AAS 19-822

CONTINGENCY PLANNING IN COMPLEX DYNAMICAL ENVIRONMENTS VIA HEURISTICALLY ACCELERATED REINFORCEMENT LEARNING

Ashwati Das-Stuart[*], Kathleen Howell[†]

Unexpected events can cause a spacecraft to significantly deviate from its nominal path, leading to undesirable impacts on the mission. In such scenarios, the capability for rapid trajectory re-design is key for mission success. This investigation leverages a reinforcement learning strategy to automate the search for a transfer route to restore the overall mission goals after a spacecraft experiences a deviance in its thrusting capabilities during nominal operations. The route is computed by exploiting natural dynamical flows and accommodating spacecraft propulsive capabilities to construct an initial guess that is then transitioned to a continuous solution via traditional optimization techniques.

INTRODUCTION

Increasing mission complexities demand effective trajectory design strategies that balance diverse objectives and constraints to ensure mission success. However, the design and re-design of trajectories in dynamically complex environments is challenging and can easily become intractable via solely manual design efforts. Thus, the problem is recast to blend machine learning techniques with traditional astrodynamics approaches to develop a rapid and flexible trajectory design framework. In particular, this investigation focuses on planning for spacecraft contingency scenarios with applications relevant to near term cislunar exploration.[1] System failure and/or degradation during flight can cause the spacecraft to significantly deviate from a nominal transfer path, for example, leading to undesirable impacts on the mission. In such scenarios, the capability for rapid trajectory re-design is key for mission success.

The construction of a rapid design framework in the cislunar region commences with a fundamental understanding of the natural dynamics in the Earth-Moon system and the option to leverage existing structures to enable mission scenarios. Folta et al.[2] offer an interactive catalog of orbit families for applications in multi-body regimes; the advantages of characterizing trajectory parameters to identify potential parking and transfer options are apparent.[3] The utility of natural arcs extends to lowering propellant costs even in efficient low-thrust regimes.[4] These hybrid low-thrust/low-energy trajectories are constructed by assessing the intersections of natural arcs with thrusting ones at a selected hyperplane crossing.[5] Additional investigations demonstrate the ability to link various natural arcs,[6] exploit orbit manifolds[7] and assess their signatures on reduced-dimension Poincaré maps[8] to generate viable transfers in the Circular Restricted Three Body Problem (CR3BP); such a first step frequently renders a suitable initial guess for higher-fidelity analysis in an ephemeris model.

The access to a wide range of natural arcs, to be coupled with powered arcs resulting from available thrust capabilities, results in the examination of an infinitely large trade space to satisfy mission constraints. Conclusions from Radice and Olmo,[9] Ceriotti and Vasile,[10] Stuart and Howell,[11] as well as Furfaro and Linares [12] demonstrate the potential of heuristic methods to be effective in alleviating this challenge in various dynamical regimes and in uncovering local optima that may have otherwise remained unknown. Approaches

[*]Ph.D. Candidate, Purdue University, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47906, das15@purdue.edu; Current: Europa Clipper Mission Operations Systems Engineer, NASA JPL California Institute of Technology, 4800 Oak Grove Dr, Pasadena, CA, 91109, ashwati.das@jpl.nasa.gov

[†]Hsu Lo Distinguished Professor of Aeronautics and Astronautics, Purdue University, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47906, (765) 494-5786, howell@purdue.edu.

employing genetic algorithms[13,14] and Artificial Neural Networks (ANNs)[15–17] are also proven as beneficial. Tsirogiannis[18] as well as Trumbauer and Villac[19] generate impulsive transfer options by constructing a framework of pre-computed natural arcs and use graph search methods to evaluate the links. The nodes then serve as waypoints in a complex dynamical regime. Simplifying assumptions in a two body model are exploited by Parrish[20] to employ heuristics in solving for low-thrust initial guesses, and Miller and Linares[21] leverage Reinforcement Learning (RL) techniques to determine control laws to transfer a low-thrust spacecraft in a multi-body regime.

The current investigation applies the framework developed in Das-Stuart et al.[16,22] that blends machine learning strategies (specifically reinforcement learning) with traditional trajectory design techniques to construct initial guesses via an automated approach for s/c recovery options, and then transitions them to continuous end-to-end solutions. The foundation is developed for a process that alleviates the challenges associated with thrust law construction, the sequencing of thrust and coast arcs, and the limited solution options resulting from narrow basins of convergence when implementing traditional numerical techniques. The broad search capability is realized via the stochastic and parallellizable nature of the RL paradigm, carried out by software 'agents'. In this analysis, reference to an 'agent' is an abstraction for the software representation of a decision-making process in lieu of a human operator.

The trajectory design framework is summarized via four individual components in Fig. 1. The first component determines the *reach* of the s/c in a multi-dimensional and infinitely large configuration space via Accessible Regions (ARs). The second component exists to render natural conditions for the s/c to evaluate within these ARs, and the third component deploys pathfinding agents to appropriately select and sequence advantageous arcs from the ARs to construct end-to-end pathways in an automated manner. The discontinuous arcs are then passed through a traditional numerical corrections process via the final component to produce a continuous solution for a specified engine model.

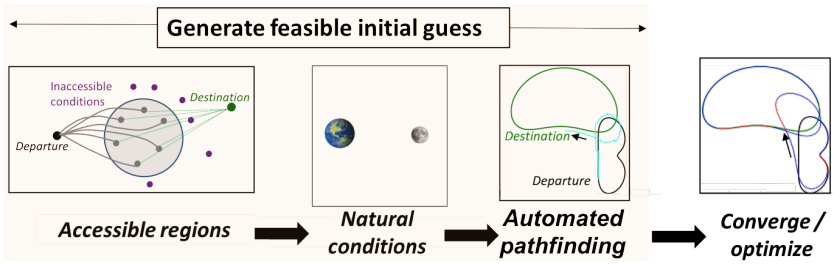


Figure 1: Overview of design framework

The investigation is organized consistent with the design framework. The theory associated with the dynamical and force models that support the generation of the accessible regions (framework component 1) and natural motion conditions (framework component 2) are initially introduced. The background on machine learning strategies employed in this investigation (framework component 3) is summarized, and the techniques associated with the numerical processes exploited to transition the initial guess to a continuous form (framework component 4) are then introduced. The implementation details and flow of the design framework (framework components 1-4) are formalized to enable sample scenarios and preliminary results.

DYNAMICAL AND FORCE MODEL

Although the dynamical sensitivity is evident, the CR3BP offers an opportunity to approximate the higher-fidelity dynamics and to exploit the natural flows that are otherwise unavailable in simpler dynamical regimes. For the design framework, a robust cislunar transportation architecture is currently a priority, so the methodology is applied within the context of the Earth-Moon CR3BP. Here, the Earth (*primary,* P_1) and the Moon (*secondary,* P_2) are assumed to revolve in circular orbits around their common barycenter.[23] The spacecraft mass is assumed to be negligible in comparison to the more massive bodies. The Equations of Motion (EOM) for the spacecraft (P_3) as viewed in a rotating frame, also incorporate a thrust force to model the physical

capabilities of an engine/thruster, i.e.,

$$\dot{\boldsymbol{\chi}} = \begin{Bmatrix} \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{v}} \\ \dot{m} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{v} \\ \underbrace{\boldsymbol{f}(\boldsymbol{r}) + \boldsymbol{g}(\boldsymbol{v})}_{\text{natural}} + \underbrace{\frac{\mathbb{T}}{m}\hat{\boldsymbol{u}}}_{\text{low-thrust}} \\ \frac{-\mathbb{T}}{Isp\, g_o} \end{Bmatrix} \tag{1}$$

The differential equations include contributions from both the natural gravitational and the thrust acceleration sources to capture the motion of the spacecraft (s/c) and its mass history over time. In these equations, $\boldsymbol{\chi}$ is the full state vector comprised of the vehicle position and velocity vectors ($\boldsymbol{r}$ and $\boldsymbol{v}$, respectively) and the vehicle mass m. The thrust magnitude is represented by $\mathbb{T}$, and the thrust direction by $\hat{\boldsymbol{u}}$ (where a caret identifies unit magnitude). The *Isp* is the engine specific impulse and g_0 is the reference gravitational acceleration. Even without the engine specific thrusting terms, there is no closed form solution to the natural EOMs. So, in the thrust-free problem, other quantities such as the period, stability and the Jacobi Constant (*JC*, an energy-like quantity) that are associated with the natural solutions aid in characterizing the motion of the s/c within the confines of the CR3BP model. The *JC* is defined as:

$$JC = 2U^* - v^2 \tag{2}$$

where U^* is a pseudo-potential term,

$$U^*(x, y, z) = \frac{1-\mu}{r_{13}} + \frac{\mu}{r_{23}} + \frac{1}{2}(x^2 + y^2) \tag{3}$$

So, U^* is a function of the s/c position relative to the barycenter (x, y, z) and relative to the two primaries r_{13} and r_{23}, as well as the mass ratio of P_2 to the total system mass, μ. Note from Eqn. (2) that the *JC* term is only constant in the autonomous system where propulsive forces are absent; its time history resulting from s/c thrusting also offers valuable insights for developing and applying the automated framework. To deliver the thrust force in Eqn. (1), a relatively new technology, advanced ion-drive propulsion, is gaining popularity due to its propellant efficiency.[24] Thus, in the current analysis, the initial guess is transitioned to a continuous Constant Specific Impulse (CSI) low-thrust solution.

NUMERICAL CORRECTIONS AND OPTIMIZATION

The construction of low-thrust optimal trajectories typically involves formulating an optimal control problem, and solving for the time histories of the thrust magnitude and the thrust force direction to meet the desired boundary conditions. In the CSI regime, the available thrust magnitude is a function of the engine power allocation ($\mathbb{P}$) and efficiency (*Isp*). The relationship is modeled as:

$$\mathbb{T} = \frac{2\mathbb{P}}{Isp\, g_0} \tag{4}$$

The constant power $\mathbb{P}$ and *Isp* parameters dictate a constant thrust magnitude, and the basis of the control authority to maneuver the s/c is an on-off engine toggle and thrust vectoring. A continuous low-thrust trajectory is constructed from a discontinuous initial guess by employing a multiple-shooting scheme. A direct method is implemented to incorporate robustness into the numerical corrections scheme, although this technique induces large dimensionality.[25] An iterative Newton-Raphson scheme is employed to compute the set of design vector variables ($\boldsymbol{X^*}$) that satisfy the specified constraints, $\boldsymbol{F}(\boldsymbol{X^*}) = \boldsymbol{0}$, i.e.,

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{\chi}_i \\ \boldsymbol{u}_i \\ \varpi_{dep} \\ \varpi_{arr} \\ \varpi_i \end{bmatrix} \quad \boldsymbol{F} = \begin{bmatrix} \boldsymbol{\chi}_{i_{desired}} - \boldsymbol{\chi}_{i_{actual}} \\ \boldsymbol{\psi}_{desired} - \boldsymbol{\psi}_{actual} \\ \boldsymbol{u}_i{}^T \boldsymbol{u}_i - 1 \\ \mathbb{T}_0 + \varpi_{dep} + \sum_{i=1}^{\mathbb{n}-1} \varpi_i - \mathbb{T}_f - \varpi_{arr} \end{bmatrix} = \boldsymbol{0} \tag{5}$$

The design vector $\boldsymbol{X}$, is comprised of the s/c states ($\boldsymbol{\chi}_i$), thrust directions ($\boldsymbol{u}_i$), propagation durations (ϖ_i) from select nodes ($i \in \mathbb{n}$), the duration (ϖ_{dep}) from the departure orbit reference point at epoch $\mathbb{t}_0$ to the spacecraft departure state, and the duration (ϖ_{arr}) from the arrival orbit reference point at epoch $\mathbb{t}_f$ to the spacecraft arrival state. In an alternative formulation, the thrust magnitude is also included as a design variable rather than pre-assigning a binary (on/off) state to the nodes and allowing the inter-node propagation times to determine the duration of the thrust and coast arcs. The constraint vector $\boldsymbol{F}$ is sought, one that meets state continuity, boundary condition ($\boldsymbol{\psi}$), thrust direction unit vector constraints at the specified nodes, and the rendezvous constraint (when rendezvous is sought). The formulation for this last constraint in this investigation assumes fixed epochs for the departure and arrival orbit reference points $\mathbb{t}_0$ and $\mathbb{t}_f$, respectively. The converged solution is passed through a Nonlinear Programming (NLP) software such as FMINCON or SNOPT to optimize a cost function that is formulated to offer a favorable solution given the mission objectives and constraints.

HEURISTICALLY ACCELERATED REINFORCEMENT LEARNING

The creation of an initial guess for a reference solution prior to numerical corrections can be a labor intensive process dependent upon human intuition. So, an automated process to develop and curate effective initial guesses are sought to aid in the exploration of large and complex trade-spaces. In this investigation, the generation of an initial guess for a trajectory that transfers a s/c from a departure location to a specific destination is cast as a routing problem. Such a route is constructed from the appropriate sequencing of available waypoints in the design space to meet global objectives. The Heuristically Accelerated Reinforcement Learning (HARL) algorithm explored in Das-Stuart et al.[16,22] to strategically assemble advantageous intermediate natural arcs from a large and complex design space to connect departure and destination when subject to a set of prescribed constraints is incorporated in this investigation.

The pathfinding is implemented via an agent interacting with its environment and is formalized as a Markov Decision Process (MDP)[26] that is constructed from a tuple: $< S, A, P, R >$; *S* is a set of states available to an agent; *A* is a set of actions available to an agent; *P* is *P*(*s*,*a*,*s*′), is the probability that action *a* in state *s* leads the agent to arrive at state *s*′, $R : S \times A \rightarrow \mathscr{R}$ is the reward received for an action *a* in state *s*. The aim is an optimal policy (π^*) that executes an action at a given state to maximize the cumulative rewards received by the agent over multiple episodes. The RL theory examined in Das-Stuart et al.[16,22,27] is summarized below, and the specific implementation details for application in the CR3BP pathfinding problem are explored later.

In complex regimes where the transition probabilities *P* defining the system model and the rewards are not known a priori, it is useful to frame the problem as a model-free, state-action-value approach. A state-action-value update function is formulated in recursive form as:

$$\mathbb{F}(s,a) \leftarrow \mathbb{F}(s,a) + \alpha[R(s,a,s') + \gamma \mathrm{U} - \mathbb{F}(s,a)] \tag{6}$$

and forms the repository of reinforcements from which the agent learns the desirable behavior. Here, the agent takes an action a in state s and is then assumed to follow a certain policy from state s' onwards; an opportunity for the agent to experience the consequences of varied actions a from state s over time is, thus, conceived. The term U is formulated as required to reflect an on-policy or off-policy update based on the policy followed from state s'.[28] The agent learning rate is specified by α, which balances the information acquired from recently rewarded pathways versus a reliance on knowledge about pathways from prior episodes. The discount factor $0 \leq \gamma \leq 1$ is a measure of the balance between rewards gained from immediate and future time-steps; smaller values of γ favor immediate rewards.

Instead of potentially time-intensive infinite visitations to state-action pairs, the Heuristically Accelerated Reinforcement Learning (HARL) approach introduces a heuristic function to accelerate the learning process by biasing the selection of an action *a*, given a state *s*, possessing a reward *R*. The associated computational time-complexity is dependent on the accuracy of the heuristics. Scenarios with large state-spaces and multiple objectives benefit from a distributed network where the agents work cooperatively by updating a centralized reinforcement repository based on the distributed knowledge (cumulatively discounted rewards) gained during a specific episode. Within the context of mission design, a RL pathfinding agent continues the search until a stopping condition is satisfied, i.e., a pre-specified search duration, a terminal condition and/or a violation

of constraints. An overview of the learning process (within the context of mission design) is illustrated in Figure 2.

The learning process within each episode is comprised of two search scenarios — exploration and exploitation. Exploration enables a training phase where the agent *learns* about likely consequences of actions in the environment; the exploitation phase enables the agent to engage in informed decisions by capitalizing on previously gained knowledge. The policy at a particular state, as influenced by the state-action pair and heuristic, is defined :[29]

$$\pi(s) = \begin{cases} \mathbb{E}[\mathbb{F}(s,a) \bowtie \mathfrak{ξ}\mathbb{H}(s,a)^{\mathfrak{β}}], & \text{if } \mathfrak{q} > p \\ a_{random}, & \text{otherwise} \end{cases} \quad (7)$$

where ($\mathbb{H} : S \times A \rightarrow \mathscr{R}$) is the heuristic function, $\bowtie$ is a math operator as determined by the RL algorithm and its implementation, $\mathfrak{ξ}$ and $\mathfrak{β}$ are weighting parameters that dictate the influence of the heuristic, and a_{random} is the action selected randomly from all those available in state *s*. In this investigation, the exploitation function $\mathbb{E}$ incorporates stochasticity to inform the nature of the policy at a state, and the implementation is discussed later when the details of the framework are presented. In Eqn. (7), *p* ($0 \leq p \leq 1$), is the trade-off parameter between exploration and exploitation, and $\mathfrak{q}$ is a random value from a uniform distribution in [0,1]. The gradual shift from exploration to exploitation is determined by diminishing the value of p to a steady-state value of p_{ss} by the k^{th} episode from a total number of episodes Ep, as determined by the relationship:[11]

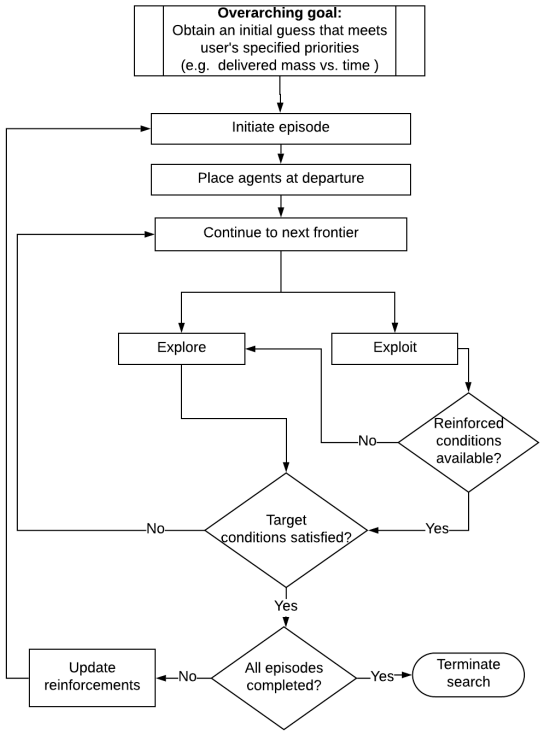


Figure 2: An implementation of HARL algorithm

$$p = p_{ss} + (1 - p_{ss})e^{(-\frac{k-1}{ln(Ep)})} \quad (8)$$

The blending of heuristics with reinforcement learning is a powerful utility in solving many NP-hard and NP-complete pathfinding problems, e.g., the Traveling Salesman Problem (TSP).

DEVELOPMENT OF THE DESIGN FRAMEWORK

A primary objective of this investigation is the application of the automated pathfinding design framework to address contingency scenarios. Furthermore, the transition of initial guesses (IG) from lower-fidelity dynamical models such as the CR3BP enhances rapid design and is effective. The first component in the framework development establishes the *reach* of a specific spacecraft within the dynamical regime for a specified time horizon, as influenced by its thrust-to-mass ratio, propellant efficiency and other performance characteristics. The s/c's *reach* then assists in identification of natural options instantaneously *accessible* within the dynamical regime. The available trajectory options and the ability of the spacecraft to transition through them dictates the subsequent component — the *sequencing* of intermediate pathways to enable a satisfactory path between the departure and destination conditions. The final component involves convergence and optimization of the initial guesses via numerical techniques to ensure that the mission constraints are not violated, and the requirements are fulfilled. The details associated with each component follow.

Computation of Accessible Regions

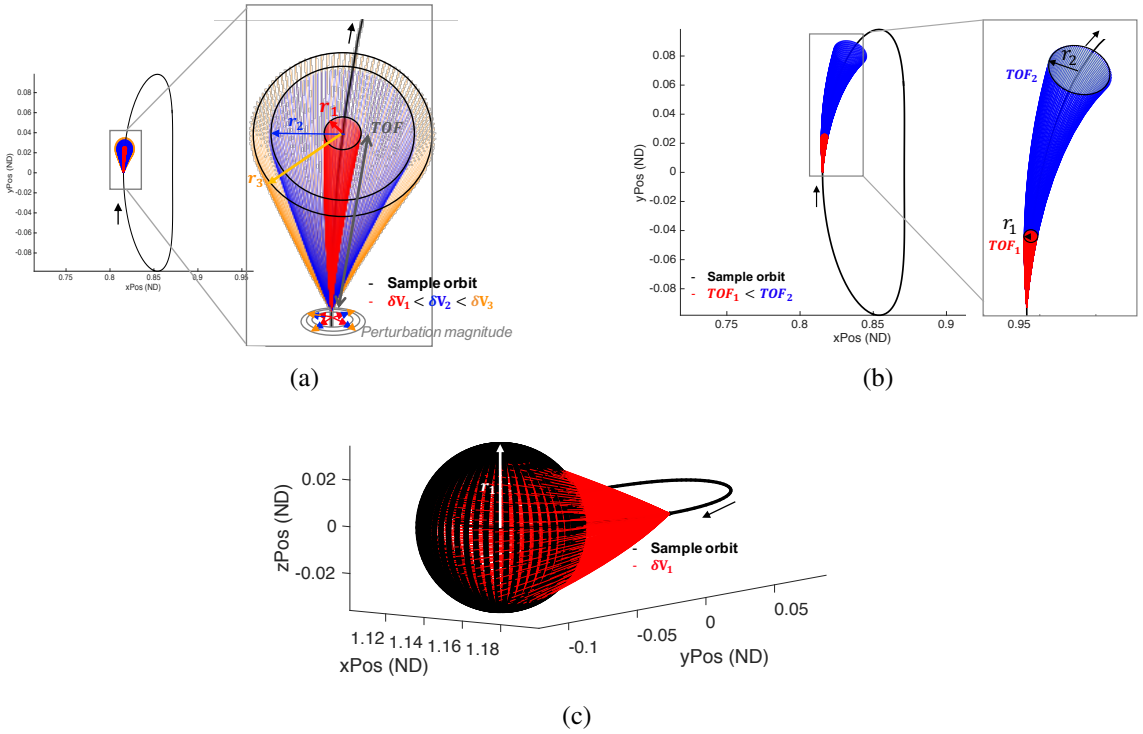
The computation of an Accessible Region (AR) from a particular node 'n', originates with a perturbation of the spacecraft current velocity within a circle/sphere (for the planar/spatial problem, respectively) with

a prescribed radius. This step is followed by a propagation of the perturbed and unperturbed states for a pre-determined duration. The resulting downstream behavior is a stretching of the perturbed states from this unperturbed natural arc due to the influence of the existing gravitational forces in the system. To simplify implementation, a measure of the deviations of the perturbed states from the end of the natural arc is mapped to a circular/spherical Accessible Region (AR) in a planar/spatial setting, respectively, rather than using the true deformations.[16,22] Also, approximating low-thrust segments via a sequence of impulsive maneuvers is effective for preliminary mission design,[30] and aids in this investigation. Thus, for the low-thrust initial guess generation, the perturbations in the s/c velocity at node 'n', (δV_n), are induced via impulsive maneuvers. The associated impulse magnitude is determined via the relationship

$$\delta V_n = Isp\ g_0\ \ln(\frac{m_0}{m_f}) \tag{9a}$$

$$\text{where, } m_f = m_0 - (\dot{m} \times \delta \mathbb{t}_n), \tag{9b}$$

once the engine characteristics (e.g., specific impulse, mass flow rate) and the user-defined engine operation time ($\delta \mathbb{t}_n$) are determined. Figure 3(a) illustrates the difference in the footprint sizes for varying perturbation magnitudes for a fixed propagation time; a larger AR enables access to more natural arcs. Inspection of Eqn. (1) and Eqn. (9) indicates that higher specific impulse values result in smaller δV_n perturbations at a given node 'n' for a fixed operation / burn time ($\delta \mathbb{t}_n$), and aids in delineating the behavior of propellant efficient low-thrust and less efficient chemical regimes. Longer propagation durations increase the footprint



(a) (b)

(c)

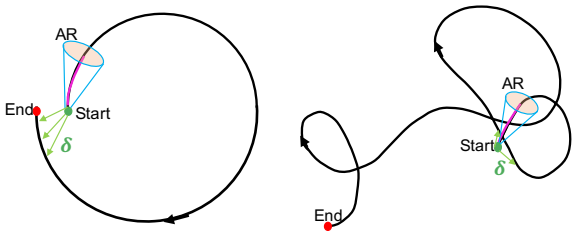
Figure 3: Influence of δV on the accessible region footprints in the (a) planar case (c) spatial case. (b) Influence of TOF on the footprint of accessible regions in the planar case.

of the AR (Fig. 3(b)), and an AR in the spatial regime is illustrated in Fig. 3(c). Note that the ARs in the search algorithm accommodate both the position and velocity states.

Instantaneous Generation of Initial Conditions

Accessible regions aid in constraining the searchable design space at any given instant in time. However, when *a priori* knowledge of the dynamics in the system is lacking, natural conditions are instan-

taneously generated within an AR, in real-time.[16] Operating under the assumptions of a complex and highly nonlinear dynamical regime, however, these conditions are a blend of both ordered and chaotic motion. Ordered motion is distinguished from chaotic by predicting the behavior over all time within the CR3BP. However, it is beneficial to exploit any available and mission-enabling conditions that do not violate mission constraints. Thus, chaotic conditions may offer suitable candidates if the resulting natural motion is predictable over some specified and acceptable time-frame. Once states are populated within the bounds specified by an AR, a receding horizon technique tests for predictability as illustrated in Fig. 4. Long-term predictability is observed by propagating a condition from within an AR for some prespecified duration; this look-ahead time-frame is a design parameter and varies based on the dynamical system and/or mission considerations. If the propagated state does not escape the system as determined by a user-specified threshold for boundedness δ, then it is set aside for selection by the s/c even if it is chaotic in nature. Note from Fig. 4 that only a small fraction along this propagation time is captured for constructing an AR. This unrestricted approach to natural state selection offers the potential to uncover non-intuitive transfer geometries otherwise not available from known and ordered natural families. Thus, the ability to select chaotic states broadens the design options and eliminates the manual time investment required to identify beneficial families to include in the searchable volume as explored in Das-Stuart et al.[16,22] The disadvantages are the diminished ability to specify the nature of various pathways traversed by the spacecraft, as well as the requirement to perform on-line propagations of a candidate state to confirm that it does not escape the system, possibly slowing the pathfinding process.



(a) (b)

Figure 4: Two sample scenarios (a) and (b) illustrating look-ahead trajectory segments from *Start* to *End*, the boundedness value, δ, and the relatively small fraction along the look-ahead segment to establish an Accessible Region (AR).

Automated Pathfinding Phase

Pathfinding is implemented to *sequence* the appropriate conditions for maneuvering a s/c from origin to destination. The available conditions are the natural states randomly populated within an AR that constitute the *frontier* of solutions at a given instant in time. The HARL algorithm selects a state from within each frontier to expand the search towards the destination. A sample schematic illustrating the search process is displayed in Fig. 5. The dimension of each computed AR is a function of the respective velocity perturbation magnitude δV_n and the propagation duration. The time required to reach any state within a particular AR is equivalent to this propagation duration required to reach its center state (the state at the end of an unperturbed natural arc propagation). The pathfinding process terminates when the destination condition falls within an AR, or when an agent violates a user-defined constraint (e.g., altitude from a primary). The destination condition could be any state belonging to a discretized arrival orbit or could be a specific user-defined destination.

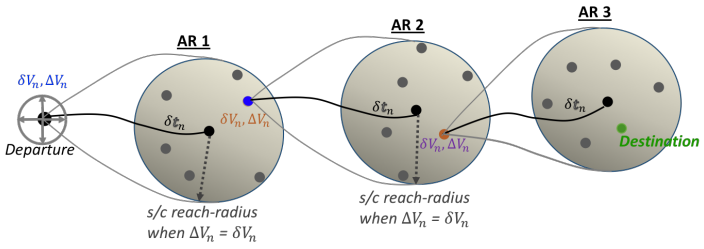


Figure 5: Schematic illustrating the roles of δV_n, ΔV_n and δt_n in single node expansions from a given AR. The colored conditions are states selected from an AR to enable a sample sequenced pathway. The black arcs represent the natural propagations, i.e., propagations when $\Delta V_n = 0\ m/s$. Note that $0\ m/s \leq \Delta V_n \leq \delta V_n\ m/s$ to reach the states populated along the radius of an AR, and $\Delta V_n = \delta V_n\ m/s$ to reach the states along its circumference.

The overarching goal for the pathfinding scheme is posed as an optimization problem. For example, the following relationship represents the prioritization of minimizing propellant consumption (m_p) versus minimization of the transfer duration (t_f) to address the persistent *mass-time* tradeoff challenge:[16,22]

$$min\ \mathbb{J} = W_t t_f + W_p m_p \quad (10)$$

where, the propellant mass m_p is

always weighted against the transfer duration, t_f, such that the weighting on TOF, W_t is always 1, and W_p is a design variable. The transfer duration t_f in Eqn. (10) is the aggregate sum of the propagation durations associated with each natural arc assembled within the transfer sequence. The total propellant consumption m_p is post-processed analytically via the rocket equation upon computation of each ΔV_n (maneuver cost to transfer from one frontier to another) in the pathfinding sequence. This approach simplifies computations for a continuously mass-varying path and supports computational efficiency as the EOM for the mass flow is not propagated for IG generation. Additionally, the approach supports a modular architecture for assessing solutions for varied s/c by adjusting the engine / thruster parameters used to compute the maneuvers.

The flexibility of the design framework admits alternative cost function formulations as well. For example, much like the mass-time trade, spacecraft rendezvous is another challenging goal in mission design. In a rendezvous problem, a chaser s/c seeks to meet the exact time-dependent states of a target object. One approach to realizing this goal, given the existing framework, is to (i) exploit the successive AR computation scheme for a chaser s/c to arrive at states along a target object's path, and (ii) simultaneously minimize the difference in epoch between the chaser s/c terminal state that also corresponds to the target state at some instant in time. The global cost function is mathematically formulated in the form:

$$min\ \mathbb{J} = |epoch_{fC} - epoch_{fT}| \tag{11}$$

where the subscripts fC and fT indicate the final conditions for the chaser s/c and target, respectively. Note that Eqn. (11) does not incorporate information about the chaser or target states. Given that satisfaction of the state and epoch component requirements for a rendezvous are independent in the implemented approach, the pathfinding algorithm does not uncover a path that truly completes the rendezvous with the target; rather, a path with minimal error in terms of the rendezvous epoch is sought. Such an implementation serves to construct a feasible initial guess for a numerical corrections process that enforces a strict rendezvous constraint. Although unexplored in this investigation, alternative implementations where the rendezvous constraint is enforced during the initial guess generation process also produce viable options.

Implementation of HARL Algorithm

A HARL algorithm does not guarantee optimality due to the inclusion of heuristics, but multiple search episodes enable convergence to a nearly-optimal solution to address the global objective. The inter-node costs (*arc* costs) are computed on-demand as an agent progresses in its search towards the destination, and the search for a transfer path is posed as an episodic problem; i.e., an agent's search duration is defined by a user-imposed upper-bound. An agent within a parallelized search episode (i) initiates the search for a suitable initial guess by randomly exploring the state-space (selecting states at random from within the ARs) and (ii) gradually employs more exploitation (capitalization on previously gained knowledge about the state-space) over subsequent episodes as specified by Eqn. (8). During 'exploitation', an agent only selects from nodes that have previously supported successful transport to the destination and are, thus, already reinforced. The selection of a particular reinforced node from an AR during the 'exploitation' phase is guided by a heuristic $\mathbb{H}_n$, one that is crafted to accommodate two aspects during the decision making process: (a) the propellant consumption required to transfer from the current state to the reinforced state within the AR m_{f_n}, and (b) a measure of 'goodness' d_n, for the selected node in terms of its proximity to the target condition(s) in both the position and velocity dimensions. These factors are incorporated mathematically into the heuristic function to yield:

$$\mathbb{H}_n = \frac{{m_f}_n^{W_m}}{{d_n}^{W_d}} \tag{12a}$$

$$\text{where, } d_n = ||\boldsymbol{\chi}_n - \boldsymbol{\chi}_T|| \tag{12b}$$

where $\boldsymbol{\chi}_n$ is the state vector (position and velocity) associated with a particular node in the AR, and $\boldsymbol{\chi}_T$ is the state vector corresponding to the target condition(s). The weights are also design variables, where W_m grants control over the desired s/c performance (delivered mass) and W_d influences the manner in which the transfer trajectory tends towards the destination. These qualities also render the heuristic applicable as a measure of reward. Specifically, the return for a state-action pair contributing to a successful transfer path is computed as the accrued sum of the heuristic values along the path ($R_n = \sum \mathbb{H}_n$). The action 'a' undertaken

from the current state 's' is the executed $\Delta\mathbf{V}_n$ and is processed as a function of the state s' selected from within the associated AR. Recall that the state s' is selected randomly during 'exploration' or as influenced via heuristics during 'exploitation'. The discount factor γ is set equal to zero as a design choice in this preliminary investigation posed as a Monte Carlo[28] based approach that computes returns for entire episodes. Future examination of the impact of variation in the value of the discount factor and the implementation of a temporal difference learning[28] may also offer additional insights.

The automated pathfinding strategy in this investigation aims to establish general desirable pathways to the destination rather than identify distinct and discrete beneficial nodes in configuration space. Thus, states neighboring those that support top ranking pathways according to the global cost function (Eqn. (10)) are also reinforced.[22] Furthermore, the quality of a node is only a guide in the overall search strategy, as stochasticity is introduced in the selection of a favorable node from within an AR with the probability $\mathscr{P}_n$ during the exploitation phase:

$$\mathscr{P}_n = \frac{\mathbb{F}_n \, \mathbb{H}_n}{\sum \mathbb{F}_n \, \mathbb{H}_n} \tag{13}$$

This probability $\mathscr{P}_n$ emphasizes two important considerations that bias the selection of a node n: (a) perceived relevance of the node to the transfer via its accrued returns ($\mathbb{F}_n$) over episodes, and (b) its current attractiveness as represented by the heuristic $\mathbb{H}_n$. The HARL design choices are enforced within a distributed, cooperative environment where the incremental knowledge acquired by each agent combines to furnish attractive transfer trajectories that are then ranked by their overall performance (e.g., Eqn. 10) over multiple episodes. The aim of this process is the extraction of the transfer solution that optimizes the global cost function to the best extent while satisfying the goals and constraints.

INITIAL GUESS TO HIGHER-FIDELITY SOLUTION

In this investigation, a final solution is defined as an end-to-end transfer trajectory that adheres to the constraints imposed within a CR3BP regime, augmented by the selected engine forces. Note that such a solution is merely an initial guess for a simulation in the CR3BP. The information associated with the arcs that are assembled by the automated pathfinding process is sufficient to initiate the numerical corrections process, namely via knowledge of the position, velocity, mass, thrust direction, thrust magnitude and time estimates along the connecting arcs. The user possesses the freedom to select a desired computational tool for executing the convergence/optimization process. Note that varying thrust magnitudes are interspersed within the lower-fidelity solution, thus, it can serve as an initial guess for a range of engine capabilities in a higher-fidelity simulation. However, the range of thrust magnitudes is bounded in the corrections updates.

SAMPLE RESULTS

The execution of the components in the design framework produce initial guesses for transfer scenarios that are then transitioned to a continuous solution via a higher-fidelity engine model. A system failure and/or any degradation during flight can cause the s/c to significantly deviate from its nominal transfer trajectory, leading to undesirable impacts on the mission. For example, a safing event for the Dawn spacecraft resulted in $\approx 4\ days$ of missed thrust that then led to a re-design of the s/c baseline trajectory approach to Ceres with an ultimate delay of $\approx 26\ days$ to capture into a particular science orbit.[31] Monte carlo simulations incorporating information from historical data in Imken et al.[32] also suggest that the typical duration of inoperability due to safing events is $\approx 3.5\ days$. In such situations, there is a critical need for rapid trajectory design solutions to minimize adverse mission outcomes. This investigation assesses the ability of the machine learning algorithms to recover mission goals during contingency scenarios via a motivating example where the engine throughput is compromised. In particular, two types of recovery scenarios are considered: (i) recovery to the arrival orbit, and (ii) recovery to rendezvous with a pre-positioned target on the arrival orbit. The terminology associated with the contingency scenarios explored in this investigation is summarized in Table 1. In this preliminary investigation, orbit determination errors and higher-fidelity ephemeris models are not incorporated.

Table 1: Definition of symbols employed in contingency-preparation-related results.

Symbols	Definition
$\left(\frac{m_f}{m_0}\right)_N$	Delivered mass fraction on nominal trajectory.
t_{0F}	Absolute epoch of failure initiation on nominal path.
t_{FC}	Absolute epoch of forced coast initiation.
t_{0N}	Absolute epoch on nominal path corresponding to absolute epoch of recovery initiation on recovery path.
t_{fN}	Absolute epoch on nominal path corresponding to its terminal state.
$t_{0R} = \text{DP}$	Absolute epoch on recovery path corresponding to the termination of the forced coast segment / initiation of the recovery path from the Decision Point (DP).
t_{fR}	Absolute epoch on recovery path corresponding to its terminal state at the target.

Recovery to Arrival Orbit Conditions

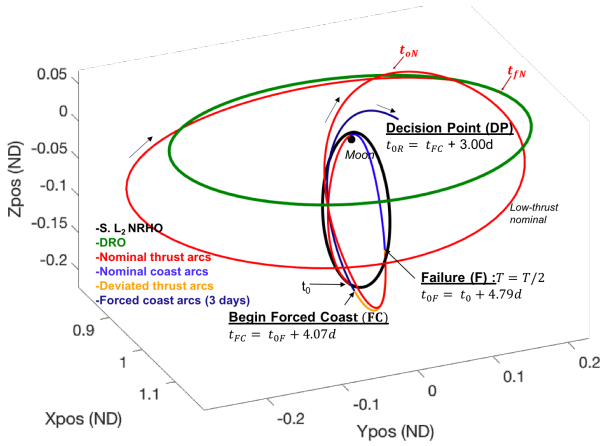
Transfer scenarios motivated by long-term scientific observations or end-of-life disposal may seek to transition a s/c to desired orbital conditions that satisfy orbit stability, communications, and other considerations. The transfer from an NRHO to a stable DRO in Fig. 6(a) is one such scenario. The nominal transfer path along with the simulated failure and post-failure routes are illustrated in Fig. 6(a). All the contingency examples are executed for a s/c initial mass of $M_0 = 1000\ kg$, with the s/c performance metrics specific to Fig. 6 summarized in Table 2. The acronym DFD for all the contingency examples refer to Days From Departure along the NRHO (i.e., from the location on the NRHO where the s/c departs on the nominal trajectory). In

Table 2: Performance metrics for Fig. 6

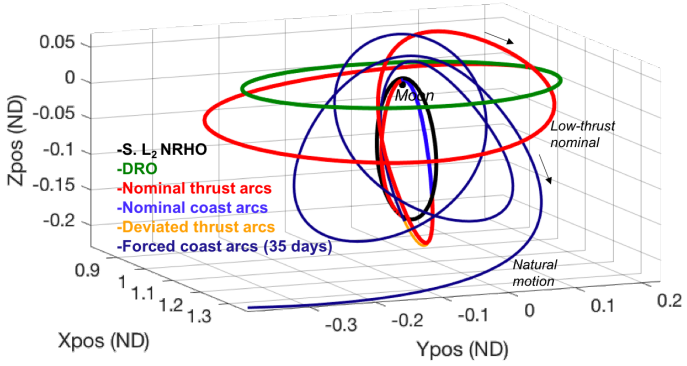
Scenario	$t_{0R}\ (DFD)$	$t_{0N}\ (DFD)$	$t_{fN}\ (DFD)$	$t_{fN} - t_{0N}\ (days)$	$\left(\frac{m_f}{m_0}\right)_N (\%)$
Fig. 6(a)	11.86	11.86	28.86	17	98.71
s/c Specifications for Nominal Trajectory					
$a_0 = 2.2 \times 10^{-4}\ m/s^2$ $\quad \mathbb{T} = 0.22\ N$ $\quad Isp = 4000\ s$					

this example, the engine throughput is diminished from $\mathbb{T} = 0.22\ N$ by 50% at $t_{0F} \approx 5\ days$ into the nominal transfer (the equivalent to losing the thrust capability of one engine if 2 engines are originally active). Such a failure leads the s/c to drift from the nominal trajectory over $\approx 4\ days$, which is followed by a commanded forced coast at $t_{FC} \approx 8.86\ days$ for a duration of 3 days to simulate a safe mode recovery operation. The Decision Point (DP) indicates the epoch t_{0R} at which the recovery trajectory is initiated. The duration of the post-failure propagation and forced coast segment in Fig. 6(a) do not accommodate orbit determination updates but are intended to introduce significant deviations from the nominal trajectory to investigate the capabilities of the design framework and machine learning techniques to construct recovery options. The ability to recover and return to the nominal transfer path after the failure is a function of the failure circumstances (e.g., where along the trajectory the failure occurs); effective response may also require feedback control. The results here do not recover the nominal path because of the extensive deviations introduced from the nominal route. The forced-coast segment in Fig. 6(a) is extended for an additional $32\ days$ in Fig. 6(b) to illustrate the spacecraft's departure from the vicinity of the destination in the absence of corrective measures over an extended duration.

The recovery of a s/c from the failure scenario in Fig. 6(a) is executed out by initial guess generation via the design framework, followed by numerical corrections to deliver a continuous end-to-end path. Since the recovery is not constrained to strict time-frames in this non-rendezvous transfer scenario, the global mass-time cost function (Eqn. (10)) is incorporated with priority placed on TOF. The recovery path resulting from restoring the s/c with full thrust at t_{0R} is produced by executing the pathfinding framework to target discretized states over 1 period ($\approx 13\ days$) of the DRO. A reverse-time waypoint seeding process via the



(a) Nominal transfer scenario with injected failure and forced coast segments



(b) Outcome from no corrective thrusting action after DP

Figure 6: Contingency scenario example — Departure: NRHO (Periapsis Altitude $= 1763.31\ km$), Arrival: DRO (JC: 2.935, Period: $\approx 13\ days$). (a) Engine failure simulated $\approx 5\ days$ after departure from NRHO, and the trajectory deviates due to diminished engine throughput, (b) Illustration of natural motion resulting from no corrective action for $35\ days$ post DP.

ARs as discussed in Das-Stuart et al.[16] is also implemented. The initial guess for the recovery transfer computed via the pathfinding agents appears in Fig. 7. The total duration for the recovery arcs reflect a

Table 3: Performance metrics for s/c recovery scenario in Fig. 7

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$\frac{m_f}{m_{0\ est}}\ (\%)$
Fig. 7(a)	11.86	32.86	21	99.02
Natural Condition Transport Approach — Free-Form Search.				
s/c Specifications Pre-Failure				
$a_0 = 2.2 \times 10^{-4}\ m/s^2$ $\quad \mathbb{T} = 0.22\ N$ $\quad Isp = 4000\ s$				
s/c Specifications Post-Forced Coast				
$a_0 = 2.205 \times 10^{-4}\ m/s^2$ $\quad \mathbb{T} = 0.22\ N$ $\quad Isp = 4000\ s$				

slightly longer TOF to that observed for the nominal path after t_{0N} (comparing Tables 2 and 3); observe that the path assumed during recovery is geometrically similar to the nominal route. The observed discrepancies are expected, as the combination of the imposed failure and forced-coast segment render a loss in altitude of $\approx 25,000\ km$ from the nominal solution at $t_{0N} = t_{0R} = 11.86\ days$ and, thus, also results in varied

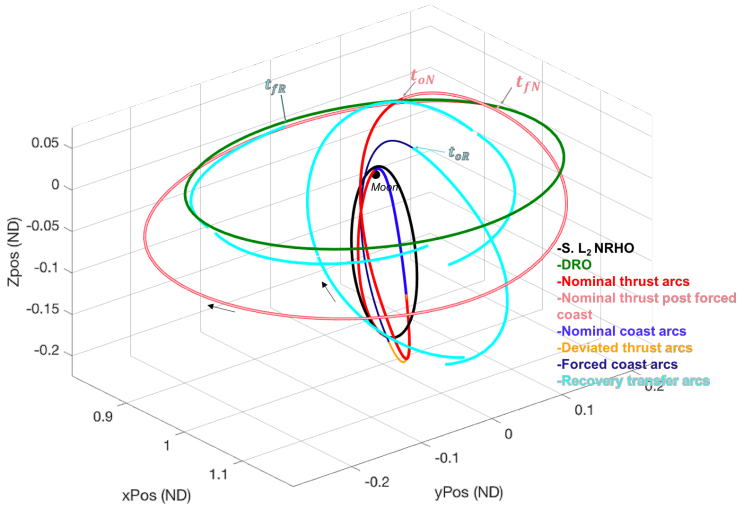


Figure 7: Initial guess to recover with $\mathbb{T} = 0.22\ N$ from failure introduced in Fig. 6(a)

trajectory options for the s/c states on the two trajectories at this epoch to continue towards the destination.

The initial guess is successfully transformed to a low-thrust solution via the process detailed in Das-Stuart et al.[22] that leverages a user-defined thrust-threshold to seed thrust and coast nodes to initiate a numerical corrections process. Note that, in this example, the discrete ΔV_n values executed to construct each arc in the initial guess (illustrated in Fig. 8) are not allowed to exceed $\approx 57\ m/s$. This value is the maximum maneuver magnitude that the low-thrust s/c is capable of imparting over each 3 day burn arc in the initial guess according to the assumptions incorporated via Eqn. (9). The user-defined threshold is a design variable that enables different local optima as demonstrated in Das-Stuart et al.[22] The nodes above the threshold are seeded as thrust nodes, and those below are initialized as coasting nodes; both types of nodes are free to move in space and time during the corrections process. In addition to the magnitudes, the selection of states from the ARs during pathfinding also delivers the s/c thrust directions to construct and maneuver from one arc to another along the initial guess.[22] The optimization process yields the mass-optimal continuous low-thrust final solution displayed in Fig. 9. The mass-optimal objective inserts coasting time along the recovery

Table 4: Performance metrics for mass optimal CSI s/c recovery in Fig. 9

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$TD\ (days)$	$\frac{m_f}{m_0}\ (\%)$
Fig. 9	11.86	36.40	24.54	23.30	98.41
s/c Specifications Pre-Failure					
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	
s/c Specifications Post-Forced Coast					
$a_0 = 2.205 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	

solution, leading the s/c to reach the destination orbit approximately a week after the nominal epoch at t_{fN}. The thrusting period of $\approx 23\ days$ in Fig. 9 enables the s/c to execute the energy and plane change required to maneuver into the DRO. This thrust duration results in a mass fraction of 98.65% and is close to the value of 99.26% estimated by the initial guess (IG). The mass fractions computed in this investigation refer to the ratio of delivered s/c mass to the initial NRHO departure mass of $M_0 = 1000\ kg$.

In practical applications, it may not be always possible to recover the s/c with full thrust capabilities after an engine malfunction. The transfer scenario in Fig. 10 explores the outcome of maintaining the thrust levels at a diminished 50% capability after t_{0N} to deliver the s/c to the DRO. The diminished thrust and, therefore, acceleration levels available to the s/c in the example in Fig. 10 lead to a prolonged recovery

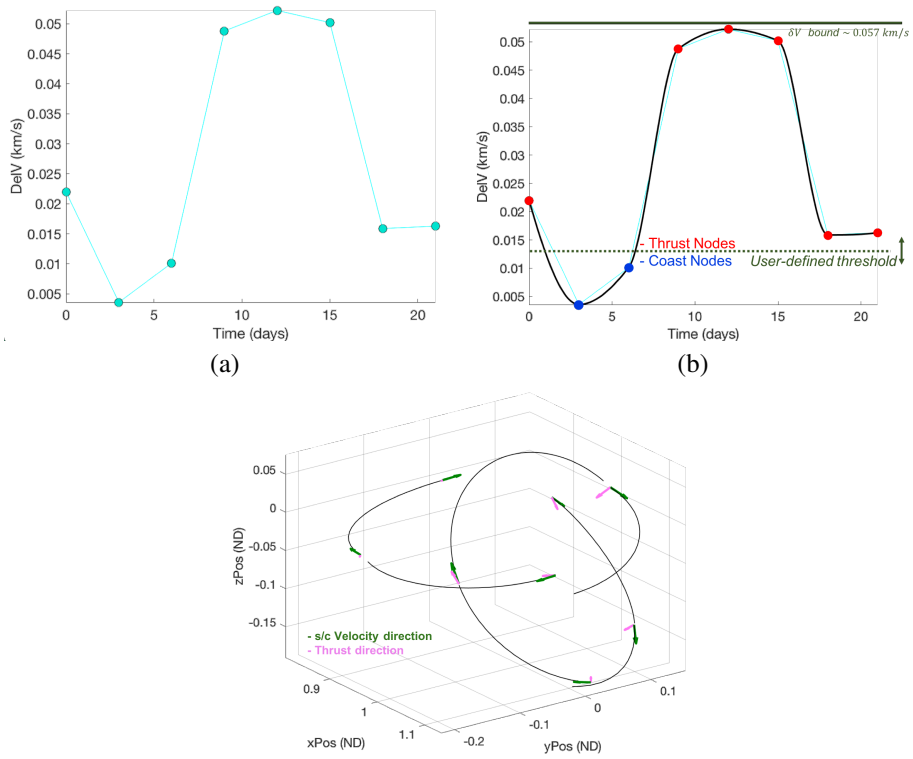


Figure 8: ΔV histories enabling construction of the initial guess for a recovery trajectory. (a) Discrete ΔV values enabling each initial guess transfer arc, (b) Interpolated ΔV curve with sample user-defined thrust threshold to aid in transitioning discrete initial guess to continuous low-thrust solution, (c) s/c velocity and maneuver / thrust directions along each arc for the initial guess.

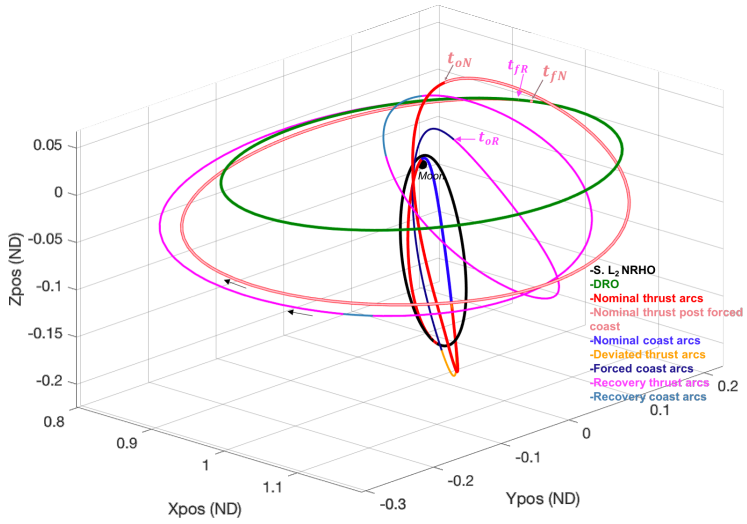


Figure 9: Low-thrust mass-optimal solution for initial guess presented in Fig. 7.

duration compared to the initial guess solution in Fig. 7. The resulting geometry and arrival locations along

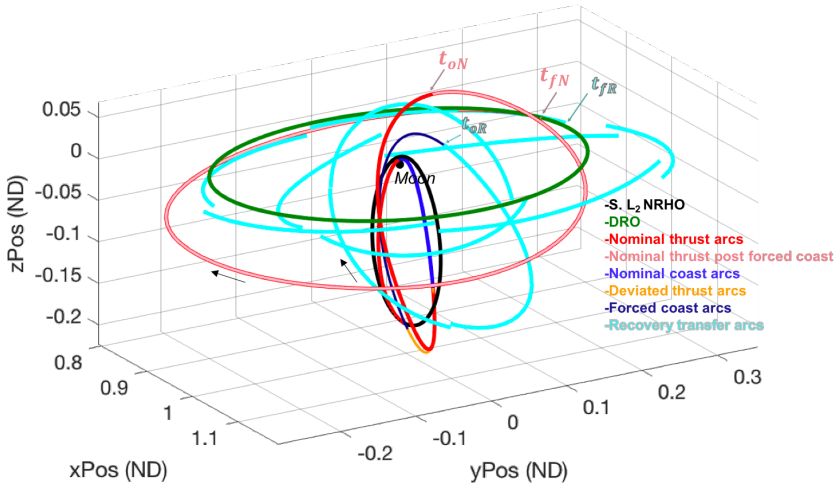


Figure 10: Initial guess to recover with $\mathbb{T} = 0.11\ N$ from failure introduced in Fig. 6(a)

Table 5: Performance metrics for s/c recovery scenario in Fig. 10

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$\frac{m_f}{m_{0\ est}}\ (\%)$
Fig. 10(a)	11.86	47.86	36	97.70
Natural Condition Transport Approach — Free-Form Search.				
s/c Specifications Pre-Failure				
$a_0 = 2.2 \times 10^{-4}\ m/s^2$ $\quad \mathbb{T} = 0.22\ N$ $\quad Isp = 4000\ s$				
s/c Specifications Post-Forced Coast				
$a_0 = 1.1025 \times 10^{-4}\ m/s^2$ $\quad \mathbb{T} = 0.11\ N$ $\quad Isp = 4000\ s$				

the DRO are also different. The associated mass-optimal solution is plotted in Fig. 11. As discussed for the

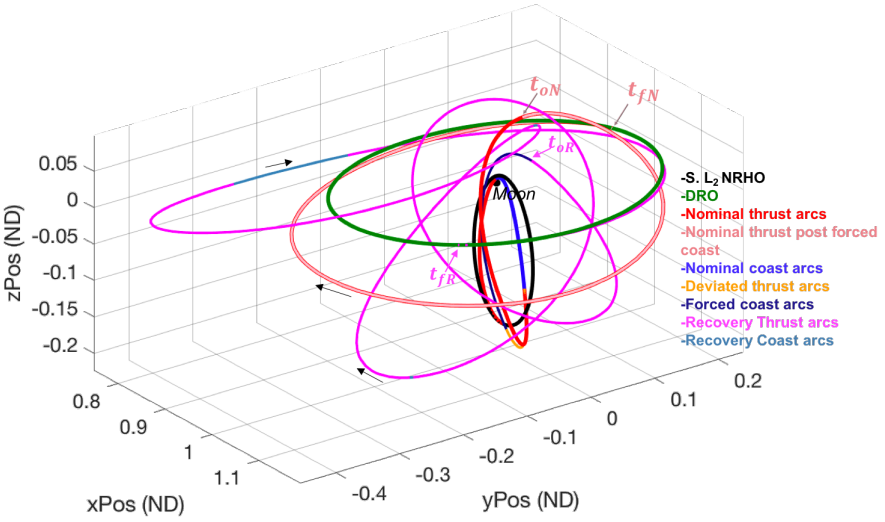


Figure 11: Low-thrust mass optimal solution for initial guess presented in Fig. 10.

Table 6: Performance metrics for mass optimal CSI s/c recovery in Fig. 11

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$TD\ (days)$	$\frac{m_f}{m_0}\ (\%)$
Fig. 11	11.86	57.11	45.25	43.53	98.48
s/c Specifications Pre-Failure					
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	
s/c Specifications Post-Forced Coast					
$a_0 = 1.1025 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.11\ N$		$Isp = 4000\ s$	

example in Fig. 9, the mass-optimal solution eliminates the discontinuities in the IG and incorporates coast arcs where possible to minimize propellant consumption. The result is a recovery path that is almost twice as long compared to the case in Fig. 7, but note that the delivered mass fraction is similar. Recall that mass-flow rate is proportional to the s/c thrust levels (Eqn. (1)). So, a degradation in thrust magnitude by 50% leads to an equally diminished rate of propellant consumption; a thrusting duration that is twice as long thus leads to similar overall propellant consumption.

The change in inclination and energy levels required to complete the recovery lead to varying thrust durations based on the s/c thrust capabilities. The history of such energy changes for both the $\mathbb{T} = 0.22\ N$ and $\mathbb{T} = 0.11\ N$ recovery scenarios are illustrated in Fig. 12 (a) and (b), respectively. The path resulting

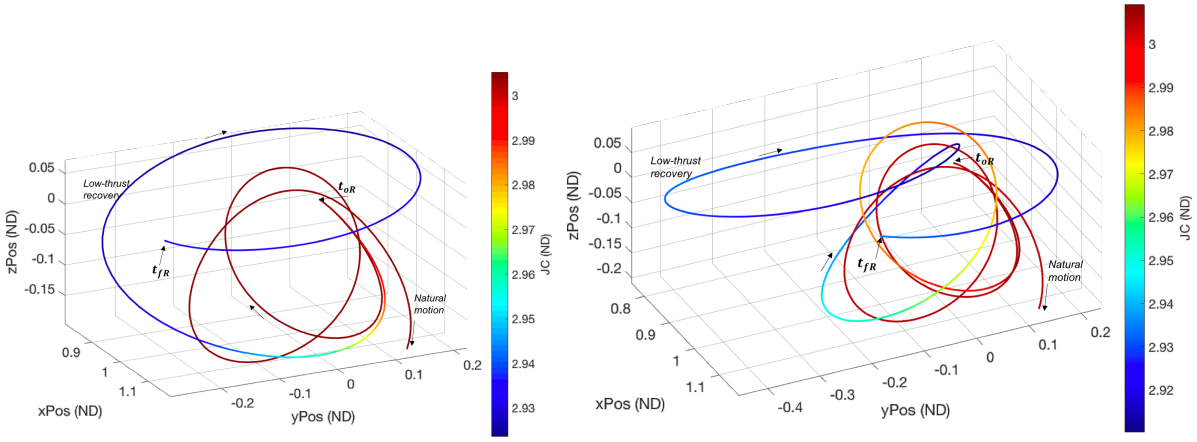


Figure 12: The Jacobi Constant (JC) history over the low-thrust flight paths in (a) Fig. 9 for s/c with $0.22\ N$ thrust during recovery and, (b) Fig. 11 for s/c with $0.11\ N$ thrust during recovery

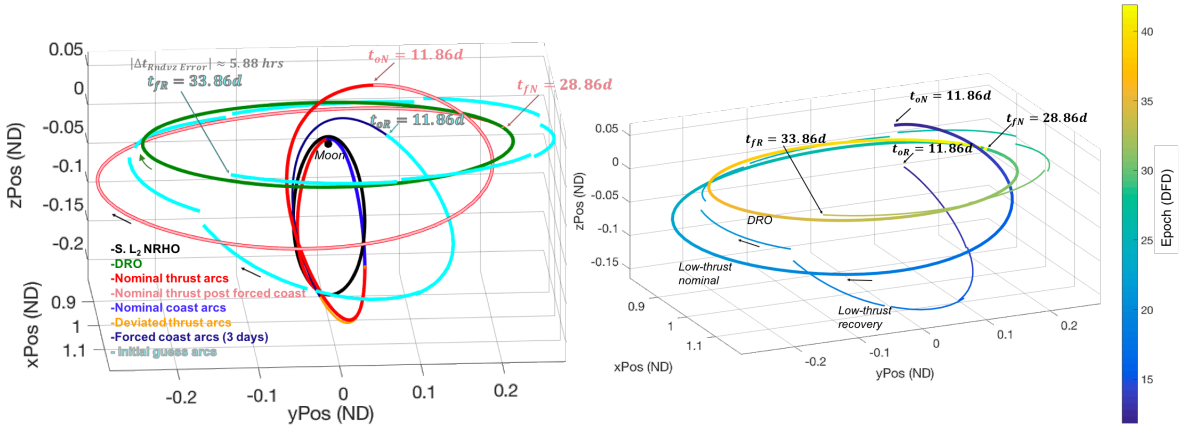
from the absence of corrective measures at t_{0R} (i.e., natural motion) is represented by the constant maroon curve indicative of a JC value > 3. Both the $0.22\ N$ and $0.11\ N$ spacecraft thrust to deviate from the natural course of motion and maneuver towards the DRO; the more powerful s/c is able to transition through the energy levels more quickly than the one at the lower thrust level ($0.11\ N$). Interestingly, both solutions maneuver through energy levels greater than that of the arrival orbit prior to orbit insertion ($JC = 2.935$).

Recovery to Complete Rendezvous on Arrival Orbit

The previous examples demonstrate the potential of the automated pathfinding algorithm to transport a s/c to the desired arrival orbit conditions. However, these examples do not incorporate rendezvous considerations that require the transport to be conducted within a strict TOF to meet the states that correspond to a specific target s/c. Thus, to rendezvous, a global cost function that seeks to uncover the transfer with the least discrepancy in epoch between the chaser and target s/c at a given state (Eqn. (11)) is implemented. The

failure scenario is the same as that considered in Fig. 6(a). Given the assumption that a target s/c is already on the DRO and continues its path along this orbit, the goal for the chaser s/c is to initiate its recovery at the DP (t_{0R}) and meet the target anywhere along the DRO such that minimal error in the rendezvous epoch is incurred. Recall that the ARs (formed instantaneously during the pathfinding search) inform the chaser s/c of its ability to *reach* the target s/c states, but does so with no consideration of the associated epoch of the target s/c states. The pathways that lead to minimal error in the rendezvous epoch are then ranked according to the global cost function, and the most attractive options are reinforced. Such a design choice, where the tracking of the target states and the epoch are considered independently, is chosen such that the design framework only seeks to uncover an initial guess via the pathfinding agents; then the result is corrected by numerical techniques. Other design approaches where the states and epoch are considered jointly for rendezvous via machine learning is beyond the scope of this preliminary investigation.

The initial guess constructed by the agents when the chaser s/c is re-equipped with full thrust at t_{0R} is displayed in Fig. 13(a) The sequence of the cyan transfer arcs in Fig. 13(a) offers an initial guess for a



(a) (b)

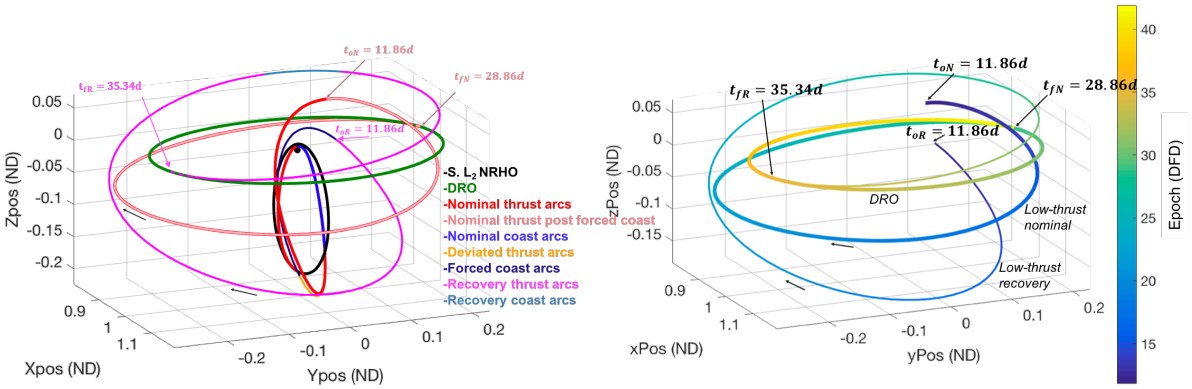
Figure 13: Sample rendezvous recovery initial guess for scenario in Fig. 6 for a chaser s/c with $\mathbb{T} = 0.22\ N$ from t_{0R}. (a) Transfer arcs, (b) Transfer arcs colored by epoch (DFD)

Table 7: Performance metrics for s/c rendezvous scenario in Fig. 13

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$\frac{m_f}{m_0}_{est}\ (\%)$
Fig. 13(a)	11.86	33.86	22	98.18
Natural Condition Transport Approach — Free-Form Search.				
s/c Specifications Pre-Failure				
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$
s/c Specifications Post-Forced Coast				
$a_0 = 2.205 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$

trajectory that delivers the chaser s/c to the target state to within $5.88\ hours$ of the associated true epoch. For example, a particular target s/c state falls within the chaser s/c AR at $t_{fR} = 33.86\ days$. However, in reality, the target s/c reaches this state at either $t_{fR} - 5.88\ hours$ or $t_{fR} + 5.88\ hours$, and thus, the chaser s/c does not deliver a 'true' rendezvous in terms of initial guess. Also, as seen in Fig. 13(b), the states along the chaser's nominal trajectory are not *reachable* by the chaser's ARs on the recovery path for a given epoch. As a consequence, in this example, the large initial deviation between the nominal and recovery trajectories prohibits the chaser s/c from recovering the nominal path such that rendezvous with the target is possible on the DRO at the nominal terminal epoch of $t_{fN} = 28.86\ days$.

The solution in Fig. 13 offers the state and epoch conditions required to initiate a numerical corrections process to eliminate the discontinuities. The targeting approach is incorporated such that the chaser does meet the required variable state and epoch of the target along the DRO. Note that the reference epochs in Eqn. (5) at the departure condition and arrival orbit are $t_{0R} = 11.86\ DFD$ and $t_{fN} = 28.86\ DFD$, respectively. Also, the departure epoch is held fixed at the departure orbit reference epoch (t_{0R}) in this recovery scenario such that $\varpi_{dep} = 0$. Thus, the last rendezvous time constraint in $\boldsymbol{F}_{CSI_{Rndzv}}$ in Eqn. (5) is modified to $\mathbb{t}_{0R} + \sum_{i=1}^{\mathbb{n}-1} \varpi_i - \mathbb{t}_{fN} - \varpi_{arr}$. Note that $\mathbb{t}_{0R}$ and $\mathbb{t}_{fN}$ are defined as the nondimensional epochs associated with the quantities t_{0R} and t_{fN}, and are constant quantities. The thrust duration of $22.41\ days$ is close



(a) (b)

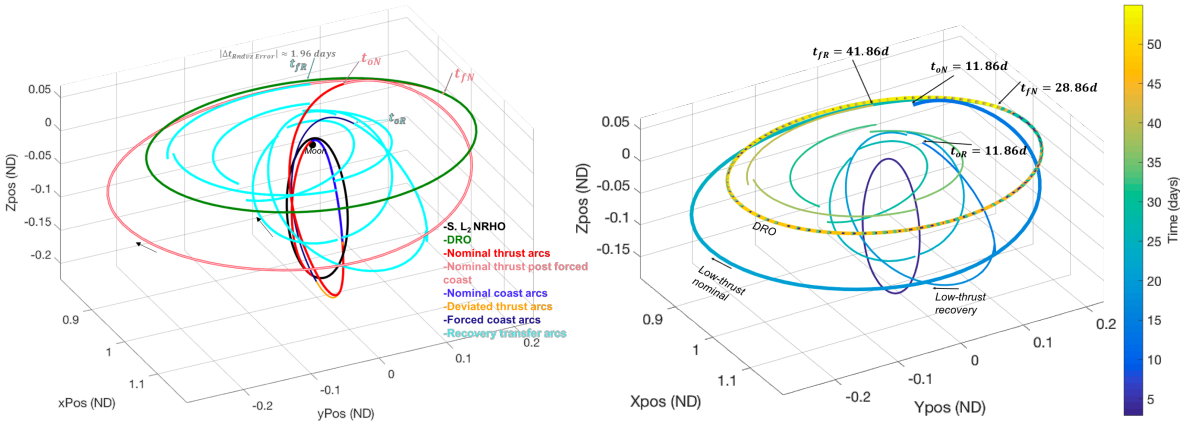
Figure 14: (a) Low-thrust mass-optimal solution for initial guess presented in Fig. 13; (b) Time history along mass-optimal low-thrust solution in (a).

Table 8: Performance metrics for mass optimal CSI rendezvous in Fig. 14

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$TD\ (days)$	$\frac{m_f}{m_0}\ (\%)$
Fig. 14	11.86	35.34	23.48	22.41	98.45
s/c Specifications Pre-Failure					
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	
s/c Specifications Post-Forced Coast					
$a_0 = 2.205 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	

to the transfer time of 22 $days$ estimated by the initial guess, and the mass-optimal transfer transports the chaser s/c to the target at 35.34 DFD with an additional $\approx 1\ day$ of coasting time incorporated. The initial guess for the delivered mass-fraction is also a reliable estimate in this example. The evolution of time/epoch along the continuous low-thrust trajectory is illustrated in Fig. 14(b); this knowledge informs the user of an $\approx 1\ week$ delay from the nominal epoch in reaching the target along the DRO due to the injected failure and the availability of a full recovery thrust magnitude of $\mathbb{T} = 0.22\ N$ from t_{0R} onwards.

The rendezvous example explored in Fig. 13 is extended to a scenario where a diminished thrust magnitude ($\mathbb{T} = 0.11\ N$) is maintained after t_{0R}. The recovery arcs transporting the chaser to the target are plotted in Fig. 15(a). Similar to the $\approx 36\ day$ transfer in Fig. 10, a recovery with a lower thrust magnitude results in a longer TOF compared to that observed in the example in Fig. 13. The look-ahead time is extended from 1 period (as with the example in Fig. 13) to 2 periods to construct a solution for the more complex rendezvous scenario with diminished thrust capabilities. The complicated nature of the transfer leads to a high rendezvous epoch error of $\approx 1.96\ days$ as illustrated in Fig. 15. However, such an initial guess offers all the state and epoch information to seed the optimization process and compute a successful continuous low-thrust rendezvous solution as illustrated in Fig. 16. The mass-optimal trajectory incorporates coast arcs to conserve propellant, and $\approx 11\ days$ of thrusting time is added to the initial guess estimate to accommodate



(a) (b)

Figure 15: Sample rendezvous recovery initial guess for scenario in Fig. 6 for a chaser s/c with $\mathbb{T} = 0.11\ N$ from t_{0R}. (a) Transfer arcs; (b) Transfer arcs colored by epoch (DFD)

Table 9: Performance metrics for s/c rendezvous scenario in Fig. 15

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$\frac{m_f}{m_{0\ est}}\ (\%)$
Fig. 15	11.86	41.86	30	99.12
Natural Condition Transport Approach — Free-Form Search.				
s/c Specifications Pre-Failure				
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$
s/c Specifications Post-Forced Coast				
$a_0 = 1.1025 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.11\ N$		$Isp = 4000\ s$

Table 10: Performance metrics for mass optimal CSI rendezvous in Fig. 16

Scenario	$t_{0R}\ (DFD)$	$t_{fR}\ (DFD)$	$t_{fR} - t_{0R}\ (days)$	$TD\ (days)$	$\frac{m_f}{m_0}\ (\%)$
Fig. 16	11.86	60.86	49	41.42	98.53
s/c Specifications Pre-Failure					
$a_0 = 2.2 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.22\ N$		$Isp = 4000\ s$	
s/c Specifications Post-Forced Coast					
$a_0 = 1.1025 \times 10^{-4}\ m/s^2$		$\mathbb{T} = 0.11\ N$		$Isp = 4000\ s$	

the large state and epoch discontinuities. The chaser s/c is delivered to the target s/c at $t_{fR} = 60.86\ days$ from departure. The extended and elongated deviation of the chaser s/c from the DRO as seen in Fig. 16(b) aids in the phasing required to complete the rendezvous as well. The time-evolution knowledge along this recovery path informs the user of an $\approx 1\ month$ delay from the nominal epoch in reaching the target on the DRO due to the injected failure and the constricted thrust magnitude of $\mathbb{T} = 0.11\ N$ from t_{0R} onwards.

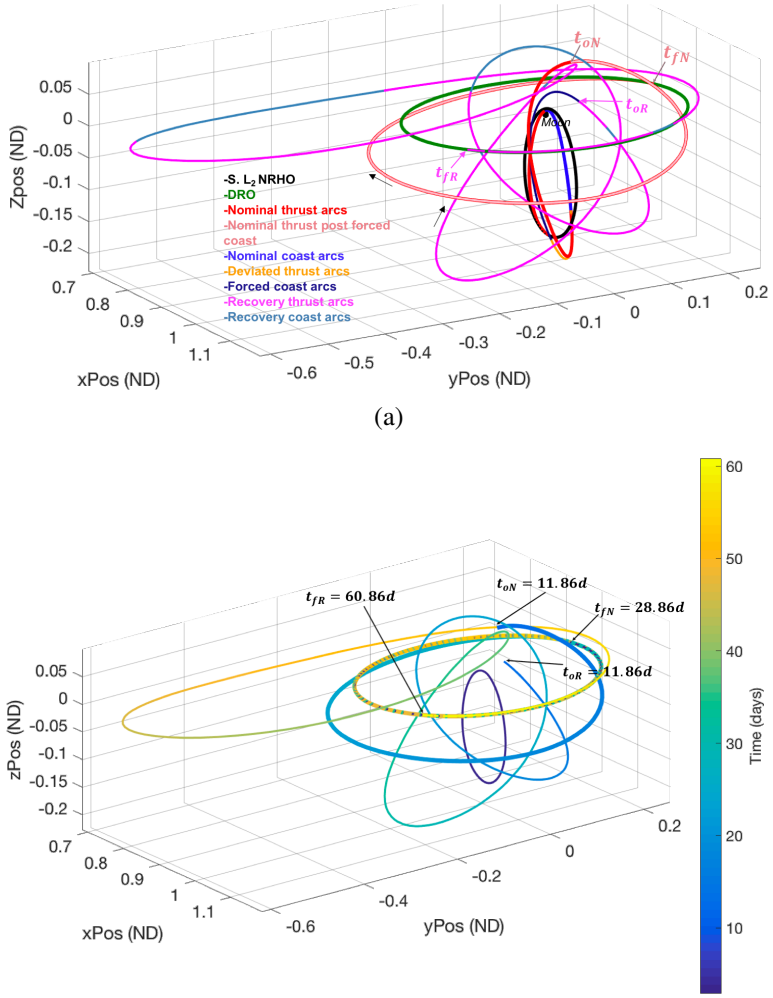


Figure 16: (a) Low-thrust mass-optimal solution for initial guess presented in Fig. 15; (b) Time history along mass-optimal low-thrust solution in (a).

CONCLUDING REMARKS

Trajectory design is a careful balance that juggles diverse constraints, priorities and requirements to enable successful missions. So, preparing for operations contingencies necessitates a rigorous mission design framework to offer solutions for varied spacecraft capabilities and objectives. Maneuvering in close proximity to the Earth and the Moon renders the CR3BP model appropriate for capturing the complex dynamical interactions in this gravitational system, and facilitates pathways for preliminary design. However, in contrast to the two body problem, the CR3BP represents a large state space for which approximations are generally not available. Thus, an infinitely large combinatorial optimization problem emerges — one that is intractable to explore thoroughly via a manual approach. So, an automated search strategy is sought where human efforts are refocussed on defining the constituents of the broader mission recovery goals, and software agents are tasked with undertaking the laborious task of filtering various recovery trajectory scenarios to then deliver only attractive options.

The design of end-to-end trajectories is facilitated by constructing a framework composed of four essential components: (1) simulation of the *reach* of the s/c to assess instantaneous accessible regions (ARs) within the dynamical regime; (2) exploitation of natural flows in the dynamical regime (CR3BP) to serve as potential

waypoints for transfer paths; (3) implementation of automated pathfinding algorithms to sequence the natural arcs and formulate a discontinuous yet complete route to the destination; and (4) transition of the solution to a higher-fidelity engine/dynamical model via a numerical corrections process.

The results from this analysis demonstrate the applicability of the reinforcement learning aided framework to deliver recovery solutions for failure scenarios, including transporting the s/c to a destination orbit or some rendezvous condition. The goals of the pathfinding process are enabled by the flexibility to incorporate varied overarching global objective functions. For example, the TOF-payload delivery trade-space is explored by posing a weighted mass-time priority global cost function. In contrast, a cost function to minimize the time-to-rendezvous error uncovers initial guesses for routes that are numerically corrected to enable a chaser s/c to rendezvous with the states and absolute time of a target object in its destination orbit. Although the HARL algorithm may not deliver globally optimal initial guesses, its parallellizable capability and stochastic nature enable the exploration of broad trade spaces to offset common restrictions such as a narrow convergence radii associated with some traditional trajectory design approaches. Furthermore, the design space is broadened by the free-form search informed solely by natural system dynamics — the agents traverse both chaotic and ordered motion to deliver unconstrained pathfinding. Consequentially, insights are also offered into the dynamical regime including natural structures for more detailed analysis in support of the particular contingency scenario.

The available state and time histories from the sequenced natural arcs are suitable initial guesses for transition to a higher-fidelity engine model. Advantageous thrust and coast locations emerge due to the freedom allowed to construct transfer segments with zero to a maximum bounded position and velocity discontinuities within the ARs. The ability to generalize the design capability across varied engine capabilities (including varied thrust levels due to failure modes) is realized by incorporating the *reach* of the spacecraft via these ARs during the automated search process. Differing implementations also render the methodology flexible to the incorporation of varied constraints during the design process, e.g., targeting desired arrival conditions or maintaining a minimum altitude relative to a primary.

The automated trajectory design framework offers a capability that responds to the challenges associated with a large trade space for mission design. The examples illustrate the potential of the framework to support the design of complex transfers in preparation for contingencies.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the facilities at the Purdue Eliasen Visualization Laboratory, and thank the Purdue Multi-Body Dynamics research group, as well as Jeffrey Stuart at NASA JPL, for valuable insights and suggestions.

REFERENCES

[1] K. Hambleton (Most recent edits), "Deep Space Gateway to Open Opportunities for Distant Destinations," 2018. Published by: National Aeronautics and Space Administration (NASA), `https://www.nasa.gov/feature/deep-space-gateway-to-open-opportunities-for-distant-destinations`, Last Accessed Date: [11 April 2019].

[2] D. C. Folta, N. Bosanac, D. Guzzetti, and K. C. Howell, "An Earth-Moon System Trajectory Design Reference Catalog," *Acta Astronautica*, Vol. 100, 2015, pp. 341–353.

[3] B. F. Kutter and G. F. Sowers, "Cislunar-1000: Transportation supporting a self-sustaining Space Economy," *AIAA Space*, Long Beach, California, USA, 2016.

[4] F. Topputo, M. Vasile, and F. Bernelli-Zazzera, "Low Energy Interplanetary Transfers Exploiting Invariant Manifolds of the Restricted Three Body Problem," *The Journal of Astronautical Sciences*, Vol. 53, No. 4, 2005, pp. 353–372.

[5] G. Mingotti, F. Topputo and F. Bernelli-Zazzera, "Efficient Invariant-Manifold, Low-Thrust Planar Trajectories to the Moon," *Communications in Nonlinear Science and Numerical Simulations*, Vol. 17, No. 2, 2012, pp. 817–831.

[6] D. Guzzetti, N. Bosanac, A. Haapala, K. C. Howell, and D. 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.

[7] M. Vaquero and K. C. Howell, "Leveraging Resonant Orbit Manifolds to Design Transfers between Libration Point Orbits in Multi-Body Regimes," *Journal of Guidance, Control and Dynamics*, Vol. 37, No. 3, 2014, pp. 1143–1157.

[8] A. Haapala and K.C. Howell, "Representations of Higher-Dimensional Poincaré Maps with Application to Spacecraft Trajectory Design," *Acta Astronautica*, Vol. 96, 2014, pp. 23–41.

[9] G. Radice and G. Olmo, "Ant Colony Algorithms for Two-Impulse Interplanetary Trajectory Optimization," *Journal of Guidance, Control and Dynamics*, Vol. 29, No. 6, 2006, pp. 1440–1444.

[10] M. Ceriotti and M. Vasile, "MGA Trajectory Planning With an ACO-Inspired Algorithm," *Acta Astronautica*, Vol. 67, No. 9-10, 2010, pp. 1202–1217.

[11] J. Stuart, K. C. Howell and R. Wilson, "Design of End-to-End Trojan Asteroid Rendezvous Tours Incorporating Potential Scientific Value," *Journal of Spacecraft and Rockets*, Vol. 53, No. 2, 2016, pp. 278–288.

[12] R. Furfaro and R. Linares, "Waypoint-Based Generalized ZEM/ZEV Feedback Guidance for Planetary Landing via a Reinforcement Learning Approach," *3rd IAA Conference on Dynamics and Control of Space Systems*, Moscow, Russia, 2017.

[13] R. Beeson, V. Shah, J. Aurich, and D. Ellison, "Automated Solution of Low Energy Trajectories," *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, Washington, 2017.

[14] S. L. McCarty, L.M. Burke, and M. L. McGuire, "Parallel Monotonic Basin Hopping for Low Thrust Trajectory Optimization," *AIAA SciTech Forum*, Kissimmee, Florida, USA, 2018.

[15] P. Witsberger and J. Longuski, "Interplanetary Trajectory Design Using a Recurrent Neural Network and Genetic Algorithm," *AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, Utah, USA, 2018.

[16] A. Das-Stuart, K. C. Howell, and D. Folta, "Rapid Trajectory Design in Complex Environments Enabled Via Supervised and Reinforcement Learning Strategies," *69th International Astronautical Congress*, Bremen, Germany, 2018.

[17] S. De Smet, D. J. Scheeres, and J. S. Parker, "Systematic Exploration of Solar Gravity Driven Orbital Transfers in the Martian System using Artificial Neural Networks," *AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, Utah, USA, 2018.

[18] G. A. Tsirogiannis, "A Graph Based Method for Mission Design," *Celestial Mechanics and Dynamical Astronomy*, Vol. 114, No. 4, 2012, pp. 353–363.

[19] E. Trumbauer and B. Villac, "Heuristic Search-Based Framework for Onboard Trajectory Redesign," *Journal of Guidance, Control and Dynamics*, Vol. 37, No. 1, 2014, pp. 164–175.

[20] N. Parrish, "A* Pathfinding for Continuous-Thrust Trajectory Optimization," *AAS 37th Annual Guidance & Control Conference*, Breckenridge, Colorado, USA, 2014.

[21] D. Miller and R. Linares, "Low-Thrust Optimal Control via Reinforcement Learning," *29th AAS/AIAA Space Flight Mechanics Meeting*, Kaánapali, Hawaii, USA, 2019.

[22] A. Das-Stuart, K. C. Howell, and D. Folta, "Rapid Trajectory Design in Complex Environments Enabled by Reinforcement Learning and Exact Methods," *Acta Astronautica (In Press: 18 April 2019)*.

[23] V. Szebehely, *Theory of Orbits*. Connecticut, USA: Academic Press, first ed., 1967.

[24] NASA, "NASA - Ion Propulsion," 2017. `https://www.nasa.gov/centers/glenn/about/fs21grc.html`, Last Accessed Date: [03 August 2019].

[25] R. P. Russell, "Primer Vector Theory Applied to Global Low-Thrust Trade Studies," *Journal of Guidance, Control and Dynamics*, Vol. 30, No. 2, 2007, pp. 460–472.

[26] D. L. Poole and A. K. Mackworth, *Artificial Intelligence, Foundations of Computational Agents*. New York, USA: Cambridge University Press, 2010.

[27] A. Das, *Artificial Intelligence Aided Rapid Trajectory Design in Complex Dynamical Environments*. PhD dissertation, School of Aeronautics and Astronautics, Purdue University, 2019.

[28] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, Massachusetts, USA: MIT Press, second ed., 2018.

[29] R. A. C Bianchi, C. H. C. Ribeiro, and A. H. R. Costa, "On the Relation between Ant Colony Optimization and Heuristically Accelerated Reinforcement Learning," *1st International Workshop on Hybrid Control of Autonomous Systems - Integrating Learning, Deliberation and Reactive Control*, Pasadena, California, USA, 2009.

[30] J. A. Sims and S. N. Flanagan, "Preliminary Design of Low-Thrust Interplanetary Missions," *Technial Report*, NASA Jet Propulsion Laboratory, California Institute of Technology, 1997.

[31] D. J. Grebow, G. J. Whiffen, D. Han, and B. M. Kennedy, "Dawn Safing Approach to Ceres Re-Design," *AIAA/AAS Astrodynamics Specialist Conference, AIAA SPACE Forum*, Long Beach, California, USA, 2016.

[32] T. Imken, T. Randolph, M. DiNicola, and A. Nicholas, "Modeling Spacecraft Safe Mode Events," *IEEE Aerospace Conference*, Big Sky, Montana, USA, 2018.