv_5	59.90
Δv_6	4.74	Δv_7	63.12	Δv_8	0	Δv_9	72.60	Δv_{10}	62.20
Δv_{11}	8.75	**Estimated Cost: 309.39 m/s**							

cost of this transfer, compared to others provided in this scenario, is that $\Delta v_8 = 0$ m/s and $\angle_8 = 0°$. This is a result of the natural sets of branches that are built when backwards edges are added to the trees during the first phase of the technical approach. As in the previous scenario, the geometry and relative angles along this initial guess support the expectation that the cumulative velocity discontinuity may significantly reduce once the initial guess is corrected and optimized.

A single-revolution initial guess between the selected boundary conditions is also recovered from a four-tree sequence. The geometry of this transfer, provided in Figure 11, mirrors that of the transfer provided in Figure 8 despite a difference in Jacobi constant. This coarse initial guess possesses the highest estimated cumulative velocity discontinuity out of all the recovered paths at 989.49 m/s with an estimated time of flight of 15.89 days. Tables 9 and 10 list the individual angle and velocity discontinuity values. The three largest discontinuities $(\Delta v_{16}, \Delta v_9, \Delta v_{15})$, all occur along the revolution about the Moon, which is consistent with higher speeds closer to the Moon. Once again,

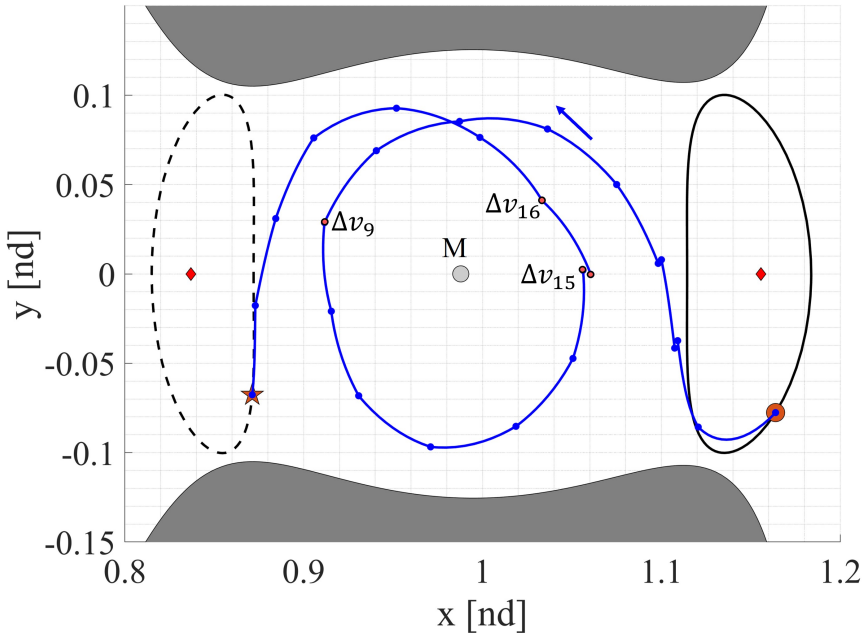

Figure 11. A coarse, one-revolution initial guess connecting boundary conditions on L_2 and L_1 Lyapunov orbits at $C_J = 3.1458$ with a cumulative velocity discontinuity of 989.49 m/s and time of flight of 15.89 days.

Table 9. Relative velocity angles in degrees for the initial guess shown in Figure 11

$\angle_1$	2.80	$\angle_2$	8.15	$\angle_3$	1.10	$\angle_4$	18.24	$\angle_5$	10.54	$\angle_6$	20.63
$\angle_7$	7.72	$\angle_8$	11.43	$\angle_9$	13.97	$\angle_{10}$	0.56	$\angle_{11}$	5.18	$\angle_{12}$	8.99
$\angle_{13}$	1.16	$\angle_{14}$	0.10	$\angle_{15}$	6.68	$\angle_{16}$	9.08	$\angle_{17}$	3.02	$\angle_{18}$	18.80
$\angle_{19}$	5.48	$\angle_{20}$	9.54	$\boldsymbol{\angle_{tot}}$**: 193.76°**							

Table 10. Velocity discontinuities in m/s for the initial guess shown in Figure 11

Δv_1	25.42	Δv_2	25.48	Δv_3	1.42	Δv_4	42.91	Δv_5	19.28
Δv_6	21.08	Δv_7	45.60	Δv_8	38.00	Δv_9	112.33	Δv_{10}	56.17
Δv_{11}	57.10	Δv_{12}	47.26	Δv_{13}	26.88	Δv_{14}	59.44	Δv_{15}	112.10
Δv_{16}	138.53	Δv_{17}	10.15	Δv_{18}	18.46	Δv_{19}	71.25	Δv_{20}	10.92
Δv_{21}	10.92	Δv_{22}	46.35	Δv_{23}	3.38	**Estimated Cost: 989.49 m/s**			

these locations do not correspond to the largest relative angles, which are located at $\angle_6$, $\angle_{18}$, and $\angle_4$.

Initial Guess for a Transfer Between L_1 Lyapunov and Distant Prograde Orbits at $C_J = 3.1683$

To further demonstrate the capabilities of the presented tree-based approach, consider paths reaching a distant prograde orbit (DPO). Consider a 12.01-day L_1 Lyapunov orbit and a 12.61-day DPO at $C_J = 3.1638$ in the Earth-Moon CR3BP with $\bar{x}_{init} = \big[0.8308, 0.052263, 0, 0.043224, 0.093865, 0\big]^T$ nondimensional units and $\bar{x}_{targ} = [1.13068, 0.010007, 0, -0.0033297, 0.092555, 0]^T$ nondimensional units. Initial guesses are successfully recovered using the hierarchical tree-based planning approach with the same parameter selections for h_{cell}, m, c_{tol}, c_{neigh}, n, and l as the other scenarios. However, the lower energy level in this scenario results in L_1 and L_2 gateways that are not as wide as the previous scenarios; therefore, only 838 trees are present in this forest.

An example of an initial guesses that connects $\bar{x}_{init}$ to $\bar{x}_{targ}$ along the selected orbits performs one revolution about the Moon before following the DPO to the target state. This path is displayed in Figure 12 and consists of four trees. The time of flight is 12.84 days, the cumulative velocity discontinuity is 601.31 m/s, and the values for the relative velocity angles and velocity discontinuities are given in Tables 11 and 12, respectively. Consistent with observations from other recovered initial guesses, the largest velocity discontinuities occur at the locations closest to the Moon yet maintain small relative velocity angles: $\Delta v_{11} = 131.53$ m/s where $\angle_{11} = 8.60°$ and $\Delta v_{12} = 107.52$ m/s where $\angle_{12} = 7.86°$. Similar to the initial guess for the zero-revolution L_2 to L_1 Lyapunov orbit transfer, this path leverages a natural edge in the forest at the location of the eighth velocity discontinuity. Despite possessing two velocity discontinuities that are significantly larger than others along this initial guess, the relative angles suggest that a lower estimated cost may be recovered after correcting and optimizing the path.

CONCLUSION

A hierarchical tree-based planning approach is presented to rapidly construct initial guesses for spacecraft trajectories in a multi-body gravitational system. The method consists of two phases:

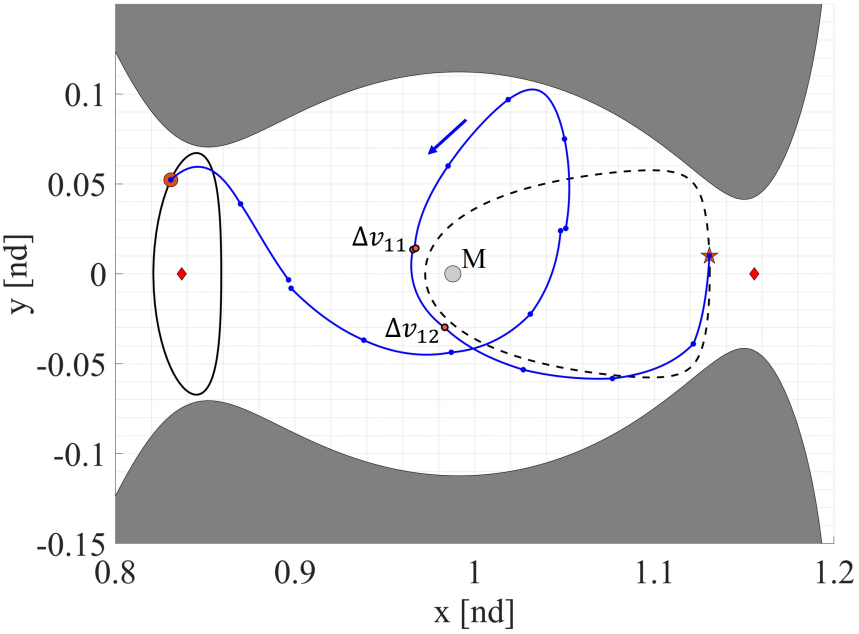

Figure 12. A coarse, one-revolution initial guess connecting boundary conditions between L_1 Lyapunov and distant prograde orbits at $C_J = 3.1683$ with a cumulative velocity discontinuity of 601.31 m/s and time of flight of 12.84 days.

(1) growing a forest of trees to explore the environment, and (2) exploiting the exploration conducted by the forest to recover a path connecting the boundary conditions. Through application to three scenarios, this approach is demonstrated to automatically recover geometrically-distinct initial guesses between sets of boundary conditions along Lyapunov and distant prograde orbits in the Earth-Moon CR3BP. The estimated costs of the recovered paths are higher than the expected costs for known transfers due to the coarse construction of the initial guesses. However, corrections and optimization are expected produce a continuous trajectory with a lower cumulative magnitude of impulsive maneuvers.

Table 11. Relative velocity angles in degrees for the initial guess shown in Figure 12

$\angle_1$	21.92	$\angle_2$	2.83	$\angle_3$	8.52	$\angle_4$	1.42	$\angle_5$	7.17	$\angle_6$	6.19
$\angle_7$	5.55	$\angle_8$	0	$\angle_9$	0.64	$\angle_{10}$	0.87	$\angle_{11}$	8.60	$\angle_{12}$	7.86
$\angle_{13}$	2.87	$\angle_{14}$	3.87	$\angle_{15}$	20.97	$\angle_{16}$	5.19	**$\angle_{tot}$: 104.47°**			

Table 12. Velocity discontinuities in m/s for the initial guess shown in Figure 12

Δv_1	40.27	Δv_2	7.69	Δv_3	42.32	Δv_4	10.88	Δv_5	74.74
Δv_6	59.44	Δv_7	45.17	Δv_8	0	Δv_9	1.84	Δv_{10}	6.72
Δv_{11}	131.53	Δv_{12}	107.52	Δv_{13}	20.20	Δv_{14}	12.96	Δv_{15}	31.42
Δv_{16}	8.60	**Estimated Cost: 601.31 m/s**							

ACKNOWLEDGMENT

This paper was supported in part by a fellowship award under contract FA9550-21-F-0003 through the National Defense Science and Engineering Graduate (NDSEG) Fellowship Program, sponsored by the Air Force Research Laboratory (AFRL), the Office of Naval Research (ONR) and the Army Research Office (ARO).

REFERENCES

[1] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Cambridge, MA: The MIT Press, 2005.

[2] *Algorithmic and Computational Robotics*, CRC Press, 2001.

[3] H. Liu, X. Zhang, J. Wen, R. Wang, and X. Chen, "Goal-Biased Bidirectional RRT Based on Curve-Smoothing," *International Federation of Automatic Control*, Vol. 52, No. 24, 2019, pp. 255–260, 10.1016/j.ifacol.2019.12.417.

[4] Y. Li, Z. Littlefield, and K. Bekris, *Algorithmic Foundations of Robotics XI*, ch. 16, pp. 263–282. Springer, 2015.

[5] M. Zucker, J. Kuffner, and M. Branicky, "Multipartite RRTs for Rapid Replanning in Dynamic Environments," *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy, 2007.

[6] A. Sintov and A. Shapiro, "Time-Based RRT Algorithm for Rendezvous Planning of Two Dynamic Systems," *2014 IEEE International Conference on Robotics and Automation*, Hong Kong, China, 2014.

[7] J. Janoš, V. Vonásek, and R. Pěnička, "Multi-Goal Path Planning Using Multiple Random Trees," *IEEE Robotics and Automation Letters*, Vol. 6, No. 2, 2021, pp. 4201–4208.

[8] T. Lai and F. Ramos, "Adaptively Exploits Local Structure With Generalised Multi-Trees Motion Planning," *IEEE Robotics and Automation Letters*, Vol. 7, No. 2, 2022, pp. 1111–1117, 10.1109/LRA.2021.3132985.

[9] Z. Sun, J. Wang, and M. Q.-H. Meng, "Multi-Tree Guided Efficient Robot Motion Planning," *Procedia Computer Science*, Vol. 209, 2022, pp. 31–39, 10.1016/j.procs.2022.10.096.

[10] J. Chen, H. Li, and Q. Zhou, "RRT-Forest: An Efficient Multi-UAV Cooperative Path Planning Method for Indoor Rescue," *2023 8th International Conference on Advanced Robotics and Mechatronics*, Sanya, China, 2023.

[11] C. Danielson and J. Kloeppel, "Rapid Construction of Safe Search-Trees for Spacecraft Attitude Planning," *2023 American Control Conference*, San Diego, CA, 2023.

[12] J. A. Starek, E. Schmerling, G. D. Maher, B. W. Barbee, and M. Pavone, "Fast, Safe, Propellant-Efficient Spacecraft Motion Planning Under Clohessy–Wiltshire–Hill Dynamics," *Journal of Guidance, Control and Dynamics*, Vol. 40, No. 2, 2017, 10.2514/1.G001913.

[13] T. Deka and J. McMahon, "Efficient Astrodynamics-Informed Kinodynamic Motion Planning for Relative Spacecraft Motion," *Advances in Space Research*, Vol. 73, No. 11, 2024, pp. 5426–5445, 10.1016/j.asr.2024.02.019.

[14] V. Szebehely, *Theory of Orbits: The Restricted Problem of Three Bodies*. Academic Press, 1967, 10.1016/B978-0-12-395732-0.X5001-6.

[15] G. S. Krishnaswami and H. Senapati, "An Introduction to the Classical Three-Body Problem: From Periodic Solutions to Instabilities and Chaos," *Resonance*, Vol. 24, No. 1, 2019, pp. 87–114.

[16] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross, *Dynamical Systems, the Three Body Problem and Space Mission Design*. Marsden Books, 2006.

[17] M. Overmars, I. Karamouzas, and R. Geraerts, *Algorithms - ESA 2008 Lecture Notes in Computer Science*, Vol. 5193, pp. 1–12. Springer, 2008.

[18] D. Nieuwenhuisen and M. H. Overmars, "Motion planning for camera movements," *IEEE International Conference on Robotics and Automation*, New Orleans, LA, April 2004.

[19] S. Aggarwal and N. Kumar, "Path Planning Techniques for Unmanned Aerial Vehicles: A Review, Solutions, and Challenges," *Journal of Computer Communications*, Vol. 149, 2020, pp. 270–299, 10.1016/j.comcom.2019.10.014.

[20] S. A. Bortoff, "Path Planning for UAVs," *Proceedings of the American Control Conference*, Chicago, IL, June 2000.

[21] S. M. LaValle and J. J. Kuffner Jr., "Randomized Kinodynamic Planning," *Proceedings of the 1999 IEEE International Confernece on Robotics and Automation*, Detroit, MI, 1999.

[22] K. L. Bruchko and N. Bosanac, "Rapid Trajectory Design in Multi-Body System using Sampling-Based Kinodynamic Planning," *AAS/AIAA Astrodynamics Specialist Conference*, Big Sky, MT, August 2023.

[23] E. Schmerling and M. Pavone, *Encyclopedia of Robotics*, pp. 1–9. Springer, 2019.

[24] R. Kindel, D. Hsu, J.-C. Latombe, and S. Rock, "Kinodynamic Motion Planning Amidst Moving Obstacles," *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, San Francisco, CA, April 2000.

[25] T.-Y. Li and Y.-C. Shie, "An Incremental Learning Approach to Motion Planning with Roadmap Management," *In Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, D.C., May 2002.

[26] S. M. LaValle, "Rapidly-exploring random trees: a new tool for path planning," *The annual research report*, 1998.

[27] R. P. Grimaldi, *Discrete and Combinatorial Mathematics: An Applied Introduction*, ch. 11. Addison-Wesley Publishing Company, 3rd ed., 1994.

[28] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, 2006.

[29] I. A. Şucan and L. E. Kavraki, "A Sampling-Based Tree Planner for Systems With Complex Dynamics," *IEEE Transactions on Robotics*, Vol. 28, No. 1, 2011, pp. 116–131, 10.1109/TRO.2011.2160466.

[30] A. Abbadi, R. Matousek, P. Osmera, and L. Knispel, "Spatial Guidance to RRT Planner Using Cell-Decomposition Algorithm," *20th International Conference on Soft Computing*, Brno, Czech Republic, 2014.

[31] T. Shifrin, *Differential Geometry: A First Course in Curves and Surfaces*, ch. 1. University of Georgia, 2023.

[32] MathWorks, "MATLAB allpaths," `https://www.mathworks.com/help/matlab/ref/graph.allpaths.html`, 2021. Last accessed 26 November 2024.