INSTITUTE FOR DEFENSE ANALYSES

NO INITIAL GUESS REQUIRED: RAPIDLY COMPUTING THE FEASIBLE SET OF FUEL-OPTIMAL ELECTRIC PROPULSION TRAJECTORIES

Prashant R. Patel, Project Leader
Daniel J. Scheeres

December 2022

Approved for public release;
distribution is unlimited.

IDA Document NS D-33273

Log: H 22-000426

INSTITUTE FOR DEFENSE ANALYSES
730 East Glebe Road
Alexandria, Virginia 22305

About This Publication

This work was conducted by the IDA Systems and Analyses Center under the Central Research Program (C7265), “Space Conference Outreach.” The views, opinions, and findings should not be construed as representing the official position of either the Department of Defense or the sponsoring organization.

Acknowledgments

Thank you to Matthew J. Roberts and James D. Thorne of IDA for performing a technical review of this document.

For More Information

Prashant R. Patel, Project Leader
prashant.patel@ida.org, (703) 575-1439

David E. Hunter, Director, Cost Analysis and Research Division
dhunter@ida.org, (703) 575-4686

INSTITUTE FOR DEFENSE ANALYSES

IDA Document NS D-33273

NO INITIAL GUESS REQUIRED: RAPIDLY COMPUTING THE FEASIBLE SET OF FUEL-OPTIMAL ELECTRIC PROPULSION TRAJECTORIES

Prashant R. Patel
Daniel J. Scheeres

NO INITIAL GUESS REQUIRED: RAPIDLY COMPUTING THE FEASIBLE SET OF FUEL-OPTIMAL ELECTRIC PROPULSION TRAJECTORIES

Prashant R. Patel,[*] and Daniel J. Scheeres[†]

We discuss our automated algorithm (no initial guess) that can map out the fuel-optimal feasible set including the reachable/controllable set for electric propulsion missions. We provide a detailed discussion of why an indirect multi-stage formulation is well suited to solving this problem over direct and indirect continuous approaches and present an in depth discussion of the theoretical and mathematical underpinnings of our formulation. Our approach also generates the fuel-optimal control law to points in the feasible set. Reachable sets for electric propulsion spacecraft are important to many problems including: dynamic re-planning, robust mission design, space situational awareness, advanced concept development, and threat assessments. Feasible solutions are important for trajectory design and maneuver. Our automated and fast algorithm has applications to many different space maneuver problems.

INTRODUCTION

In this paper we develop an automated algorithm for estimating the feasible set, including the reachable (controllable) boundary, and the associated near-optimal fuel-minimum control laws for electric propulsion spacecraft. The algorithm is fast and requires only one first order integration to generate the control law for all feasible points, allowing us to map out the full maneuver envelop of the spacecraft without a user supplied initial guess. This feature makes our algorithm useful for many space domain awareness and path planning algorithms. This paper is focused on discussing the theoretical and formulation choices we make in developing our algorithm.

Electric propulsion is now widely used for civil, military, and commercial space missions. There are currently over 2000 spacecraft using Hall effect thrusters, a form of electric propulsion. Electric propulsion is or will soon be the dominant form of propulsion for spacecraft. Estimating the full maneuver envelope for electric propulsion spacecraft remains a difficult problem. Solving this problem can help with automated path planning/maneuver and space domain awareness applications, which will become increasingly important as space becomes contested and economically significant.

Our contribution is that we use a fuel-optimal formulation and extract reachable set through the switching structure, which allows us to recover the reachable set and fuel-optimal feasible trajectories. In addition, our approach is fully automated. We also employ an indirect multi-stage formulation, which is not widely used in astrodynamics. Finally, we derive various boundary conditions so

[*]Research Staff Member, Cost Analysis and Research Division, Institute for Defense Analyses, 730 Glebe Road, Alexandria VA. 22305

[†]A. Richard Seebass Chair Professor, Ann and H.J. Smead Aerospace Engineering Sciences, University of Colorado, 3775 Discovery Drive, Boulder CO 80303

the algorithm can be used with space domain awareness applications and for chemical and electric propulsion path planning problems.

We define the maneuver envelope to be the full range of orbits a spacecraft can achieve. The reachable set is the outer envelope of states that are achievable. The feasible set consists of the states in the maneuver envelope but not in the reachable set. In practice, the reachable set is a subset of the feasible set, and the feasible set should be equal to the maneuver envelope. We treat the feasible and reachable set as distinct to help with communication and clarity.

We will begin our paper by reviewing prior work and discussing our goals. Next, we will review and discuss the indirect multi-state formulation for solving optimal control problems. This review will include a comparison with indirect continuous formulations. We will then derive our formulation and provide a discussion about the specific equations we use and how alternative formulations complicate the development of a fast and efficient algorithm.

DESIGN OBJECTIVES AND LITERATURE REVIEW

Automation in space operations is becoming commonplace and will continue to be required in the future. The large number of current and future satellites using electric propulsion in orbit is driving the need for algorithmic speed and scalability. Accordingly, we set our goal to develop a fast, automated (no initial guess), and parallelizable algorithm that can estimate the fuel-optimal maneuver envelope of electric propulsion spacecraft.

Fundamentally, estimating the maneuver envelope of spacecraft is a challenge because location independent range curves for spacecraft do not exist, requiring us to rely on numerical methods to estimate the maneuver envelope of electric propulsion spacecraft. The most common ways to estimate the reachable set of electric propulsion spacecraft is to write the problem as an optimal control problem. There are many ways to write and solve optimal control problems[1, 2] with direct and continuous indirect methods being the most common. Both these methods are widely used throughout the astrodynamics community to solve a variety of problems.

An indirect continuous approach uses calculus of variations[1], which results in the control being eliminated and parameterized in terms of the co-states and the states. With the control eliminated the problem is solved when the co-states are found such that the various boundary conditions are satisfied. This situation results in a low dimensional problem, but one that is highly sensitive. A variety of numerical techniques attempt to solve these problems[2]. Clever formulation or insights can also be used to solve these problems. Thorne[3] leveraged insight to create accurate estimates of the initial co-states, thus enabling a good initial guess.

Indirect continuous approaches are used to solve reachable problems. They have been widely applied to space domain awareness (SDA) applications[4, 5]. Holzinger[4] derived the continuous indirect equations for the electric propulsion reachability problem with ellipsoid uncertainty and bounded acceleration. These methods were then shown to work on SDA applications[5]. This results in a two point boundary value problem (TPBVP) that requires an initial guess. They also use a distance based cost function which limits the ability to apply these methods to feasible trajectories.

Generating functions[6] can also be used to map out the reachable set. Bando[7] used a power series expansion of the generating function to identify low thrust transfers in the Hill frame. Generating functions provide the solution flow; however, current techniques work only on dynamical systems with polynomial forms, such as the Hill problem[6].

Direct formulations are a common alternative to indirect continuous formulations. Direct formulations parameterize the control then attempt to directly minimize the cost function, eliminating the need for guessing the Lagrange multipliers and reducing the sensitivity of the problem. Direct methods have been combined with stochastic solvers [8, 9] to generate optimal trajectories. These approaches use the reachability problem as an input into their trajectory optimization solver, but they require the switching structure has to be specified.

One gap in the current literature is connecting fuel-optimal trajectory design and reachability analysis. Prior methods[4, 5, 8, 9] use a distance measure as their objective function. Reachability and fuel-optimal problems are closely related; when the switching structure is fully activated for fuel-optimal trajectories, they are also on the reachable set. Our approach will close this gap and show how we can use the switching structure to estimate the reachable and feasible trajectories.

Both indirect continuous and direct formulations are widely used to solve fuel-optimal trajectories. Indirect continuous methods are used in reachability problems. However, for our goal of developing automated approaches, both methods have significant downsides, which led us to use an indirect multi-stage formulation.

We will show that setting up the reachable problem as a fuel-minimum problem allows us to rapidly solve for the reachable set and the feasible fuel-optimal set. We will exploit an indirect multi-stage formulation to generate an automated algorithm. This method allows us to estimate both the reachable set and fuel-optimal trajectories using a single, fast, automated algorithm.

INDIRECT MULTI-STAGE FORMULATION

To create our automated, fast, and scalable algorithm we begin by formulating the trajectory optimization problem as an indirect multi-stage problem[1]. An indirect multi-stage formulation breaks the trajectory into a sequence of segments or stages. The inputs into a stage are the states and the control. The output is the updated states (see Figure 1 for a graphical example). In the multi-

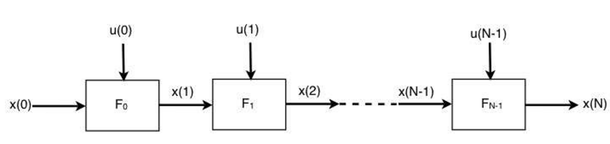

Figure 1. The multi-stage formulation.

stage problem, each stage can be considered a black box where the dynamics inside the box can change so long as certain conditions are met. We consider a controlled dynamical system defined by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t, \boldsymbol{u}, \boldsymbol{p}) \tag{1}$$

Where a ***bold*** indicates a vector or matrix while a $normal$ font indicates a scalar quantity. $\boldsymbol{x}$ represents the state vector; $\boldsymbol{u}$ is the control vector, which is in general a function of time; t is time, and $\boldsymbol{p}$ are a set of parameters that are constant in the system.

This continuous dynamical system can be transformed into a set of indirect multi-stage formulation equations [1]. A trajectory segment 'i' is governed by

$$\boldsymbol{x}^{i+1} = \boldsymbol{F}^i(\boldsymbol{x}^i, t^i, \boldsymbol{u}^i, \boldsymbol{p}) \tag{2}$$

where

$$\boldsymbol{F}^i = \boldsymbol{x}^i + \int_{t_i}^{t_{i+1}} \dot{\boldsymbol{x}} dt = \boldsymbol{x}^i + \int_{t_i}^{t_{i+1}} \boldsymbol{f}(\boldsymbol{x}, t, \boldsymbol{u}, \boldsymbol{p}) dt \tag{3}$$

A trajectory is then composed of a series of N trajectory segments, governed by Equation 2 and Equation 3, where $N \geq 1$. For a multi-segment trajectory, we can integrate Equation 2 from i to $i+1$, then use $i+1$ to obtain $i+2$, and so on. The dynamics here represent a finite time interval. If N is very large, then in the limit we can think of a stage as a single integration step. For the indirect continuous formulation, the dynamics are the terms under the integral.

In the indirect multi-stage formulation, the control parameters $\boldsymbol{u}^i$ are fixed on each segment, which is also how direct formulations are generally written. The fixed nature of $\boldsymbol{u}^i$ is not a significant limitation as time varying inputs can be modeled by having $\boldsymbol{u}^i$ be the coefficients to a time varying function. The indirect multi-stage formulation's dynamical and control structure are similar to the direct formulation as both use fixed controls over each stage and parameterize the trajectory in stages.

The cost function for a multi-stage formulation is given by

$$J = \phi(\boldsymbol{x}^N, \boldsymbol{x}^0, N) + \sum L(i, \boldsymbol{x}^i, \boldsymbol{u}^i) \tag{4}$$

where ϕ is the initial and terminal cost function and $L(i, \boldsymbol{x}, \boldsymbol{u}) = L^i$ is the integral cost function evaluated at the ith stage. Once again, the cost function is analogous to the indirect continuous and direct formulations. If the summation of the Lagrangian is replaced by an integral then it would reflect the indirect continuous formulation. As written the cost function can also be used in direct formulations.

The Hamiltonian is then

$$H = \sum H^i \tag{5}$$

$$H^i = \boldsymbol{\lambda}^{i+1} \cdot \boldsymbol{F}^i + L^i + \sum_k \nu^k C^k \tag{6}$$

For notational convenience later, we specify the co-states, $\boldsymbol{\lambda}$, with an index $i+1$, with C representing the per stage constraints. The Hamiltonian structure is analogous to the indirect continuous formulation.

Taking the partial with respect to $\boldsymbol{x}$ yields the dynamics equations for the co-states.

$$H^i_{\boldsymbol{x}} = \boldsymbol{\lambda}^i = \boldsymbol{F}^{iT}_{\boldsymbol{x}} \boldsymbol{\lambda}^{i+1} + L^i_{\boldsymbol{x}} + \sum_k \nu^k C^k_{\boldsymbol{x}} \tag{7}$$

The control law is then given by Pontryagin's minimum principle. Taking the partial of the Hamiltonian with respect to the control and solving for when Equation 8 is zero (and hence an extremal) yields

$$H^i_{\boldsymbol{u}} = \boldsymbol{0} = \boldsymbol{F}^{iT}_{\boldsymbol{u}} \boldsymbol{\lambda}^{i+1} + L^i_{\boldsymbol{u}} + \sum_k \nu^k C^k_{\boldsymbol{u}} \tag{8}$$

We notice several interesting features of Equation 7 and Equation 8. One interesting observation is that Equation 7 is set up to be computed backwards in the stages, where $i+1$ is on the right hand side and the prior co-state multipliers i are on the left hand side. In general, we have no guarantee that $\boldsymbol{u}$ is analytically separable in Equation 2, Equation 7, or Equation 8. Thus, we have to assume that $F_{\boldsymbol{x}}$ and $F_{\boldsymbol{u}}$ are functions of $\boldsymbol{u}$.

The indirect multi-stage initial and terminal conditions are given as

$$\boldsymbol{C}^0(\boldsymbol{x}) = 0 \quad \boldsymbol{C}^N(\boldsymbol{x}) = 0 \tag{9}$$

which result in the initial and terminal co-states being

$$\boldsymbol{\lambda}^0 = \phi_{\boldsymbol{x}^0} + \nu^0 \cdot \boldsymbol{C}^0_{\boldsymbol{x}} \quad \boldsymbol{\lambda}^N = \phi_{\boldsymbol{x}^N} + \nu^N \cdot \boldsymbol{C}^N_{\boldsymbol{x}} \tag{10}$$

These boundary conditions mirror the indirect continuous case*. These equations share a similar structure to the indirect continuous approach; taken together, they represent the general equations for the multi-stage optimization problem. We will now explore the differences between the two indirect formulations.

DIFFERENCES BETWEEN CONTINUOUS AND MULTI-STAGE FORMULATION

We will now explore the differences between the indirect multi-stage and the indirect continuous formulation, by deriving the multi-stage formulation from first principles. The results will mirror those in Bryson and Ho[1], although the derivation is not found in Bryson and Ho.

Indirect Continuous Co-state Derivation

In order to highlight why the indirect multi-stage method is different. Much of the formulation will follow from Bryson and Ho[1]. We will begin by defining the augmented cost function and then defining the terms we use. Next, we will take the partials and solve for the conditions. Finally, we will make some observations.

We begin by defining the augmented cost function as

$$\bar{J} = \boldsymbol{\phi}^f + \boldsymbol{\psi}^{f,T}\boldsymbol{\nu}^f + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \int_{t^0}^{t^f} \left(L(t) + (\boldsymbol{f}(t) - \dot{\boldsymbol{x}}(t))^T \boldsymbol{\lambda}(t)\right) dt \tag{11}$$

The costs are given as $J = \phi^f + \phi^0 + \int_{t^0}^{t^f} L dt$, where ϕ^f is the terminal cost, ϕ^0 is the initial cost, and L represents the change in costs over the trajectory. For brevity we simplify notation and omit the dependencies on states and controls. ψ is the initial and terminal constraints. f represents they dynamics that returns the change in the states, given by $\dot{x}$. The states are given by x. The dynamic co-states are given by λ, and ν represents the Lagrange multipliers at initial and terminal time. All terms under the integral are computed at the same point in time.

The Hamiltonian is defined as

$$H(t) = L(t) + \boldsymbol{f}^T(t)\boldsymbol{\lambda}(t) \tag{12}$$

Rewriting and expanding the augmented cost function we get

$$\bar{J} = \phi^f + \boldsymbol{\psi}^{f,T}\boldsymbol{\nu}^f + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \int_{t^0}^{t^f} \left(H(t) - \dot{\boldsymbol{x}}^T(t)\boldsymbol{\lambda}(t)\right) dt \tag{13}$$

$$\bar{J} = \phi^f + \boldsymbol{\psi}^{f,T}\boldsymbol{\nu}^f + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \int_{t^0}^{t^f} H(t)dt - \int_{t^0}^{t^f} \dot{\boldsymbol{x}}^T(t)\boldsymbol{\lambda}(t)dt \tag{14}$$

*The terminal conditions can be found in Bryson and Ho, but the initial conditions are not

Next, we apply integration by parts to expand the term under the second integral and then collect similar terms.

$$\bar{J} = \phi^f + \boldsymbol{\psi}^{f,T}\boldsymbol{\nu}^f + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \int_{t^0}^{t^f} H(t)dt - \boldsymbol{x}^{f,T}\boldsymbol{\lambda}^f + \boldsymbol{x}^{0,T}\boldsymbol{\lambda}^0 + \int_{t^0}^{t^f} \boldsymbol{x}^T(t)\dot{\boldsymbol{\lambda}}(t)dt \quad (15)$$

$$\bar{J} = (\phi^f + \boldsymbol{\psi}^{f,T}\boldsymbol{\nu}^f - \boldsymbol{x}^{f,T}\boldsymbol{\lambda}^f) + (\phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{x}^{0,T}\boldsymbol{\lambda}^0) + \int_{t^0}^{t^f} H(t) + \boldsymbol{x}^T(t)\dot{\boldsymbol{\lambda}}(t)dt \quad (16)$$

We now consider the variation in $\bar{J}$ due to a variation in the control vector, u for fixed time, giving

$$\delta\bar{J} = (\phi_x^f + \boldsymbol{\psi_x}^{f,T}\boldsymbol{\nu}^f - \boldsymbol{\lambda}^f)\boldsymbol{\delta x}_{t=t^f} + (\phi_x^0 + \boldsymbol{\psi_x}^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{\lambda}^0)\boldsymbol{\delta x}_{t=t^0} + \int_{t^0}^{t^f} \left(\left(H_x + \dot{\boldsymbol{\lambda}}\right)\boldsymbol{\delta x} + H_u\boldsymbol{\delta u}\right)dt \quad (17)$$

We then choose the co-state values and dynamics such that the variations in δx vanish, which removes the need to create a dependency between δx and δu. Doing so, we obtain the classical co-states values as

$$\boldsymbol{\lambda}^f = \phi_x^f + \boldsymbol{\psi}_x^{f,T}\boldsymbol{\nu}^f \quad (18)$$

$$\boldsymbol{\lambda}^0 = -\phi_x^0 + \boldsymbol{\psi}_x^{0,T}\boldsymbol{\nu}^0 \quad (19)$$

$$\dot{\boldsymbol{\lambda}} = -H_x = -L_x(t) - \boldsymbol{f}_x^T(t)\boldsymbol{\lambda}(t) \quad (20)$$

The optimal control is then given when $H_u = 0$ which is

$$0 = H_u = L_u(t) + \boldsymbol{f}_u^T(t)\boldsymbol{\lambda}(t) \quad (21)$$

Interestingly, the optimal control and the co-state dynamics, in these equations, depend only on the current state, dynamics, and their partials. Stated another way, for the continuous indirect case no future or past information is needed!

Indirect Multi-Stage Co-State Derivation

In this subsection we will derive the indirect multi-stage case. This formulation is less widely used in the astrodynamics community. This derivation will deviate from Bryson and Ho[1], as we use finite differences for our derivation and include initial costs and constraints, which are not in Bryson and Ho.

We begin our derivation with finite differences and include initial costs and constraints in our derivation. In our formulation, $\boldsymbol{\Delta x}^{i+1} = \boldsymbol{f}^i = \boldsymbol{x}^{i+1} - \boldsymbol{x}^i$. We will mirror the steps for the indirect continuous case to allow for comparisons later. The purpose of using the finite difference approach is to gain insights.

We begin by defining the augmented cost function as

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1}\left(L^i + \left(\boldsymbol{f}^i - \boldsymbol{\Delta x}^{i+1}\right)^T\boldsymbol{\lambda}^{i+1}\right) \quad (22)$$

The costs are given as $J = \phi^N + \phi^0 + \sum_{i=0}^{N-1} L^i$, where ϕ^N is the terminal cost, ϕ^0 is the initial cost, and L^i represents the change in cost when moving from state i to $i+1$. Once again, for brevity we simplify notation and omit the dependencies on states and controls. ψ is the initial and terminal constraints. f represents the dynamics that return the finite difference in the states, given by Δx. The states are given by x. The co-states are given by λ, and ν represents the Lagrange multipliers at initial and terminal stage. The $i+1$ associated with the co-states under the summation is to follow the convention used by Bryson and Ho.

We begin by defining the Hamiltonian as

$$H^i = L^i + \boldsymbol{f}^{i,T}\boldsymbol{\lambda}^{i+1} \tag{23}$$

Rewriting and expanding the augmented cost function, we get

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1}\left(L^i + \left(\boldsymbol{f}^i - \boldsymbol{\Delta x}^{i+1}\right)^T\boldsymbol{\lambda}^{i+1}\right) \tag{24}$$

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1} H^i - \boldsymbol{\Delta x}^{i+1,T}\boldsymbol{\lambda}^{i+1} \tag{25}$$

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1} H^i - \sum_{i=0}^{N-1}\boldsymbol{\Delta x}^{i+1,T}\boldsymbol{\lambda}^{i+1} \tag{26}$$

Reordering the indices on the second summation, we get

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1} H^i - \sum_{i=1}^{N}\boldsymbol{\Delta x}^{i,T}\boldsymbol{\lambda}^{i} \tag{27}$$

Next, we apply summation by parts to expand the term under the second summation and then collect similar terms. The summation by parts rule is

$$\sum_{i=j+1}^{n} a^i\Delta b^i = a^n b^n - a^j b^j - \sum_{i=j+1}^{n} b^{i-1}\Delta a^i \tag{28}$$

where $\Delta a^i = a^i - a^{i-1}$. Applying summation by parts to the augmented cost function, we obtain

$$\bar{J} = \phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N + \phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \sum_{i=0}^{N-1} H^i - \boldsymbol{\lambda}^N\boldsymbol{x}^N + \boldsymbol{\lambda}^0\boldsymbol{x}^0 + \sum_{i=1}^{N}\boldsymbol{\Delta\lambda}^i\boldsymbol{x}^{i-1} \tag{29}$$

Reordering the indices and collecting terms, we obtain

$$\bar{J} = (\phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N - \boldsymbol{\lambda}^N\boldsymbol{x}^N) + (\phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{\lambda}^0\boldsymbol{x}^0) + \sum_{i=0}^{N-1} H^i + \sum_{i=1}^{N}\boldsymbol{\Delta\lambda}^i\boldsymbol{x}^{i-1} \tag{30}$$

$$\bar{J} = (\phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N - \boldsymbol{\lambda}^N\boldsymbol{x}^N) + (\phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{\lambda}^0\boldsymbol{x}^0) + \sum_{i=0}^{N-1} H^i + \sum_{i=0}^{N-1}\boldsymbol{\Delta\lambda}^{i+1}\boldsymbol{x}^{i} \tag{31}$$

$$\bar{J} = (\phi^N + \boldsymbol{\psi}^{N,T}\boldsymbol{\nu}^N - \boldsymbol{\lambda}^N\boldsymbol{x}^N) + (\phi^0 + \boldsymbol{\psi}^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{\lambda}^0\boldsymbol{x}^0) + \sum_{i=0}^{N-1} (H^i + \boldsymbol{\Delta\lambda}^{i+1}\boldsymbol{x}^{i}) \tag{32}$$

We now consider the variation in $\bar{J}$ due to a variation in the control vector u, for fixed stage, which gives us

$$\delta\bar{J} = \left(\phi_x^N + \boldsymbol{\psi_x}^{N,T}\boldsymbol{\nu}^N - \boldsymbol{\lambda}^N\right)\boldsymbol{\delta x}^N + \left(\phi_x^0 + \boldsymbol{\psi}_x^{0,T}\boldsymbol{\nu}^0 + \boldsymbol{\lambda}^0\right)\boldsymbol{\delta x}^0 + \sum_{i=0}^{N-1}\left((H_x^i + \boldsymbol{\Delta\lambda}^{i+1})\boldsymbol{\delta x}^i + H_u^i\boldsymbol{\delta u}^i\right) \quad (33)$$

We then choose the co-state values and dynamics such that the variations in δx vanish, removing the need to create a dependency between δx and δu. We now solve for the boundary conditions and λ. The boundary conditions are

$$\boldsymbol{\lambda}^N = \phi_x^N + \boldsymbol{\psi_x}^{N,T}\boldsymbol{\nu}^N \quad (34)$$

$$\boldsymbol{\lambda}^0 = -\phi_x^0 - \boldsymbol{\psi_x}^{0,T}\boldsymbol{\nu}^0 \quad (35)$$

As we see, the equations mimic the continuous case. We now solve for the evolution of λ. We begin by writing out the finite difference equation

$$\boldsymbol{\Delta\lambda}^{i+1} = -H_x^i \quad (36)$$

This equation reduces the augmented variation cost function to

$$\delta\bar{J} = \sum_{i=0}^{N-1} H_u^i \delta u^i \quad (37)$$

which gives us the optimal control law as

$$0 = H_u^i = L_{u^i}^i + \boldsymbol{f}_{u^i}^{i,T}\boldsymbol{\lambda}^{i+1} \quad (38)$$

$$0 = L_{u^i}^i + (\boldsymbol{F}_{u^i}^{i,T} - \boldsymbol{0})\boldsymbol{\lambda}^{i+1} \quad (39)$$

$$0 = L_{u^i}^i + \boldsymbol{F}_{u^i}^{i,T}\boldsymbol{\lambda}^{i+1} \quad (40)$$

The equations are similar to to the continuous condition. However, as we will see there are some fundamentally differences between the continuous and multi-stage formulation. To see the differences, we need to expand the finite difference equation for $\boldsymbol{\Delta\lambda}^{i+1}$.

$$\boldsymbol{\Delta\lambda}^{i+1} = -H_x^i \quad (41)$$

$$-\boldsymbol{\Delta\lambda}^{i+1} = H_x^i \quad (42)$$

$$-\boldsymbol{\lambda}^{i+1} + \boldsymbol{\lambda}^i = L_x^i + \boldsymbol{f}_x^{i,T}\boldsymbol{\lambda}^{i+1} \quad (43)$$

$$-\boldsymbol{\lambda}^{i+1} + \boldsymbol{\lambda}^i = L_x^i + (\boldsymbol{F}_x^{i,T} - \boldsymbol{I}^T)\boldsymbol{\lambda}^{i+1} \quad (44)$$

$$\boldsymbol{\lambda}^i = L_x^i + (\boldsymbol{F}_x^{i,T} - \boldsymbol{I}^T)\boldsymbol{\lambda}^{i+1} + \boldsymbol{\lambda}^{i+1} \quad (45)$$

$$\boldsymbol{\lambda}^i = L_x^i + \boldsymbol{F}_x^{i,T}\boldsymbol{\lambda}^{i+1} - \boldsymbol{I}^T\boldsymbol{\lambda}^{i+1} + \boldsymbol{\lambda}^{i+1} \quad (46)$$

$$\boldsymbol{\lambda}^i = L_x^i + \boldsymbol{F}_x^{i,T}\boldsymbol{\lambda}^{i+1} \quad (47)$$

Our indirect multi-stage derivation recovers and extends the result from Bryson and Ho. As we have seen, taking a finite difference approach mirrors the continuous formulation and recovers the classical results. We now explore some insights gained from this derivation.

Indirect Multi-Stage Formulation Observations

One major value of the indirect continuous formulation is that the control law and dynamics depend on information only at the current time. This feature (normally) allows us to eliminate the controls and write the problem as an initial value problem that is a function of the states and co-states. We can then use various methods to solve the resulting TPBVP[2].

In this section we will explore to see if there is an analogous approach for the multi-stage case. Specifically, we are exploring to see if we can compute the states and co-states in the same direction and if this offers any advantages. To do this we attempt to solve H_u as a function of λ^i. We begin by solving for λ^{i+1}

$$\lambda^i = L_x^i + F_x^{i,T}\lambda^{i+1} \tag{48}$$
$$\lambda^i - L_x^i = F_x^{i,T}\lambda^{i+1} \tag{49}$$
$$F_x^{i,-T}(\lambda^i - L_x^i) = \lambda^{i+1} \tag{50}$$
$$\lambda^{i+1} = F_x^{i,-T}\lambda^i - F_x^{i,-T}L_x^i \tag{51}$$

Subsisting this result into the condition for the optimal control law, we obtain

$$0 = L_{u^i}^i + F_{u^i}^{i,T}\lambda^{i+1} \tag{52}$$
$$0 = L_{u^i}^i + F_{u^i}^{i,T}(F_x^{i,-T}\lambda^i - F_x^{i,-T}L_x^i) \tag{53}$$
$$0 = (L_{u^i}^i - F_{u^i}^{i,T}F_x^{i,-T}L_x^i) + F_{u^i}^{i,T}F_x^{i,-T}\lambda^i \tag{54}$$
$$0 = (L_{u^i}^i - (F_x^{i,-1}F_{u^i}^i)^T L_x^i) + (F_x^{i,-1}F_{u^i}^i)^T\lambda^i \tag{55}$$

So, for computational purposes we can write the co-state equations similar to an initial value. In general, we cannot isolate and eliminate the control using this form. Therefore, using this form requires having the controls, initial states, and co-states, and computing a **matrix inverse**. In the prior form, where the co-states are computed backwards, we require the controls, initial states, and co-states. Thus both forms carry the same information, but it is computationally more efficient to compute the co-states backwards in stages.

This difference between the continuous and multi-stage co-state equations make logical sense if we interpret the co-states as the change in the cost to go with respect to the states. From dynamic programming, we recognize that the cost-to-go is readily computed working from the end to the beginning, and we can show this relationship by starting with the terminal boundary conditions for the co-states (assuming no terminal state constraints). We have

$$\lambda^N = \frac{\partial \phi}{\partial x^N} = \frac{\partial J}{\partial x^N} = J_x^N \tag{56}$$
$$\lambda^i = L_x^{N-1} + F_x^{N-1,T}\lambda^{i+1} \tag{57}$$
$$\lambda^{N-1} = J_x^{N-1} = L_x^i + F_x^{i,T}\frac{\partial J}{\partial x^N} \tag{58}$$

With this formulation we can more clearly see the relationship to the cost-to-go and why the co-states are computed backwards in stages. With a deeper understanding of the indirect multi-stage formulation, we now derive the specific equations we will use and develop an algorithm for estimating the reachable and feasible set.

FUEL MINIMUM EQUATIONS

In this section we begin by deriving the fuel-minimum optimal control problem using a multi-stage formulation. The objective of this section is to derive the specific equations we use, discuss how they will be used, and highlight interesting features and comparisons with other possible approaches.

The equations of motion we use in this paper are stated in an inertial frame with an arbitrary origin

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{v}} \\ \dot{n} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ \sum_j -\frac{\mu^j}{|\boldsymbol{r}-\boldsymbol{r}^j|^3}(\boldsymbol{r}-\boldsymbol{r}^j) + n\boldsymbol{T} \\ n^2 \frac{T_{mag}}{c} \end{bmatrix} \tag{59}$$

where μ^j is the gravitational parameter of the attracting body located at $\boldsymbol{r}^j$, $\boldsymbol{T}$ is the thrust vector (magnitude and direction), T_{mag} is the magnitude of thrust, $\boldsymbol{r}$ is the position vector, $\boldsymbol{v}$ is the velocity, n is the inverse mass, and c is one Earth gravity times the specific impulse. In Equation 59 we use the inverse of mass in order to make the equations more linear, which eliminates negative powers on the mass and makes the derivation and computation of $F_{\boldsymbol{x}}$ and $F_{\boldsymbol{u}}$ easier. The control vector is

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{T} \\ T_{mag} \end{bmatrix} \tag{60}$$

We also separate out the control, $\boldsymbol{T}$, associated with the position and velocity dynamics from the control, T_{mag} used to track the inverse mass of the spacecraft. There are many possible formulation we can use here. One formulation commonly used is, simply to have the controls be $\boldsymbol{T}$, (no T_{mag}) and the dynamics associated with mass be $\dot{m} = -\frac{|\boldsymbol{T}|}{c}$. This formulation is non-differentiable when $\boldsymbol{T} = \boldsymbol{0}$ which is a well known issue[6] and has several mitigations. One approach is to write the mass equation as

$$\dot{m} = -\frac{|\boldsymbol{T} + \epsilon|}{c} \tag{61}$$

where ϵ is a small (but not too small) positive number, thereby reducing accuracy but gaining numerical efficiency. Another approach is to separate the magnitude and direction of the thrust vector, which results in the control being written as $\begin{bmatrix} \hat{\boldsymbol{T}}^T & T_{mag} \end{bmatrix}$, where $\hat{\boldsymbol{T}}$ is the unit vector of thrust. Substituting the above into the dynamics we get $\dot{\boldsymbol{v}} = \sum_j -\frac{\mu^j}{|\boldsymbol{r}-\boldsymbol{r}^j|^3}(\boldsymbol{r}-\boldsymbol{r}^j) + n\hat{\boldsymbol{T}}T_{mag}$, resulting in a cubic term that is counter to our goal. Because we are trying to get Equation 59 to be as linear as possible, we choose Equation 60[6]. We will then apply a constraint to ensure the magnitude of the thrust vector, $|\boldsymbol{T}|$, is equal to T_{mag}. Although we use these dynamics in our examples, we do not exploit any special structure associated with them. Therefore, any dynamical coordinate system or frame should work, using our approach.

Optimal control problems also require a cost function. One of our goals is to compute fuel minimum trajectories. There are several ways to write a cost function. All of the following forms are valid, and theoretically equivalent:

$$J = n \tag{62}$$

$$J = \sum_i \int n^2 \frac{T_{mag}}{c} \tag{63}$$

$$J = \sum_i \frac{T_{mag}}{c} \Delta t \tag{64}$$

The cost function we will use is

$$J = \sum_i \frac{T^i_{mag}}{c^i} \Delta t \tag{65}$$

We will discuss why we chose this particular form in the algorithm section of this paper. The constraints we apply are $|\boldsymbol{T}| = T_{mag}$ and $0 \leq |\boldsymbol{T}| \leq T_{max}$. These conditions require the thrust must be less than the maximum allowable thrust, T_{max}, and that the total thrust and T_{mag} be equal. The latter requirement simply enforces the constraint that the thrust used to generate acceleration be equal to the thrust supplied by the engine.

The Hamiltonian is then

$$H^i = \boldsymbol{F}^{iT} \boldsymbol{\lambda}^{i+1} + \frac{T^i_{mag} \Delta t}{c^i} + \nu_1 (|\boldsymbol{T}| - T_{mag}) + \nu_2 (\frac{1}{2} T^2_{mag} - \frac{1}{2} T^2_{max}) \tag{66}$$

The co-state equation is

$$H^i_{\boldsymbol{x}} = \boldsymbol{\lambda}^i = \boldsymbol{F_x}^{iT} \boldsymbol{\lambda}^{i+1} \tag{67}$$

and the control law is defined by

$$H_{\boldsymbol{u}} = 0 = \boldsymbol{F_u}^T \boldsymbol{\lambda}^{i+1} + \nu_1 \begin{bmatrix} \frac{\boldsymbol{T}}{|\boldsymbol{T}|} \\ -1 \end{bmatrix} + \nu_2 \begin{bmatrix} 0 \\ T_{mag} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{\Delta t}{c} \end{bmatrix} \tag{68}$$

The term $\boldsymbol{F^i_x}$ is the state transition matrix at the end of the segment and $\boldsymbol{F^i_u}$ is similar to the impulse response matrix for linear systems. Defining $\Phi = F_{\boldsymbol{x}}$ and $\Omega = F_{\boldsymbol{u}}$ we can solve for them by recognizing that $\dot{\boldsymbol{u}} = \boldsymbol{0}$ and treating $\boldsymbol{u}$ like a state. The equations for $\dot{\Phi}$ and $\dot{\Omega}$ are

$$\begin{bmatrix} \dot{\boldsymbol{\Phi}} & \dot{\boldsymbol{\Omega}} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{x}} & \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{T}} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi} & \boldsymbol{\Omega} \\ 0 & \boldsymbol{I} \end{bmatrix} \tag{69}$$

The identity matrix in Equation 69 comes from the fact that $\frac{\partial u}{\partial u}$ is constant ($\dot{\boldsymbol{u}} = \boldsymbol{0}$) and equal to the identify matrix, which also results in the zeros on the left hand side. The specific forms of $\boldsymbol{\Phi}$ and $\boldsymbol{\Omega}$ are

$$\boldsymbol{F^i_x} = \frac{\partial \boldsymbol{x}^{i+1}}{\partial \boldsymbol{x}^i} = \Phi(t^{i+1}, t^i) = \int_{t^i}^{t^{i+1}} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \frac{\partial \dot{\boldsymbol{v}}}{\partial \boldsymbol{r}} & \boldsymbol{0} & \boldsymbol{T} \\ 0 & 0 & 2n \frac{T_{mag}}{c} \end{bmatrix} \Phi(t, t^i) dt \tag{70}$$

and

$$\boldsymbol{F^i_u} = \frac{\partial \boldsymbol{x}^{i+1}}{\partial \boldsymbol{u}^i} = \Omega(t^{i+1}, t^i) = \int_{t^i}^{t^{i+1}} \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \frac{\partial \dot{\boldsymbol{v}}}{\partial \boldsymbol{r}} & \boldsymbol{0} & \boldsymbol{T} \\ 0 & 0 & 2n \frac{T_{mag}}{c} \end{bmatrix} \Omega(t, t^i) + \begin{bmatrix} \boldsymbol{0} & 0 & 0 & \boldsymbol{0} \\ - & diag(\boldsymbol{n}) & - & \boldsymbol{0} \\ 0 & 0 & 0 & \frac{n^2}{c} \end{bmatrix} dt \tag{71}$$

The initial conditions are $\boldsymbol{\Phi}(i,i) = \boldsymbol{I}$ and $\boldsymbol{\Omega}(i,i) = \boldsymbol{0}$.

Returning to our derivation and solving for the control law, we first solve Equation 68 for $\frac{\boldsymbol{T}}{|\boldsymbol{T}|} = \hat{\boldsymbol{T}}$, the thrust unit vector. This part of Equation 68 is

$$0 = \boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1} + \nu_1 \frac{\boldsymbol{T}}{|\boldsymbol{T}|} \tag{72}$$

We rearrange and isolate $\hat{\boldsymbol{T}}$.

$$-\boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1} = \nu_1 \frac{\boldsymbol{T}}{|\boldsymbol{T}|} \tag{73}$$

$$-\frac{\boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1}}{\nu_1} = \frac{\boldsymbol{T}}{|\boldsymbol{T}|} = \hat{\boldsymbol{T}} \tag{74}$$

By inspection we can see that ν_1 must equal $|\boldsymbol{F_T}^{i,T} \boldsymbol{\lambda}^{i+1}|$ in order for the right hand side to be a unit vector, which gives us

$$\hat{\boldsymbol{T}} = -\frac{\boldsymbol{F_T}^{i,T} \boldsymbol{\lambda}^{i+1}}{|\boldsymbol{F_T}^{i,T} \boldsymbol{\lambda}^{i+1}|} \tag{75}$$

and

$$\nu_1 = \left|\boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1}\right| \tag{76}$$

$\boldsymbol{F_T}$ represents the columns of $\boldsymbol{F_u}$ that are associated with the $\boldsymbol{T}$. We now need to solve for T_{mag}. The equation for this is

$$\boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} + \frac{\Delta t}{c} - \nu_1 + \nu_2 T_{mag} \tag{77}$$

We have already solved for ν_1 and still have two unknowns, ν_2 and T_{mag} but only one equality equation. The second condition is the inequality constraint, $\frac{1}{2}T_{mag}^2 - \frac{1}{2}T_{max}^2 \leq 0$.

We begin by assuming that the inequality constraint is binding ($T_{mag} = T_{max}$) and solving for ν_2. We will then check if ν_2 has a valid value ($\nu_2 \geq 0$). If $\nu_2 < 0$ then the inequality constraint is not binding and we set $T_{mag} = 0$ since the solution structure is bang-bang. We begin by setting $T_{mag} = T_{max}$, which gives us

$$\boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} + \frac{\Delta t}{c} - \nu_1 + \nu_2 T_{max} \tag{78}$$

Rearranging and solving for ν_2 we obtain

$$\nu_2 = \frac{\nu_1 - \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c}}{T_{max}} \geq 0 \tag{79}$$

Because the cost function is linear in T_{mag} it is bang-bang making the equation for ν_2 our switching function. For the switching function, we care about preserving its sign and where it crosses 0. Thus we are free to scale it arbitrarily, as the aforementioned conditions are met. We now can simplify by removing T_{max} since being a positive scalar factor does not change the sign or zero crossing. We now have

$$\nu_1 - \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c} > 0 \tag{80}$$

Substituting in our solution for ν_1 we obtain

$$|\boldsymbol{F_T} \lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c} > 0 \tag{81}$$

We now define SF, the value of the switching function, as

$$SF^{i,\boldsymbol{\lambda}^N} = |\boldsymbol{F_T}\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c} \tag{82}$$

The control law, when $SF^{i,\boldsymbol{\lambda}^N} > 0$, is now given as:

$$\begin{aligned} \boldsymbol{T} &= -T_{max}\frac{\boldsymbol{F_T}^{i,T}\boldsymbol{\lambda}^{i+1}}{|\boldsymbol{F_T}^{i,T}\boldsymbol{\lambda}^{i+1}|} \\ T_{mag} &= T_{max} \end{aligned} \tag{83}$$

In the case when $SF^{i,\boldsymbol{\lambda}^N} < 0$ the constraint is not binding and we set $\nu_2 = 0$, resulting in $T_{mag} = 0$. We have now solved for the control law and the co-state equations and will use them to develop an automated algorithm that can generate a feasible optimal trajectory to any reachable point.

REACHABLE, CONTROLLABLE, AND FEASIBLE TRAJECTORIES

In this section we derive an algorithm that can automatically generate near-optimal control laws to the reachable (controllable) set and interior feasible points. In the previous section we derived the equations for a fuel minimum optimal control using an indirect multi-stage formulation. We can leverage these equations recognizing that a fuel minimum trajectory where the switching function is > 0 over the entire trajectory is also on the reachable set. The fuel minimum formulation different from distance based cost function[4, 8, 9] as it exploits the switching structure. Thus our use of the switching function enables us to compute feasible fuel-optimal trajectories in addition to just reachable trajectories.

Algorithm Overview

This subsection covers the formulation of our algorithm and expands on how we use the prior equations to develop an algorithm to estimate the fuel-optimal control law along with the maneuver envelope.

To start, we will integrate forward along a reference trajectory, then solve for the updated control law backwards in stages. We can then apply this control law to estimate the reachable and feasible states. This algorithm outline has two outstanding issues. First, how do we select the initial reference trajectory? Second, what co-state values should we select? We will address these questions in order.

We begin by recognizing that there is a trivial solution to the optimal control problem we wrote. That solution is $\boldsymbol{\lambda}^N = 0$, $\boldsymbol{T} = 0$, and $T_{mag} = 0$. We can see this simple solution by setting $\boldsymbol{\lambda}^N = 0$. From this we know the switching function, Equation 82 is equal to $-\frac{\Delta t}{c}$, meaning that $T_{mag} = 0$ and $\boldsymbol{T} = 0$. The last stage is unpowered or a natural trajectory. We can use Equation 67 to solve for the $N-1$ stage. Equation 67 is linear in $\boldsymbol{\lambda}$. Therefore, when $\boldsymbol{\lambda^{i+1}} = 0$, $\boldsymbol{\lambda^{i}} = 0$. From these results we deduce that when $\boldsymbol{\lambda}^N = \boldsymbol{0}$, all $\boldsymbol{\lambda}^i = \boldsymbol{0}$ resulting in $T_{mag} = 0$ and $\boldsymbol{T} = 0$ for all segments. Therefor, an unpowered or natural trajectory is a valid solution to the optimal control problem we specified. The unpowered trajectory can act as our initial reference trajectory since it is consistent and optimal. Our first issue of defining the reference trajectory is solved by setting $\boldsymbol{\lambda}^N = \boldsymbol{0}$, $T_{mag} = 0$, and $\boldsymbol{T} = 0$ and integrating forward to obtain $\boldsymbol{F_x}$ and $\boldsymbol{F_u}$.

The next challenge is to compute the updated control laws, which requires solving for the terminal co-states. We will begin by explaining our solution, discuss why our particular cost function choice makes it possible, and discuss how we can use this formulation to identify the maneuver envelope.

In order to determine what the terminal co-states should be, we begin by examining Equation 83, which does not depend on the magnitude of the co-states just the unit direction. From a control law perspective we can sweep over the unit ball of the co-states to identify all the thrust directions. This procedure was used by Patel[10] to identify the reachable set. However, with our new formulation we also have to check the switching function. We see that Equation 82 is affine in the co-states meaning that as we scale up the magnitude of the co-states the switching function will become positive until it is fully activated over all stages. A fully activated switching structure corresponds with being on the reachable set. Trajectories where the switching function is < 0 for some stages would correspond with solutions on the feasible set. Using this information, we can develop a procedure for the co-states.

We first choose $\boldsymbol{\lambda}^N$ from a 6-dimensional unit ball, such that $\lambda_n = 0$ and $|\boldsymbol{\lambda}^N| = 1$. λ_n is set equal to 0 because we do not have a defined constraint for the state n. Rather, we allow n to be an optimal value. Using $\boldsymbol{\lambda}^N$ we can then compute the thrust directions as defined by Equation 75.

The next step is to identify the magnitudes of $|\boldsymbol{\lambda}^N|$ that will cause a change in the switching structure. To do this, we start solving and storing the switching values for a given $\boldsymbol{\lambda}^N$ on the unit ball.

$$SF^{i,\boldsymbol{\lambda}^N} = |\boldsymbol{F_T}\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c} \tag{84}$$

We can solve for the values of the co-states that make each segment of $SF^{i,\boldsymbol{\lambda}^N} = 0$. We do this by introducing a scaling factor, $\gamma^{i,\boldsymbol{\lambda}^N}$, on $\boldsymbol{\lambda}$ and setting each part of the switching function equal to 0. Doing so gives us

$$0 = |\boldsymbol{F_T}(\gamma^{i,\boldsymbol{\lambda}^N})\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T(\gamma^{i,\boldsymbol{\lambda}^N})\boldsymbol{\lambda}^{i+1} - \frac{\Delta t}{c} \tag{85}$$

Collecting terms, we get

$$0 = \gamma^{i,\boldsymbol{\lambda}^N}\left(|\boldsymbol{F_T}\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1}\right) - \frac{\Delta t}{c} \tag{86}$$

Multiplying by c and rearranging we get

$$\Delta t = c\gamma^{i,\boldsymbol{\lambda}^N}\left(|\boldsymbol{F_T}\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1}\right) \tag{87}$$

Solving for $\gamma^{i,\boldsymbol{\lambda}^N}$ we get

$$\gamma^{i,\boldsymbol{\lambda}^N} = \Delta t\left(c\left(|\boldsymbol{F_T}\lambda^{i+1}| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1}\right)\right)^{-1} \tag{88}$$

We can also write this equation in terms of the switching function as

$$\gamma^{i,\boldsymbol{\lambda}^N} = \Delta t\left(cSF^{i,\boldsymbol{\lambda}^N} + \Delta t\right)^{-1} \tag{89}$$

$\gamma^{i,\boldsymbol{\lambda}^N}$ is the scaling factor that we can iterate over to obtain different switching structures and hence different trajectories for a given $\hat{\boldsymbol{\lambda}}^N$. In this formulation, $\boldsymbol{\lambda}^N = \gamma^{i,\hat{\boldsymbol{\lambda}}^N}\hat{\boldsymbol{\lambda}}^N$. This equation addresses our second question of how to select the terminal co-states, $\boldsymbol{\lambda}^N$.

We can formally write out the algorithm, Algorithm 1, as

Algorithm 1: Feasible Set Algorithm

```
x^0 ← initial conditions
t^0 ← start time
dt ← segment duration
Seg ← number of segments
T ← 0
i ← 0
for (i = 0; i < Seg; i ← i + 1) do
    x^{i+1}, F_x^i, F_u^i ← F^i(x^i, T^i, t_0, t_0 + dt)
    t_0 ← t_0 + dt
end
N ← Number of sample points
for (j = 0; j < N; j ← j + 1) do
    i ← Seg − 1
    λ^{Seg} ← from unit ball
    for (i = Seg − 1; i ≥ 0; i ← i − 1) do
        T^{i,j} ← Equation 83
        λ^i ← Equation 67
        γ^{λ^{Seg},i} ← Equation 89
    end
end
for ∀λ̂ ∈ λ^{Seg} do
    for ∀γ ∈ γ^{Seg,i} do
        λ ← γλ̂
        SF^{Seg,i} ← Equation 82
    end
    x^{λ^{Seg},γ^{Seg,i}} ← Trajectory using activated T profile and SF^{Seg,i}
end
```

Revisiting the Cost Function

The algorithm and procedure we defined heavily depends on the linearity of many equations and being able to scale the co-states by a positive factor for all stages. We now revisit the various cost function options we had to highlight why the other options are not useful.

For Algorithm 1 to work, we require that Equation 67 be linear and not affine. We also require that we can arbitrarily scale $\boldsymbol{\lambda}^N$ by a positive factor. If we had used Equation 63, then Equation 67 would become, $H^i_{\boldsymbol{x}} = \boldsymbol{\lambda}^i = \boldsymbol{F_x}^{iT}\boldsymbol{\lambda}^{i+1} + L_{\boldsymbol{x}}$ and no longer be linear because it contains an affine term due to the state dependency. When we attempt to solve for an arbitrary $\boldsymbol{\lambda}^i$, we get a solution of the form

$$\boldsymbol{\lambda}^i = \left(\prod F_x^{Tj}\right)\boldsymbol{\lambda}^N + \sum\left(\prod F_x^{Tj-1}L_x^j\right) \quad (90)$$

When this equation is plugged into the switching structure it results in a form that is much more complicated.

The Mayer form, Equation 62, results in a terminal boundary condition of $\lambda_n = 1$. Given our solution procedure, having a known condition would appear beneficial. However, Equation 67 then becomes

$$\boldsymbol{\lambda}^i = \left(\prod_{N-1}^{i} F_x^{jT}\right)\left(\kappa \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}\right)\boldsymbol{\lambda}^N \tag{91}$$

This form is easier to handle, although it also makes solving for κ in the switching structure more complicated. While all these forms are theoretically equivalent, we see that the Lagrangian with the states and the Mayer form both result in a more complicated set of equations for the switching function and the co-state dynamics. For these reasons, we used Equation 65 as our cost function.

INITIAL BOUNDARY CONDITIONS

Our approach can also accommodate various initial boundary conditions. In this section, we derive various initial constraints to demonstrate that they can be analytically incorporated into the algorithm. The initial constraints we consider are position and/or velocity uncertainty with ellipsoid uncertainty[4] and ΔV maneuvers. In our previous work[10], we made simplifying assumptions since we were only concerned with the reachable/controllable set. For this paper, we remove those assumptions to account for interior points and the reachable/controllable boundary.

Earlier we derived how to account for initial constraints as well as initial costs. In this section, we exploit another feature of indirect multi-stage formulations. While the states must be consistent across the stages, the dynamics do not have to be, nor does the cost function.

Initial Hamiltonian For Additional Constraints

We begin by defining the dynamics, controls, and the Hamiltonian that converts the reference trajectory (0) to the optimal initial states (0+) as

$$\begin{bmatrix} r^{0+} \\ v^{0+} \\ n^{0+} \end{bmatrix} = \begin{bmatrix} r^{0} \\ v^{0} \\ n^{0} \end{bmatrix} + \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & I & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta r \\ \delta v_1 \\ \delta v_2 \\ \delta n \end{bmatrix} \tag{92}$$

where $[\delta r \delta v_1 \delta v_2 \delta n]$ are controls that represent the changes in position, change in velocity (ΔV), and change in inverse mass. δr and δv_1 are associated with the ellipsoid uncertainty constraints, whereas δv_2 and δn are associated with the ΔV inequality constraint. This approach is flexible and can be used to handle a wide variety of constraints.

Next, we need a cost function that accounts for the mass consumed in order to be consistent with the overall problem formulation. We write the cost function as the mass consumed rather than simply ΔV because we want the proper weighting between all the segments.

$$L^{0+} = m_0 - m_0 e^{-\frac{\Delta V}{c}} \tag{93}$$

This leads to a Hamiltonian expressed as

$$H^{0+} = \lambda^T \left(\begin{bmatrix} r^{0} \\ v^{0} \\ n^{0} \end{bmatrix} + \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & I & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta r \\ \delta v_1 \\ \delta v_2 \\ \delta n \end{bmatrix} \right) + m_0 - m_0 e^{-\frac{\Delta V}{c}} + \sum_i \nu_i C_i \tag{94}$$

Ellipsoid Uncertainty

We write the constraints for the ellipsoid uncertainty for position and velocity as

$$\frac{1}{2}\delta r^T E_r \delta r - \frac{1}{2} r_0^2 = 0 \tag{95}$$

$$\frac{1}{2}\delta v_1^T E_r \delta v_1 - \frac{1}{2} v_0^2 = 0 \tag{96}$$

In Equation 95 and Equation 96, δr represents the change in the position and velocity, respectively, from the initial conditions. Solving for δr we get

$$H_{\delta r} = 0 = \lambda_r + \nu_1 E_r \delta r \tag{97}$$

$$\delta r = -\nu_1 E_r^{-1} \lambda_r \tag{98}$$

$$0 = \frac{1}{2} r_0^2 - \frac{1}{2} \frac{\lambda_r^T E_r^{-1} \lambda_r}{\nu_1^2} \tag{99}$$

$$\nu_1 = \frac{\lambda_r^T E_r^{-1} \lambda_r}{r_0^2} \tag{100}$$

$$\delta r = -r_0 \frac{E_r^{-1} \lambda_r}{\sqrt{\lambda_r^T E_r^{-1} \lambda_r}} \tag{101}$$

If we follow the same steps, we get δv_1

$$H_{\delta v_1} = 0 = \lambda_v + \nu_2 E_v \delta v_1 \tag{102}$$

$$\delta v_1 = -v_0 \frac{E_v^{-1} \lambda_v}{\sqrt{\lambda_v^T E_v^{-1} \lambda_v}} \tag{103}$$

Initial Instantaneous Maneuver Inequality Constraint

The equations for an initial ΔV maneuver are

$$\frac{1}{2}\delta v_2^T \delta v_2 - \Delta V_{max}^2 \leq 0 \tag{104}$$

$$\delta n - \frac{1}{m_0 e^{-\frac{\Delta V}{c}}} + \frac{1}{m_0} = 0 \tag{105}$$

In these equations, δv_2 is the change in velocity associated with the ΔV maneuver, and δn is the change in $\frac{1}{m}$ due to the ΔV maneuver. For the following equations, $\Delta V = |\delta v_2|$.

$$H_{\delta v_2} = 0 = \lambda_v + \frac{\delta v_2}{\Delta V} \frac{m_0}{c} e^{-\frac{\Delta V}{c}} + \nu_3 \delta v_2 + \nu_4 \frac{\delta v_2}{\Delta V} \frac{1}{m_0 c} e^{\frac{\Delta V}{c}} \tag{106}$$

$$H_{\delta n} = 0 = \lambda_n + \nu_4 \tag{107}$$

This formulation gives us $\nu_4 = -\lambda_n$. Letting $m_+ = m_0 e^{-\frac{\Delta V}{c}}$ we can transform Equation 106 into

$$0 = \lambda_v - \frac{\delta v_2}{\Delta V} \frac{m_+}{c} + \nu_3 \delta v_2 + \nu_4 \frac{\delta v_2}{\Delta V} \frac{1}{m_+ c} \tag{108}$$

A solution process with an inequality constraint will first assume the constraint is binding (treat the inequality as an equality constraint) and then check to see if $\nu_3 \geq 0$. If not, we then set $\nu_3 = 0$ and solve. Applying this process and substituting in, $\nu_4 = -\lambda_n$ and $\Delta V = \Delta V_{max}$, we get

$$0 = \lambda_v + \left(\frac{m_+}{\Delta V_{max} c} + \nu_3 - \lambda_n \frac{1}{\Delta V_{max} m_+ c} \right) \delta v_2 \tag{109}$$

We can extract δv_2 and by inspection see it must be parallel to λ_v and have a magnitude equal to ΔV_{max}, which gives us

$$\delta v_2 = -\Delta V_{max} \frac{\lambda_v}{|\lambda_v|} \tag{110}$$

Next, we need to solve for ν_3. To do this, we see that the term inside the parenthesis from Equation 109 must be equal to $\frac{|\boldsymbol{\lambda_v}|}{\Delta V_{max}}$. Using this fact, we obtain

$$\frac{|\lambda_v|}{\Delta V_{max}} = \frac{m_+}{\Delta V_{max} c} + \nu_3 - \lambda_n \frac{1}{\Delta V_{max} m_+ c} \tag{111}$$

Solving for ν_3 we get

$$\nu_3 = \frac{|\lambda_v|}{\Delta V_{max}} + \frac{\lambda_n}{\Delta V_{max} m_+ c} - \frac{m_+}{\Delta V_{max} c} \geq 0 \tag{112}$$

Since we care to know only if $\nu_3 \geq 0$ rather than its actual value we can further reduce this condition. We multiply the right hand side by $\Delta V_{max} m_+ c$ and get

$$-m_+^2 + |\lambda_v| m_+ c + \lambda_n \geq 0 \tag{113}$$

If this condition is true, then the constraint is binding and Equation 110 is the optimal solution. If Equation 110 is not true then the constraint is not binding and we set $\nu_3 = 0$ and resolve getting

$$0 = \lambda_v + \left(\frac{m_+}{\Delta V c} - \lambda_n \frac{1}{\Delta V m_+ c} \right) \delta v_2 \tag{114}$$

We then pull out ΔV and solve for $\hat{\delta v_2}$

$$0 = \lambda_v + \left(\frac{m_+}{c} - \frac{\lambda_n}{m_+ c} \right) \frac{\delta v_2}{\Delta V} \tag{115}$$

$$\frac{\delta v_2}{\Delta V} = -\frac{\lambda_v}{|\lambda_v|} \tag{116}$$

This equation means that the term inside the parenthesis must equal $|\boldsymbol{\lambda_v}|$.

$$|\boldsymbol{\lambda_v}| = \left(\frac{m_+}{c} - \frac{\lambda_n}{m_+ c} \right) \tag{117}$$

We can now solve and get

$$0 = \frac{m_+}{c} - \frac{\lambda_n}{m_+ c} - |\boldsymbol{\lambda_v}| \tag{118}$$

$$0 = m_+^2 - |\boldsymbol{\lambda_v}| c m_+ - \lambda_n \tag{119}$$

Applying the quadratic equation and solving for m_+ we obtain

$$m_+ = \frac{|\boldsymbol{\lambda_v}|c \pm \sqrt{|\boldsymbol{\lambda_v}|^2 c^2 + 4\lambda_n}}{2} \tag{120}$$

Because we are maximizing m_+, thus minimizing fuel consumed, we take the positive sign and can now solve for ΔV giving us

$$\Delta V = -c \ln\left(\frac{|\boldsymbol{\lambda_v}|c + \sqrt{|\boldsymbol{\lambda_v}|^2 c^2 + 4\lambda_n}}{2m_0}\right) \tag{121}$$

$$\delta v_2 = c \ln\left(\frac{|\boldsymbol{\lambda_v}|c + \sqrt{|\boldsymbol{\lambda_v}|^2 c^2 + 4\lambda_n}}{2m_0}\right)\frac{\boldsymbol{\lambda_v}}{|\boldsymbol{\lambda_v}|} \tag{122}$$

We show that our formulation remains analytic and avoids the need for iterative numerical methods under a variety of initial boundary conditions from uncertainty in position/velocity to potential ΔV maneuvers. This important result allows for the rapid and automatic estimation of all feasible trajectories without risking numerical failures. Finally, we note that because our new (0) depends only on the co-states at (0+) our prior algorithmic assumptions regarding linearity and the switching function are valid. If we also had a stage before (0), our assumptions would be violated.

EQUATIONS FOR 2-NORM COST FUNCTION

In this section, we derive the equations using a $||.||_2$ and discuss the implications and value of this formulation for automated spacecraft maneuvers.

We begin by defining the Hamiltonian with the 2-norm.

$$H^i = \boldsymbol{F}^{iT}\boldsymbol{\lambda}^{i+1} + \frac{1}{2}T_{mag}^2 + \nu_1(|\boldsymbol{T}| - T_{mag}) + \nu_2(\frac{1}{2}T_{mag}^2 - \frac{1}{2}T_{max}^2) \tag{123}$$

Following the same procedure as before, we obtain the control. The control law has the same form as that for the fuel-minimum formulation.

$$H_{\boldsymbol{u}} = 0 = \boldsymbol{F_u}^T\boldsymbol{\lambda}^{i+1} + \nu_1\begin{bmatrix}\frac{\boldsymbol{T}}{|\boldsymbol{T}|}\\ -1\end{bmatrix} + \nu_2\begin{bmatrix}0\\ T_{mag}\end{bmatrix} + \begin{bmatrix}0\\ T_{mag}\end{bmatrix} \tag{124}$$

$$\hat{\boldsymbol{T}} = -\frac{\boldsymbol{F_T}^{i,T}\boldsymbol{\lambda}^{i+1}}{|\boldsymbol{F_T}^{i,T}\boldsymbol{\lambda}^{i+1}|} \tag{125}$$

where

$$\nu_1 = \left|\boldsymbol{F_T}^T\boldsymbol{\lambda}^{i+1}\right| \tag{126}$$

We now solve by first assuming the constraint is binding. We assume that $T_{mag} = T_{max}$ then solve for ν_2.

$$0 = \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1} - \nu_1 + (\nu_2 + 1)T_{max} \tag{127}$$

$$\nu_2 = \frac{\left|\boldsymbol{F_T}^T\boldsymbol{\lambda}^{i+1}\right| - \boldsymbol{F}_{T_{mag}}^T\boldsymbol{\lambda}^{i+1}}{T_{max}} - 1 > 0 \tag{128}$$

If ν_2 is greater than 0, then $T_{mag} = T_{max}$. Otherwise,

$$0 = \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} - \nu_1 + T_{mag} \tag{129}$$

$$0 = \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} - \left|\boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1}\right| + T_{mag} \tag{130}$$

$$T_{mag} = \left|\boldsymbol{F_T}^T \boldsymbol{\lambda}^{i+1}\right| - \boldsymbol{F}_{T_{mag}}^T \boldsymbol{\lambda}^{i+1} \tag{131}$$

This formulation results in a similar structure and we can employ the same technique to sweep over a range of $\boldsymbol{\lambda}^N$ values to generate a map of the feasible set. A more practical and computationally efficient approach is to begin by setting T_{mag} to Equation 131, and then if $T_{mag} > T_{max}$ let $T_{mag} = T_{max}$.

Applicability for Automated Spacecraft Maneuver

The $||.||_2$ norm has features that make it useful for automated spacecraft maneuver. Recall that for the fuel-minimum formulation, we skipped over when the switching function would hit a 0. We did this because we cannot easily solve for the control when it is 0, which would require taking additional partials. The implication is that the fuel-minimum formulation is useful for many applications but not for rapid automated spacecraft maneuvers.

The $||.||_2$ ameliorates this issue, as we can cleanly recover the control law for any co-state value. Since the $||.||_2$ uses the same algorithm it is fast and efficient and therefor a prime candidate for use in automated spacecraft maneuver applications.

CONCLUSIONS

We developed a fast and efficient algorithm for computing the reachable set, including the fuel-optimal control law, for electric propulsion missions. Our approach does not require an initial guess, thus making it useful in automated applications. In our paper, we exploited the link between fuel-optimal and time optimal trajectories and leveraged an indirect multi-stage formulation, which is not commonly used in aerodynamics. This approach allows our single algorithm to compute the reachable set and other feasible trajectories. We show that our approach can accommodate various initial boundary including ΔV maneuvers and ellipsoid uncertainty, making it useful for many other applications. We also derive the $||.||_2$ version, which is particularly well suited for automated spacecraft maneuver.

ACKNOWLEDGEMENTS

The authors thank the Institute for Defense Analyses for providing funding for the conference and publication. We also appreciate the comments of the reviewers.

REFERENCES

[1] A. E. Bryson and Y.-C. Ho, *Applied optimal control: optimization, estimation, and control*. Routledge, 2018.

[2] A. V. Rao, "A survey of numerical methods for optimal control," *Advances in the Astronautical Sciences*, Vol. 135, No. 1, 2009, pp. 497–528.

[3] J. D. Thorne and C. D. Hall, "Minimum-time continuous-thrust orbit transfers," *The Journal of the astronautical sciences*, Vol. 45, No. 4, 1997, pp. 411–432.

[4] M. J. Holzinger and D. J. Scheeres, "Reachability results for nonlinear systems with ellipsoidal initial sets," *IEEE transactions on aerospace and electronic systems*, Vol. 48, No. 2, 2012, pp. 1583–1600.

[5] D. Aguilar Marsillach and M. J. Holzinger, "Spacecraft Custody Maintenance and Maneuver Detection Using Robotic Telescopes and Reachable Sets," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 4, 2021, pp. 667–683.

[6] P. Gurfil, *Modern astrodynamics*. Elsevier, 2006.

[7] M. Bando and D. J. Scheeres, "Nonlinear attractive and reachable sets under optimal control in three-body problem," *Journal of guidance, control, and dynamics*, Vol. 41, No. 8, 2018, pp. 1766–1775.

[8] C. M. Chilan and B. A. Conway, "A reachable set analysis method for generating near-optimal trajectories of constrained multiphase systems," *Journal of Optimization Theory and Applications*, Vol. 167, No. 1, 2015, pp. 161–194.

[9] Q. Chen, D. Qiao, and C. Wen, "Minimum-Fuel Low-Thrust Trajectory Optimization via Reachability Analysis and Convex Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 5, 2021, pp. 1036–1043.

[10] P. Patel and D. Scheeres, "Rapidly and Automaticlly Estimating the Reachability of Electric Propulsion Spacecraft," *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2022, p. 571.

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION

1. REPORT DATE	**2. REPORT TYPE**	**3. DATES COVERED**	
xx-12-2022	Final	**START DATE** Jun 2022	**END DATE** Oct 2022

4. TITLE AND SUBTITLE

NO INITIAL GUESS REQUIRED: RAPIDLY COMPUTING THE FEASIBLE SET OF FUEL-OPTIMAL ELECTRIC PROPULSION TRAJECTORIES

5a. CONTRACT NUMBER	**5b. GRANT NUMBER**	**5c. PROGRAM ELEMENT NUMBER**
HQ0034-19-D-0001		
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**
C7265		

6. AUTHOR(S)

Patel, Prashant, R.
Scheeres, Daniel, J.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Institute for Defense Analyses /
Systems and Analyses Center Research Division
730 East Glebe Road
Alexandria, Virginia 22305

8. PERFORMING ORGANIZATION REPORT NUMBER

IDA Document NS D-33273
Log: H 22-000426

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Institute for Defense Analyses /
Systems and Analyses Center Research Division
730 East Glebe Road
Alexandria, Virginia 22305

10. SPONSOR/MONITOR'S ACRONYM(S)

IDA / SAC

11. SPONSOR/MONITOR'S REPORT NUMBER

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We discuss our automated algorithm (no initial guess) that can map out the fuel optimal feasible set including the reachable/controllable set for electric propulsion missions. We provide a detailed discussion of why an indirect multi-stage formulation is well suited to solving this problem over direct and indirect continuous approaches and present an in depth discussion of the theoretical and mathematical underpinnings of our formulation. Our approach also generates the fuel-optimal control law to points in the feasible set. Reachable sets for electric propulsion spacecraft are important to many problems including: dynamic re-planning, robust mission design, space situational awareness, advanced concept development, and threat assessments. Feasible solutions are important for trajectory design and maneuver. Our automated and fast algorithm has applications to many different space maneuver problems.

15. SUBJECT TERMS

Space; trajectory; fuel-optimal; rapid; automated; optimal control; electric propulsion

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	28

19a. NAME OF RESPONSIBLE PERSON	**19b. PHONE NUMBER**
Patel, Prashant, R.	(703) 575-1439