AIAA SciTech Forum
8-12 January 2024, Orlando, FL
AIAA SCITECH 2024 Forum

10.2514/6.2024-0848

Feature Learning Based Optimal Control Method for Cislunar Trajectory Design

Roha Gul*, Mitchell Dominguez †, Ran Dai‡, Kathleen C. Howell §
Purdue University, West Lafayette, IN, 47907

The paper investigates close range rendezvous, nearby orbital transfer, and collision avoidance maneuvers for a chaser spacecraft moving relative to a target in a 9:2 L_2 Near-Rectilinear Halo Orbit (NRHO). A feature learning optimal control method (L-OCM) is proposed to solve optimal control problems corresponding to the maneuvers to minimize fuel consumption. First, the optimal control problems for each operation are formulated using nonlinear relative dynamics for the circular restricted three-body problem (CR3BP), where the controller is a two-finite-burn maneuver. Then, a nonlinear programming (NLP) solver is used to solve the problems offline for a range of initial conditions. The NLP solutions are also compared to solutions from existing methods, i.e., differential corrections. The set of solutions is used to generate datasets with initial conditions as inputs and the identified critical features as outputs. These features are extracted from the optimal solution and are used to reconstruct the entire optimal solution. A deep neural network is trained off-line to map the complex, nonlinear relationship between the inputs and outputs, and then implemented to find on-line solutions to any initial condition. The L-OCM method provides fuel-optimal, real-time solutions that can be implemented by a spacecraft performing operations in cislunar space.

I. Introduction

The advent of NASA's Artemis program has brought much attention to the notion of a sustained human presence in cislunar space, which is rapidly becoming the focus for expansion of long-term space infrastructure. A continuous human presence beyond low Earth orbit is a key stepping stone towards the feasibility of human exploration of the Moon, Mars, and beyond. One concept to achieve this presence is the NASA Gateway, an integral piece of the Artemis program. Gateway will be maintained in a cislunar orbit and serve as both a permanent human habitat as well as a staging point for future space exploration missions and activities. Over the course of its operational lifetime, Gateway is also expected to be a part of a dynamic infrastructure involving multiple other spacecraft. These spacecraft must perform a myriad of operations including rendezvous and docking, stationkeeping, collision avoidance, and other such maneuvers. Owing to limitations for fuel capacity and computational capabilities onboard the spacecraft, minimum fuel-optimal, on-line solutions to generate control commands for these maneuvers are necessary.

The focus of this investigation is close range rendezvous (CRR), nearby orbit transfers, and collision avoidance maneuvers for spacecraft on or in the vicinity of the baseline orbit for Gateway, the 9:2 synodic resonant Earth-Moon southern L_2 Near Rectilinear Halo Orbit (NRHO). These maneuvers are accomplished by controlling a chaser spacecraft to follow a path relative to a passive target. The CRR maneuver is accomplished by controlling the chaser to intercept the target, i.e., by bringing the chaser's relative position and velocity with respect to the target to zero. The nearby transfer uses rephasing maneuvers where the chaser transfers to the same orbit as the target at a desired phase either ahead or behind. The collision avoidance maneuver also uses rephasing to bring the chaser ahead in phase relative to the target, while avoiding a collision sphere around the target.

The Cartesian representation of the equations of relative motion for a chaser spacecraft in orbit near a passive target in a two-body orbit were first described by the Hill-Clohessy-Wiltshire (HCW) equations [1], and this formulation remains widely in use today. The HCW equations are used to deliver optimal trajectories for rendezvous, assuming impulsive maneuvers involving both fixed and free final time conditions [2–4]. Additionally, extensions to the HCW equations involve parameterization of the relative motion in terms of relative orbital elements [5] or introduce approximations

*Graduate Research Associate, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47907
†Graduate Research Associate, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47907
‡Associate Professor, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47907
§Hsu Lo Distinguished Professor, School of Aeronautics and Astronautics, 701 W. Stadium Ave., West Lafayette, IN, 47907

for the state transition matrices for elliptical orbits [6]. However, these and other related strategies all generally fail to accurately represent relative dynamics in the cislunar region.

The major difference between the two-body orbits modeled by the HCW equations and trajectories in the multi-body dynamical environment of cislunar space is that there is no underlying analytical representation of the orbit. However, equations of relative motion are still formulated in a manner analogous to the HCW equations [7] within the framework of the circular-restricted three-body problem (CR3BP). Linear variational approaches to the cislunar rendezvous problem have been undertaken, where the method of differential corrections is used to construct rendezvous and formation flying scenarios in the vicinity of the Gateway orbit [8]. The method of differential corrections has also been employed within the cislunar environment to compute families of locally optimal rendezvous and rephasing maneuvers [9], and also to compute passively safe rendezvous trajectories [10]. Furthermore, linearized dynamics within the CR3BP are the basis for computing debris avoidance maneuvers [11] and producing transfers between members of the same orbit family [12]. In addition, two and three impulse strategies for phasing maneuvers to transfer to nearby halo orbits have also been implemented using shooting methods in the CR3BP [13].

There are also recent advances in the use of modern control techniques for spacecraft applications in cislunar space. Linear optimal control techniques, such as linear quadratic regulator (LQR) and linear model predictive control (MPC), have been used for rendezvous and loitering problems using linearized dynamics in the CR3BP [14–16]. Reinforcement Learning (RL), a branch of Machine Learning (ML), is also employed to estimate initial guesses for more conventional iterative targeting methods for the purposes of stationkeeping [17].

However, there exists a gap in the literature when it comes to using nonlinear dynamics and non-impulsive maneuvers in the CR3BP. Linearized dynamics are an approximation, and thus used as a first initial guess towards constructing a trajectory giving way to more high-fidelity methods. Though linearization provides a good initial guess, the equations fail to capture the underlying dynamics for longer times of flight when moving further away from a reference path. Moreover, most targeting methods used to solve these linear problems cannot include or modify path constraints with ease, which is required for maneuvers like collision avoidance. The nonlinear relative dynamics between the chaser and target in the CR3BP employs a control acceleration required by the chaser to maneuver itself. Using control acceleration instead of an impulsive maneuver reformulates the relative trajectory problem by implementing a finite burn maneuver. The finite burn assumes a constant burn velocity over a finite burn time, which is a more accurate representation of these maneuvers and their implementation in real-world applications. By implementing the nonlinear relative equations of motion and a finite burn maneuver to formulate a nonlinear optimal control problem (OCP), this investigation aims to quantify and examine the nonlinear behavior of a spacecraft in cislunar space.

This investigation aims to address two important issues being considered for the Artemis program for control system requirements onboard the spacecraft. The first being limited amount of fuel, and the second being a need for control systems that have lower computational times and storage requirements. Since a limited amount of propellant is available for all maneuvers required by the spacecraft, minimum fuel consumption (or minimum burn velocity) is desired. Hence, the optimal trajectory problem is also a minimum fuel nonlinear OCP, set up as two-finite-burn maneuvers, modeled using nonlinear relative dynamics between chaser and target. Unlike the relative two-body problem, the relative three-body problem does not possess an analytical solution, and must be solved using numerical methods. This limitation, accompanied by the inclusion of highly nonlinear dynamics, presents a challenging task to determine the fuel-optimal control maneuvers onboard. It requires computational resources needed for other tasks. Therefore, an optimal guidance algorithm that can generate real-time fuel-optimal control solutions needed to perform multiple operations is necessary. The traditional numerical approaches to solve such types of problems, such as direct collocation formulations solved using nonlinear programming (NLP) cannot guarantee robust convergence and real-time computational performance. Hence, a feature learning optimal control method (L-OCM) is proposed to solve the fuel-optimal control problems for real-time cislunar trajectory optimization.

The proposed method utilizes a dataset generated from NLP solutions of the OCP mentioned above to train a Deep Neural Network (DNN). The DNN maps the complex, nonlinear relationship between the inputs representing initial conditions and outputs representing key features of the optimal solution. These features are used to reconstruct the entire optimal solution. Although DNNs have been applied in a variety of guidance and control problems to identify the underlying nonlinear relationship that maps from system inputs to outputs [18], the extensive efforts required for training a large dataset are time and resource-consuming. By identifying features that represent the entire optimal solution, the learning space is condensed, resulting in a substantially reduced computational load and thereby facilitating real-time computation [19]. To validate the improved computational performance, results obtained from the proposed L-OCM are compared to solutions from NLP.

In the following, Section II introduces the dynamic models in the restricted three-body problem. Section III describes

the formulation of the optimal control problems for close range rendezvous, nearby transfer, and collision avoidance. Section IV introduces the feature learning optimal control method (L-OCM). Section V provides simulation results for the problems solved using linear targeting methods and nonlinear programming, and training and testing results of the DNNs.

II. Dynamical Models

This investigation employs the CR3BP to model spacecraft dynamics in cislunar space. Relative dynamics are modeled with nonlinear and linear variational equations derived from the CR3BP equations of motion. Additionally, spacecraft maneuvers are modeled as either impulsive maneuvers that instantaneously change velocity, or as finite burns modeled as control accelerations augmenting the CR3BP dynamics.

A. Circular Restricted Three-Body Problem

When using the CR3BP to model cislunar space, two primary bodies, the Earth (P_1) and the Moon (P_2), are modeled as point masses assumed to be moving in circular orbits about their common barycenter (B). The motion of P_1 and P_2 is denoted as the primary system. The spacecraft (P_3) with mass m is under the gravitational influence of the primary bodies, with the masses of the three bodies assumed to obey $m \ll M_2 < M_1$. Since the mass of the spacecraft is assumed to be infinitesimally small relative to the mass of each primary, it is assumed that the spacecraft does not affect the motion of the primary system.

A reference frame $\mathcal{S} = \{\hat{x}_s, \hat{y}_s, \hat{z}_s\}$, i.e., the synodic frame, that is centered at B and rotates with the primary system, describes the motion of the spacecraft. The unit vector $\hat{x}_s$ is directed from the barycenter towards P_2, $\hat{z}_s$ is parallel to the angular momentum direction of the primary system, and $\hat{y}_s$ completes the dextral orthonormal triad. Additionally, an inertial frame $\mathcal{I} = \{\hat{X}, \hat{Y}, \hat{Z}\}$ is defined, with its origin at B. Both the span of $\hat{x}_s$ and $\hat{y}_s$ and the span of $\hat{X}$ and $\hat{Y}$ define the orbital plane of the primary system, and $\hat{z}_s = \hat{Z}$. Since the primary system is circular, the synodic frame rotates at a constant angular velocity of $\omega_{\mathcal{S}/\mathcal{I}} = n\hat{z}_s$ relative to the inertial frame, where n is the mean motion of the circular primary system. A diagram of the CR3BP depicting the relationship between the $\mathcal{S}$ and $\mathcal{I}$ frames is shown in Fig. 1.

It is advantageous to express the motion of P_3 in terms of non-dimensional coordinates. For the purposes of nondimensionalization, characteristic quantities are introduced. The characteristic mass (m^*) is the sum of the primary masses, the characteristic length (l^*) is the distance between the primaries, and the characteristic time (t^*) is the inverse of the primary mean motion. Using the nondimensionalized mass, any CR3BP system can then be characterized by a mass ratio $\mu = M_2/m^*$, and the distance from B to P_1 and B to P_2 can be written as $R_1 = -\mu$ and $R_2 = 1 - \mu$, respectively. For the Earth-Moon system, the value of μ is roughly 0.01215. In nondimensionalized synodic frame coordinates, the position of the spacecraft with respect to the barycenter is

$$\boldsymbol{r}_3 = x\hat{x}_s + y\hat{y}_s + z\hat{z}_s \tag{1}$$

and the distance of the spacecraft from each of P_1 and P_2, respectively denoted d and r, is

$$d = \sqrt{(x+\mu)^2 + y^2 + z^2}, \qquad r = \sqrt{(x-1+\mu)^2 + y^2 + z^2} \tag{2}$$

The second-order scalar nondimensional equations of motion of the spacecraft in the synodic frame are then

$$\ddot{x} = 2\dot{y} + \frac{\partial U^*}{\partial x}, \qquad \ddot{y} = -2\dot{x} + \frac{\partial U^*}{\partial y}, \qquad \ddot{z} = \frac{\partial U^*}{\partial z} \tag{3}$$

where U^* is a pseudo-potential function dependent only on the position of the spacecraft. The pseudo-potential is defined as

$$U^* = \frac{1-\mu}{d} + \frac{\mu}{r} + \frac{1}{2}\left(x^2 + y^2\right) \tag{4}$$

Eqs. (3) admit one integral of the motion, known as the Jacobi constant, which is defined as

$$C = 2U^* - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) \tag{5}$$

As can be seen from Eq. (5), the Jacobi constant is dependent on the position of the spacecraft and the magnitude of the rotating frame velocity, and thus can be considered an "energy-like" term. Any thrusting maneuver performed by the

spacecraft will change its energy, whether the maneuver is modeled as an instantaneous change in velocity or as an acceleration added to the equations of motion. The theoretical minimum instantaneous change in velocity, or Δv_{th}, to bridge an energy gap from C_i to C_f at any given point is computed with

$$\Delta v_{th} = \sqrt{2U_i^* - C_f} - v_i \tag{6}$$

where subscript i indicates the initial values, and f indicates the final values. The calculation of theoretical minimum Δv assumes that the impulsive maneuver is applied tangentially to the synodic frame state. While the optimal maneuvers that are computed in this investigation are frequently nontangential, Δv_{th} still provides a useful benchmark with which to compare results.

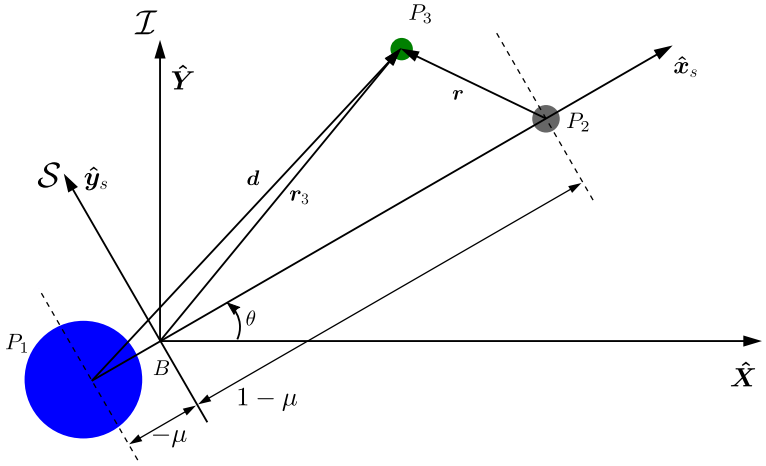


Fig. 1 Three-body system in the synodic frame

B. Periodic Orbits in the CR3BP

The baseline orbit for NASA's Gateway mission is an Earth-Moon southern L_2 NRHO with a period of roughly 6.5 days, such that it exhibits a 9:2 lunar synodic resonance. The spacecraft undergoes 9 orbital revolutions for every 2 synodic lunar periods. Over the course of its mission, the station must perform a variety of maneuvers to rephase, avoid collisions, and maintain its orbit. Additionally, any spacecraft that interacts with Gateway must also perform rendezvous, loitering, and collision avoidance maneuvers. For these operations, the natural dynamical structures that are present in the vicinity of the 9:2 NRHO, including other nearby members of the southern L_2 halo family, must be utilized.

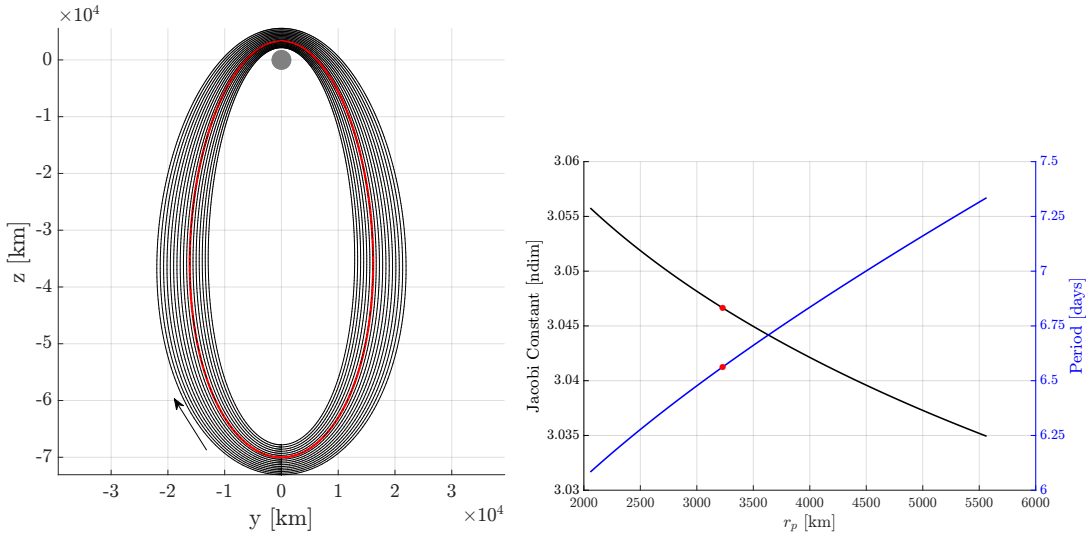
Periodic orbits in the CR3BP exist in families, where properties such as orbital period and energy level vary smoothly from family member to family member. Fig. 2 demonstrates trajectories of several L_2 halos in the vicinity of the 9:2 NRHO as well as the evolution of orbital period and energy level throughout this range of orbits. Nearby periodic orbits, in particular, offer predictable methods of rephasing spacecraft, as differences in orbital period naturally drift spacecraft apart in phase. The advantages of rephasing on periodic orbits at greater period differences are offset by the costs to span energy differences between the orbits. In this investigation, optimal transfers between nearby periodic orbits are explored.

C. Nonlinear Relative Dynamics

The scenarios examined in this investigation feature the motion of two spacecraft, known as the target and chaser, within the Earth-Moon CR3BP problem. The target is assumed to be passive with an accurately represented orbit. The problem can then be categorized as active-passive rendezvous, where the chaser spacecraft actively tries to rendezvous with a passive target spacecraft. Vectors $\boldsymbol{r}_t$ and $\boldsymbol{r}_c$ denote position vectors for the target and chaser spacecraft with respect to the barycenter, respectively.

The relative dynamics of the chaser are modeled using the variational equations based on the CR3BP equations of motion listed in Eq. (3). The relative position of the chaser with respect to the target is defined as

$$\boldsymbol{\rho} = \boldsymbol{r}_c - \boldsymbol{r}_t = \rho_x \hat{x}_s + \rho_y \hat{y}_s + \rho_z \hat{z}_s \tag{7}$$



(a) (b)

Fig. 2 (a) Selected members of the L_2 halo family in the vicinity of the 9:2 NRHO. (b) Evolution of Jacobi constant and period within the selected L_2 halo orbits as functions of perilune radius. In each figure, red indicates the 9:2 NRHO.

The nonlinear equations of relative motion for the chaser, also known as the nonlinear variational equations, are then

$$\ddot{\boldsymbol{\rho}} = \ddot{\boldsymbol{r}}_c - \ddot{\boldsymbol{r}}_t + \boldsymbol{u} \tag{8}$$

where each of $\ddot{\boldsymbol{r}}_c$ and $\ddot{\boldsymbol{r}}_t$ are given by Eq. (3) and $\boldsymbol{u} = u_x\hat{x}_s + u_y\hat{y}_s + u_z\hat{z}_s$ represents the control acceleration applied by the chaser as a function of time. The components of $\boldsymbol{u}$ in the $\mathcal{S}$ frame are defined as

$$u_x = \|\boldsymbol{u}\|\cos\theta\cos\phi \qquad u_y = \|\boldsymbol{u}\|\cos\theta\sin\phi \qquad u_z = \|\boldsymbol{u}\|\sin\theta \tag{9}$$

where θ represents the angle between the control vector $\boldsymbol{u}$ and the $\hat{x}_s$-$\hat{y}_s$ plane and ϕ represents the angle between the projection of $\boldsymbol{u}$ on the $\hat{x}_s$-$\hat{y}_s$ plane and $\hat{x}_s$ axis, as apparent in Fig. 3.

The nonlinear equations of relative motion for the chaser in Eq. (8) are then numerically integrated, with the absolute equations of motion for the target from Eq. (3), to simultaneously produce the absolute trajectory of the target spacecraft and the relative trajectory of the chaser. The relative state vector is $\boldsymbol{s} = [\boldsymbol{\rho}, \dot{\boldsymbol{\rho}}]^T$ and the relative nonlinear dynamics can be expressed as $\dot{\boldsymbol{s}} = \boldsymbol{f}(\boldsymbol{s}, \boldsymbol{u}, \mu)$.

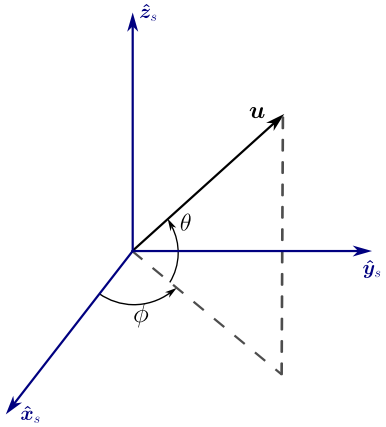


Fig. 3 Control vector in the synodic frame

D. Linear Relative Dynamics

The nonlinear equations of relative motion are also linearized relative to the target trajectory to yield the linear variational equations. From the Taylor expansion about the target trajectory, the linear variational equations, i.e., the linear equations of relative motion, are derived in the form

$$\dot{\boldsymbol{\rho}}(t) = A(t)\boldsymbol{\rho}(t) \tag{10}$$

Since Eq. (10) is a linear, time-varying system, where $A(t)$ is the Jacobian matrix of the CR3BP dynamics evaluated on the reference trajectory, its solutions depend only on the initial conditions. The general solution to Eq. (10) is

$$\boldsymbol{\rho}(t) = \phi(t, t_0)\boldsymbol{\rho}(t_0) \tag{11}$$

where $\phi(t, t_0)$ is the state transition matrix (STM) that predicts relative states at the time of interest t given an initial state at time t_0. The STM is propagated using

$$\dot{\phi} = A\phi \tag{12}$$

with initial condition $\phi(t_0, t_0) = I_6$, where I_6 is the 6×6 identity matrix. From a practical standpoint, the STM is obtained by integrating Eq. (12) alongside the nonlinear equations of motion in Eq. (3).

E. Differential Corrections

A free-variable/constraint formulation of a differential corrections scheme is implemented to locate feasible solutions for scenarios such as the rendezvous problem. This targeting method is set up by defining an $n \times 1$ free variable vector and an $m \times 1$ constraint vector as $\boldsymbol{X} = [X_1, \ldots, X_n]^T$ and $\boldsymbol{F}(\boldsymbol{X}) = [F_1(\boldsymbol{X}), \ldots, F_m(\boldsymbol{X})]^T$. The $m \times n$ Jacobian matrix $DF(\boldsymbol{X}) = \partial \boldsymbol{F} / \partial \boldsymbol{X}$ is then used to iteratively update the free variables until $\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{0}$ within a user-defined numerical tolerance. If $m = n$, then the inverse of $DF(\boldsymbol{X})$ can be directly computed, and the free variable vector is updated using

$$\boldsymbol{X}_i = \boldsymbol{X}_{i-1} - DF(\boldsymbol{X}_{i-1})^{-1}\boldsymbol{F}(\boldsymbol{X}_{i-1}) \tag{13}$$

However, if $m < n$, then the problem is underconstrained and the minimum-norm solution for the free variable update is computed using

$$\boldsymbol{X}_i = \boldsymbol{X}_{i-1} - DF(\boldsymbol{X}_{i-1})^T \left[DF(\boldsymbol{X}_{i-1}) \cdot DF(\boldsymbol{X}_{i-1})^T\right]^{-1} \boldsymbol{F}(\boldsymbol{X}_{i-1}) \tag{14}$$

If $m > n$, then the problem is considered overconstrained and will exhibit great reluctance to converge on a solution. The partial derivatives in the DF matrix are constructed analytically or numerically.

This algorithm, of course, is a linear approach that requires an adequate initial guess to converge on a feasible solution that satisfies the constraints. Additionally, there is no general guarantee of optimality for solutions. The differential corrections strategy produces locally optimal solutions, however, by inserting an optimality constraint, as demonstrated by McCarthy et al. [9]. Assume that a non-optimal targeting scheme is set up such that it has n free variables and $m = n - 1$ constraints contained in $\boldsymbol{X}_a$ and $\boldsymbol{F}_a$. The associated Jacobian matrix DF_a of the targeter has dimension $(n-1) \times (n)$. To ensure local optimality, an additional constraint is added to $\boldsymbol{F}_a$ such that the targeting problem becomes

$$\boldsymbol{X}_b = \boldsymbol{X}_a, \qquad \boldsymbol{F}_b(\boldsymbol{X}_a) = \begin{bmatrix} \boldsymbol{F}_a \\ F_{opt} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_a \\ \nabla f \cdot \text{null}(DF_a) \end{bmatrix} = \boldsymbol{0} \tag{15}$$

where $f(\boldsymbol{X}_a)$ is a scalar cost function to be minimized and $\text{null}(DF_a)$ is the 1-dimensional nullspace of DF_a. The nullspace of DF_a represents the linear approximation of free variable vectors $\boldsymbol{X}_a$ that satisfy the constraints $\boldsymbol{F}_a$, and F_{opt} is driven to zero when $\boldsymbol{X}_a$ is at a value such that the gradient of the cost function is orthogonal to the constraint manifold, thus ensuring that the solution of the targeting problem lies at a critical point. It is the responsibility of the trajectory designer to evaluate this solution to ensure that it lies at the desired type of critical point (local minimum or local maximum). The targeter for local optimality is, thus, a square problem, with n free variables and $m = n$ constraints.

Once a solution to a differential corrections problem is determined, nearby solutions are located via pseudo-arclength continuation (PALC). As with the optimality targeter, assume that the original targeting scheme is underconstrained, with n free variables and $m = n - 1$ constraints contained in $\boldsymbol{X}_c$ and $\boldsymbol{F}_c$. The associated Jacobian matrix DF_c of the targeter has dimension $(n-1) \times (n)$. An additional free variable and constraint are added to the targeting problem, such that

$$\boldsymbol{X}_d = \boldsymbol{X}_c, \qquad \boldsymbol{F}(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{F}_c \\ F_{palc} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_c \\ \left(\boldsymbol{X_i} - \boldsymbol{X}^*_{i-1}\right)^T \mathcal{N}_{i-1} - \delta s \end{bmatrix} = \boldsymbol{0} \tag{16}$$

where $\boldsymbol{X}^*_{i-1}$ is the previously converged solution and δs is a user-provided step size. The one-dimensional nullspace $\mathcal{N}_{i-1}$ of $D\boldsymbol{F}_c$ is the matrix of partial derivatives of $\boldsymbol{F}_c$ with respect to $\boldsymbol{X}^*_{i-1}$. The choice of X_{palc} is problem-dependent, and affects the solutions that are obtained via PALC. From Eq. (16), it is evident that the targeter used for PALC is also a square problem, with n free variables and $m = n$ constraints. Additionally, note that, for problems with sufficient quantity of free variables and constraints, PALC is effective to continue families of optimal solutions that were produced using Eq. (15). The advantage of the differential corrections method is that, given a reasonable initial guess, a family of optimal solutions can be located, with each solution giving insight into nearby solutions. In this way, the solution space for a given problem is explored using a relatively simple and transparent process without reliance on individual point solutions.

There are two primary limitations of the differential corrections method. First, the optimality constraint only ensures a locally optimal solution. While this issue is somewhat mitigated by the ability to explore the solution space with PALC, it is still infeasible to ensure that a given solution is truly the globally optimal. The other major drawback of differential corrections is due to the linear update for the free variables, as in Eqs. (13) and (14). While many simple constraints, such as continuity and time of flight constraints, are generally conducive to providing linear updates for the free variables, more complicated constraints could exhibit much more nonlinear behaviors that hinder the convergence of the differential corrections method. In such cases, more sophisticated methods may be necessary to provide the desired solutions.

III. Formulation of the Optimal Control Problem

The most basic OCP considered is the rendezvous between the chaser and target spacecraft in the CR3BP. Both the nearby transfer and the collision avoidance problems are an extension of the rendezvous problem with additional path constraints and/or different boundary conditions. The rendezvous OCP aims to find a finite control law, in this case, an optimal burn velocity for a two-finite-burn maneuver required for the chaser to intercept the target for a fixed time of flight, denoted by ToF. The state of each spacecraft at any time t is defined as $\boldsymbol{s} = [\boldsymbol{\rho}, \dot{\boldsymbol{\rho}}]^T \in \mathbb{R}^6$, where $\boldsymbol{\rho}$ and $\dot{\boldsymbol{\rho}}$ are the relative position and velocity vector respectively, as measured from the synodic frame. This is a two-point boundary value problem with an initial relative state $\boldsymbol{s}_0$ between the chaser and target at initial time t_0. At t_f, the rendezvous condition satisfies $\dot{\boldsymbol{\rho}}(t_f) = \boldsymbol{\rho}(t_f) = \boldsymbol{0}$. The total burn velocity, denoted by $\Delta v = \Delta v_1 + \Delta v_2$, which represents the integral of the control acceleration over the operational time period, i.e., $\Delta v \int_{t_0}^{t_f} \boldsymbol{u}\, dt$. The final time t_f is given, and the burn time interval Δt is also fixed, where time intervals for both burns are assumed to be equal $\Delta t_1 = \Delta t_2 = \Delta t$. The first burn is to initialize the rendezvous, and the second is to bring excess relative velocity to zero at the point of interception of the spacecraft. Hence, the control maneuver is only applied at t_0 and $t_f - \Delta t$. The chaser is passive for the remainder of the time. The rendezvous problem is formulated as

$$\min_{u} \int_{t_0}^{t_f} \boldsymbol{u}\, dt \tag{17a}$$

$$\text{subject to } \dot{\boldsymbol{s}} = \boldsymbol{f}(\boldsymbol{s}, \boldsymbol{u}, \mu) \tag{17b}$$

$$\boldsymbol{s}(t_0) = \boldsymbol{s_0} \tag{17c}$$

$$\boldsymbol{s}(t_f) = \boldsymbol{0} \tag{17d}$$

$$u_{min} \leq ||\boldsymbol{u}|| \leq u_{max} \tag{17e}$$

where Eq. (17a) is the objective function, Eq. (17b) is the system dynamics, Eqs. (17c)-(17d) are boundary conditions at the initial and final time, and Eq. (17e) is the upper and lower bound on control as u_{max} and u_{min}, respectively.

For the nearby transfer problem, a two-finite-burn phasing maneuver is implemented using the nondimensional phase variable θ, defined as the fraction of orbit period elapsed since the last perilune passage:

$$\theta = \frac{t - t_p}{\mathbb{P}} \in [0, 1) \tag{18}$$

where $t \in [t_0 \quad t_f]$, t_p is the time at perilune, and $\mathbb{P}$ is the orbital time period. The phase indicates the state of the spacecraft in orbit, where θ_{c_0} defines the state of the chaser at t_0 as $\boldsymbol{x}_{c_0} = \left[\boldsymbol{r}_{c_0}, \boldsymbol{v}_{c_0}\right]^T$ in the nearby orbit, θ_{t_0} defines the state of the target at t_0 as $\boldsymbol{x}_{t_0} = \left[\boldsymbol{r}_{t_0}, \boldsymbol{v}_{t_0}\right]^T$, θ_{c_f} defines the state of the chaser at t_f as $\boldsymbol{x}_{c_f} = \left[\boldsymbol{r}_{c_f}, \boldsymbol{v}_{c_f}\right]^T$, and θ_{t_f} defines state of chaser at t_f as $\boldsymbol{x}_{t_f} = \left[\boldsymbol{r}_{t_f}, \boldsymbol{v}_{t_f}\right]^T$. Hence, the problem is a two-point boundary value problem with fixed final

time t_f. Via phasing maneuver, the chaser is free to insert itself into the NRHO either ahead, behind, or on the target (rendezvous conditions). This decision is defined by a new fixed variable δt, the phasing parameter, which is used to determine the phase of the chaser at t_f and nondimensionalized by t^*. Given the initial states and time of flight ToF the state of the chaser at t_f is determined by

$$t_0 = \theta_{t_0} \mathbb{P}_{NRHO} + t_p \tag{19a}$$

$$t_f = t_0 + ToF \tag{19b}$$

$$\theta_{c_f} = \frac{t_f + \delta t - t_p}{\mathbb{P}_{NRHO}} \tag{19c}$$

where a $\delta t = 0$ implies rendezvous conditions $\boldsymbol{x}_{c_f} = \boldsymbol{x}_{t_f}$. The OCP stays the same except for the terminal conditions that are redefined as $\boldsymbol{s}(t_f) = \boldsymbol{s}_f$.

For the collision avoidance problem, the same two-finite-burn phasing maneuver is implemented but only for the NRHO. The chaser spacecraft implements a maneuver to change its phase in the orbit and reinsert itself ahead of the target back into the orbit, under an additional path constraint. The target is further modelled as a sphere of avoidance with a radius r_{avoid} and center $c_{avoid} = [x_t, y_t, z_t]^T$, where x_t, y_t, z_t are the position vector components for the target measured in the synodic frame. For any point during the time of flight, the chaser is not allowed to enter the collision sphere. This is modelled as the path constraint $\|\boldsymbol{\rho}\| - r_{avoid} \geq 0$. Then, the corresponding OCP is a two-point boundary value problem with initial and final states and a fixed value of δt, which will always place the chaser ahead of the target at t_f.

IV. Feature Learning based Optimal Control Method

An L-OCM is proposed to solve the cislunar trajectory design problems. This method employs DNNs to map the underlying behavior between the initial conditions and the control accelerations of the finite burn maneuver. The CR3BP relative dynamics are highly nonlinear, which prevents finding an analytical solution when considering CR3BP relative dynamics in a trajectory optimization problem. With the addition of a minimization objective, solving the nonlinear OCP in real-time faces the challenges of heavy computational load and time restraints to the missions. When DNNs are employed to solve OCPs, they typically require large datasets to represent the entire optimal solutions, e.g., system states along the optimized trajectory, which gives rise to computationally expensive learning spaces. However, if the optimal control solution could be described by a few critical features, then learning only these features can recreate the entire solution. Learning the identified features instead of the entire optimal solution significantly reduces the computational load by reducing the learning space and storage, which are the claimed advantages of the proposed L-OCM.

The schematic in Fig. 4 explains the overall framework for this method. The L-OCM consists of two stages. The first, off-line stage includes generating a dataset and training the DNN. The data set is comprised of the discrete optimal solutions to the nonlinear OCP. Specifically, it is a pair between the initial states (conditions) and the set of features identified for the optimal control solution. The nonlinear OCP in Eqs. 17a-17e is solved using an NLP solver for a range of initial conditions, and the features of the optimal solution are identified and extracted from that solution. This step is analogous to a parameter optimization problem, where the features (parameters) lead to a solution of the OCP. After all OCPs are solved and features determined off-line, the dataset is constructed, where the inputs correspond to the initial states and outputs correspond to the features. Using the dataset, the DNNs are constructed and trained off-line to map the underlying relationship between the inputs and outputs. After the DNNs are trained, the second, on-line stage is used to find the corresponding features for a given set of initial conditions. Finally, the features are used to reconstruct the optimal control solution.

A. Feature Identification

For the two-finite-burn maneuvers, feature identification is straightforward, since the control acceleration is only applied at two points during the time of flight, i.e., at t_0 and $t_f - \Delta t$. The control magnitude and direction at these two points are identified as the features needed to recreate the entire control history and corresponding states along the trajectory. Hence, the features are selected as the components of the control acceleration vector at t_0 and $t_f - \Delta t$:

1) Components of control vector at t_0: u_{x0}, u_{y0}, and u_{z0}
2) Components of control vector at t_f: u_{xf}, u_{yf}, and u_{zf}

Hence, the features for the two-finite-burn maneuver are $X_{output} = \left[u_{x0}, u_{xf}, u_{y0}, u_{yf}, u_{z0}, u_{zf}\right]^T$.

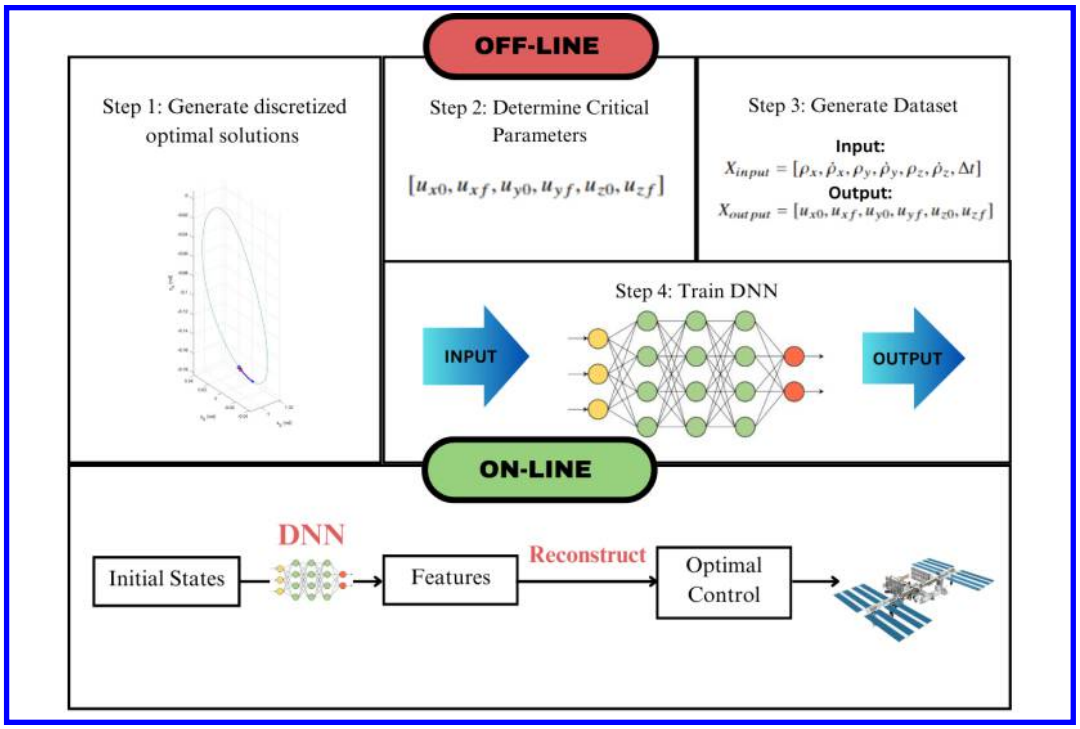


Fig. 4 Framework for the L-OCM

B. Training of the DNNs

After generating the dataset for training, DNNs are constructed to map the relationship between X_{input} and X_{output}. The architecture for the DNN is shown in Fig. 5, which is composed of the input, hidden, and output layers. Layer 1, which is the input layer with m_1 neurons, receives data from X_{input} and passes it on to the next layer with a certain set of weights. The output layer is Layer L, whose neurons are determined by the length of X_{output}. The remaining layers are called hidden layers, each with m_i neurons.

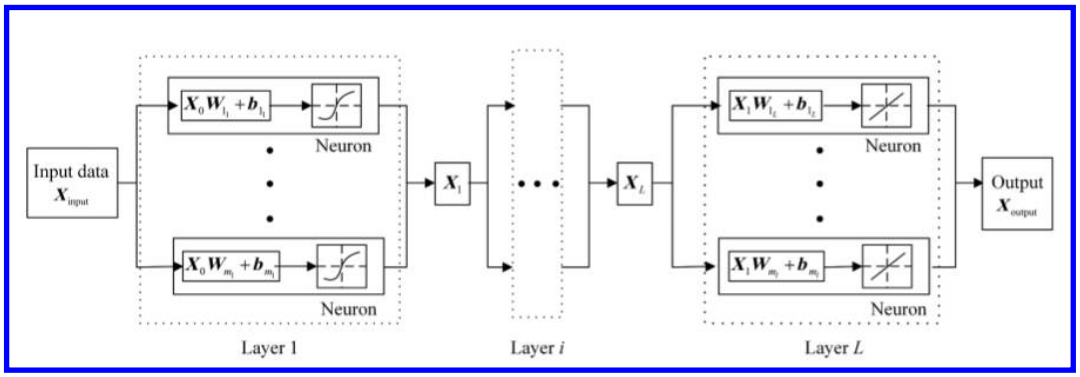


Fig. 5 The architecture of a DNN [20]

In order to capture the nonlinear dynamics of the CR3BP system more accurately, individual DNNs are constructed for each identified feature. Consequently, each DNN will produce only one identified feature as its output, which means $m_L = 1$ for all DNNs. The DNNs for each feature are constructed with a total of five layers, including three hidden layers, and the number of neurons in each layer is set as 64, 32, 16, 8, 1 respectively. In addition, corresponding activation functions include the rectified linear unit (ReLU) function and the linear function. The ReLU is a popular activation function used in neural networks, known for its simplicity and effectiveness. It is defined as $f(x) = \max(0, x)$, which means that it outputs the input value if it is positive and zero otherwise. One of the key benefits of the ReLU function lies in its ability to introduce non-linearity to the neural network while being computationally efficient. The linear activation

function, also known as the identity activation function, simply outputs the weighted sum of the inputs without applying any non-linear transformation $f(x) = wx$, where w are the weights tuned by the DNN. The parameters for the DNN are optimized using an adaptive moment estimation algorithm [21], known for its efficiency, fast convergence, and ability to handle noisy or sparse gradients. To guarantee high accuracy and training performance of the DNNs, the generated dataset is divided into 80% training data, 10% validation data, and 10% test data, all within the range of the initial conditions selected for the NLP.

V. Simulation and Results

To verify the effectiveness of the proposed L-OCM, simulation results for the close range rendezvous, nearby transfer, and collision avoidance maneuvers are provided in this section. All simulations are run in Matlab environments on a 3.5 GHz computer with 16GB RAM. The nonlinear OCPs are solved using a commercial NLP solver integrated with Matlab and the DNNs are trained using machine learning libraries in Python. The datasets for the three separate problems are generated for a range of predefined initial conditions listed in Table 1. The initial conditions generally define the state of the chaser and target, and the time of flight.

The differential corrections (targeting) method is also applied to the close range rendezvous and nearby transfer problems, both to validate NLP results as well as to gain insight into the convergence basins inherent to each scenario. For each set of initial conditions listed in Table 1, the targeting and NLP solutions are compared. Additionally, locally optimal solutions are found by applying Eq. (15) with the impulsive $\Delta\boldsymbol{v}$ at departure and time of flight as free variables, and position continuity at the arrival location as the constraint. The cost function for the local optimality constraint is then

$$f = \|\Delta v_1\|^2 + \|\Delta v_2\|^2 \tag{20}$$

where Δv_2 is the impulsive maneuver applied at arrival. Note that because of the necessity of an initial guess and the application of the linear update in the targeting process, the optimality targeter can fail to converge. In such cases, the time of flight of the transfer is removed as a free variable, and the square targeting problem is solved for a feasible solution. While this setup is no longer explicitly targeting optimality, it can be shown that the resulting transfers maintain a measure of optimality, by virtue of the fact that there is frequently only one feasible solution for a fixed time of flight in the vicinity of the initial guess. However, the fixed time of flight targeter cannot adjust time of flight to find the minimum Δv transfer in the local solution region. In such cases, a continuation method must be applied. Using the same free variables as the optimality targeter, but substituting the optimality constraint in favor of the pseudo-arclength constraint from Eq. (16), rendezvous solutions can be continued to fill out the solution space and find the minimum cost solutions. The result of applying the differential corrections method is that the solution space can be efficiently explored, with the pseudo-arclength continuation method providing a means of outlining the convergence basins that exist for each scenario.

It is important to note that implementing path constraints within the targeting formulation is in general nontrivial, and consequently the targeting method is not applied to the collision avoidance problem. The partial derivative matrix of the constraint with respect to the free variables must be explicitly provided, and is either calculated analytically or numerically. Additionally, since the update to the free variables in the targeting algorithm is derived from a linear approximation, the addition of complex constraints with highly nonlinear behavior can prevent convergence. These inherent challenges with the targeting method stand in contrast to optimal control formulations like NLP, which can be beneficial because they possess flexibility when it comes to the addition and modification of different types of constraints involving state and control variables.

Close Range Rendezvous	
Parameter	**Range**
Mean Anomaly (MA)	$[144, 162]^\circ$
ρ_0	$[0.3, 1000]\ km$
Time of Flight ToF	$[8, 50]\ hrs$
Nearby Transfer	
Parameter	**Range**
θ_{t_0}	$[0.0031, 0.1253]$
δt	$[-2, 2]\ hrs$
Time of Flight ToF	$[1, 100]\ hrs$
Collision Avoidance	
Parameter	**Range**
θ_{t_0}	$[0.0282, 0.1003]$
θ_{c_0}	$[0.0031, 0.0627]$
Time of Flight ToF	$[50, 200]\ hrs$

Table 1 Range of Initial Conditions for Dataset

A. Close Range Rendezvous

For the CRR problem, the objective is to minimize the burn velocity required by the chaser to rendezvous with the target, i.e., to match their position and velocity at t_f. The target and chaser are in a string of pearls formation on the NRHO, where the chaser is behind the target at t_0. The parameters for the initial conditions are specified as the Mean Anomaly (MA) of the target, initial relative distance ρ_0, and time of flight ToF.

An initial investigation of the solution space is conducted using differential corrections for a sample scenario with target mean anomaly of 150°, with the chaser starting 2 hours behind in phase, 897 km away. The solution space for rendezvous times of flight up to 2.6 target revolutions is filled in, and the hodograph of maneuver cost as a function of time of flight can be viewed in Fig. 6a. The most salient result from the figure is that there are several solution basins, with a local minimum cost rendezvous appearing at the bottom of each trough. Because these convergence basins are separated by steep walls and complex behavior that is difficult for the linear continuation process to step along, the continuation process had to be stopped and restarted at different times of flight in order to fill out the plot. It is important to note, however, that the apparent gaps in the plot in Fig. 6a do not indicate a lack of solutions at those times of flight. The corresponding results at those times of flight in general possess prohibitively large maneuver costs and undesirable geometry.

The analysis of the convergence basins also predicts the general trend that local minimum total maneuver costs decrease as the time of flight is increased. For example, the total Δv for the 0.51 revolution time of flight is nearly 3 times as expensive as the total Δv for the 2.49 revolution time of flight (11.21 m/s vs. 3.80 m/s). The trajectories of these two solutions, as well as another at an intermediate time of flight, are shown in Fig. 7. The decreasing Δv over increasing time of flight is to be expected, as phase changes in the vicinity of the NRHO can be accomplished more slowly over a longer period of time. The decreasing trend in local minimum CRR solutions does not indicate that all solutions get less expensive as the time of flight is increased, however. The presence of very expensive solutions is indicated by the steep walls in the Δv vs. time of flight plot. CRR solutions exist over a wide range, but in certain regions of time of flight, such as around 0.95 and 2.1 revolutions, the available solutions are prohibitively expensive and can prove difficult to locate. Additionally, some of the basins from the figure appear to overlap. The apparent intersection points are not the same solution, however, and instead indicate the presence of two solutions of equivalent cost and time of flight that fly on different trajectories.

Fig. 6b demonstrates the comparison between the converged solutions and the theoretical minimum Δv for each scenario. Theoretical minimum Δv is calculated by summing Δv_{th} from the NRHO onto the transfer leg and Δv_{th} from the transfer leg back onto the NRHO. The costs for every converged transfer are greater than that predicted by the theoretical minimum, but are within the same order of magnitude, indicating that the solutions are likely global optima for the selected times of flight.

A total of 8,000 converged cases are generated using NLP. Fig. 8a shows the terminal position and velocity error

for all cases in the dataset. It can be seen that position error $e_p < 0.08m$ and velocity error $e_v < 3 \times 10^{-8} m/s$ for all cases. For all NLP solutions, corresponding solutions are generated using the targeting method. Fig. 9a shows the relationship between Δv and ToF for the first 4000 cases in the dataset, where an inverse trend can be seen between the dependent and independent variables. The burn magnitude decreases as the time of flight increases for the same initial conditions, which is a result predicted from the linear analysis. The times of flight considered in this investigation mostly lie within the first solution basin shown in Fig. 6a. The linear results also predict the sparsity of results for times of flight greater than around 70 hours, as this is approaching the region where the first solution basin ends and solutions greatly increase in cost. It is important to note that the finite burn solutions produced by the NLP tend to have lower Δv values as compared to the impulsive results from the targeting method, with the largest differences appearing with larger Δv values.

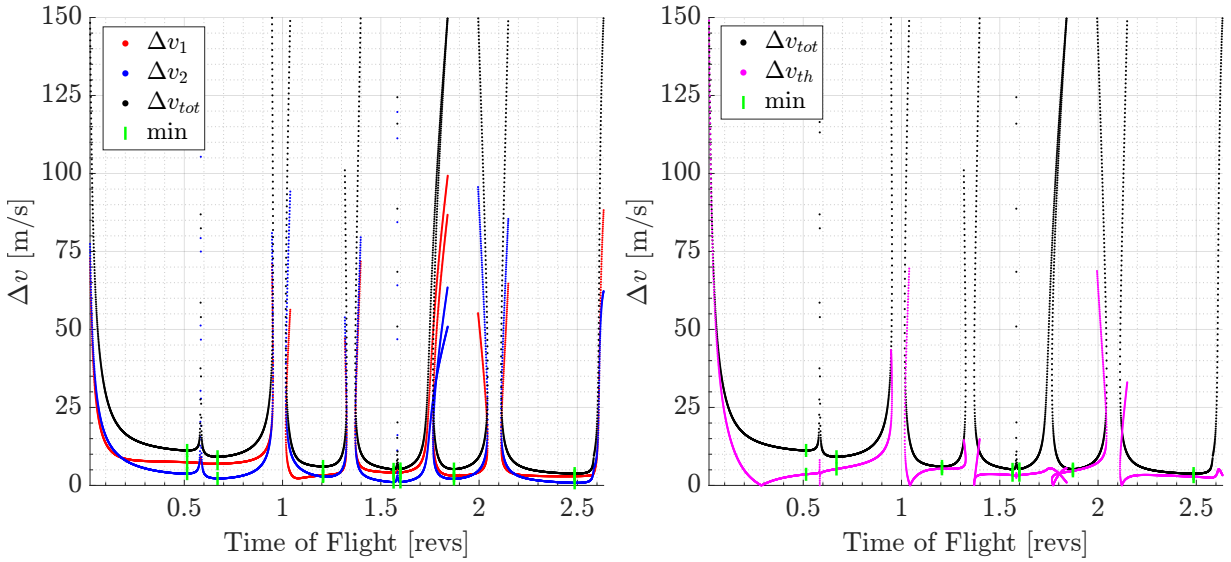


Fig. 6 Δv vs. time of flight for the CRR scenario where the target and chaser are initialized one hour after and before apolune, respectively. In (a), individual maneuver costs are plotted with their corresponding total Δv, and in (b), the total maneuver cost is plotted with the corresponding theoretical minimum Δv. Green vertical bars indicate the locations of local minimum cost transfers.

Fig. 10d shows the true and predicted values from the test dataset for one identified feature, i.e., u_{z0}. This case demonstrates that the DNNs generate accurate mapping relationships. The average R_2 score calculated for the test dataset for all six DNNs is 0.9985, which conveys that on average, the trained models are able to explain 99.85% variation in the data. In order to test the L-OCM method, a case from the test dataset is chosen, for which the initial relative states and burn time is given to the trained models as input $X_{input} = \left[\rho_{x0}, \dot{\rho}_{x0}, \rho_{y0}, \dot{\rho}_{y0}, \rho_{z0}, \dot{\rho}_{z0}, \Delta t\right]^T$. The model predicts values for the control accelerations at t_0 and t_f, from which the optimal solution is reconstructed using forward integration. The solution is compared to the discretized optimal solution from the NLP, where Figs. 10a- 10c show the time history of the relative position, relative velocity, and trajectory found from the NLP and the L-OCM. The values for burn velocities and computational time are listed in Table 2. The results show very close results for the time histories and the total burn velocities produced by both methods. The L-OCM also lowers the computational time, effectively proving the advantages of using the proposed L-OCM to generate optimal control solutions.

B. Nearby Transfer

The nearby transfer problem considers two reference orbits, i.e., the NRHO and a nearby orbit from the L_2 halo family. The selected nearby halo orbit has an orbital time period and a Jacobi constant near to the values for the 9:2 L_2 NRHO. Hence, both orbits are similar to one another in size and energy. The parameters that define the initial conditions are the phase of the target at initial time θ_{t_0}, phase change parameter at final time δt, and time of flight ToF. The initial chaser phase θ_{c_0} is kept constant, which leads to a dataset large enough with adequate variation for learning

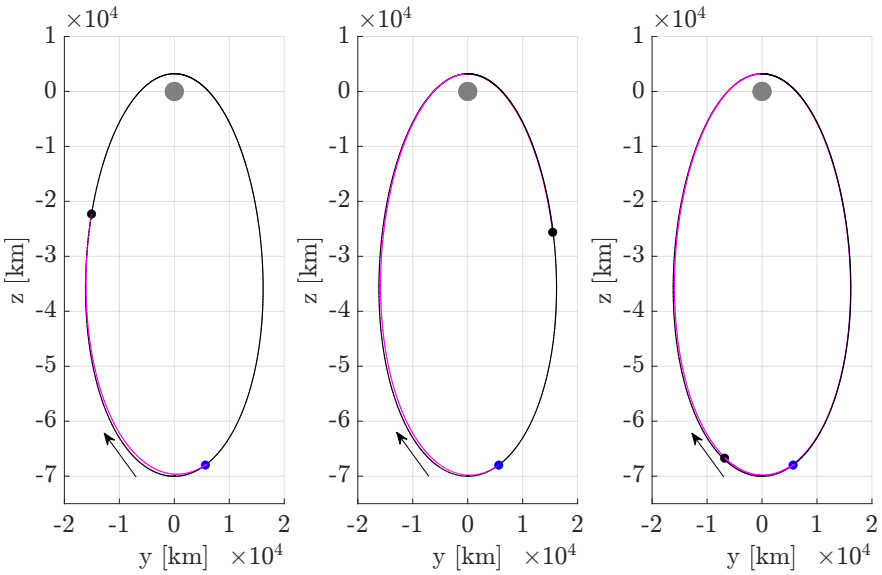


Fig. 7 Sample CRR trajectories where the target and chaser are initialized one hour after and before apolune, respectively. The times of flight for these transfers are 0.51 (left), 1.56 (center), and 2.49 (right) revolutions of the 9:2 NRHO. The total impulsive manuever costs for these rendezvous scenarios are 11.2, 5.15, and 3.80 m/s, respectively. Blue markers indicate the starting location of the chaser, black markers indicate the arrival location, and the magenta line represents the transfer trajectory. The Moon is plotted to scale in grey.

Close Range Rendezvous		
Parameter	**NLP**	**L-OCM**
Burn Velocity Δv	5.5174 m/s	5.5327 m/s
Computation Time	3.439 s	1.05 s
Nearby Transfer		
Parameter	**NLP**	**L-OCM**
Burn Velocity Δv	124.9 m/s	125.5 m/s
Computation Time	4.37s	0.913745 s
Collision Avoidance		
Parameter	**NLP**	**L-OCM**
Burn Velocity Δv	532.3691 m/s	533.9 m/s
Computation Time	9.87 s	2.137078 s

Table 2 Results from NLP and L-OCM

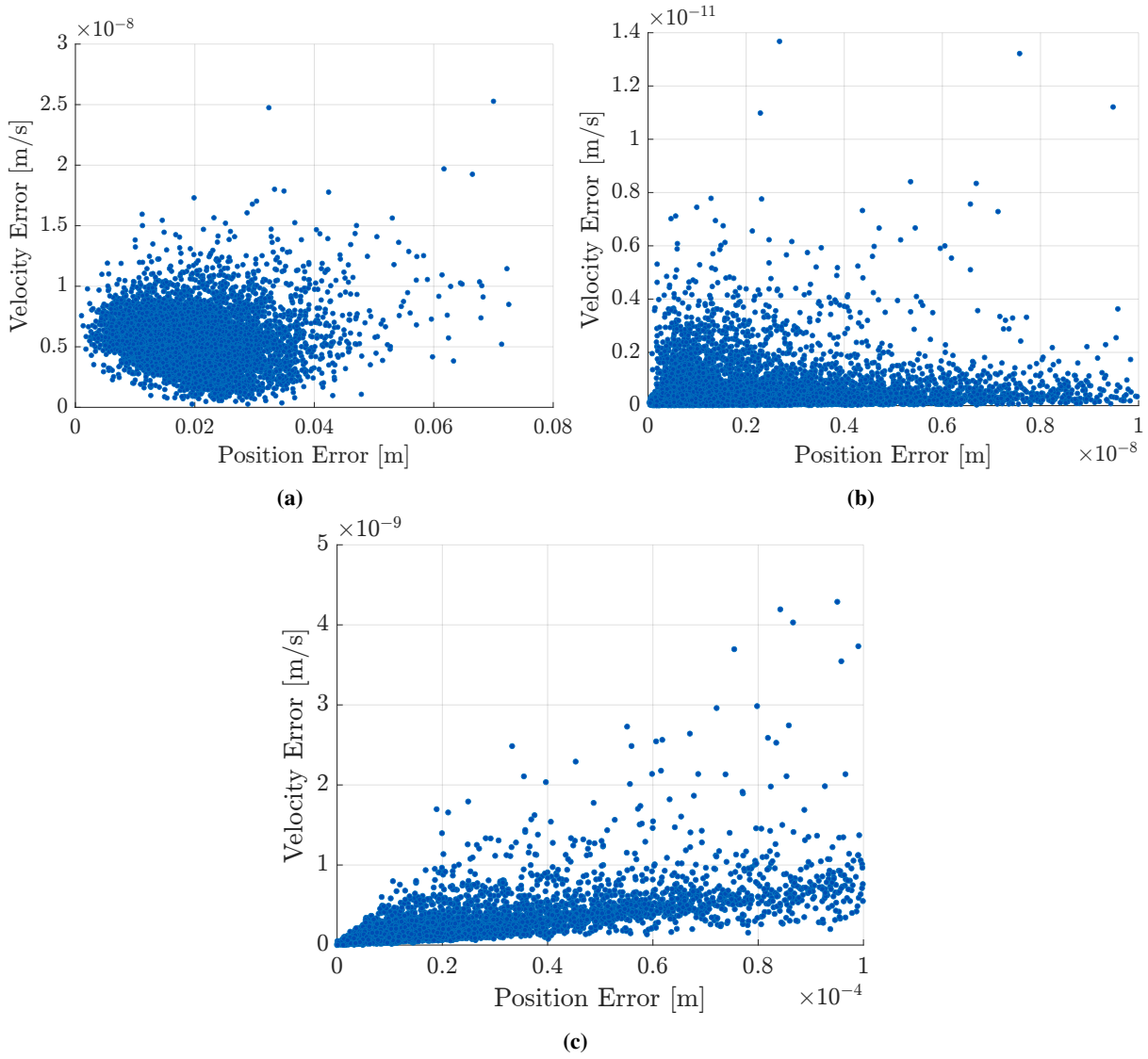


Fig. 8 Position error (m) vs. velocity error (m/s) at t_f of generated datasets for: (a) CRR (b) nearby transfer (c) collision avoidance

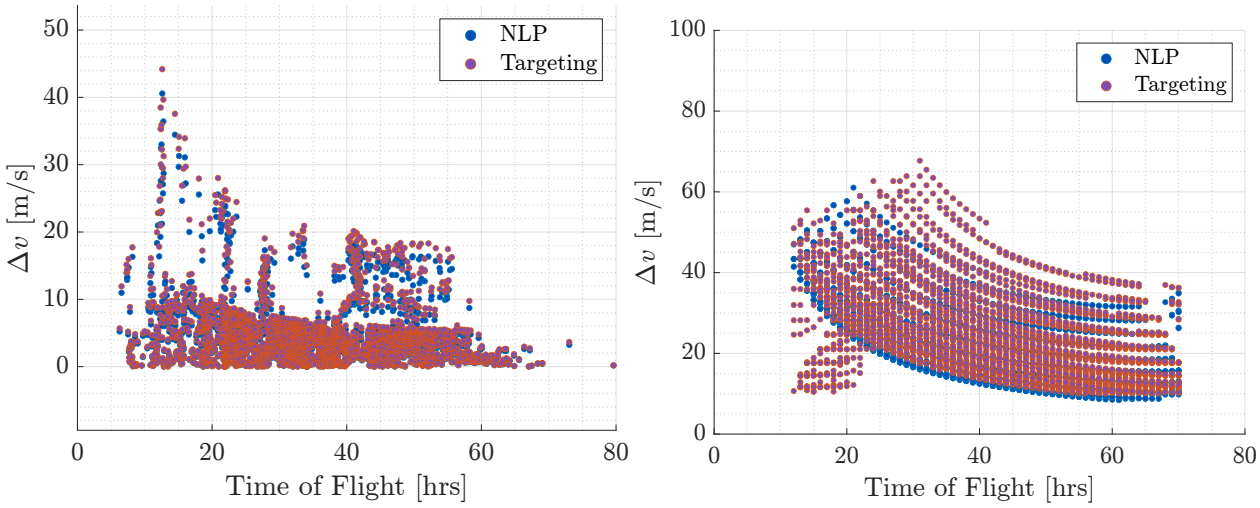


Fig. 9 Burn velocity Δv (m/s) vs ToF (hrs) for: (a) CRR (b) nearby transfer

and taking into account computation capabilities. The range for these initial conditions can always be increased to produce larger-size datasets.

As with the CRR problem, the solution space of the nearby transfer problem is investigated using differential corrections. The nearby orbit has a period 2.89 hours longer than the 9:2 NRHO, the target and chaser are both initialized 1 hour past apolune in their respective orbits, and $\delta t = 0$ for targeting a rendezvous. The initial relative separation between the target and chaser is 976 km. Fig. 11 displays the Δv vs. time of flight plots for this scenario. As with the CRR problem, the primary feature of these plots is that the solution space is broken into several convergence basins separated by steep walls, with only high cost solutions with poor geometry bridging the gaps. The widths of these convergence basins roughly correspond to half a revolution of the target, with the walls occurring roughly at apolune and perilune. Additionally, much like with the CRR problem, the theoretical minimum Δv is smaller than the converged results, but within the same order of magnitude, indicating that the targeting results are likely global optima for their respective times of flight. Selected trajectories from the linear analysis for the nearby transfer are shown in Fig. 12.

The major difference with the linear analysis from the CRR problem is that there is no trend of decreasing Δv for increasing time of flight. This is likely due to the chaser being initialized at a different energy level than the target. While the initial relative distance is not significantly different between the CRR and nearby transfer cases, bridging the energy gap between the two periodic orbits for the nearby transfer requires leveraging different dynamical structures than for the CRR case.

A total of 7,700 converged cases were generated using NLP. Fig. 8b shows the terminal position and velocity error for all cases in the dataset, where the position error $e_p < 10^{-8}m$ and the velocity error $e_v < 1.4 \times 10^{-11}m/s$ for all cases. Similar to CRR, Fig. 9b shows the relationship between Δv and ToF for the first 4000 cases in the dataset, where an inverse trend can be seen between Δv and time of flight, as predicted by the linear analysis. Fig. 13d shows the true and predicted values from the test dataset for one identified feature, i.e., u_{x0}. The average R_2 score calculated for the test dataset for all six DNNs is 0.9999, which conveys that on average, the trained models are able to explain 99.99% variation in the data. In order to validate the advanced computational performance of the L-OCM method, one case from the test dataset is chosen, for which the initial phase of the target in the NRHO, time of flight, and the phasing parameter are given to the trained model as input $X_{input} = \left[\theta_{t_0}, ToF, \delta t\right]^T$. Figs. 13a- 13c show the time histories of the relative position, relative velocity, and trajectory found from the NLP and the L-OCM for the selected case. The values for burn velocities and computational time are listed in Table 2. Similar to the CRR, the results show close results and reduced computational time for the L-OCM.

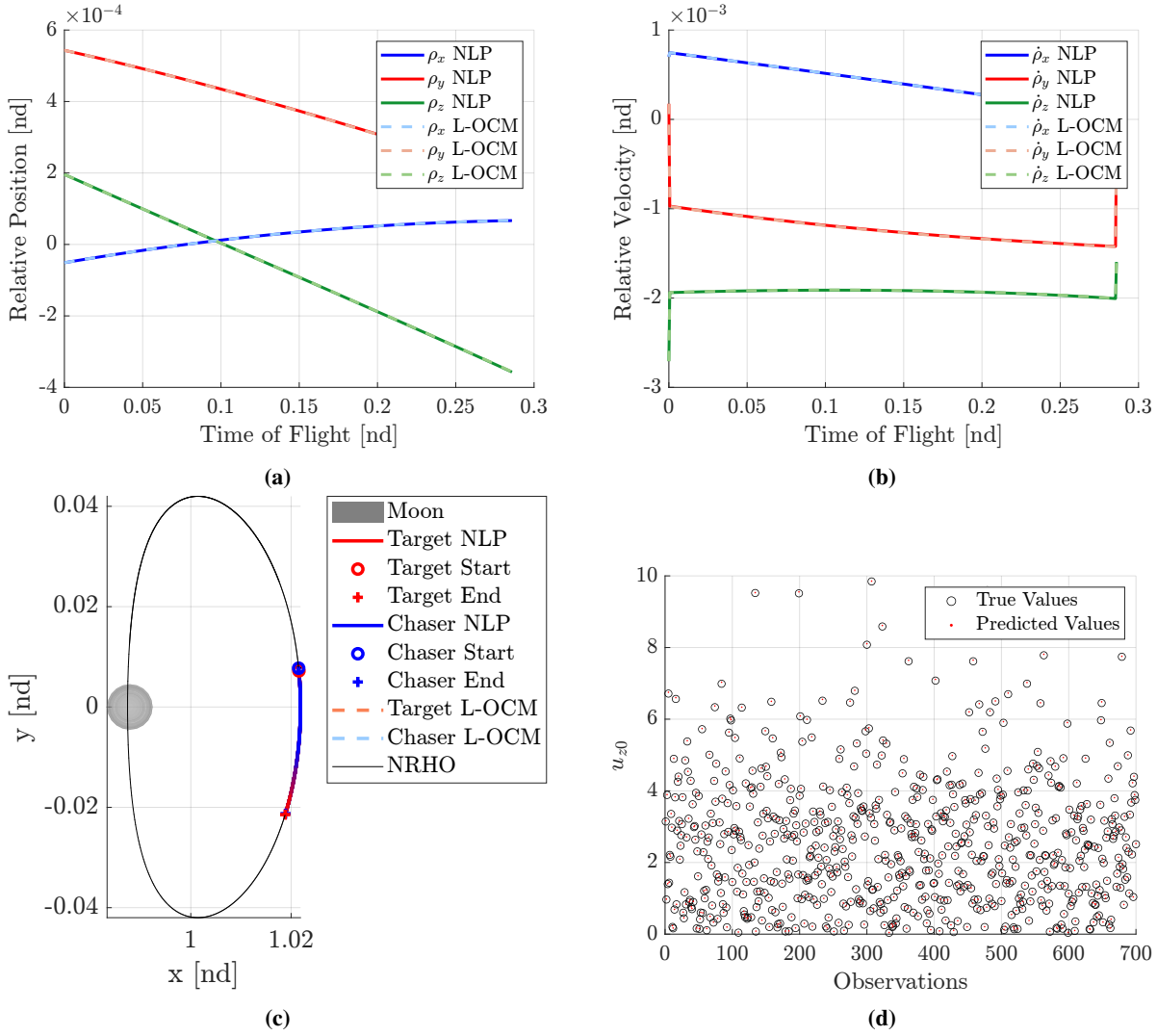


Fig. 10 Comparison of results from NLP and L-OCM for CRR: (a) relative position time history. (b) relative velocity time history. (c) trajectory time history. (d) comparative results of DNNs and test dataset for feature u_{z0}.

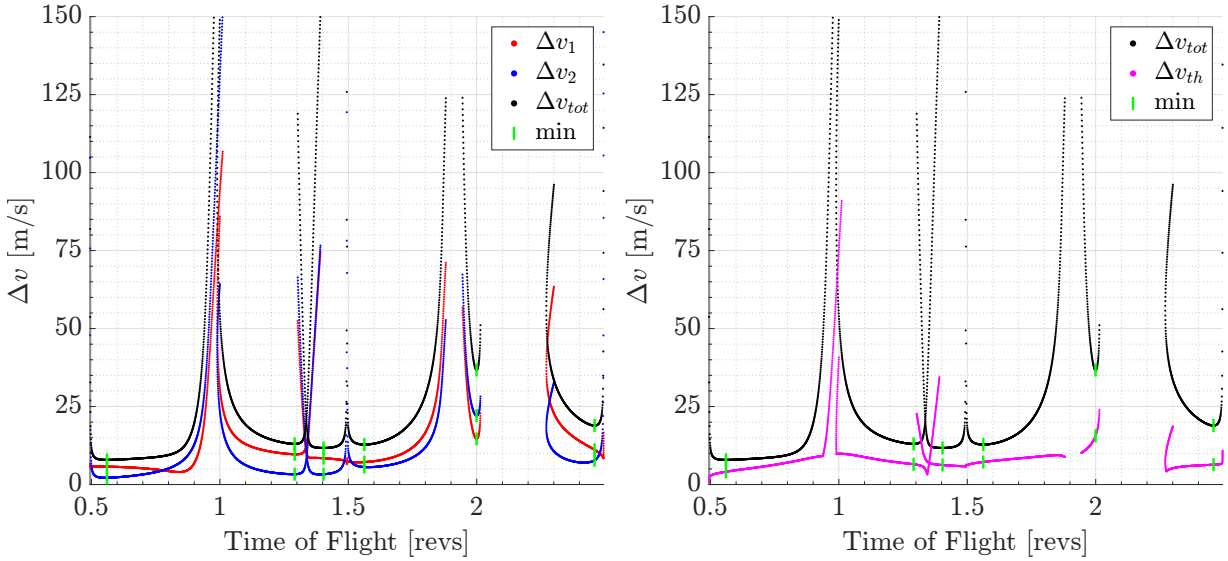


Fig. 11 Δv **vs. time of flight for the nearby transfer scenario where the chaser and target are initialized 1 hour past apolune in their respective original orbits. In (a), individual maneuver costs are plotted with their corresponding total** Δv**, and in (b), the total maneuver cost is plotted with the corresponding theoretical minimum** Δv**. Individual scatter points indicate the locations of local minimum cost transfers.**

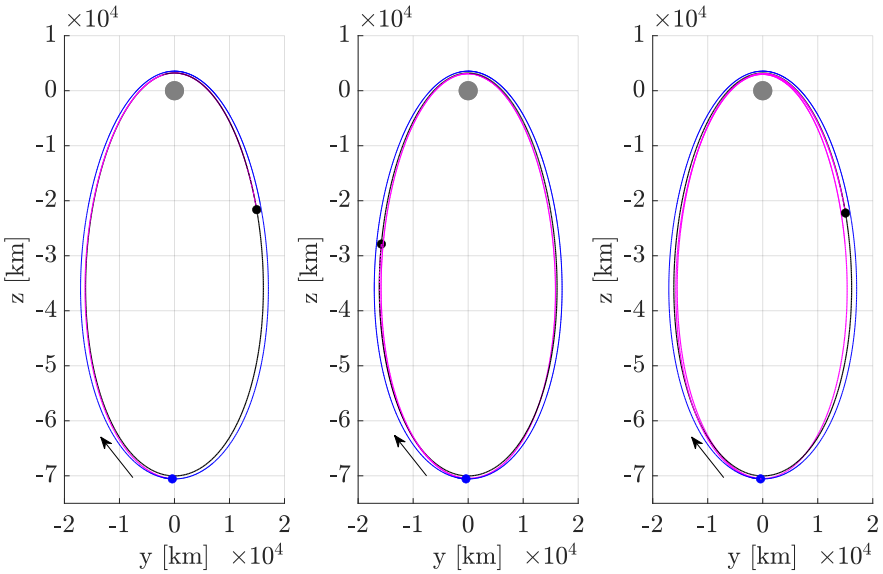


Fig. 12 Sample nearby transfer scenario where the chaser and target are initialized 1 hour past apolune in their respective original orbits. The times of flight for each case are 0.56 (left), 1.40 (center), and 1.56 (right) revolutions of the target orbit. The total impulsive maneuver costs are 7.91 m/s, 11.73 m/s, and 12.82 m/s, respectively. Blue markers indicate the starting location of the chaser, black markers indicate the arrival location, and the magenta line represents the transfer trajectory. The Moon is plotted to scale in grey.

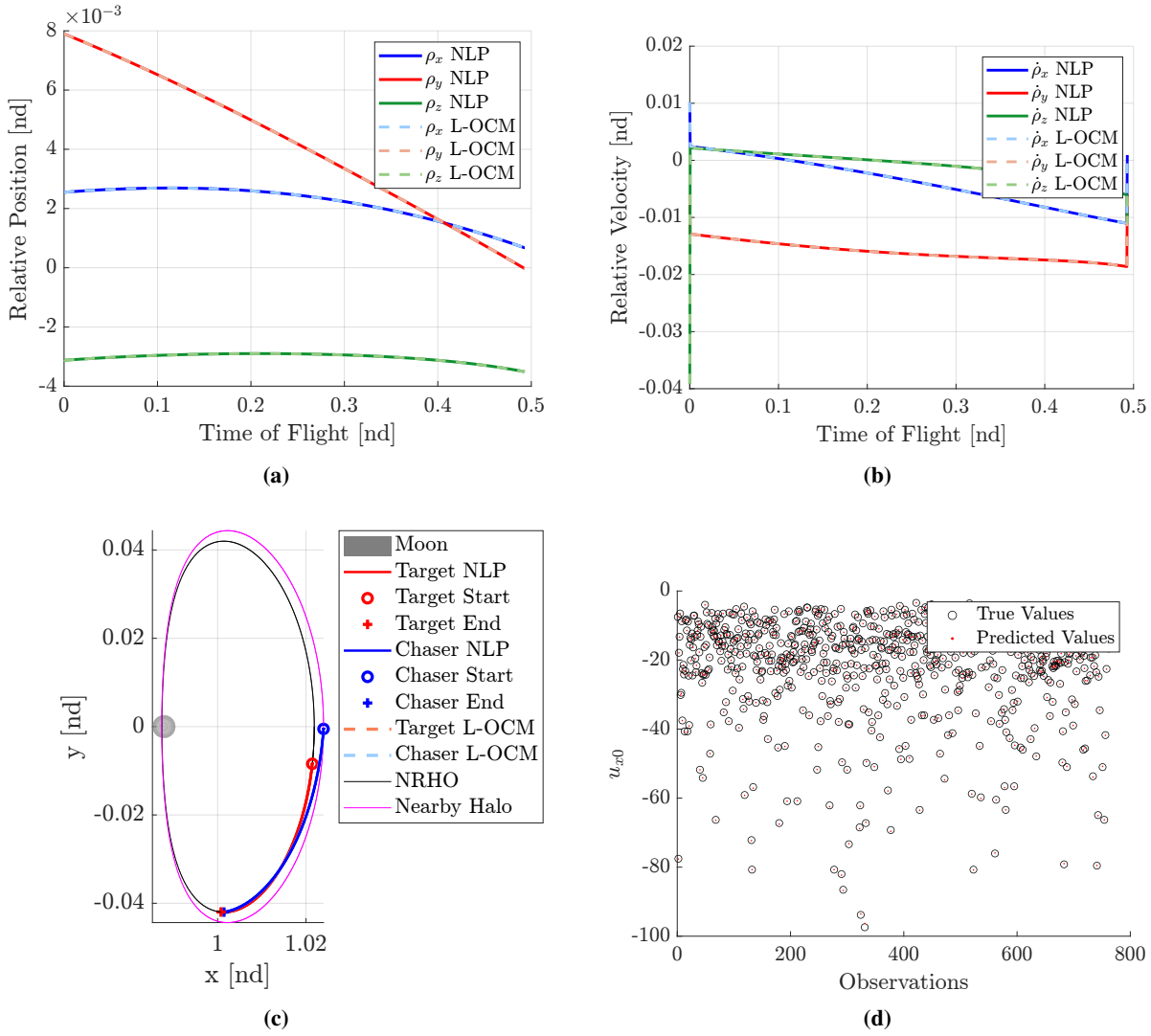


Fig. 13 Comparison of results from NLP and L-OCM for the Nearby Transfer Problem: (a) relative position time history (b) relative velocity time history (c) trajectory time history (d) comparative results of DNNs and test dataset for feature u_{x0}.

C. Collision Avoidance

In the collision avoidance problem, the chaser must perform a phase change maneuver to get ahead of the collision sphere centered at the target in the NRHO, all while ensuring that it does not enter the sphere at any point in time. The radius of the collision sphere is r_{avoid} = 1000 km. The parameters for the initial conditions are listed as phase of target at initial time θ_{t0}, the phase of chaser at initial time θ_{c0}, and time of flight ToF. The phasing parameter δt is set to a constant value of 3 hrs, indicating that the chaser needs to be three hours ahead of the target at t_f. A total of 7000 converged cases were generated. Fig. 8c shows the terminal position and velocity errors for all cases in the dataset, where the position error $e_p < 1 \times 10^{-4}$ and the velocity error $e_v < 5 \times 10^{-9} m/s$ for all cases. Fig. 13d shows the true and predicted values from the test dataset for one identified feature, i.e., u_{xf}. The average R_2 score calculated for the test dataset for all six DNNs is 0.998, which conveys that on average, the trained models are able to explain 99.8% variation in the dataset.

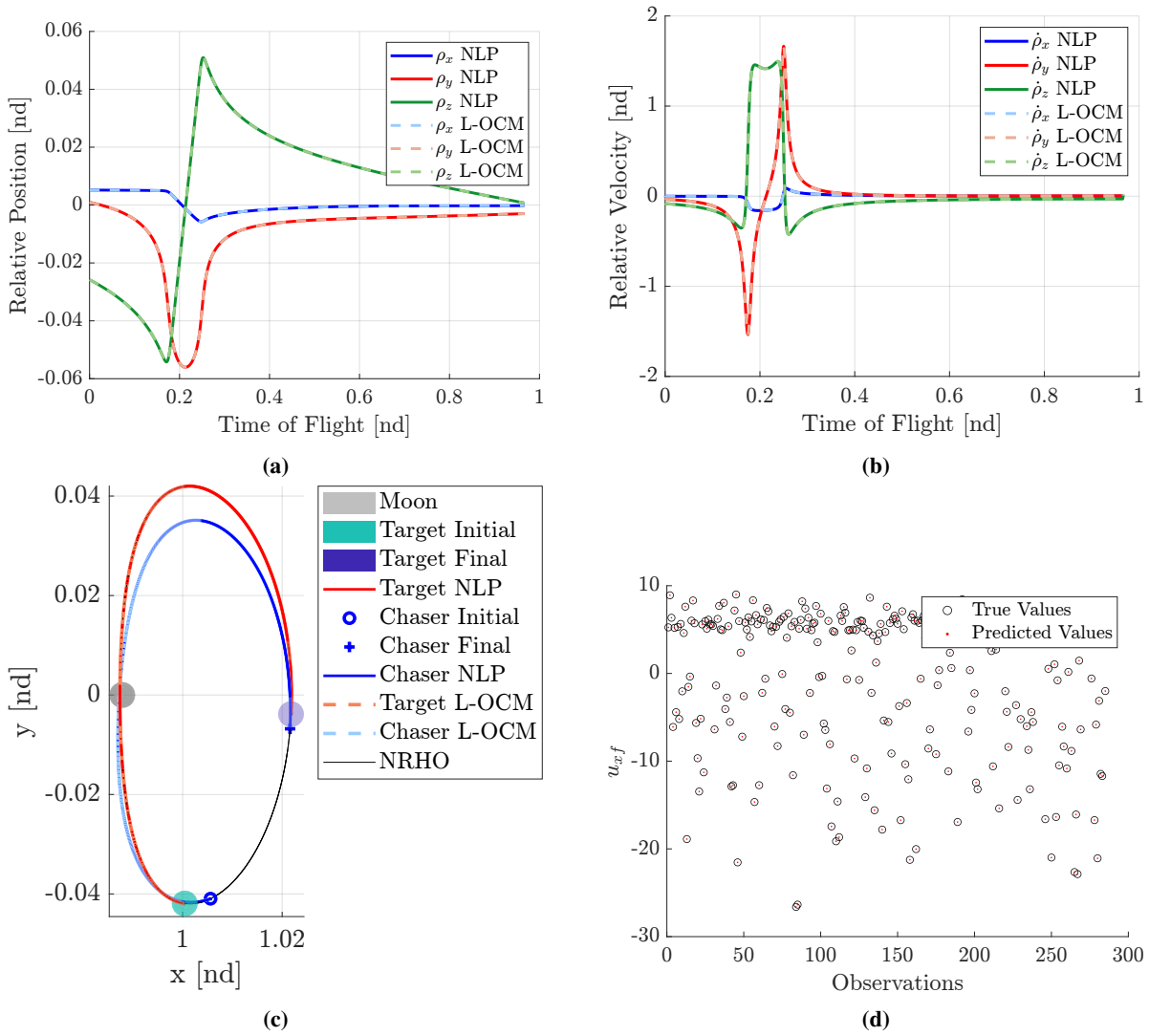


Fig. 14 Comparison of results from NLP and L-OCM for the Collision Avoidance Problem: (a) relative position time history (b) relative velocity time history (c) trajectory time history (d) comparative results of DNNs and test dataset for feature u_{xf}

In order to validate the computational performance of the L-OCM method, a case from the test dataset is chosen, for which the initial phase of the target in the NRHO, time of flight, and phasing parameter is given to the trained model as input $X_{input} = \left[\theta_{t_0}, \theta_{c0}, ToF\right]^T$. Figs. 14a- 14c show the time histories of the relative position, relative velocity, and trajectory obtained from NLP and L-OCM. The results are close to each other, resulting in almost identical time histories and total burn velocities as listed in Table 2. These results complete the validation of the proposed L-OCM.

VI. Conclusions

This paper develops a feature learning based optimal control method (L-OCM) for the close range rendezvous, nearby orbit transfer, and collision avoidance problems while quantifying and studying the behavior of chaser spacecraft moving relative to a target for the nonlinear circular restricted three-body problem (CR3BP). The nonlinear programming (NLP) solutions to the problems for a specified range of initial conditions are validated from widely known and used targeting methods. The burn velocities follow similar inverse trends with the time of flight, also indicating feasible regions for the solutions. As seen with the collision avoidance problem, the formulation of the nonlinear optimal control problems also proves to be more flexible to accommodate path constraints than the targeting method. The NLP solutions

are used to generate datasets to construct the deep neural networks (DNNs), which map the relationship from the initial conditions to the features representing the entire optimal control solutions. The proposed method requires significantly reduced training data compared to existing learning methods, which leads to reduced computational load and time. The method provides real-time optimal solutions which can be implemented onboard. The effectiveness and robustness of the L-OCM has been verified using extensive simulations.

VII. Acknowledgements

The authors would like to thank the Purdue University School of Aeronautics and Astronautics for providing the financial support necessary to perform this work. Additionally, the authors would like to thank Stephen Scheuerle, Beom Park, Chaoying Pei, and other members the of Multibody Dynamics Research Group and the Automation and Optimization Laboratory at Purdue, for lending their insights.

References

[1] Clohessy, W., and Wiltshire, R., "Terminal guidance system for satellite rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–658.

[2] Prussing, J. E., "Optimal four-impulse fixed-time rendezvous in the vicinity of a circular orbit." *AIAA Journal*, Vol. 7, No. 5, 1969, pp. 928–935.

[3] Prussing, J. E., "Optimal two-and three-impulse fixed-time rendezvous in the vicinityof a circular orbit," *AIAA Journal*, Vol. 8, No. 7, 1970, pp. 1221–1228.

[4] Jezewski, D., and Donaldson, J., "An analytic approach to optimal rendezvous using Clohessy-Wiltshire equations," *Journal of the Astronautical Sciences*, Vol. 27, 1979, pp. 293–310.

[5] Gaias, G., and D'Amico, S., "Impulsive Maneuvers for Formation Reconfiguration Using Relative Orbital Elements," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 6, 2015, pp. 1036–1049. https://doi.org/10.2514/1.G000189, URL https://arc.aiaa.org/doi/10.2514/1.G000189.

[6] Yamanaka, K., and Ankersen, F., "New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 60–66. https://doi.org/10.2514/2.4875, URL https://arc.aiaa.org/doi/10.2514/2.4875.

[7] Franzini, G., and Innocenti, M., "Relative Motion Dynamics in the Restricted Three-Body Problem," *Journal of Spacecraft and Rockets*, Vol. 56, No. 5, 2019, pp. 1322–1337. https://doi.org/10.2514/1.A34390, URL https://arc.aiaa.org/doi/10.2514/1.A34390.

[8] Khoury, F., and Howell, K. C., "Orbital Rendezvous and Spacecraft Loitering in the Earth-Moon System," *AAS/AIAA Astrodynamics Specialist Conference*, Lake Tahoe, California, 2020, p. 20.

[9] McCarthy, B., Scheuerle, S., Zimovan-Spreen, E., Davis, D., and Howell, K., "Rephasing and Loitering Strategies in the Gateway Near Rectilinear Halo Orbit," Big Sky, Montana, 2023.

[10] Blazquez, E., Beauregard, L., Lizy-Destrez, S., Ankersen, F., and Capolupo, F., "Rendezvous design in a cislunar near rectilinear Halo orbit," *The Aeronautical Journal*, Vol. 124, No. 1276, 2020, pp. 821–837. https://doi.org/10.1017/aer.2019.126, URL https://www.cambridge.org/core/product/identifier/S000192401900126X/type/journal_article.

[11] Davis, D. C., Zimovan-Spreen, E. M., Scheuerle, S. T., and Howell, K. C., "Debris Avoidance and Phase Change Maneuvers in Near Rectilinear Halo Orbits," Breckenridge, Colorado, 2022, p. 16.

[12] Gómez, G., Jorba, A., Masdemont, A., and Simó, C., "Study of the transfer between halo orbits," *Acta Astronautica*, Vol. 43, No. 9-10, 1998, pp. 493–520. https://doi.org/10.1016/S0094-5765(98)00177-5, URL https://linkinghub.elsevier.com/retrieve/pii/S0094576598001775.

[13] Fossà, A., Bucchioni, G., Blazquez, E., Canalias, E., Lizy-Destrez, S., Bertrand, R., Lamy, A., and Goester, J.-F., "Two and three impulses phasing strategy with a spacecraft orbiting on an Earth–Moon NRHO," *Acta Astronautica*, Vol. 198, 2022, pp. 669–679.

[14] Cuevas del Valle, S., Urrutxua, H., Solano-López, P., Gutierrez-Ramon, R., and Sugihara, A. K., "Relative Dynamics and Modern Control Strategies for Rendezvous in Libration Point Orbits," *Aerospace*, Vol. 9, No. 12, 2022, p. 798.

[15] Bucchioni, G., and Innocenti, M., "Rendezvous in cis-lunar space near rectilinear halo orbit: Dynamics and control issues," *Aerospace*, Vol. 8, No. 3, 2021, p. 68.

[16] Galullo, M., Bucchioni, G., Franzini, G., and Innocenti, M., "Closed Loop Guidance During Close Range Rendezvous in a Three Body Problem," *The Journal of the Astronautical Sciences*, Vol. 69, No. 1, 2022, pp. 28–50.

[17] LaFarge, N. B., Howell, K. C., and Foltac, D. C., "Adaptive Closed-Loop Maneuver Planning for Low-Thrust Spacecraft using Reinforcement Learning," 2022.

[18] Sánchez-Sánchez, C., and Izzo, D., "Real-time optimal control via deep neural networks: study on landing problems," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 5, 2018, pp. 1122–1135.

[19] You, S., Wan, C., Dai, R., and Rea, J. R., "Learning-based onboard guidance for fuel-optimal powered descent," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 3, 2021, pp. 601–613.

[20] You, S., Wan, C., Dai, R., and Rea, J. R., "Onboard fuel-optimal guidance for human-Mars entry, powered-descent, and landing mission based on feature learning," *Acta Astronautica*, Vol. 195, 2022, pp. 129–144.

[21] Kingma, D. P., and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.