

Multi-Impulse to Time Optimal Finite Burn Trajectory Conversion

Joshua A. Fogel¹, Maxon Widner², Jacob Williams³ and Amelia Batcha¹
NASA Johnson Space Center, Houston, TX, 77058, USA

A novel algorithm is presented that provides an improvement over a traditional parameter optimization method when solving time-optimal, finite-burn pseudo-rendezvous spacecraft trajectory problems. A hybrid optimization procedure is described that converts a set of multiple-impulses, representing high- or low-thrust maneuvers, to an exact time-optimal finite-burn trajectory for a thrust limited, constant exhaust velocity spacecraft. The Hybrid Method applies a control law derived from the Euler-Lagrange system of equations within the classical Indirect Method to a modern Direct Method. An iterative adjoint-control transformation and an evolving constraint vector are introduced to solve the optimal control two-point boundary value problem. This method requires no prior knowledge of the solution, which adds simplicity to the trajectory design process and aids automation. Examples are shown for low-thrust apogee raise maneuvers, non-coplanar Earth orbit transfers, and a modified Deep Space 1 low-thrust trajectory. A numerically significant improvement to objective cost is shown across all application problems compared to a traditional solution method, as well as a significant improvement to convergence speed for a select class of problems.

Nomenclature

FB	Finite-Burn	SNOPT	Sparse Nonlinear Optimizer ^[26]
LEP	Low-thrust Electric Propulsion	SOC	Sub-Optimal Control
SEP	Solar Electric Propulsion	OCT	Optimal Control Theory
LEO	Low Earth Orbit	P-R	Pseudo-Rendezvous
GEO	Geosynchronous Orbit	SQP	Sequential Quadratic Programming
TPBVP	Two-Point Boundary Value Problem	SP	Sub-Problem
NLP	Non-Linear Programming	DS1	Deep Space 1
ΔV	Delta-Velocity	NEA	Near Earth Asteroid
V_∞	Hyperbolic Excess Velocity		

I. Introduction

Recent technological advancements in the realm of low-thrust electric propulsion (LEP) have allowed such systems to offer significant performance benefits over other in-space propulsion systems in certain applications. Shorter flight times, increased payload mass fraction, and smaller launch vehicles are all potential benefits to LEP missions [1]. The growing dominance of LEP systems requires the development of robust, accurate, and practical

¹ Flight Mechanics and Trajectory Design Branch, National Aeronautics and Space Administration, Houston, TX

² Jacobs Technology, JSC Engineering, Technology and Science (JETS) Contract, Houston, TX

³ ERC, Inc., JSC Engineering, Technology and Science (JETS) Contract, Houston, TX

methods of computing optimal low-thrust trajectories, and more generally, finite-burn trajectories. The performance index typically minimized in these problems is transfer time or propellant expenditure. Unfortunately, finding global optimal solutions for these problems is a formidable challenge, as no analytic, closed-form solutions exist for the full perturbed N-body problem. While many indirect and direct numerical optimization methods have been developed [2], they are often tailored for specific transfer problems, with sensitive convergence properties as the objective functions are often highly nonlinear.

The motivation of the work is to bridge the gap between indirect and direct optimization methods. Many industry astrodynamists rely on Direct Methods for designing spacecraft trajectories due to their adaptability, ease of use, and relative high-fidelity solutions. There is a desire, however, to find improved solutions which yield a savings in flight time, propellant, computational efficiency, or other mission resources. In order to leverage higher-fidelity solutions available from Indirect Methods, a new methodology must be derived.

The research presented herein provides an improvement over traditional parameter optimization methods when solving time-optimal, finite-burn spacecraft trajectory problems based on existing solutions computed by readily available techniques. The new conversion algorithm takes an approximate finite-burn trajectory comprised of n -number of discretized, impulsive nodes, or sub-optimal parametrically-steered finite burns, and through automated procedure, replaces the approximate trajectory with a set of time-optimal finite-burns separated by ballistic arcs, whose pre- and post- maneuver states match that of the original approximation.

The methodology of this conversion algorithm can be broken down into the single and multi-impulse cases. Inspired by existing state-of-the-art work [3], the single impulse case formulates an optimal control problem that is treated as a bounded, pseudo-rendezvous (P-R), targeting initial and final states upstream and downstream of a single ΔV on the initial impulsive trajectory. This becomes a Two-Point Boundary Value Problem (TPBVP), which can be solved via hybrid optimization by applying the control law from the classical indirect method, utilizing calculus of variations and primer vector theory to minimize the Hamiltonian with respect to the controls.

The multi-impulse case is an extension of the first to a more complex conversion problem. This new approach chains together single-impulse conversions, separated by ballistic arcs, thereby forming a larger parameter optimization problem around the sequence of individually solved burn arcs. To complete the conversion, focus must be given to mitigating the finite-burn arc overlap problem, which occurs when interior ballistic arcs between optimized burns result in negative times of flight. This is mitigated by evolving the boundary constraints across multiple sub-problems as part of an iterative adjoint-control transformation. This hybrid conversion algorithm requires no prior knowledge of the solution beyond the supply of an approximate finite-burn trajectory, which adds simplicity to the design process and aids automation.

The presented work enables a robust capability that is otherwise unavailable to industry mission designers, promising a moderate savings in objective cost (flight time or propellant) when compared to conventional Direct Methods for a wide variety of trajectory problems. Moreover, the new Hybrid Method is shown to yield superior convergence behavior in terms of iterations and convergence time for conversion problems where both the burn time is greater than half the orbit period, and the burn is changing inclination. This specific class of problem will therefore reap larger benefits than other problems when the Hybrid Method is employed in lieu of the classical Direct.

II. Single-Impulse Conversion

The following highlights the single-impulse-to-finite-burn conversion method developed by Jesick and Ocampo [3]. This serves as the foundation upon which the presented multi-impulse conversion algorithm is built. First we must define the impulsive maneuver that we wish to convert into a finite-burn (Fig. 1). Consider the state vector:

$$\bar{\mathbf{X}} = [\bar{\mathbf{r}}^\top \bar{\mathbf{v}}^\top m]^\top \quad (1)$$

with associated state equations:

$$\dot{\bar{\mathbf{X}}} = \begin{bmatrix} \bar{\mathbf{v}}(t) \\ \bar{\mathbf{g}}(t, \bar{\mathbf{r}}) + \frac{T}{m} \hat{\mathbf{u}}(t) \\ -\frac{T}{c} \end{bmatrix} \quad (2)$$

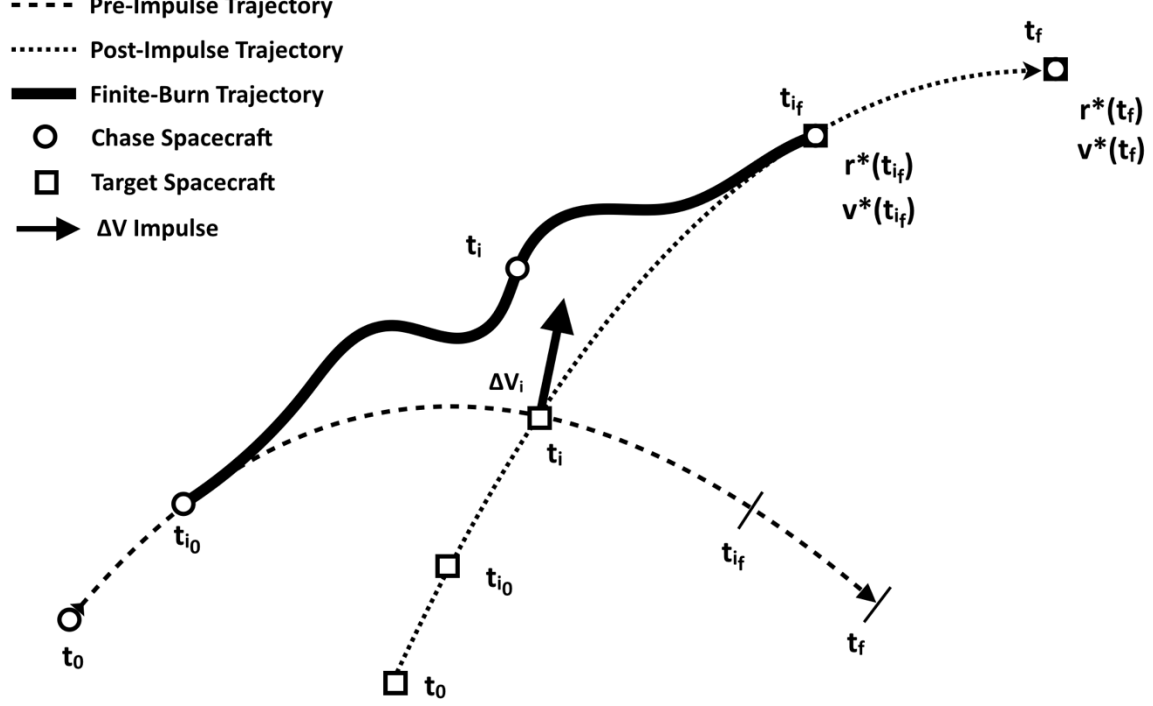


Fig. 1: Single Impulse to Finite-Burn Conversion via Pseudo-Rendezvous targeting.

where $\vec{g}(t, \vec{r})$ is the gravitational acceleration term, T/m is the thrust acceleration term, and $\hat{u}(t)$ is the parameterized thrust direction unit vector defined by spherical angles α and β :

$$\hat{u}(t) = \begin{bmatrix} \cos \alpha & \cos \beta \\ \sin \alpha & \cos \beta \\ \sin \beta \end{bmatrix} \quad (3)$$

Additional perturbing accelerations may be included in the state dynamics as well, but for the simplicity of this work, only two-body gravitational and thrust acceleration terms are included. The initial conditions at t_0 are $\vec{r}_0$, $\vec{v}_0$, and m_0 . The spacecraft's thrust magnitude is bounded:

$$0 \leq T \leq T_{max} \quad (4)$$

This model assumes two arcs of pure ballistic motion ($T = 0$), separated by discontinuities in m and $\vec{v}$ at some interior point $t_i \geq t_0$, representing an impulsive change in velocity and mass. This impulse occurs at position $\vec{r}_i$. The states immediately before and after the impulse are $[\vec{r}_i^- \ \vec{v}_i^- \ m_i^-]^\top$, and $[\vec{r}_i^+ \ \vec{v}_i^+ \ m_i^+]^\top$, respectively.

The impulsive $\Delta \vec{V}$ is then defined as:

$$\vec{v}_i^+ = \vec{v}_i^- + \Delta \vec{V}_i \quad (5)$$

$$m_i^+ = m_i^- + e \frac{\Delta v_i}{c} \quad (6)$$

$$m_i^- = m_0 \quad (7)$$

with the rocket's effective exhaust velocity c . Since $\Delta V_i > 0$, the instantaneous change in mass $\Delta m_i < 0$, such that:

$$\Delta m_i = m_i^+ - m_i^- \quad (8)$$

The final time $t_f > t_i$ is appropriately chosen such that the yet-to-be-found finite-burn with $T > 0$ that will replace the impulse will terminate at some $t_f \leq t_i$.

The pre-impulse ballistic trajectory is defined as a continuous arc from t_0 to t_i , and the post-impulse ballistic trajectory is defined as a continuous arc from t_i to t_f . This is important, as these pre- and post-impulse ballistic arcs (expressed as propagated functions of time) will become the boundary targeting constraints once the finite-burn TPBVP is formulated. Herein the point-mass spacecraft on the pre-impulse and post-impulse trajectories will be referred to as the chase and target spacecraft, respectively.

The goal is to replace the impulse with a finite burn with bounded thrust magnitude T . In general, T is typically taken to be a control variable. However, it is well known that for any minimum time finite-burn problem, $T = T_{Max}$ [4].

We seek an estimate for the finite burn duration:

$$\Delta t_i = \frac{\Delta m_i}{\frac{T_{max}}{c}} \quad (9)$$

Estimates of the finite burn start and stop times can be similarly computed, under the simplifying assumption that the time of impulse t_i is also at the midpoint of the finite burn:

$$t_{i0} = t_i - \frac{\Delta t_i}{2} \quad (10)$$

$$t_{if} = t_i + \frac{\Delta t_i}{2} \quad (11)$$

The state on the post-impulse trajectory at t_{if} is $\begin{bmatrix} \vec{r}_{if}^* & \vec{v}_{if}^* & m_{if}^* \end{bmatrix}$, where $m_{if}^* = m_i^+$. The total transfer time is $t_{if} - t_0$, and the total finite burn time is $t_{if} - t_{i0}$. The optimization variables are then simply t_{i0} and t_{if} since the target state is a function of time.

The optimization problem is to find the minimum-time finite burn maneuver that begins on the pre-impulse trajectory at t_{i0} and matches exactly the position and velocity of the target spacecraft at t_{if} . Since the trajectory of the target spacecraft is defined by a pair of ballistic arcs separated by a pre-defined discontinuity in velocity and mass, its state is a known function of time: $\vec{r}(t)$ and $\vec{v}(t)$. As such, the finite-burn optimization problem is more accurately described as a Pseudo-Rendezvous (P-R) problem, as the boundary conditions are known functions of time.

The unknowns of our problem are the start and stop times of the finite burn, t_{i0} and t_{if} , respectively, and the thrust direction as a function of time $\hat{u}(t)$ for $t_{i0} \leq t \leq t_{if}$. The target conditions for the finite-burn (denoted with *) are:

$$\vec{r}(t_{if}) = \vec{r}^*(t_{if}) \quad (12)$$

$$\vec{v}(t_{if}) = \vec{v}^*(t_{if}) \quad (13)$$

The cost functional J which we seek to maximize is chosen to be a negative quantity:

$$J = \max k(t_{i0} - t_{if}) \quad (14)$$

for a given $k > 0$, $\vec{g}(t, \vec{r})$, t_0 , $\vec{X}_0$, t_i , $\vec{\Delta V}_i$, T_{max} , and c .

While this method can be derived using any general, time and position dependent gravity field $\vec{g}(t, \vec{r})$, a central body inverse square force model is employed for simplicity. The two-body gravitational acceleration is defined as:

$$\vec{g}(t, \vec{r}) = -\frac{\mu}{r^3} \vec{r} \quad (15)$$

with its associated gravity gradient matrix:

$$\mathbf{G}(\vec{\mathbf{r}}) = \frac{\partial \vec{\mathbf{g}}(\vec{\mathbf{r}})}{\partial \vec{\mathbf{r}}} = \frac{\mu}{r^3} \left(\frac{3 \vec{\mathbf{r}} \vec{\mathbf{r}}^\top}{r^2} - \mathbf{I} \right) \quad (16)$$

This minimum time, pseudo-rendezvous problem, with the optimal control function $\hat{\mathbf{u}}(t)$ and the finite-burn start and stop times (t_{i_0} and t_{i_f}), can be solved using direct or indirect methods. The application of the direct method is a parameter finite-burn model which uses a simple set of parameters to define the thrust directions $\hat{\mathbf{u}}(t)$, whose sub-optimal solution can be found using standard NLP algorithms via multi-shooting numerical integration. The indirect method leads to a TPBVP associated with the optimal control problem. Either a nonlinear equation algorithm or an NLP algorithm can be used to solve for the time-optimal controls.

Using established theory, one of the fundamental optimal control problems in orbital mechanics can be formulated [4-6]. Expressing the cost functional in the Mayer form:

$$J = \max \phi(t_{i_0}, t_{i_f}) \quad (17)$$

Assuming there exists some $u^*(t)$ that maximizes J , then the Euler-Lagrange theorem states that there must exist a time-varying Lagrange multiplier (i.e. “co-state”) vector $\vec{\lambda}^\top(t) = [\lambda_1, \lambda_2, \dots, \lambda_n]$ and a constant Lagrange multiplier vector $\vec{\nu}^\top = [\nu_1, \nu_2, \dots, \nu_q]$ such that a Hamiltonian function, H , can be formed for which first order necessary conditions for optimality are valid. For our P-R problem, the Hamiltonian can be expressed as:

$$H = \vec{\lambda}^\top \vec{f} \quad (18)$$

where the time-varying Lagrange multiplier (co-state) vector, $\vec{\lambda}^\top(t)$, and state dynamics, $\vec{f} = \dot{\vec{X}}$, are defined as:

$$\vec{\lambda} = \begin{bmatrix} \vec{\lambda}_r^\top & \vec{\lambda}_v^\top & \lambda_m \end{bmatrix} \quad (19)$$

$$\vec{f} = \dot{\vec{X}} = \begin{bmatrix} \vec{v} \\ \vec{g}(\vec{r}) + \frac{T}{m} \hat{\mathbf{u}} \\ -\frac{T}{c} \end{bmatrix} \quad (20)$$

Note that the position and velocity vectors, and mass scalar, each have associated co-states. Hence the Hamiltonian becomes:

$$H = \vec{\lambda}_r^\top \vec{v} + \vec{\lambda}_v^\top [\vec{g}(\vec{r}) + \frac{T}{m} \hat{\mathbf{u}}] + \lambda_m \frac{T}{c} \quad (21)$$

The co-states are required to satisfy the following necessary conditions:

$$\dot{\vec{\lambda}}_r = - \left(\frac{\partial H}{\partial \vec{r}} \right)^\top = - \mathbf{G} \vec{\lambda}_v \quad (22)$$

$$\dot{\vec{\lambda}}_v = - \left(\frac{\partial H}{\partial \vec{v}} \right)^\top = - \vec{\lambda}_r \quad (23)$$

$$\dot{\lambda}_m = - \left(\frac{\partial H}{\partial m} \right)^\top = - \frac{T}{m^2} \vec{\lambda}_v^\top \hat{\mathbf{u}} \quad (24)$$

Equations (22—24) form the co-state equations of motion, or “co-state dynamics”, which are propagated along with the state dynamics’ equations of motion. The necessary condition for optimality of the controls, $\hat{\mathbf{u}}(t)$, that maximizes the Hamiltonian is the primer vector unit direction [44]:

$$\hat{\mathbf{u}}(t) = + \frac{\vec{\lambda}_v(t)}{\lambda_v(t)} \quad (25)$$

The Hamiltonian can be rewritten in terms of the switching function, $\$(t)$:

$$H = \vec{\lambda}_r^\top \vec{v} + \vec{\lambda}_v^\top \vec{g}(\vec{r}) + \$(t)T \quad (26)$$

with

$$\$(t) = \frac{\lambda_v}{m} - \frac{\lambda_m}{c} \quad (27)$$

and the thrust throttling function T determined from the following bang-bang control law:

$$T = \begin{cases} 0 & \text{if } \$ < 0 \\ T_{max} & \text{if } \$ > 0 \\ 0 \leq T \leq T_{max} & \text{if } \$ = 0 \text{ (i.e. singular arc, undetermined)} \end{cases} \quad (28)$$

The augmented state vector to be numerically integrated is then:

$$\vec{X} = \left[\vec{r}^\top \quad \vec{v}^\top \quad m \quad \vec{\lambda}_r^\top \quad \vec{\lambda}_v^\top \quad \lambda_m \right]_{14 \times 1}^\top \quad (29)$$

governed by the corresponding first order vector differential equation:

$$\dot{\vec{X}} = \begin{bmatrix} \vec{v} \\ \vec{g}(\vec{r}) + \frac{T}{m} \frac{\vec{\lambda}_v(t)}{\lambda_v(t)} \\ -\frac{T}{c} \\ -\vec{G} \vec{\lambda}_v \\ -\vec{\lambda}_r \\ -\frac{T}{m^2} \lambda_v \end{bmatrix}_{14 \times 1} \quad (30)$$

Note that the explicit thrust direction $\hat{u}(t)$ has been successfully removed from the problem, and replaced with the primer vector, governed by the co-state dynamics.

III. Multi-Impulse to Finite-Burn Hybrid Conversion

A. Assumptions

Conversion Solution Structure: The conversion algorithm is derived for a pseudo-rendezvous optimal control problem composed of multiple burns. The output solution to this algorithm is constructed with a coast-thrust-coast structure that repeats for the number of burns input to the algorithm. The coast arcs between the computed finite burns are allowed to have flight times greater than or equal to zero.

Input Trajectory: The trajectory input to the conversion algorithm is assumed to be comprised of a set of impulsive and/or sub-optimal, parametrically steered finite-burns, separated by ballistic arcs. The spacecraft is assumed to have a constant specific impulse, variable thrust engine model.

Force Model: A two-body, point-mass gravity field is used to derive the dynamics and boundary constraints associated with the pseudo-rendezvous optimal control problem.

Primer Vector Alignment: The initial thrust directions provided by the input trajectory are assumed to be closely aligned with the optimal Primer Vector direction, as dictated by the co-state dynamics control law. This enables a series of adjoint-control transformations to be used to estimate the initial co-states from the input trajectory. This represents the fundamental assumption of this work, which then requires the input trajectory to be a reasonable approximation of the true time-optimal finite-burn solution. This does not require a priori knowledge of the solution beyond a simple calculation to ensure that the engine model has enough time to provide the input ΔV before the next impulse. Even when this is not the case, the algorithm is designed to mitigate the resulting overlap problem its omission creates.

Optimization Settings: The software program *Copernicus 4.6* [7] was used to simulate all trajectories. A variable order, variable step-size Adams-Basforth-Moulton integration method [8] was utilized for propagation, with absolute and relative error tolerances of 1.0×10^{-12} . The constrained optimization system SNOPT [9-10] was employed with the following key settings which were found to improve convergence:

<i>Major Optimality Tolerance:</i>	1.0×10^{-3}
<i>Major Feasibility Tolerance:</i>	1.0×10^{-6}
<i>Major Step Limit:</i>	1.0×10^{-2}
<i>Gradient Perturbation Step Size, Difference Interval:</i>	1.0×10^{-7}
<i>Gradient Perturbation Step Size, Central Difference Interval:</i>	1.0×10^{-6}

B. The Hybrid Conversion Algorithm

A novel conversion algorithm is presented that extends the hybrid optimization method by Jesick & Ocampo [3] to a new problem domain. When attempting to solve general optimal control finite-burn trajectory optimization problems (e.g. low-thrust orbit transfers) using a control law derived from the Euler-Lagrange equations and iterating on the initial co-state vector at ignition, difficulty arises in autonomously deriving a sufficient initial guess to enable solution convergence. Due to the extreme sensitivity of convergence to these initial co-states (i.e. the control), such optimal control methods are often difficult to generalize and automate, as prior knowledge of the problem domain is required.

The presented method mitigates the difficulty in generating the approximate initial co-state vector by evolving the optimal control TPBVP across multiple solution iterations, culminating in the desired problem definition. The algorithm does this by iteratively applying adjoint control transformations and slowly introducing the full constraint vector with each iterate. This provides an increasingly more accurate approximation of the initial co-states, which by the end of the process become sufficiently close to the optimal values, enabling solution convergence for the desired problem. The method requires no prior knowledge of the solution, merely a multi-impulse or suboptimal finite-burn initial guess, which aids automation.

An algorithm is herein described that converts a set of multiple-impulses or parameter steered sub-optimal finite-burns, representing the entirety or a portion of a high- or low-thrust maneuver, to an exact time optimal finite-burn trajectory for a thrust limited, constant exhaust velocity spacecraft. This can be called a Pseudo-Rendezvous (P-R) problem, as it yields a finite-burn solution whose final time, position and velocity state is equal to that of the original post-multi-impulse trajectory.

A conceptual overview of the conversion procedure is shown in Fig. 2. First, an approximate multi-impulse (or sub-optimal finite-burn) trajectory is used as input to the algorithm. Each burn is then replaced with a time-optimal finite-burn computed using the single impulse conversion method by Jesick and Ocampo [3], targeting the ballistic arcs between neighboring burns. In some circumstances, this may yield a Finite-Burn Arc Overlap, which is said to exist when the coast time between two sequential finite-burn segments is negative. This problem can arise when patching together a series of single-impulse conversions with large ΔV s too close together in time (relative to the engine performance), meaning the bounded time between impulses is not sufficient for the engine to produce the required finite-burn ΔV .

The Finite-Burn Arc Overlap problem is mitigated through a process of introducing additional boundary constraints, and by relaxing the pseudo-rendezvous state targeting on all interior conversions, leaving the state targeting for only the final impulse conversion. These redefinitions of the optimal control TPBVP allow the finite-burns to be re-optimized, while ensuring the final P-R targeting is upheld, which is the stated goal of the procedure. This process also has the benefit of improving optimality for the entire sequence of burns, as by definition each finite-burn was optimized individually, not as a member of the larger set of conversions. By evolving the targeting constraints for the entire set, regardless if any overlaps exist, the optimizer is afforded greater freedom when minimizing the full multi-finite-burn problem, as the burn out states of each finite-burn are no longer constrained to be on the inherently sub-optimal impulsive input trajectory.

The adjoint-control transformation central to the solution method represents a mapping across three fundamental burn models: Impulsive, Sub-Optimal Control (SOC), and Optimal Control Theory (OCT) control laws [Fig. 3]. The impulsive model has the thrust vector comprised of ΔV magnitudes and spherical angular directions, with instantaneous mass change. The SOC model represents a sub-optimal finite-burn, with thrust directions described as a function of polynomial-based spherical steering angles. The OCT model represents an optimal control finite-burn, which requires the numerical integration of the Euler-Lagrange system of equations from the standard formulation of the optimal control TPBVP.

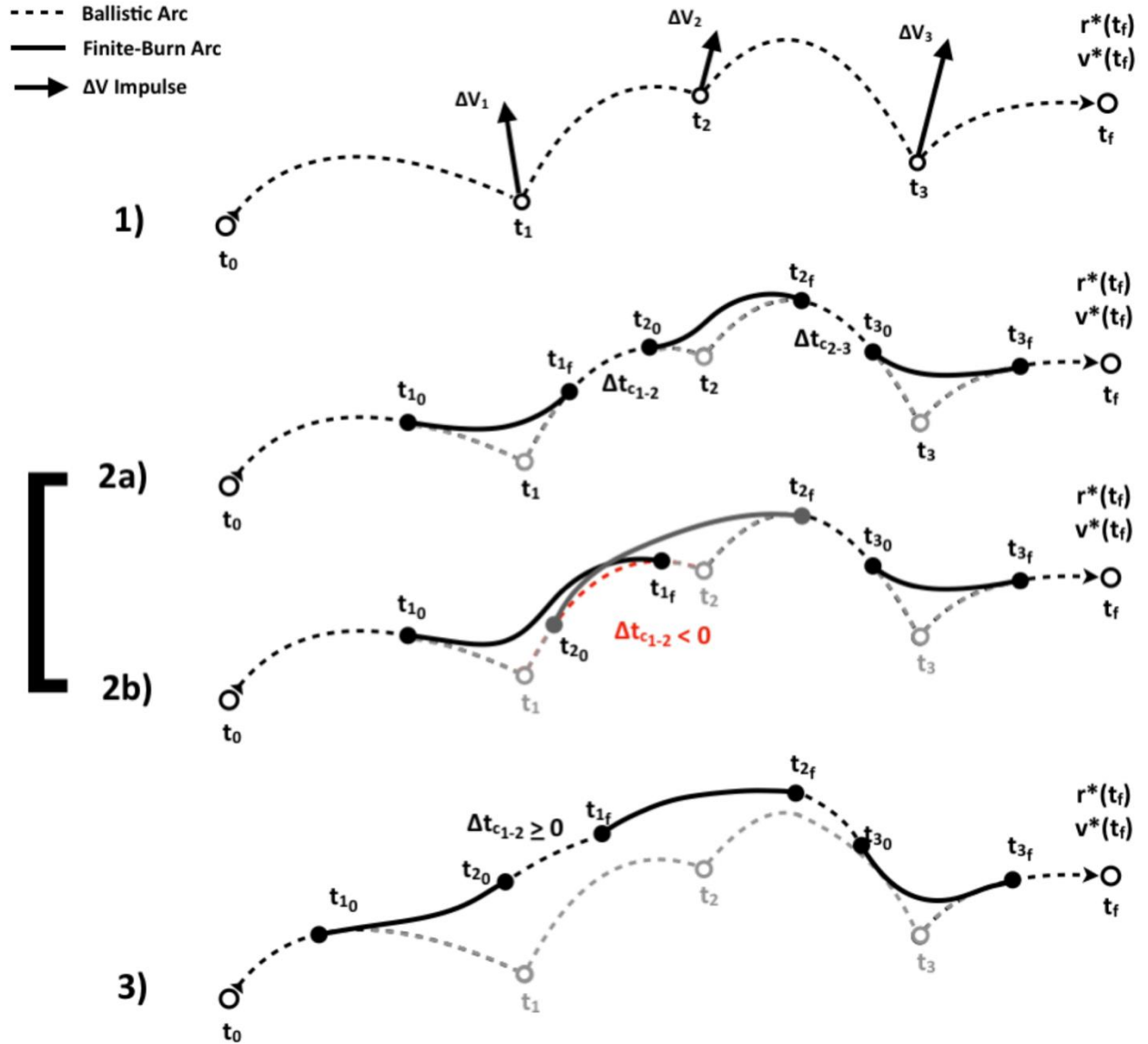


Fig. 2: Conceptual Overview of Multi-Impulse to Multi-Finite-Burn Conversion Algorithm. 1) Multi-impulse trajectory input, 2a) Set of converted impulses, targeting interior ballistic arcs, 2b) Existence of finite-burn arc overlap, with negative coast time on interior ballistic arc, 3) Re-targeted solution, relaxing targeting constraints and adding positive interior coast time constraints.

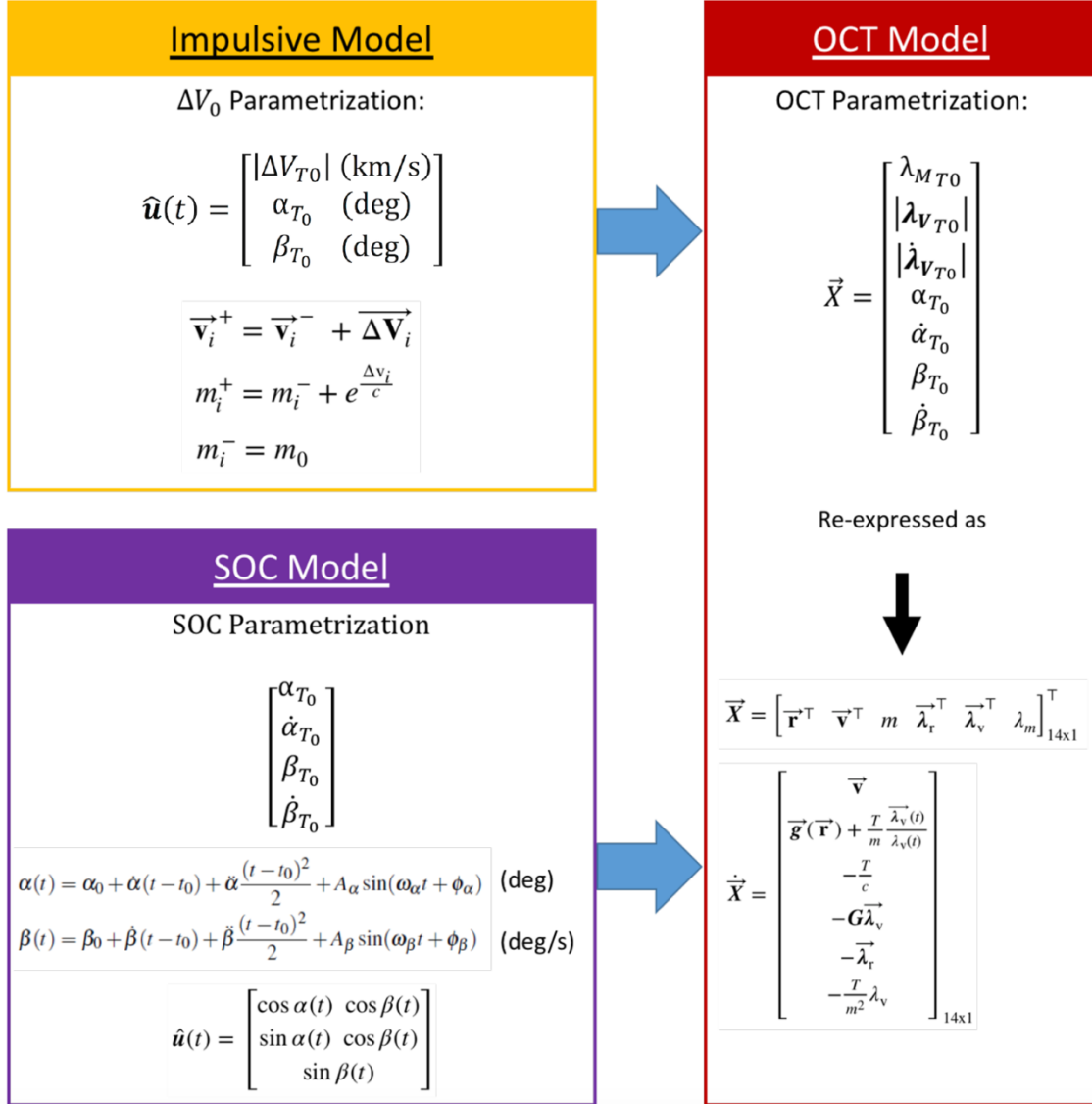


Fig. 3: Conversion algorithm burn model parameterizations. Mapping from Impulsive & SOC thrust control to OCT initial co-state vector.

Selecting the proper parameterizations within these models allows for the transfer of solution information from one to another, which can be used to estimate the initial co-state vector for each finite-burn conversion. Instead of attempting to estimate the full vector (7x1), selecting the parameterization shown in Fig. 3 replaces either 2 or 4 of these values with thrust angles which can be inferred from the control parameterizations of the input trajectory. This leaves only 3 additional values that are needed to complete the estimate of the initial co-state vector, one of which is known from Jesick & Ocampo's single impulse conversion strategy [3]: $|\lambda_{v_{i0}}| = 1.0$. Through experimentation, it was found that setting $|\dot{\lambda}_{v_{i0}}| = \lambda_{m_{i0}} = 0$ provided suitable convergence for the problems explored. Monotonic Basin Hopping, i.e. a bounded randomization of the control variables [11], was utilized to encourage convergence in the event that the optimizer exited prematurely.

The last piece of theory needed to construct the multi-finite-burn conversion algorithm is how to express the P-R targeting within the TPBVP. As previously discussed, the post-impulse or SOC ballistic arc is assumed to be defined by the input trajectory. As such, this ballistic arc can be treated as a known function of time, enabling a simplification of the boundary constraints. However, to implement this within the conversion algorithm, a pseudo segment was added to the trajectory structure to provide a variable state for the finite-burn to target. This *Pseudo-Rendezvous Segment* is

shown in Fig. 4. The P-R segment is defined as inheriting the time and state immediately after the ΔV impulse and has a final time TF equal to the TF of the OCT finite-burn segment. This means the OCT burn-out time will always be identical to the TF of the P-R segment. With identical TF s, the OCT FB segment can now target its final state to be equal to that of the P-R segment. Satisfying this constraint ensures that the post-OCT FB state and time are identical to that of the original input trajectory, satisfying the requirement that the conversion yield a pseudo-rendezvous trajectory.

Now the full multi-finite-burn conversion algorithm can be constructed, as shown in Fig. 5. The algorithm is comprised of 3 key evolutions of the optimal control problem (Tables 1—3), which each seed the next Sub-Problem (SP). This structure was found to aid in solution convergence, as each interim solution guides the optimizer closer to the optimal initial co-state vector. Attempts to skip immediately to the final optimization sub-problem were met with peril due to the sensitivity of the initial co-states, and significant difference between the input trajectories and the true solutions to the TPBVP.

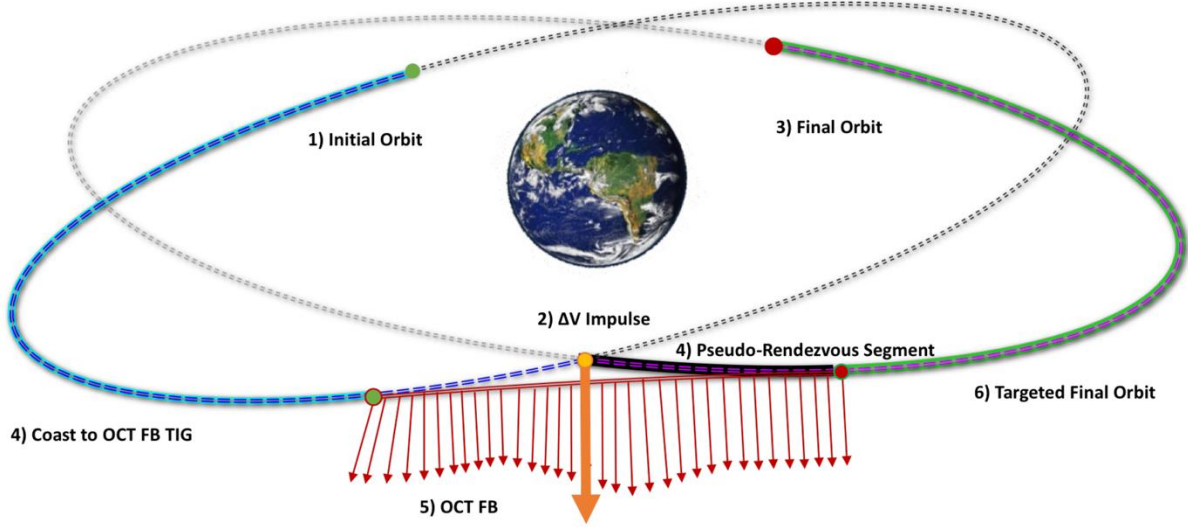


Fig. 4: Implementation of Single Impulse Conversion, with added Pseudo-Rendezvous Segment used for burn-out targeting.

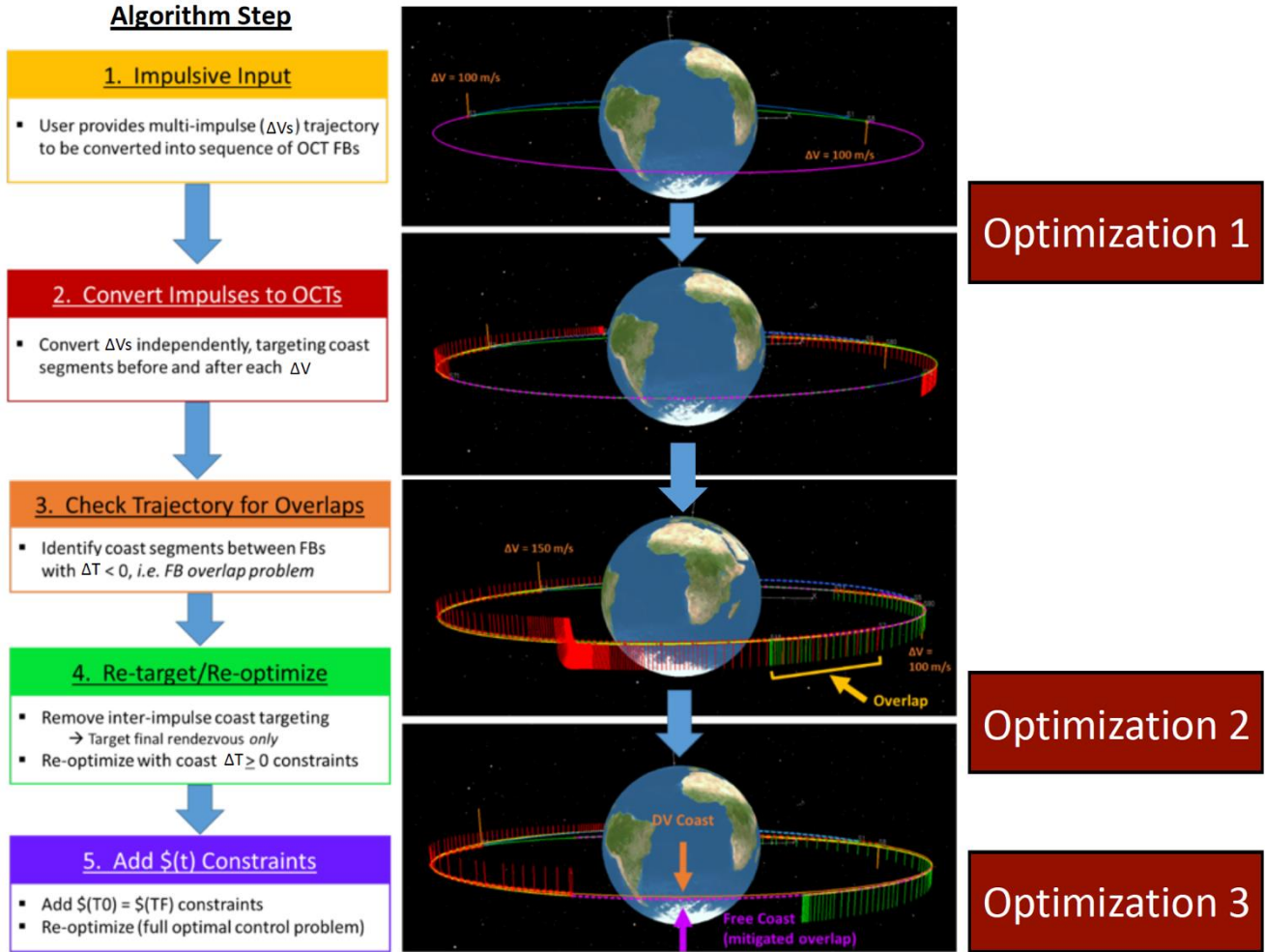


Fig. 5: Multi-Finite-Burn Conversion Algorithm, with example low-thrust inclination change problem. Note the addition of the final equality constraint, switching function $\$(t)$, in Step 5, to avoid over-constraining the earlier - optimization sub-problems. The 3 Optimization Sub-Problems are identified (right).

Table 1: Optimization Sub-Problem 1, with state & co-state dynamics as defined in Section II, for a set of n-number of burn conversions. The selection of constraints represents a majority of the full TPBVP constraint vector, reduced to not over-constrain the problem. Note an equal number of controls and constraints (i.e., problem is square).

EXAMPLE: For a 3-burn conversion, the control vector would be 24x1, with the constraint vector 24x1.

Objective Function		Control Vector	
$J = \min \sum (t_{i_f} - t_{i_0})_n$	Variable	Initial Guess	
	t_{i_0}	t_{i_0} estimate from input	
	Δt	$(t_{i_f} - t_{i_0})$ estimate from input	
	$\lambda_{M_{i_0}}$	0.000	
	$ \vec{\lambda}_{v_{i_0}} $	0.000	
	α_{i_0}	α_{i_0} from input	
	$\dot{\alpha}_{i_0}$	0 OR $\dot{\alpha}_{i_0SOC}$ from SOC input	
	β_{i_0}	β_{i_0} from input	
	$\dot{\beta}_{i_0}$	0 OR $\dot{\beta}_{i_0SOC}$ from SOC input	
	Size of Control Vector:		(8 x # of burns) x 1
Constraint Vector		Size	
$\vec{X}_{i_f} = \vec{X}_{PR_f}$		(6 x # of burns) x 1	
$\lambda_{M_{i_f}} = 0.0$		(1 x # of burns) x 1	
$ \vec{\lambda}_{v_{i_f}} \geq 0.0$		(1 x # of burns) x 1	
Size of Constraint Vector:		(8 x # of burns) x 1	

Table 2: Optimization Sub-Problem 2, with state & co-state dynamics as defined in Section II, for a set of n-number of burn conversions. The P-R targeting constraints are removed for all but the final burn, and a constraint is added to ensure interior coast times are positive. This frees the optimizer to find cheaper neighboring solutions to those found in Sub-Problem 1, and reduces the number of constraints to always be less than the number of controls (for multiple-finite-burn conversions), in preparation for Sub-Problem 3.

EXAMPLE: For a 3-burn conversion, the control vector would be 24x1, with the constraint vector 14x1.

Objective Function		Control Vector	
$J = \min \sum (t_{if} - t_{i0})_n$	Variable	Initial Guess	
	t_{i0}	from Sub-Problem 1 solution	
	Δt	$(t_{if} - t_{i0})$ from Sub-Problem 1 solution	
	$\lambda_{M_{i0}}$	from Sub-Problem 1 solution	
	$ \vec{\lambda}_{v_{i0}} $	from Sub-Problem 1 solution	
	α_{i0}	from Sub-Problem 1 solution	
	$\dot{\alpha}_{i0}$	from Sub-Problem 1 solution	
	β_{i0}	from Sub-Problem 1 solution	
	$\dot{\beta}_{i0}$	from Sub-Problem 1 solution	
Size of Control Vector:		(8 x # of burns) x 1	
Constraint Vector		Size	
$\vec{X}_{i_{fFinal-Burn}} = \vec{X}_{PR_{fFinal-Burn}}$		6 x 1 (final burn only)	
$\Delta T_{coast} \geq 0$		(1 x # of burns - 1) x 1	
$\lambda_{M_{if}} = 0.0$		(1 x # of burns) x 1	
$ \vec{\lambda}_{v_{if}} \geq 0.0$		(1 x # of burns) x 1	
Size of Constraint Vector:		(3 x # of burns + 5) x 1	

Table 3: Optimization Sub-Problem 3, with state & co-state dynamics as defined in Section 2, for a set of n-number of burn conversions. Note the addition of the switching function constraint, completing the multiple-finite-burn optimal control TPBVP. The reduced P-R constraints are maintained, ensuring the number of constraints will still be always less than the number of controls (for multi-finite-burn conversions), and the interior coast time constraint is maintained.

EXAMPLE: For a 3-burn conversion, the control vector would be 24x1, with the constraint vector 17x1.

Objective Function		Control Vector	
$J = \min \sum (t_{if} - t_{io})_n$	Variable	Initial Guess	
	t_{i0}	from Sub-Problem 2 solution	
	Δt	$(t_{if} - t_{i0})$ from Sub-Problem 2 solution	
	$\lambda_{M_{i0}}$	from Sub-Problem 2 solution	
	$ \vec{\lambda}_{v_{i0}} $	from Sub-Problem 2 solution	
	α_{i0}	from Sub-Problem 2 solution	
	$\dot{\alpha}_{i0}$	from Sub-Problem 2 solution	
	β_{i0}	from Sub-Problem 2 solution	
	$\dot{\beta}_{i0}$	from Sub-Problem 2 solution	
Size of Control Vector:		(8 x # of burns) x 1	
Constraint Vector		Size	
$\vec{X}_{i_{fFinal-Burn}} = \vec{X}_{PR_{fFinal-Burn}}$		6 x 1 (final burn only)	
$\Delta T_{coast} \geq 0$		(1 x # of burns - 1) x 1	
$\lambda_{M_{if}} = 0.0$		(1 x # of burns) x 1	
$ \vec{\lambda}_{v_{if}} \geq 0.0$		(1 x # of burns) x 1	
$\$(t_{if}) = \$(t_{io})$		(1 x # of burns) x 1	
Size of Constraint Vector:		(4 x # of burns + 5) x 1	

IV. Application Problems

For the subsequent application problems, the conversion algorithm was modeled in NASA JSC's in-house orbit trajectory optimization software *Copernicus 4.6* [7], however the procedure may be implemented within any trajectory optimization schema. Examples are shown for low-thrust apogee raise maneuvers, non-coplanar Earth orbit transfers, a modified Deep Space 1 low-thrust trajectory. Results are compared to conventional Direct SOC solutions. The Hybrid OCT solutions are shown to offer a numerically significant improvement to objective cost savings across all application problems, as well as a notable improvement to convergence speed for a select class of problems.

A. Low Thrust: Apogee Raise

The first introductory test problem is a simple two-burn, low-thrust apogee raise maneuver sequence. An input trajectory was generated with two ΔV impulses occurring at perigee on consecutive revolutions, starting from an initial orbit with elements: $a_0 = 16,378.14$ km, $e_0 = 0.0$, $i_0 = 28.474$ deg, $\Omega_0 = \omega_0 = 0.0$ deg, with a final orbit with elements: $a_f = 17,855.220$ km, $R_{Pf} = 16,378.14$ km, $R_{Af} = 19,332.299$ km, $e_f = 0.082725$, $i_f = 28.474$ deg, $\Omega_f = 0.0$ deg, $\omega_f = 170.856$ deg. The ΔV vectors are aligned in the velocity vector direction, ensuring pure apogee raise. The identical ΔV magnitudes provide a total change in apogee altitude of 2,954.159 km. A SOC solution and OCT solution were generated using the multi-finite-burn conversion algorithm, whose solutions are detailed in Fig. 6 and Table 4. For this simple problem, the thrust profiles are nigh identical, however the OCT solution manages to save 7.2 sec of burn time, or an equivalent 5 grams of propellant mass. While not a substantial savings in cost, this does validate the conversion algorithm as being able to provide a solution at least as good as a SOC solution, and in this case marginally improved. Moreover, it was found that converted finite-burns whose burn Δt are less than about a quarter of a revolution are more likely to yield comparable SOC and OCT solutions. Only when longer duration burns (relative to orbit period) or more complex maneuvers are demanded does the hybrid conversion algorithm yield appreciable benefits over SOC-type solutions.

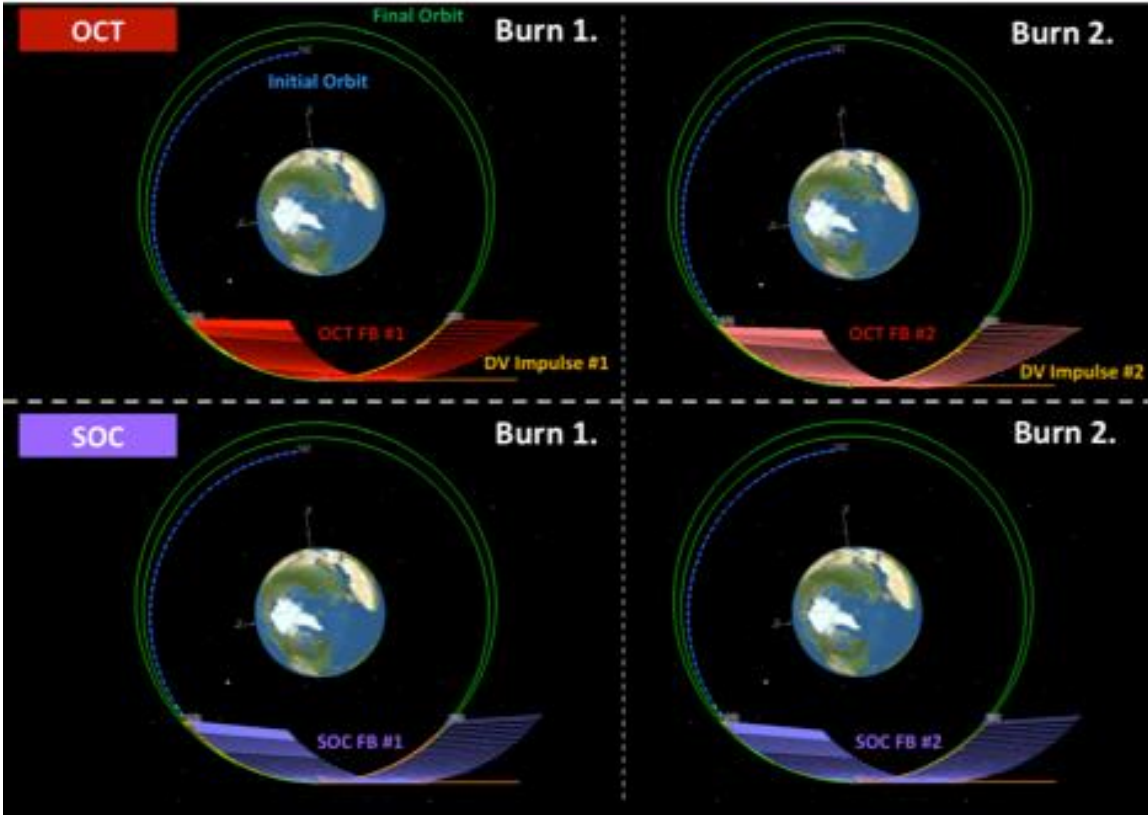


Fig. 6: Solution comparison of two-burn, apogee raise conversion from multi-impulse input.

Table 4: Two-burn, low-thrust apogee raise solution burn parameters.

Impulsive Parameters (per impulse)		
Δi	1.146 deg	
$ \overrightarrow{\Delta V} $	100 m/s	
α_{i_o}	0.0 deg	
β_{i_o}	-/+ 90.0 deg	
SOC Parameters		OCT Parameters
Thrust	19.2 N	19.2 N
I_{sp}	2,500 sec	2,500 sec
Total Burn Δt	3 hrs 18 min 57.2 sec	3 hrs 18 min 13.8 sec
M_{i_o}	1,000 kg	1,000 kg
Propellant Used	9.349 kg	9.315 kg
Effective ΔV	229.8 m/s	229.51 m/s

B. Low-Thrust: Inclination Change

The next application problem is a two-impulse to low-thrust inclination change conversion from an initial circular orbit with elements: $a_0 = 16,378.14$ km, $e_0 = 0.0$, $i_0 = 28.474$ deg, $\Omega_0 = \omega_0 = 0.0$ deg. Two ΔV impulses (100 m/s each) were placed half a revolution apart, with thrust directions -/+ 90 deg out-of-plane to yield a change in inclination of $\Delta i_{imp} = + 1.146$ deg per impulse, or 2.292 deg total. The optimization SP-1 solution is shown in Fig. 7 and Table 5. As with the first application problem, the OCT and SOC solutions are comparable, with a slight benefit to OCT in terms of iteration time and optimality. Note the OCT solution's savings in cost of 43.4 sec of burn time, or 34 grams of propellant, over the SOC solution. This represents a larger cost savings compared to the apogee raise problem, since the converted finite-burns for this problem have longer durations for the same spacecraft model.

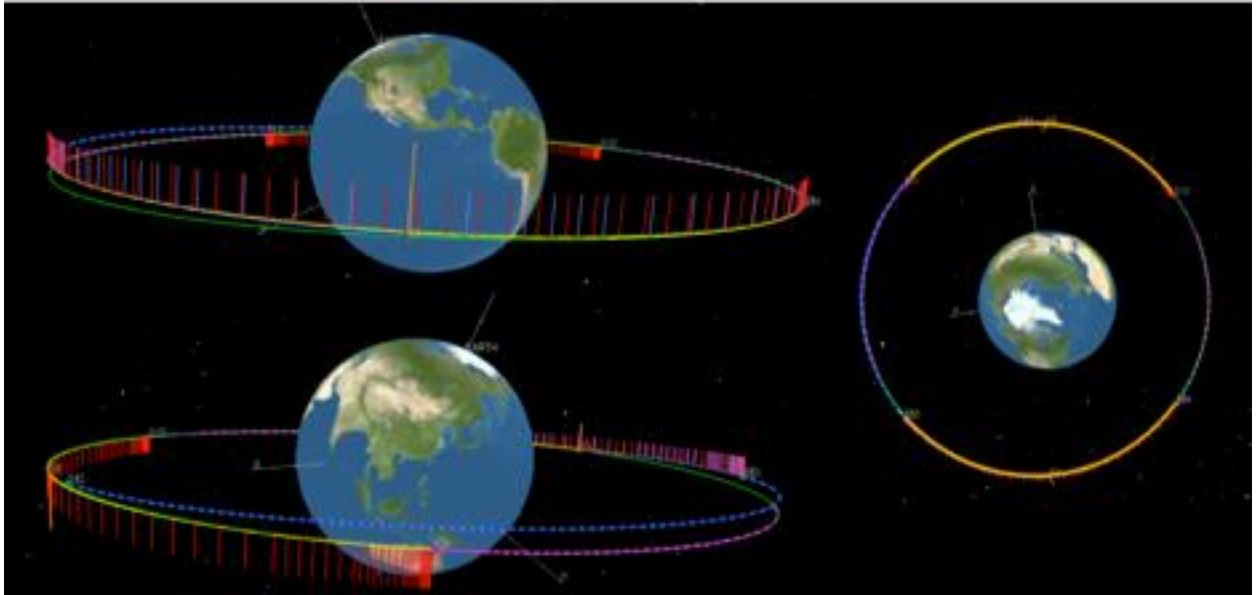


Fig. 7: Two-impulse inclination change conversion, Optimization Sub-Problem 1 solution. OCT thrust vectors (red) and SOC thrust vectors (purple) are shown, with initial orbit (blue dashed) and final orbit (magenta dashed). A planar projection (right) shows the two finite-burn arcs (yellow).

Table 5: Two-burn, low-thrust inclination change Sub-Problem 1 solution burn parameters.

Impulsive Parameters (per impulse)		
Δi	1.146 deg	
$ \overrightarrow{\Delta \mathbf{V}} $	100 m/s	
α_{i_o}	0.0 deg	
β_{i_o}	$-/+ 90.0$ deg	
	SOC Parameters	OCT Parameters
Thrust	19.2 N	19.2 N
I_{sp}	2,500 sec	2,500 sec
Total Burn Δt	3 hrs 18 min 57.2 sec	3 hrs 18 min 13.8 sec
M_{i_o}	1,000 kg	1,000 kg
Propellant Used	9.349 kg	9.315 kg
Effective ΔV	229.8 m/s	229.51 m/s

To fully exercise the conversion algorithm, it was desired to induce a finite-burn arc overlap. This was done by increasing the first ΔV impulse to 150 m/s, resulting in a longer finite-burn and larger change in inclination ($\Delta i = +1.718$ deg). The overlap is shown in Figure 8 (SP-1 solution). After solving SP-2 and 3 (evolving the constraint vector by relaxing the P-R targeting, and then adding the switching function constraints), the overlap is eliminated and the full multi-finite-burn TPBVB solved (Figure 9, Table 6). Note that in addition to mitigating the finite-burn arc overlap problem, the Hybrid OCT SP-3 solution also yields a lower cost compared to that of SP-1, illustrating the benefits of solving all 3 sub-problems, regardless of the existence of an overlap. Note that the Direct SOC solution was unable to converge to the first sub-problem but was able to converge to SP-2. This is discussed in greater detail in subsequent sections.

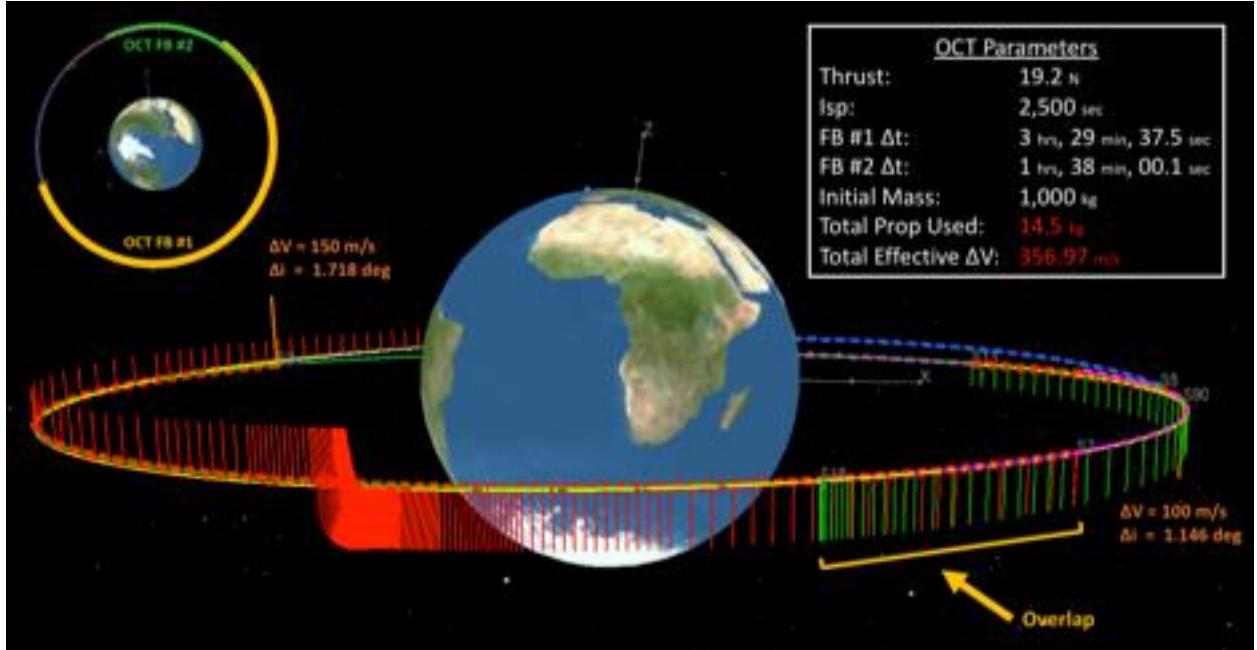


Fig. 8: Finite-Burn Arc Overlap Problem for two-impulse inclination change conversion. Optimization Sub-Problem 1 solution. Note the inefficiency in the first burn thrust direction slew (red).

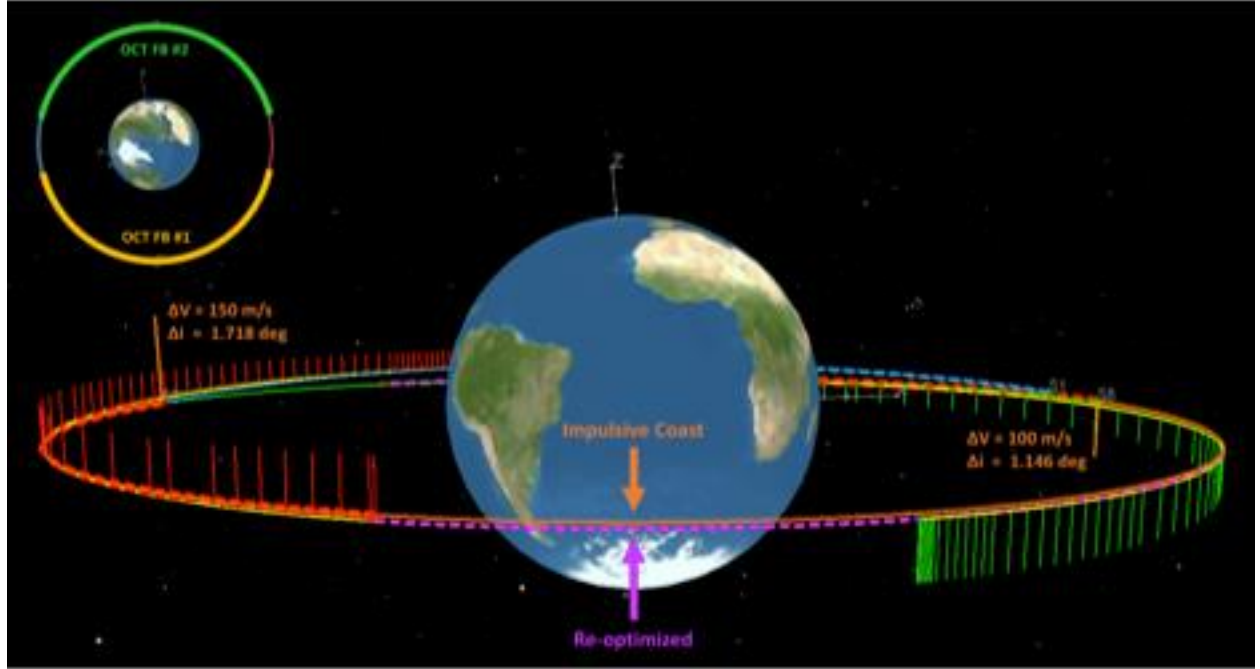


Fig. 9: Mitigated Finite-Burn Arc Overlap (Optimization Sub-Problem 3 solution) for two-impulse inclination change conversion.

Table 6: Two-burn, low-thrust inclination change Sub-Problem 3 solution burn parameters.

Impulsive Parameters (per impulse)		
Δi	1.146 deg, 1.718 deg	
$ \overrightarrow{\Delta V} $	150 m/s, 100 m/s	
α_{i_o}	0.0 deg	
β_{i_o}	-/+ 90.0 deg	
SOC Parameters		OCT Parameters
Thrust	19.2 N	19.2 N
I_{sp}	2,500 sec	2,500 sec
Finite-Burn Δt_1	2 hrs 27 min 55.7 sec	2 hrs 27 min 55.1 sec
Finite-Burn Δt_2	2 hrs 27 min 46.1 sec	2 hrs 27 min 44.2 sec
M_{i_o}	1,000 kg	1,000 kg
Propellant Used	14.013 kg	13.892 kg

Table 7 compares optimization solution data between the Hybrid and Direct Methods for the 2-burn LEO Δi with overlap. The first observation is that the Hybrid Method converges for all three of the sub-problems, while the Direct Method only converges for the second sub-problem (and is incapable of solving the third, as it involves adding the switching function constraint, which is not a part of the SOC control law). As such, the remainder of the table only compares solutions after the first and second sub-problems. Despite solving two problems instead of one, the Hybrid Method still converges to a final SP-2 solution in a total of 128 iterations, 25 iterations faster than the Direct Method.

Table 7: Comparison of optimization parameters between Hybrid OCT and Direct SOC for 2-burn LEO Δi application problem w/ FB overlap. Green = superior solution, red = lesser solution. Note the faster convergence of the Hybrid OCT solution in terms of iteration count and computer sim time, as well its cheaper propellant mass solution.

	Hybrid OCT	Direct SOC
SP-1 Solution Convergence	YES	NO
SP-2 Solution Convergence	YES	YES
SP-3 Solution Convergence	YES	N/A
SP-1 # of Iterations	57	N/A
SP-2 # of Iterations	71	153
Total # of Iterations after SP-2	128	153
SP-1 Convergence Time	4.82 sec	N/A
SP-2 Convergence Time	4.50 sec	15.16 sec
Total Convergence Time after SP-2	9.32 sec	15.16 sec
Min Prop Mass after SP-2	13.926 kg	14.013 kg

This also equates to a 38.5% reduction in convergence time, a substantial benefit when it comes to automation. This improvement can be attributed to the overlap inherent to the SP-1 solution, which also explains the inability of the Direct Method to converge on SP-1. The overlap in this problem yields an OCT SP-1 solution with a burn spanning greater than half a revolution:

$$\Delta T_{\text{burn}} > \frac{T_{\text{Period}}}{2} \quad (31)$$

This means that due to the targeted plane change with each burn roughly centered around the line of nodes, the converged OCT thrust vector slews nearly 180° during the burn arc as the spacecraft nears the opposite node to maintain consistent change in inclination (which can be seen in Fig. 8). This rapid change in thrust direction is easily handled by the Hybrid OCT control law (derived from the Euler-Lagrange equations), but the linearly-steered SOC control law is incapable of such a slew in such a short fraction of the orbit period. This demonstrates that the Hybrid OCT Method will yield superior convergence behavior for conversion problems where both the burn is less than half a revolution, and the burn is changing inclination. This specific class of problem will therefore reap larger benefits than other problems when the Hybrid OCT Method is employed in lieu of Direct SOC.

C. Low-Thrust: Modified Deep Space 1 Mission to Near-Earth Asteroid 9969 Braille

In order to further validate the capabilities of Hybrid OCT Method, an interplanetary conversion problem was derived based closely on the flown trajectory of NASA's Deep Space 1 (DS1), launched October 24th, 1998 [12-14]. The technology demonstration mission was designed to test high risk systems, namely an electrostatic ion thruster. During the mission, DS1's propulsion system was shown to provide 90 mN of thrust in vacuum, with a demonstrated specific impulse of 3,100 seconds [14]. The launch mass of the vehicle was 486 kg, of which there was 82 kg of Xenon propellant. DS1 was launched using a Boeing Delta II 7326 rocket, stacked with a Star-37FM third-stage apogee kick motor (AKM). This third-stage provided the ΔV necessary to escape from its LEO parking orbit and onto a heliocentric trajectory. The Star-37FM stage provided 63 seconds of burn time at a thrust of 47.26 kN, a specific impulse of 289.8 seconds, a launch mass of 1,147.4 kg, and a burn-out mass of 81.5 kg for the stage.

The first destination of DS1 was a fly-by of the near-Earth asteroid (NEA) 9969 Braille. During the first outbound interplanetary transfer leg, DS1 fired its ion thruster twice, first for only 4.5 min on November 10th, 1998, and then for 14 days starting November 24th, 1998. DS1 made its closest approach to 9969 Braille on July 29, 1999 at a relative distance and velocity of 26 km and 15.5 km/s, respectively. For the purposes of this dissertation, two larger burns were designed for this outbound leg to better exercise the hybrid conversion method, while maintaining a comparable fly-by geometry of 9969 Braille. Note that the goal was not to replicate the DS1 trajectory, but to develop a test problem comparable to a flown mission.

Taking inspiration from the DS1 trajectory, a set of Earth-escape orbital parameters was computed using the spacecraft design of DS1 and the Star-37FM AKM. Two impulsive ΔV s of equal magnitude (based on achievable propellant expenditure during the outbound time interval) were then set for November 10, 1998, and March 22, 1999. The spherical angles α and β , dictating the ΔV and V_∞ directions in heliocentric space within a spacecraft-centered frame, were then optimized to target a fly-by geometry close to that of the flown DS1 trajectory. The parameters of this computed impulsive trajectory are shown below.

$$\text{Earth Escape } V_\infty = 2.9489 \text{ km/s}$$

$$\alpha_\infty = -14.8047 \text{ deg}$$

$$\beta_\infty = -62.9735 \text{ deg}$$

where α_∞ and β_∞ are expressed in the Sun J2000 VUW coordinate frame, as defined by the spacecraft's velocity, angular momentum cross velocity, and angular momentum directions, respectively.

$$\Delta V1 = \Delta V2 = 1.650806 \text{ km/s}$$

$$\alpha_1 = 103.6443 \text{ deg}$$

$$\beta_1 = 3.1642 \text{ deg}$$

$$\alpha_2 = 97.5617 \text{ deg}$$

$$\beta_2 = 102.7469 \text{ deg}$$

where $\alpha_{1,2}$ and $\beta_{1,2}$ are expressed in the Sun J2000 IJK frame.

This two-burn impulsive trajectory can be seen in Fig. 10, along with the orbit of asteroid 9969 Braille. The approximation of the modified DS1 trajectory was then input to the Hybrid Conversion Algorithm, and separately solved using the traditional Direct SOC Method. The SP-1 Hybrid OCT solution is shown in Fig. 11, while the SP-2 solution compared to the Direct SOC solution is shown in Fig. 12 and Table 8.

It should be noted that the modified DS1 impulsive trajectory was designed to purposely induce a FB overlap to further exercise the conversion method, as seen in Fig. 11. The overlap between burns 1 and 2 is found to be 39.069 days, significantly longer than the overlap in application problem 2. For comparison, Fig. 13 shows the first leg of the flown DS1 trajectory (note the former name of asteroid 9669 Braille, asteroid 1992 KD). After executing two burns with its electrostatic thruster during this outbound segment (4.5 min and 14 days, respectively), this equates to roughly 3.581 kg of propellant used, compared to over 50 kg in the computed trajectories. Recall that the purpose of this work was not to replicate the DS1 trajectory, but to use it as a foundation to validate the conversion algorithm with a realistic, flown spacecraft. This required increasing the burn time to make the conversion process more significant. That being said, the modified trajectory is still comparable in terms of the mission objectives considered for this study (targeting the NEA fly-by) and does not exceed the flown propellant mass budget. After SP-2 conversion, the difference in used propellant shows a 0.12% improvement when using the Hybrid OCT Method over Direct SOC. The numerical validity of this will be investigated in the next section.

Before that critical discussion, however, it must first be emphasized that the Direct SOC solution failed to converge for SP-1 and SP-2. This is due to the inability of the SOC control law (linearly steered) to model the thrust direction flip shown to be associated with low-thrust FB overlaps for burns spanning a significant portion of a revolution. The only way the modified DS1 trajectory conversion was able to be solved using Direct SOC (in SP-2) was to use the converged Hybrid OCT SP-2 solution in a reverse-adjoint control transformation (i.e. use the Hybrid OCT solution as an initial guess for the Direct SOC solution). This re-seeded, "assisted" Direct SOC solution is presented in Table 8, but it should be noted that such a solution would not be available were it not for the successful performance of the Hybrid OCT Method. This means that for all practical considerations, the Direct SOC Method completely failed to solve the modified DS1 conversion problem, while the Hybrid OCT Method showed no issue, once again demonstrating its superiority for the classes of problems explored.

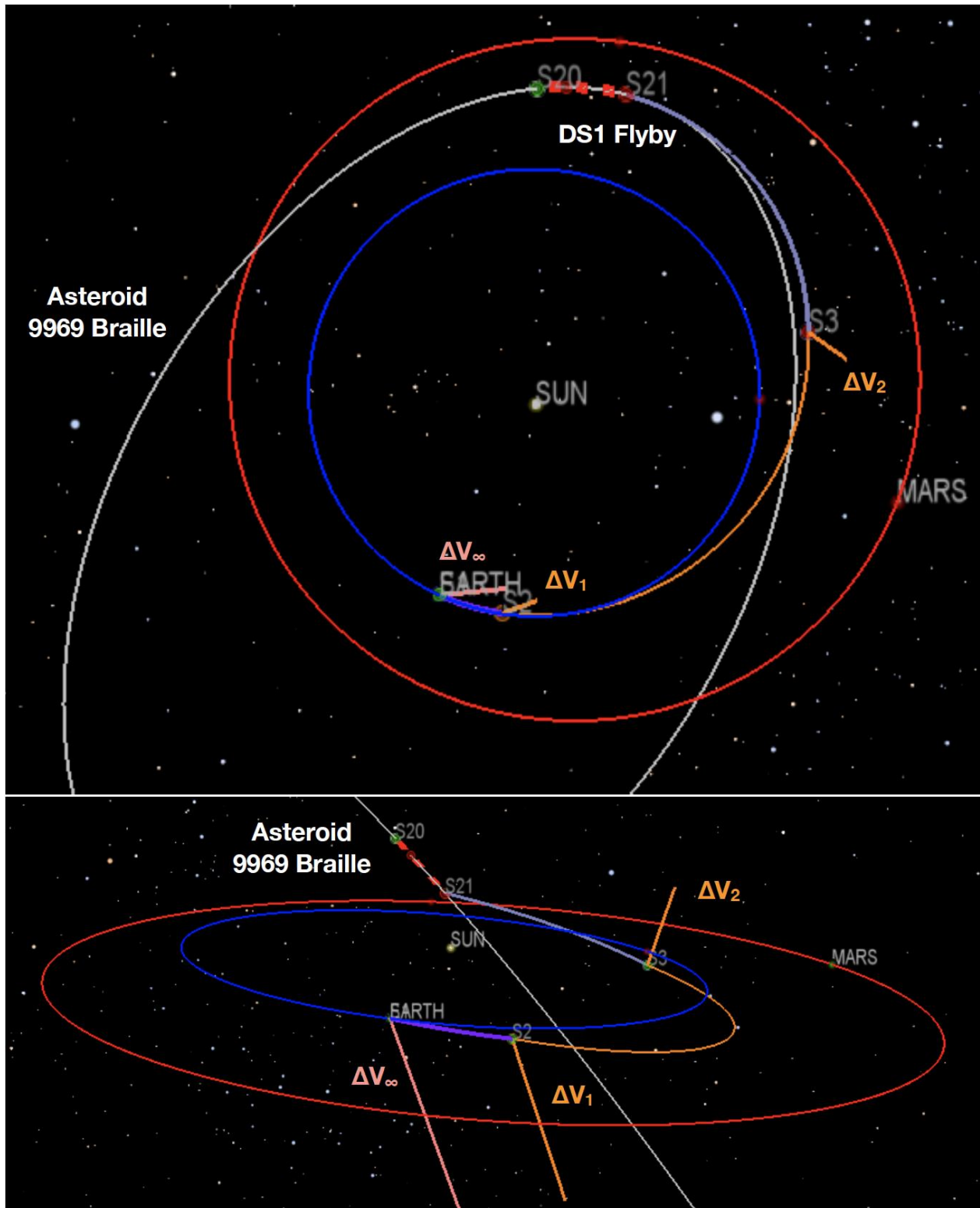


Fig. 10: Impulsive Approximation of Modified DS1 Trajectory, with increased ΔV . Earth, Mars, and asteroid 9969 Braille orbits are shown in blue, red, and white, respectively. V_∞ post-AKM departure burn is shown.

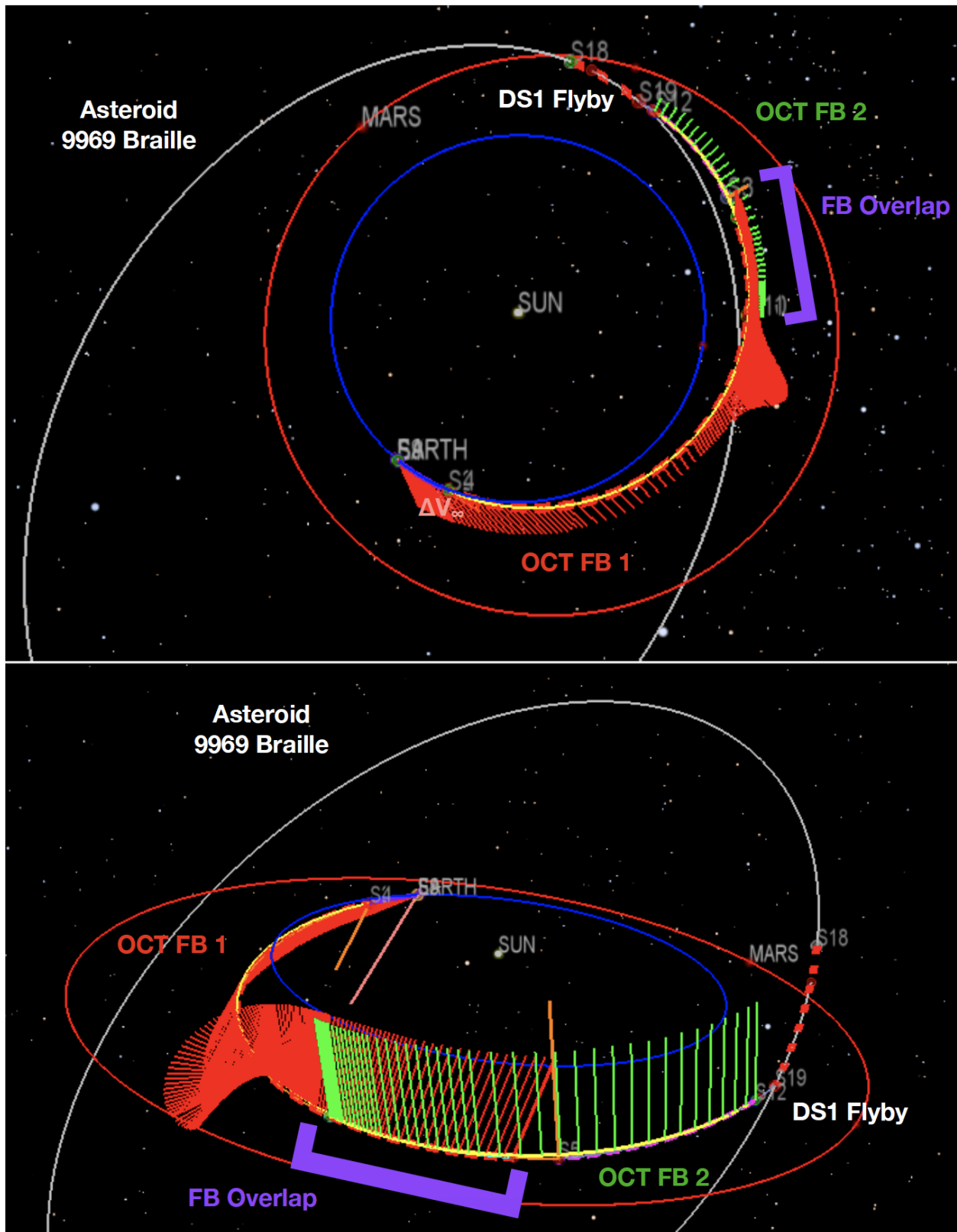


Fig. 11: Modified DS1 Trajectory, SP-1 Hybrid OCT Solution. Note the FB overlap of 39.069 days.

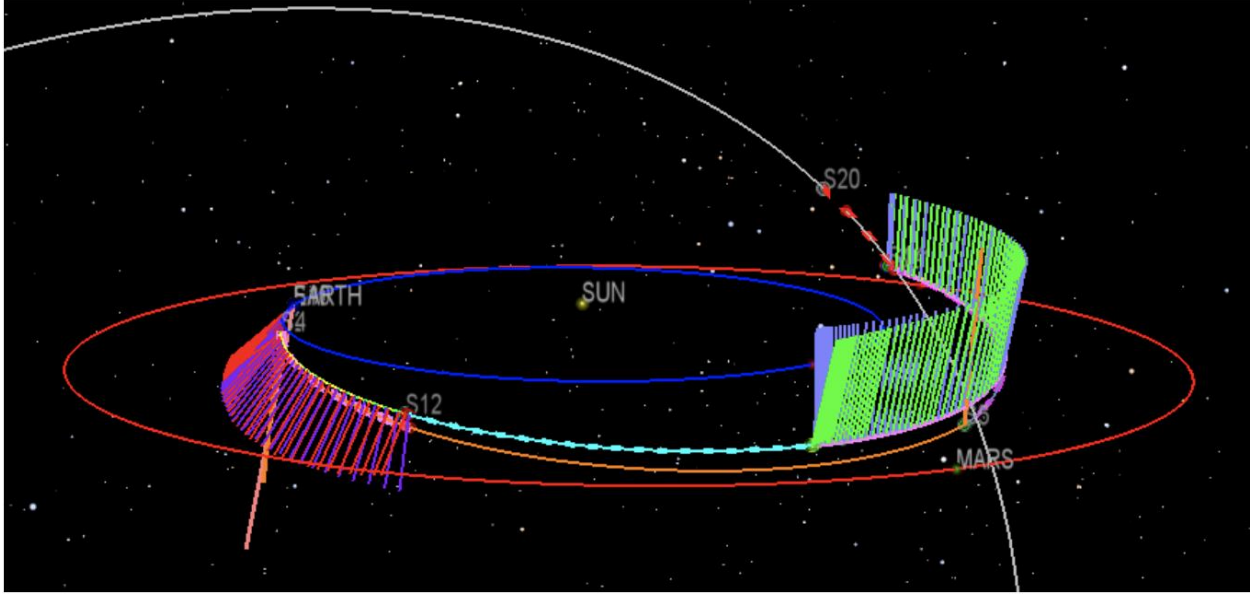


Fig. 12: Modified DS1 Trajectory, SP-2 Hybrid OCT Solution (red & green), and assisted SP-2 Direct SOC Solution (purple).

Table 8: Modified DS1 Trajectory, SP-2 Hybrid OCT and Assisted Direct SOC Solution Burn Parameters.

Impulsive Parameters		
$ \overline{\Delta \mathbf{V}}_1 $	1,650.806 m/s	
$ \overline{\Delta \mathbf{V}}_2 $	1,650.806 m/s	
$ \overline{\Delta \mathbf{V}}_{\text{Total}} $	3,301.612 m/s	
SOC Parameters (SP-2)		OCT Parameters (SP-2)
Thrust	90 mN	90 mN
I_{sp}	3,100 sec	3,100 sec
Finite-Burn Δt_1	61.113 days	60.574 days
Finite-Burn Δt_2	138.052 days	138.357 days
M_{i_o}	486 kg	486 kg
Propellant Used	50.943 kg	50.883 kg

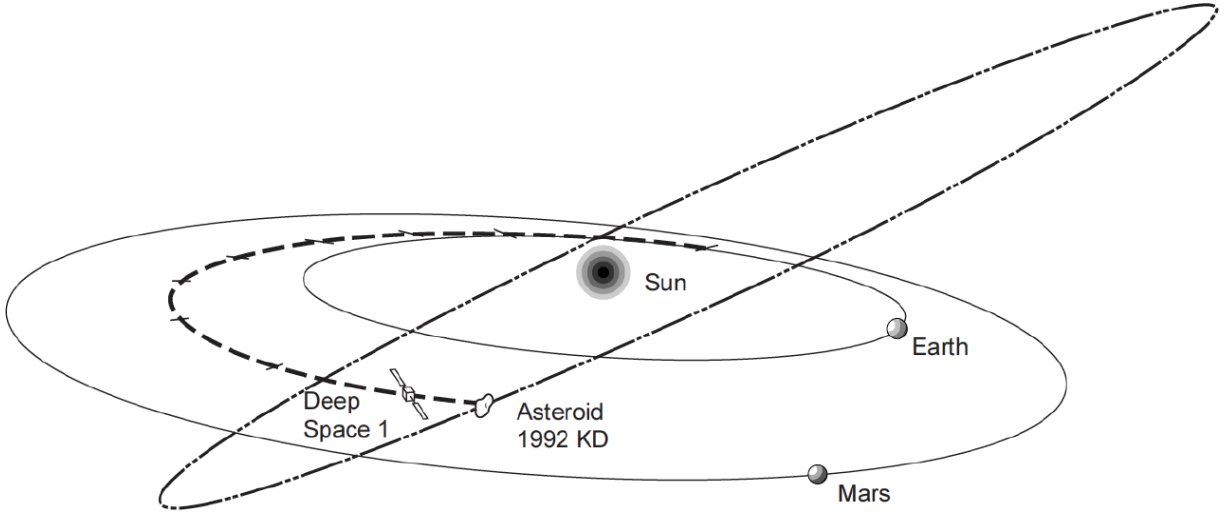


Figure 13: Flown Deep Space 1 Trajectory to Asteroid 1992 KD (now called 9969 Braille) [12].

D. Proof of Numerical Significance

It has been shown that using the Hybrid OCT Method instead of the traditional Direct SOC Method yields a roughly consistent objective cost savings of $\sim 0.1\%$ for all problems explored. In order for this generally small improvement of the Hybrid OCT Method to be considered numerically significant, i.e. not a result of numerical error or noise in the simulation, focus must be given to the integration error tolerances, and the optimization convergence tolerances.

First, as described in the Section III A), a variable order, variable step-size Adams-Basforth-Moulton integration method [8] was utilized for propagation, with absolute and relative error tolerances of 1.0×10^{-12} . The Simulink user's guide [15] describes these standard integration error tolerances as follows: during each time step, the variable-step solver computes the state values at the end of the step in order to determine the local error, i.e. the estimated error of the state values. The solver then compares the local error to the acceptable error, which is a function of both the relative tolerance (rtol) and the absolute tolerance (atol). If the local error is greater than the acceptable error for any one state, the solver reduces the step size and tries again. The relative tolerance measures the error relative to the size of each state, and represents a percentage of the state value. For example, a rtol value of 1×10^{-3} means that the computed state is accurate to within 0.1%. The absolute tolerance is a threshold error value, representing the acceptable error as the value of the measured state approaches zero.

This tells us that by setting these integration error tolerances to 1.0×10^{-12} , the computed state is accurate to within 0.0000000001%. As this error tolerance is nine orders of magnitude smaller than $\sim 0.1\%$, the difference in objective cost between the Hybrid OCT Method and Direct SOC Method cannot be attributed to integration error or noise in the propagated trajectory.

The next pieces needed to prove numerical significance are the optimization convergence tolerances, namely, we wish to verify that the converged solutions have acceptable accuracy relative to the observed difference in cost between solution types. The optimization package SNOPT utilizes an SQP algorithm with two key tolerances. First, a feasibility metric: the measure of how infeasible the constraint vector is at each iterate, or in other words, the maximum constraint residual. This specifies how accurately the nonlinear constraints are satisfied. The maximum nonlinear constraint violation is normalized by the size of the solution, and is required to satisfy:

$$\max_i \text{viol}_i / \|x\| \leq \epsilon_r \quad (32)$$

Where viol_i is the violation of the i^{th} nonlinear constraint, and ϵ_r is the major feasibility tolerance. The default value of 1×10^{-6} is appropriate when the linear and nonlinear constraints contain data to about that accuracy [64]. This violation is recorded as the feasibility metric for each iterate.

The second metric is the optimality tolerance (set to 1×10^{-3}). This metric specifies the final accuracy of the dual variables (Lagrange multipliers for the general constraints) based on an estimate of the maximum complementarity slackness variable (maxComp). This maxComp is computed from the final QP solution using the reduced gradients

derived from the objective gradient, the associated column of the constraint matrix, and the set of the QP dual variables. Both of these convergence metrics approach zero near a solution (i.e. local minima).

What this means, in essence, is that we must compare the feasibility and optimality metrics of the converged solutions with the size of the observed difference in objective cost. Table 9 presents the feasibility and optimality metrics for the final iterate of each of the application problems presented (i.e. the converged solutions for each). It can be clearly seen that without exception, all converged solutions boast feasibility and optimality violations that are several orders of magnitude smaller than the observed difference in objective cost ($\sim 0.1\%$), indicating sufficient convergence. This fact combined with the demonstrated numerical integration accuracy puts a nail in the coffin for the notion that the difference in cost between solutions is a result of numerical error or noise. It can now be confidently stated that the Hybrid OCT Method yields a numerically significant improvement in objective cost over the Direct SOC Method for all conversion problems explored. This benefit is in addition to the aforementioned 38.4% improvement in convergence time for conversion problems where both the burn is less than half a revolution, and the burn is changing inclination.

Table 9: Comparison of optimization convergence violations via Hybrid OCT and Direct SOC Methods for all four application problems: apogee raise, LEO ΔI w/ FB overlap, and Modified DS1. Note that for all problems, feasibility tolerance = $1.0E-06$, and optimality tolerance = $1.0E-03$.

Apogee Raise	Feasibility	Optimality
Hybrid OCT Method	9.2×10^{-7}	3.3×10^{-5}
Direct SOC Method	6.6×10^{-7}	1.5×10^{-5}
LEO ΔI, w/ FB Overlap	Feasibility	Optimality
Hybrid OCT Method	2.1×10^{-7}	7.3×10^{-4}
Direct SOC Method	1.9×10^{-8}	5.6×10^{-5}
Modified Deep Space 1	Feasibility	Optimality
Hybrid OCT Method	1.0×10^{-6}	8.8×10^{-7}
Direct SOC Method	1.9×10^{-8}	1.1×10^{-6}

V. Conclusion

A novel finite-burn trajectory optimization algorithm has been presented that offers objective cost savings, and optimization convergence benefits for certain class of problems, over a conventional method. The procedure combines the high-fidelity control law of an indirect optimization method with the generality of a direct method to improve upon solving time-optimal, finite-burn (FB) Pseudo-Rendezvous (P-R) spacecraft trajectory problems. The traditional difficulty in approximating the initial co-state vector within the optimal control Two-Point Boundary Value Problem (TPBVP) is mitigated by a unique series of adjoint-control transformations. These mappings evolve the constraint vector across multiple sub-problems (SP), culminating in the desired problem definition. This provides an increasingly more accurate approximation of the initial co-state vector, which by the end of the process become sufficiently close to the optimal values, enabling solution convergence for the desired P-R problem.

Focus was given to solving the FB Arc Overlap problem, which can arise when patching together a series of single-impulse conversions with insufficiently spaced ΔV magnitudes, relative to engine performance. The FB overlap problem results in the ballistic arc connecting two converged finite-burns having a negative time of flight, yielding an infeasible trajectory. The conversion algorithm mitigates this through a process of introducing additional boundary constraints, and by relaxing the P-R state targeting on all interior conversions, leaving the state constraints on only the final impulse conversion. This process also has the benefit of improving optimality for the entire sequence of burns, as by definition each FB was first optimized individually, not as a member of the larger set of conversions, which is the true problem at hand. By evolving the targeting constraints and growing the problem to encompass the entire set, regardless if any overlaps exist, the optimizer is afforded greater freedom when

minimizing the full multi-FB problem. This is because the burn-out states of each FB are no longer constrained to be on the inherently sub-optimal impulsive input trajectory.

The conversion method requires no prior knowledge of the solution, merely a multi-impulse or sub-optimal finite-burn initial guess, which aids automation. Numerical examples were shown for a variety of 3D orbit transfers, illustrating the adaptability of the method. These include a co-planar apogee raise maneuver, low-thrust non-coplanar Earth orbit transfers with and without FB arc overlaps, and a modified low-thrust interplanetary trajectory based upon the flown Deep Space 1 mission. Hybrid conversion results are shown with comparisons to standard sub-optimal control (SOC) solutions. The Hybrid OCT solutions offer faster convergence under certain circumstances, and in all cases provide moderately cheaper propellant cost when compared to SOC solutions. Simulations were run using common industry tools, demonstrating the potential new capability that can be added to conventional mission design practice by applying the presented method.

The presented work can be continued in a multitude of directions. The first step would be to apply the hybrid conversion algorithm to more complex transfer problems with a larger number of impulses than those presented. A perfect example of this is a multi-impulse representation of a low-thrust spiral with many revolutions. This represents a key transfer problem found in the realm of low-thrust and solar electric propulsion mission design, and one that would benefit substantially from the robust convergence properties of the hybrid conversion algorithm. As low-thrust spirals are often employed to place commercial satellites in their operational orbits, a savings in objective cost (transfer time and propellant) could represent a significant savings in financial cost as well. This would be thanks to the satellite's reduction in launch mass, and more importantly reaching its operational orbit sooner than if a sub-optimal spiral trajectory was employed.

Another way this research can be continued is to focus on automation and implementation. First, effort could be given to derive and compute the analytic gradients of the objective and constraints, as opposed to leaving them to the optimization routine to estimate. This can be done by applying linear perturbation theory, namely the computation of the State Transition Matrix. This has the potential to save considerable computer processing time [3]. Additional automation can be achieved by exploring the conversion of SOC-specific trajectories, with and without ΔV impulses.

Next, due to the inherently parallelizable nature of the first conversion sub-problem (individually optimizing a set of independent P-R problems), significant improvements to computational processing and convergence times may be achieved by leveraging subroutines such as the Modified-Picard Chebyshev Method (MPCM) [16] for propagation. Since the hybrid conversion algorithm computes trajectories comprised of discretized, sub-revolution sized impulsive and finite-burns, it is a perfect application for the MPCM, which performs best on partial orbital segments as opposed to many revolutions. Previous efforts demonstrated the ability to improve runtime performance of SNOPT by over a factor of three when using the MPCM over MATLAB's ODE113 for numerical propagation [17]. Such a subroutine combined with the cost savings and case-dependent convergence improvements afforded by the hybrid conversion algorithm will likely yield substantial overall benefits compared to using standard industry optimization tools for FB problems.

References

- [1] Williams, S.N., Coverstone-Carroll, V., "Benefits of Solar Electric Propulsion for the Next Generation of Planetary Exploration Missions," *The Journal of the Astronautical Sciences*, Vol. 45, No. 2, April-June 1997, pp. 143-159.
- [2] Fogel, J., Koblitz, D., Shah, N., "Survey of Low-Thrust Gravity Assist Trajectory Optimization Methods, with Comparisons to a Novel, Multi-Impulse Discretization Approach", International Astronautical Congress, Oct 2015.
- [3] Jesick, M., Ocampo, C., "Automated Design of Optimal Finite Thrust Orbit Insertion with Ballistic Flyby Constraints", *Journal of Spacecraft and Rockets*, Vol. 51, No. 6, 2014.
- [4] Longuski, M., Guzmán, J., Prussing, J., *Optimal Control with Aerospace Applications*, Space Technology Library, Microcosm Press and Springer Science, 2014.
- [5] Bryson, A., Ho, Y., *Applied Optimal Control*, Hemisphere Publishing 1975.
- [6] Hull, D., *Optimal Control Theory for Applications*, Springer Verlag, 2003.
- [7] Ocampo, C., Senent, J., Williams, J., "Theoretical Foundation of Copernicus: A Unified System for Trajectory Design and Optimization", *4th International Conference on Astrodynamics Tools and Techniques*, 3-6 May 2010, Madrid, Spain.
- [8] Shampine, L.F., Watts, H.A., "DEPAC - Design of a User Oriented Package of ODE Solvers", *Technical Report SAND-79-2374*, Sandia National Labs, September 1980.
- [9] Gill, P.E., Murray, W., Saunders, M.A., "User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming," June 16, 2008.

- [10] Gill, P.E., Murray, W., Saunders, M.A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," Society for Industrial and Applied Mathematics, Vol. 47, No. 1, 2005.
- [11] Englander, J.A., Englander, A.C., "Tuning Monotonic Basin Hopping: Improving the Efficiency of Stochastic Search as Applied to Low-Thrust Trajectory Optimization", *Technical Report GSF-E-DAA-TN14154*, NASA Technical Report Server, July 30, 2014.
- [12] "Deep Space 1 Launch Press Kit October 1998," National Aeronautics and Space Administration, October 1998.
- [13] Rayman, M., Chadbourne, P., Culwell, J., Williams, S., "Mission Design for Deep Space 1: A Low-Thrust Technology Validation Mission," *Acta Astronautica*, Vol. 45, Nos. 4-9, 1999.
- [14] Siddiqi, A., "Beyond Earth: A Chronicle of Deep Space Exploration, 1958-2016," United States, NASA History Division, NASA SP-2018-4041, Vol. 45, Nos. 4-9, 1999.
- [15] "Simulink User's Guide" MathWorks R2019b, The Math Works, Inc., 2019.
- [16] Koblick, D., Shankar, P., "Evaluation of the Modified Picard-Chebyshev Method for High-Precision Orbit Propagation," *Journal of Aerospace Engineering*, 2014.
- [17] Koblick, D., Xu, S., Fogel, J., Shankar, P., "Low Thrust Minimum Time Orbit Transfer Nonlinear Optimization Using Multi-Impulse Discretization via the Modified Picard-Chebyshev Method", International Conference on Computational & Experimental Engineering and Sciences, 2015.