

# Low Thrust Minimum Time Orbit Transfer Nonlinear Optimization Using Impulse Discretization via the Modified Picard–Chebyshev Method

Darin Koblick<sup>1,2,3</sup>, Shujing Xu<sup>4</sup>, Joshua Fogel<sup>5</sup> and Praveen Shankar<sup>1</sup>

**Abstract:** The Modified Picard-Chebyshev Method (MPCM) is implemented as an orbit propagation solver for a numerical optimization method that determines minimum time orbit transfer trajectory of a satellite using a series of multiple impulses at intermediate waypoints. The waypoints correspond to instantaneous impulses that are determined using a nonlinear constrained optimization routine, SNOPT with numerical force models for both Two-Body and  $J_2$  perturbations. It is found that using the MPCM increases run-time performance of the discretized low-thrust optimization method when compared to other sequential numerical solvers, such as Adams-Bashforth-Moulton and Gauss-Jackson 8<sup>th</sup> order methods.

**Keywords:** Low Thrust, Picard–Chebyshev, Optimization, Multi-Impulse, Boundary Value Problem.

## 1 Introduction

Low thrust propulsion systems provide an efficient means of orbit transfer for both interplanetary trajectories as well as Earth-centric trajectories where mission time is not significantly important [Keaton (2002)]. Low thrust propulsion systems, however, operate continuously during the mission and hence depend on having an optimal trajectory defined for the orbit transfer. The optimization of low thrust trajectories using indirect methods encounter highly sensitive two-point boundary value problems with unknown co-state boundary conditions. In the case of direct methods the control history is parameterized leading to a high-dimensional nonlinear programming problem. This results in poor convergence results as the dimensionality of the parameter space grows larger. Thus both direct and indirect methods

---

<sup>1</sup> California State University - Long Beach, CA, USA.

<sup>2</sup> Millennium Space Systems Inc., El Segundo, CA, USA.

<sup>3</sup> Claremont Graduate University, CA, USA.

<sup>4</sup> University of California - San Diego, CA, USA.

<sup>5</sup> University of Southern California, Los Angeles, CA, USA.

typically lead to challenging numerical difficulties. There have been a number of different approaches proposed to solve the general optimization problem of low constant thrust trajectories between orbits [Betts (2000); Kechichian (2007); Yang (2009); Falck and Dankanich (2012); Graham and Rao (2015)]. These approaches generally fall into either direct or indirect methods. The direct methods typically convert the low thrust trajectory into a parameter optimization problem which can be solved using a nonlinear programming method. Indirect methods are developed around calculus of variations or the maximum principle [Bryson and Ho (1975)]. Using an indirect method can be difficult as there are typically six (resp. four) costate variables for three-dimensional (resp. two-dimensional) cases, which must be determined and they are not only extremely sensitive but difficult to guess for generic orbit transfer problems [Ulybyshev (2009)]. An extensive review of the low thrust trajectory optimization based on direct, indirect and stochastic methods and its application can be found in [Kim (2005)].

The continuous low thrust trajectory optimization problem can be discretized into waypoints or nodes with the solution consisting of impulses computed at each of the waypoints using an optimization routine over the entire trajectory. Such a method is described as multi-impulse low thrust trajectory optimization. Similar to the performance of most well behaved numerical methods, the discretized solution approaches the continuous solution as the number of nodes are increased. The method of using multiple impulses to approximate a continuous low thrust trajectory has been addressed by several studies [Xiao-yong, Yi-jun, Hong-bo, and Guo-jian (2013); Polsgrove, Kos, Hopkins, and Crane (2006)].

One of the requirements of the multi-impulse method of low thrust optimization is a numerical solver to propagate the orbit between the waypoints since an analytical solution may not exist for most transfer problems utilizing numerical force models. The choice of this numerical solver plays a significant role in the run-time performance of the multi-impulse method, especially when the number of waypoints are increased to improve accuracy. This paper presents the implementation of the Modified Picard–Chebyshev Method as the numerical solver for orbit propagation within a multi-impulse method of low thrust optimization. The low thrust optimization utilizes the solution from a direct or indirect optimization [Kechichian (1997a)] method as an initial guess and discretizes this guess into evenly distributed waypoints. The method then applies a nonlinear optimization routine, the Sparse Nonlinear OPTimizer (SNOPT)[Gill, Murray, and Saunders (2002)], to determine the instantaneous change in velocity,  $\Delta V$ , at each of these waypoints while minimizing the total  $\Delta V$  over the entire trajectory. It is important to emphasize here that the novelty in the proposed method is the utilization of the Modified Picard–Chebyshev Method [Bai (2010)] for both the boundary value and initial value problem solutions required for the optimization. This implementation im-

proves the run-time performance as compared to typical numerical solvers, such as the Adams–Bashforth–Moulton (ODE113) [Bradley (2012)] and Gauss–Jackson 8<sup>th</sup> (GJ8) [Berry and Healy (2004)] methods.

The rest of the paper is organized as follows: Section 2 addresses some of the fundamental concepts and background. Section 3 describes the multi-impulse low thrust optimization method that utilizes the MPCM as a numerical solver for the initial value and boundary value problems. Section 4 presents the results of utilizing the method on two orbit transfer problems. The results are divided into two parts. The multi-impulse method is compared to an indirect optimization method proposed by [Kechichian (1997a)] and a numerical method based on Edelbaum’s analytical circular transfer problem [Kechichian (1997b)] to show the fidelity of the discretization strategy with the implementation of the Modified–Picard Chebyshev Method. Then, the run-time performance of the method with the Adams–Bashforth–Moulton and Gauss–Jackson 8<sup>th</sup> order solvers is compared to that of the implementation with the MPCM. Section 5 provides a discussion of the results and the characteristics of the method and is followed by conclusions and future work.

## 2 Background

This section discusses some of the fundamental concepts that will be utilized in developing and implementing the multi-impulse low thrust orbit transfer method.

### 2.1 Optimal Control Problem

Consider the nonlinear system

$$\dot{x} = f(x, u, t), \quad t \in [t_0, t_f]. \quad (1)$$

The optimal control problem is defined as the process of determining the control input  $u^*$  such that an associated cost function

$$J = \phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x, u, t) dt \quad (2)$$

is minimized subject to the constraints

$$\Psi(x(t_0), x(t_f), t_0, t_f) = 0. \quad (3)$$

In the case of spacecraft trajectory control, these constraints are generally related to the maximum thrust available or the total time of the mission. This optimal control problem can be solved by both direct and indirect methods. In direct methods, the optimal control problem is converted to a nonlinear programming problem which can be solved using a variety of methods. While the direct methods tend to be

highly robust, the solutions obtained are generally not very accurate [Von Stryk and Bulirsch (1992)].

Indirect methods are based on calculus of variations and Pontryagin's minimum principle [Lewis, Vrabie, and Syrmos (2012); Geering (2007)]. These methods modify the cost function to include the equations of motion and state and control variable constraints. The Hamiltonian function,  $H$ , that results from applying the calculus of variations to the augmented cost function under the constraints can be written as

$$H = L(x, u, t) + \lambda^T f(x, u, t), \quad (4)$$

where  $\lambda$  is a vector of costates associated with the equation of motion constraints. The optimality conditions are obtained by computing the first variation of  $J$  resulting in

$$\dot{x} = \frac{\partial H}{\partial \lambda} = f(x, u, t), \quad (5)$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = -\frac{\partial L(x, u, t)}{\partial x} - \lambda^T \frac{\partial f(x, u, t)}{\partial x}. \quad (6)$$

The above equations represent a Boundary Value Problem (BVP) and while the boundary conditions of the state variables are given, the boundary conditions for the costates are normally unknown. Optimization does not require minimization of the costates  $\lambda$ , but their determination is required for satisfying optimality constraints through Eq. 4. There are several methods that have been discussed for solving such BVPs, for example, the shooting method; but since the initial values of the costates are unknown, these methods become extremely cumbersome to implement.

## 2.2 Orbit Equations of Motion

Equinoctial orbit elements are frequently used in solving low thrust optimization problems as they help in avoiding singularities involved with using the classical Keplerian orbital elements [Chobotov (1996); Broucke and Cefola (1972)]. The mapping to convert classical elements to equinoctial elements is

$$\begin{bmatrix} h \\ k \\ p \\ q \\ \lambda \end{bmatrix} = \begin{bmatrix} e \sin(\omega + \Omega) \\ e \cos(\omega + \Omega) \\ \tan(\frac{1}{2}i) \sin \Omega \\ \tan(\frac{1}{2}i) \cos \Omega \\ M + \omega + \Omega \end{bmatrix}, \quad \begin{bmatrix} e \\ i \\ \Omega \\ \omega \\ M \end{bmatrix} = \begin{bmatrix} \sqrt{h^2 + k^2} \\ 2 \tan^{-1} \sqrt{p^2 + q^2} \\ \tan^{-1}(p/q) \\ \tan^{-1}(h/k) - \Omega \\ \lambda - \tan^{-1}(h/k) \end{bmatrix}. \quad (7)$$

The two-body equations of motion for low thrust based on the equinoctial orbital elements can then be written as

$$\dot{a} = \left( \frac{\partial a}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad \dot{h} = \left( \frac{\partial h}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad \dot{k} = \left( \frac{\partial k}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad (8)$$

$$\dot{p} = \left( \frac{\partial p}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad \dot{q} = \left( \frac{\partial q}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad \dot{\lambda} = n + \left( \frac{\partial \lambda}{\partial \dot{r}} \right)^T \cdot \hat{u} \bar{a}_t, \quad (9)$$

where

$$\begin{aligned} \frac{\partial a}{\partial \dot{r}} &= \frac{2}{n^2 a} (\dot{X}_1 \hat{f} + \dot{Y}_1 \hat{g}), \\ \frac{\partial h}{\partial \dot{r}} &= \frac{\sqrt{1-h^2-k^2}}{na^2} \left[ \left( \frac{\partial X_1}{\partial k} - \frac{h\beta}{n} \dot{X}_1 \right) \hat{f} + \left( \frac{\partial Y_1}{\partial k} - \frac{h\beta}{n} \dot{Y}_1 \right) \hat{g} \right] + \frac{k(qY_1 - pX_1)}{na^2 \sqrt{1-h^2-k^2}} \hat{w}, \\ \frac{\partial k}{\partial \dot{r}} &= -\frac{\sqrt{1-h^2-k^2}}{na^2} \left[ \left( \frac{\partial X_1}{\partial h} + k\beta \frac{\dot{X}_1}{n} \right) \hat{f} + \left( \frac{\partial Y_1}{\partial h} + k\beta \frac{\dot{Y}_1}{n} \right) \hat{g} \right] - \frac{h(qY_1 - pX_1)}{na^2 \sqrt{1-h^2-k^2}} \hat{w}, \\ \frac{\partial p}{\partial \dot{r}} &= \frac{(1+p^2+q^2)Y_1}{2na^2 \sqrt{1-h^2-k^2}} \hat{w}, \\ \frac{\partial q}{\partial \dot{r}} &= \frac{(1+p^2+q^2)X_1}{2na^2 \sqrt{1-h^2-k^2}} \hat{w}, \\ \frac{\partial \lambda}{\partial \dot{r}} &= \frac{1}{na^2} \left[ -2X_1 + 3\dot{X}_1 t + \sqrt{1-h^2-k^2} \left( h\beta \frac{\partial X_1}{\partial h} + k\beta \frac{\partial X_1}{\partial k} \right) \right] \hat{f} \\ &\quad + \frac{1}{na^2} \left[ -2Y_1 + 3\dot{Y}_1 t + \sqrt{1-h^2-k^2} \left( h\beta \frac{\partial Y_1}{\partial h} + k\beta \frac{\partial Y_1}{\partial k} \right) \right] \hat{g} \\ &\quad + \frac{qY_1 - pX_1}{na^2 \sqrt{1-h^2-k^2}} \hat{w}, \end{aligned}$$

$\bar{a}_t$  is the magnitude of the acceleration vector due to a constant thrust and  $\hat{u}$  corresponds the direction (unit vector) of the applied constant thrust. The set of equinoctial orbital elements (denoted by  $z$ ) is provided in matrix form:

$$M(z, F) = \left[ \frac{\partial a}{\partial \dot{r}} \quad \frac{\partial h}{\partial \dot{r}} \quad \frac{\partial k}{\partial \dot{r}} \quad \frac{\partial p}{\partial \dot{r}} \quad \frac{\partial q}{\partial \dot{r}} \quad \frac{\partial \lambda}{\partial \dot{r}} \right]^T \quad (10)$$

The Euler–Lagrange equations can then be determined through numerical integration of

$$\dot{\lambda}_z = -\lambda_z^T \frac{\partial M}{\partial z} \bar{a}_t \hat{u} - \lambda_\lambda \frac{\partial n}{\partial z}. \quad (11)$$

Note that the partial matrices,  $\partial M / \partial z$ , are provided in the appendices of [Kechichi-an (1997a)].

### 2.3 Low Thrust Optimal Orbit Transfer Problem

The low thrust orbit transfer method can be described as a mission that continuously utilizes low thrust to transfer a satellite from an initial position to a final position. The unique nature of such a mission as compared to an impulse transfer is that the orbit (orbital elements) continuously changes during the mission. The constant low thrust optimal control problem involves the determination of the unit vector  $\hat{u}$  along the spacecraft trajectory that minimizes either transfer time or total  $\Delta V$  with the assumption that the thrust  $\bar{a}_t$  is held constant over the transfer.

This section reviews two methods for constant low thrust transfer optimization based on Euler–Lagrange multipliers and Edelbaum’s method. These two methods are also used as a benchmark for the proposed multi-impulse method.

#### 2.3.1 Minimum Time Continuous Thrust Using Euler–Lagrange Multipliers

Under certain assumptions, there exists a function of adjoint variables  $\lambda(t)$  and a vector function  $v(t)$  in which the first order necessary conditions for optimality are valid; see [Von Stryk and Bulirsch (1992)]. This is known as the Hamiltonian function of the multi-point boundary value problem defined by Eqs. 12–14 and is shown generically as

$$H = \lambda(t)^T f + v(t)^T g. \quad (12)$$

where  $f$  and  $g$  are some force functions which describe a dynamic system of nonlinear ODEs (e.g.  $f(x(t), u(t), t)$ , and  $g(x(t), u(t), t)$ ). The corresponding coupled differential equations for  $t_0 \leq t \leq t_f$  are defined as

$$\dot{x} = \frac{\partial H}{\partial \lambda} = f, \quad (13)$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = -\lambda^T \frac{\partial f}{\partial x} - v^T \frac{\partial g}{\partial x}. \quad (14)$$

For the low thrust minimum time, Eq. 12 can be expressed as [Kechichian (1997a)]

$$H(t) = \lambda_z^T \dot{z} = \lambda_z^T M(z, F) f_t \hat{u} + \lambda_\lambda n, \quad (15)$$

where  $z = (a, h, k, p, q, \lambda)$ ,  $\dot{z}$  is the derivative of the equinoctial elements,  $\lambda_z$  describes all of the Euler–Lagrange Multipliers,  $F$  is the eccentric longitude,  $f_t$  is the thrusting force and  $n$  is the mean motion. The unit acceleration vector,  $\hat{u}$ , is parallel to  $\lambda_z^T M(z, F)$  in order to maximize the Hamiltonian, and is obtained from

$$\hat{u} = \frac{(\lambda_z^T M)^T}{|\lambda_z^T M|}. \quad (16)$$

To minimize the total flight time, and indirectly the  $\Delta V$ , the performance index  $J$  can be minimized:

$$J = \int_{t_0}^{t_f} dt = t_f - t_0. \quad (17)$$

For fixed  $t_0$ , minimization of  $t_f$  or maximization of  $-t_f$  and  $H(t) = \lambda_z^T \dot{z}$  gives rise to the transversality condition,  $H(t_f) = 1$  [Kechichian (1997a); Kechichian (1997c); Chobotov (1996)] resulting in time-optimality.

### 2.3.2 Edelbaum's Solution For Non-Coplanar Circular Orbit Transfers

Edelbaum (1961) developed an analytical solution for the optimization of low thrust orbit transfer problems that could be applied to circular trajectories. This section details a numerical solution based on Edelbaum's method that is valid for inclination changes of less than  $114.6^\circ$  and assumes that the acceleration magnitude is small enough such that the transfer orbits are circular [Kechichian (1997b); Eagle (2013)]. Much like the acceleration unit vector that can be determined by the costates in Eq. 11, the acceleration can be found for Edelbaum's solution as well. In this case, the acceleration unit vector is determined from

$$\hat{u}_T(t) = \begin{bmatrix} \sin(\beta(t)) \\ \cos(\beta(t)) \cos(\psi(t)) \\ \cos(\beta(t)) \sin(\psi(t)) \end{bmatrix} = \begin{bmatrix} U_{T_r} \\ U_{T_t} \\ U_{T_n} \end{bmatrix}, \quad (18)$$

where  $[\hat{r}, \hat{t}, \hat{n}]$  is the spacecraft-centered radial-tangential-normal coordinate system defined by the following transformation matrices:

$$\begin{bmatrix} \hat{r} \\ \hat{t} \\ \hat{n} \end{bmatrix} = [\mathbf{R}_{rtn}^{eci}]^T \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix}, \quad \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} = \mathbf{R}_{rtn}^{eci} \begin{bmatrix} \hat{r} \\ \hat{t} \\ \hat{n} \end{bmatrix}, \quad \mathbf{R}_{rtn}^{eci} = \begin{bmatrix} \frac{\vec{r}}{|\vec{r}|} & \frac{(\vec{r} \times \vec{v}) \times \vec{r}}{|\vec{r} \times \vec{v}| |\vec{r}|} & \frac{\vec{r} \times \vec{v}}{|\vec{r} \times \vec{v}|} \end{bmatrix}. \quad (19)$$

Note that the acceleration unit vector shown in Eq. 16 is in the equinoctial frame  $[\hat{f}, \hat{g}, \hat{w}]$  and can be transformed to the inertial frame through the following matrices [Chobotov (1996)]:

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} = \mathbf{R}_{fgw}^{eci} \begin{bmatrix} \hat{f} \\ \hat{g} \\ \hat{w} \end{bmatrix}, \quad \begin{bmatrix} \hat{f} \\ \hat{g} \\ \hat{w} \end{bmatrix} = [\mathbf{R}_{fgw}^{eci}]^T \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix}, \quad \begin{bmatrix} \hat{f} \\ \hat{g} \\ \hat{w} \end{bmatrix} = [\mathbf{R}_{fgw}^{eci}]^T \mathbf{R}_{rtn}^{eci} \begin{bmatrix} \hat{r} \\ \hat{t} \\ \hat{n} \end{bmatrix}, \quad (20)$$

where

$$\mathbf{R}_{fgw}^{eci} = \begin{bmatrix} c_\Omega^2 + c_i s_\Omega^2 & c_\Omega s_\Omega - s_\Omega c_i c_\Omega & s_\Omega s_i \\ s_\Omega c_\Omega - s_\Omega c_\Omega c_i & s_\Omega^2 + c_i c_\Omega^2 & -c_\Omega s_i \\ -s_\Omega s_i & s_i c_\Omega & c_i \end{bmatrix}. \quad (21)$$

Note that the Keplerian orbital elements  $\Omega$  and  $i$  can be found from equinoctial elements as shown in Eq. 7:  $\Omega = \tan^{-1}(p/q)$ , and  $i = 2 \tan^{-1}\left(\sqrt{p^2 + q^2}\right)$ .

The initial yaw angle of the acceleration vector  $\beta_0$  is defined by

$$\beta_0 = \tan^{-1}\left(\frac{\sin\left(\frac{\pi}{2}\Delta i\right)}{V_0/V_f - \cos\left(\frac{\pi}{2}\Delta i\right)}\right), \quad (22)$$

where  $V_0$  is the magnitude of velocity in the initial circular orbit, and  $V_f$  is the velocity in the desired circular orbit (e.g., final orbit after the maneuver). The yaw angle and inclination with respect to time can be found by

$$\beta(t) = \tan^{-1}\left(\frac{V_0 \sin \beta_0}{V_0 \cos \beta_0 - \bar{a}_t t}\right). \quad (23)$$

The total change in velocity required for the transfer,  $\Delta V$ , is given by

$$\Delta V = \sqrt{V_0^2 - 2V_0V_f \cos\left(\frac{\pi}{2}\Delta i\right) + V_f^2}. \quad (24)$$

The change in inclination with respect to time is

$$\Delta i(t) = \frac{2}{\pi} \left[ \tan^{-1}\left(\frac{\bar{a}_t t - V_0 \cos \beta_0}{V_0 \sin \beta_0}\right) + \frac{\pi}{2} - \beta_0 \right]. \quad (25)$$

The instantaneous velocity can be computed at anytime during the orbital transfer

$$V(t) = V_0 \sin \beta_0 \frac{\sqrt{1 + \tan^2(\beta(t))}}{\tan(\beta(t))}. \quad (26)$$

Using the orbital-energy-invariance law (vis-viva equation), the semi-major axis with respect to time can be determined from the velocity magnitude

$$a(t) = \frac{\mu}{V(t)^2}, \quad (27)$$

where  $\mu$  is the standard gravitational parameter of the central body.

The out of plane yaw-angle,  $\beta$ , and the in-plane pitch angle,  $\psi$  are related to the equinoctial frame in the following way:

$$\beta = \sin^{-1}(U_{T_r}) \quad (28)$$

$$\psi = \tan^{-1}(U_{T_n}, U_{T_t}) \quad (29)$$



## 2.4 Overview of Numerical Force Models for Orbit Propagation

One benefit of using numerical techniques when modeling satellite orbit dynamics is that it is easy to incorporate additional acceleration components into the equations of motion. Cowell's formulation for the two-body equation of motion [Vallado and McClain (2001)] is

$$\ddot{\vec{r}} = -\frac{\mu}{r^3}\vec{r} + \vec{a}_{perturbed}, \quad (30)$$

where  $\vec{a}_{perturbed}$  is the acceleration caused by additional forces acting on the object. These forces can be due to gravitational perturbations, a low thrust propulsion system, atmospheric drag, solar radiation pressure, etc. In this paper, the low thrust optimization problem is solved for the case of the non-spherical Earth modeled by  $J_2$  perturbations.

### 2.4.1 Modeling $J_2$ Perturbations

The second zonal harmonic,  $J_2$ , is among the most dominant of the harmonics and the perturbation acceleration due to  $J_2$  is significant for orbits at and below GEO altitude [Larson and Wertz (1999)]. Eq. 31 shows the magnitude of this term when other zonal harmonics are three orders of magnitude smaller (e.g.,  $10^{-6}$  or less):

$$J_2 = 1.082629 \times 10^{-3}. \quad (31)$$

The acceleration acting on the satellite due to  $J_2$  effects used in this study is provided with respect to the inertial reference frame in Cartesian coordinates [Schaub and Junkins (2003)]

$$a_{J_2} = -\frac{3}{2} \frac{\mu J_2 R_e^2}{r^4} \begin{bmatrix} \left(1 - 5 \frac{z^2}{r^2}\right) \frac{x}{r} \\ \left(1 - 5 \frac{z^2}{r^2}\right) \frac{y}{r} \\ \left(3 - 5 \frac{z^2}{r^2}\right) \frac{z}{r} \end{bmatrix}. \quad (32)$$

## 2.5 Lambert Two-Body Problem

Lambert's problem involves the solution of the orbit between two position vectors and the time of flight required to propagate a satellite from one position to another. The problem can be formulated in the following way: Given  $\vec{r}_1$ ,  $\vec{r}_2$ ,  $t_1$ , and  $t_2$ , find  $\vec{v}(t_1)$  and  $\vec{v}(t_2)$ . Note that  $\vec{v}(t_1)$  is the required velocity for the satellite at  $\vec{r}_1$  to intercept  $\vec{r}_2$  at  $t_2$ . The velocity required to rendezvous with  $\vec{r}_2$  is described as  $\vec{v}_2$  and the  $\Delta\vec{V}$  required is shown as  $\Delta\vec{V}_1 = \vec{v}(t_1) - \vec{v}_1$  and  $\Delta\vec{V}_2 = \vec{v}_2 - \vec{v}(t_2)$ , respectively.

See Fig. 1 for an illustration.  $\vec{v}_1$  is the velocity of the satellite at position  $\vec{r}_1$  before the maneuver and  $\vec{v}_2$  is the velocity of the satellite at position  $\vec{r}_2$  after the maneuver.

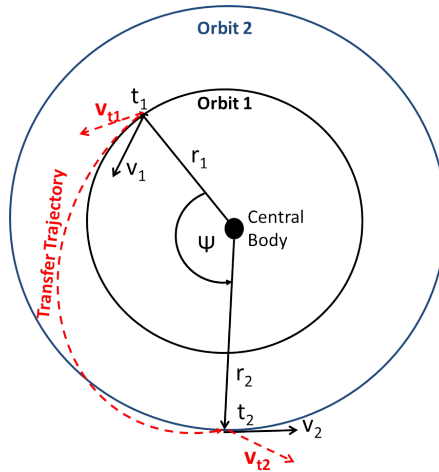


Figure 1: Illustration of a Two-Impulse Orbit Transfer Using a Two-Body Lambert Solver

There are many different methods to solve a Lambert Problem: Gauss' Method [Gauss (1857)], Thorne's Solution [Thorne (1990)], Battin's Solution [Battin (1999)], Godal's Method [Godal (1961)], Lancaster's Method [Lancaster and Blanchard (1969)], and Sun's Method [Sun (1979)]. The Superior Lambert Solver [Der (2011)] is used in this analysis to obtain the initial thrust vector direction and magnitude.

## 2.6 Modified Picard–Chebyshev Method

The vectorized matrix formulation of the Picard–Chebyshev Method is a modern concept. It was originally conceived in Feagin and Nacozy (1983); and later it was adapted by Fukushima to run on a vector computer to solve astrodynamical IVPs and BVPs of first order ODEs [Fukushima (1997); Fukushima (1999)]. Since his findings, it has been evaluated as a viable candidate for parallel orbit propagation [Neta (1998)] and further enhanced through the work of [Bai (2010)]. In this section we provide a brief background on the MPCM, which has been vectorized and parallelized for IVPs on multiple processors in our previous studies [Koblick (2012); Koblick, Poole, and Shankar (2012); Koblick and Shankar (2014)].

The formulation for the Modified Picard–Chebyshev Method presented in this section is applicable to certain types of first order BVPs and IVPs, based on that of

Bai (2010), Bai and Junkins (2011a) and Bai and Junkins (2012a). The MPCM can be used to augment the Lambert solution in solving a two-impulse transfer under additional perturbations, as well as to replace the numerical ODE solver which is needed for propagating the solution from one node to the next.

Consider a system of coupled nonlinear first order differential equations which may represent either an IVP or a BVP:

$$\frac{d\mathbf{x}}{dt} = f(t, \mathbf{x}), \quad t \in [t_0, t_f]. \quad (33)$$

It is desired to obtain a solution  $\mathbf{x}(t)$  to Eq. 33. Let  $N$  be the number of data points within the interval  $[t_0, t_f]$  at which the solution is calculated. The independent variable, time  $t$ , can be transformed to another domain,  $\tau$ , where  $\tau \in [-1, 1]$  via

$$\tau_j = -\cos \frac{j\pi}{N}, \quad j = 0, 1, \dots, N. \quad (34)$$

The simple relationship between the two domains is  $t_j = \omega_2 \tau_j + \omega_1$ , where  $\omega_2 = \frac{1}{2}(t_f - t_0)$  and  $\omega_1 = \frac{1}{2}(t_f + t_0)$ . After substitution, Eq. 33 can be written as

$$\frac{d\mathbf{x}}{d\tau} = \omega_2 f(\omega_2 \tau + \omega_1, \mathbf{x}) = g(\tau, \mathbf{x}). \quad (35)$$

For the IVP, the initial conditions  $f(t_0, \mathbf{x})$  are related to  $g(\tau_0, \mathbf{x})$  through

$$g(\tau_0, \mathbf{x}) = \omega_2 f(t_0, \mathbf{x}). \quad (36)$$

The Picard iteration which serves to update the initial approximation can be computed through

$$\mathbf{x}^i(\tau) = \sum_{k=0}^N \beta_k^i T_k(\tau), \quad (37)$$

where  $T_k(\tau) = \cos(k \cos^{-1}(\tau))$  is the Chebyshev polynomial of degree  $k$ , and  $'$  indicates that the first term in the summation is divided by two. For the IVP, the coefficients needed for the Picard iteration approximation outlined in Eq. 37 are

$$\beta_r = \frac{1}{2^r} (F_{k-1} - F_{k+1}), \quad r = 1, \dots, N-1, \quad \beta_N = \frac{F_{N-1}}{2N}, \quad (38)$$

$$\beta_0 = 2x_0 + 2 \sum_{k=1}^{k=N} (-1)^{k+1} \beta_k, \quad (39)$$

where

$$F_k = \frac{2}{N} \sum_{j=0}^{j=N} g(\tau_j, \mathbf{x}^i(\tau_j)) T_k(\tau_j). \quad (40)$$

Here " denotes that the first and last terms of the summation are divided by two. For the BVP where the initial and final positions are known, the coefficients needed to find the updated positions with respect to time can be computed from

$$\beta_r = \frac{1}{2r}(F_{k-1} - F_{k+1}), \quad r = 2, \dots, N-1, \quad \beta_N = \frac{F_{N-1}}{2N}, \quad (41)$$

$$\beta_0 = x_0 + x_f - 2 \sum_{k=1}^{k=N} \beta_{2k}, \quad \beta_1 = \frac{x_f - x_0}{2} - \sum_{k=1}^{k=N} \beta_{2k+1}. \quad (42)$$

To compute the updated velocity components (the first derivative of position with respect to time), different coefficients are needed to approximate the Picard iteration:

$$\dot{x}^j(\tau) = \sum'_{k=0}^{k=N} \alpha_k^j T_k(\tau), \quad (43)$$

where

$$\alpha_r = \frac{1}{2r}(F_{k-1} - F_{k+1}), \quad r = 1, 2, \dots, N-1, \quad \alpha_N = \frac{F_{N-1}}{2N}, \quad (44)$$

$$\alpha_0 = \frac{2}{\omega_1 + \omega_2} \sum_{k=1}^{k=N} \beta_k k^2 (-1)^{(k+1)} + 2 \sum_{k=1}^{k=N} (-1)^{k+1} \alpha_k. \quad (45)$$

The Picard iteration is performed for  $j$  iterations until the relative difference between solutions is less than some desired error tolerance  $\varepsilon$ , i.e.,

$$|x^j(\tau) - x^{j-1}(\tau)| \leq \varepsilon. \quad (46)$$

### 3 Optimal Multi-Impulse Constant Low Thrust Orbit Transfer Method

Like any unperturbed trajectory, a continuous low thrust trajectory can be approximated by discretization. For higher continuous acceleration levels, the discretization can be chosen with respect to the true anomaly while for lower continuous acceleration levels, the discretization can be chosen with respect to time. The multi-impulse discretization approach can be simply described as approximating a continuous thrust trajectory with multiple impulses. Similar techniques have been described and suggested by Xiao-yong, Yi-jun, Hong-bo, and Guo-jian (2013) and Polsgrove, Kos, Hopkins, and Crane (2006). The novelty in this proposed method is to use the Modified Picard–Chebyshev Method not only as a BVP solver, but also an IVP solver within the nonlinear constraint function to support multiple gravitational bodies and complicated numerical force models. The steps necessary in the optimization of a continuous low thrust problem when using the multi-impulse method are described below:

- **Step One: Low-Thrust Orbit Transfer Approximation** Edelbaum's solution for Non-Coplanar Circular Orbit Transfers is used as a good initial approximation for solving continuous low thrust trajectories. It is important to note that there are limitations on the magnitude of the acceleration since the analytic solution starts to break down as the orbit transfer becomes more and more elliptic. In certain cases better initial solutions available through indirect methods such as Kechichian's Euler–Lagrange Multipliers are used.
- **Step Two: Discretization of Orbit Transfer Approximation** The orbit transfer trajectory developed in step one is discretized into  $N$  nodes. Either a two-body boundary value problem solver, such as Lambert's solution, or another numerical boundary value problem solver such as the Modified Picard–Chebyshev Method is used to compute the instantaneous impulse required to travel from point  $r_{N-1}$  to point  $r_N$ . If the MPCM is used, the Lambert solver is used as an initial guess for the MPCM.
- **Step Three: Nonlinear Optimization** Given the desired constraints developed in the Constraints function, a nonlinear optimization routine, such as SNOPT [Gill, Murray, and Saunders (2002)], fmincon [Waltz, Morales, Nocedal, and Orban (2006)], or IPOPT [Wächter and Biegler (2006)], is used to determine the  $\Delta V$  necessary for each impulse, as well as the time of flight in order to minimize the objective function (which can be either time of flight or total  $\Delta V$ ). A numerical solver (MPCM, Adams–Bashforth–Moulton or Gauss-Jackson 8<sup>th</sup> order) is utilized to propagate the orbit between waypoints within the optimization process.
- **Step Four: Solution Refinement** Once a coarse solution is obtained from step three, this sequence can be repeated using the coarse solution as the initial guess.

Since this approach takes a continuous trajectory and discretizes it into multiple segments, it creates a perfect application for the Modified Picard–Chebyshev Method (the MPCM performs best on partial orbital segments as opposed to many revolutions [Koblick and Shankar (2014)]). The primary result was that the run-time performance of SNOPT could be significantly improved when using the MPCM for numerical propagation as compared to other solvers.

### 3.1 Initial Set-up

The time of flight,  $t_f$ , is discretized into  $N$  intervals. The impulse vector for each interval  $\Delta \mathbf{V}_i$  is represented as  $\Delta \mathbf{V}_i = [\eta_i, \phi_i, \delta_i]$ , where  $\phi$  and  $\delta$  are spherical azimuth and elevation angles with respect to the ECI coordinate frame and  $\Delta V_i = \eta_i t_f \bar{a}_i / N$ .

$\eta_i$  is a multiplier which should approach one when the acceleration is evenly distributed around the orbit transfer. The optimization variables are

$$\mathbf{Z} = [\Delta\mathbf{V}_1, \Delta\mathbf{V}_2, \Delta\mathbf{V}_3, \dots, \Delta\mathbf{V}_N, t_f]^T. \quad (47)$$

Using a spherical coordinate system allows the search space to be within the range of  $0-\pi$  radians in elevation and  $0-2\pi$  radians in azimuth. Upper and lower bounds on  $\eta_i$  can be provided to remove very large impulses out of the feasible region.

### 3.2 Objective Function

This method can be utilized to minimize either the total  $\Delta V$  or the time of flight  $t_f$  (but not both as the maximal  $\Delta V$  corresponds to the minimal  $t_f$  for the multiple impulse approximation). The performance index for the minimum  $\Delta V$  objective is

$$J = \int_0^{t_f} |\Delta V(t)| dt \approx \sum_{i=1}^N \Delta V_i. \quad (48)$$

Subsequently, the minimum time of flight performance index is described in Eq. 17. For the optimization performed in the following examples, the minimum time of flight was chosen as the objective function with a penalty for  $\Delta V_i$  when larger than the maximum  $\Delta V_i$  for the segment, i.e., when  $\eta_i > 1$ .

### 3.3 Constraints

The constraints associated with an orbital maneuver depend on which objectives are important to the designer. The initial conditions will remain the same regardless of which optimization objectives one has in mind:

$$\begin{bmatrix} r(t_0) \\ v(t_0) \end{bmatrix} = \begin{bmatrix} r_0 \\ v_0 \end{bmatrix}. \quad (49)$$

For full orbital rendezvous optimization, it may be necessary to fix the final state vector such that the position and velocity at the end of the transfer match the target spacecraft,

$$\begin{bmatrix} r(t_f) \\ v(t_f) \end{bmatrix} = \begin{bmatrix} r_f \\ v_f \end{bmatrix}. \quad (50)$$

For orbit transfers similar to those described by Kechichian, it may be desirable to fix only five of the six orbital elements as show in Eq. 51 (this can also be written

in equinoctial elements to avoid singularities):

$$\begin{bmatrix} a(t_f) \\ e(t_f) \\ i(t_f) \\ \Omega(t_f) \\ \omega(t_f) \end{bmatrix} = \begin{bmatrix} a_f \\ e_f \\ i_f \\ \Omega_f \\ \omega_f \end{bmatrix}, \quad \begin{bmatrix} a(t_f) \\ h(t_f) \\ k(t_f) \\ p(t_f) \\ q(t_f) \end{bmatrix} = \begin{bmatrix} a_f \\ h_f \\ k_f \\ p_f \\ q_f \end{bmatrix}. \quad (51)$$

For both the minimum  $\Delta V$  and minimum  $t_f$  optimization problems, the low thrust constraint is satisfied by

$$\Delta V_i \leq \eta_i \bar{a}_t \frac{t_f}{N}, \quad (52)$$

where  $\eta_i$  is the maximum ratio (typically around 1) where  $\eta_i = N\Delta V_i / \Delta V_{tot}$ .

### 3.4 Optimization Routine

A 64-bit MATLAB executable of SNOPT, was used for the minimum time of flight optimization shown in the results section. While both SNOPT and fmincon were used to solve the same constrained nonlinear optimization problems, SNOPT showed better convergence characteristics for this specific implementation.

## 4 Results

The optimal method developed in this paper is used to perform low thrust orbit maneuvers for two cases:

**Case 1** Minimum Time Non-Coplanar Circular Transfer

**Case 2** Minimum Time LEO to GEO Transfer

### 4.1 Non-Coplanar Circular Transfer

For this example, the constant low thrust transfer of a satellite from inclined circular GEO to equatorial circular GEO is optimized. Initial conditions are  $a_0 = 42000$  km,  $e_0 = 0.00$ ,  $i_0 = 10.00^\circ$ ,  $\Omega_0 = 0.00^\circ$ ,  $\omega_0 = 0.00^\circ$ ,  $M_0 = -220^\circ$ , with a constant acceleration  $\bar{a}_t = 1 \times 10^{-5}$  km/s<sup>2</sup>. The final conditions of the target orbit are  $a_f = 42000$  km,  $e_f = 0.00$ ,  $i_f = 0.00^\circ$ ;  $\Omega_f$ ,  $\omega_f$ , and  $M_f$  are determined by the initial approximation;  $t_f$  is to be minimized. Using Eq. 24, the analytical solution to this problem corresponds to a  $\Delta V$  of 0.8419 km/s and a time of flight of 84193.88 seconds. The effects on optimization when a different number of impulses are chosen with respect to each orbit revolution are shown in Table 1. It is interesting to note that there is an optimal setting to achieve high feasibility and the most realistic low

thrust segmentation ( $\eta \leq 1.000$ ); this occurs around eight impulses/revolutions. It should be noted that this paper uses the same definitions for optimality and feasibility as outlined in the SNOPT User's Guide [Gill, Murray, and Saunders (2006)].

Table 1: Number of Impulses vs Total  $\Delta V$ , Time of Flight, and Feasibility.

Impulses	Total $\Delta V$	Time of Flight	Feasibility	Optimality	Maximum $\eta_i$
4	0.93086 km/s	93085.5503 sec	1.40E-09	7.50E-10	1.0000
5	0.87948 km/s	87122.3347 sec	2.30E-04	1.50E-01	1.0474
6	0.85429 km/s	85428.8107 sec	1.20E-06	2.40E-05	1.1696
7	0.83944 km/s	83943.5045 sec	1.60E-04	3.40E-03	1.00000
<b>8</b>	<b>0.84183 km/s</b>	<b>84183.1071 sec</b>	<b>2.70E-07</b>	<b>9.10E-07</b>	<b>1.0000</b>
9	0.86207 km/s	86207.3741 sec	2.80E-09	1.30E-06	1.0000
10	0.84288 km/s	84288.3501 sec	3.30E-05	4.20E-03	1.0000
11	0.85478 km/s	85477.876 sec	2.20E-04	8.30E-03	1.0000
12	0.89109 km/s	89108.6533 sec	3.20E-07	1.70E-06	1.0634

A top view of the noncoplanar orbit transfer is shown in Fig. 2; Spline interpolated thrust vectors are overlaid in Fig. 3.

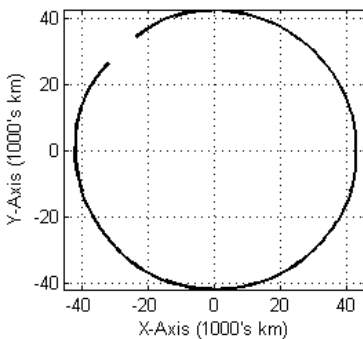


Figure 2: Birds Eye View of Transfer

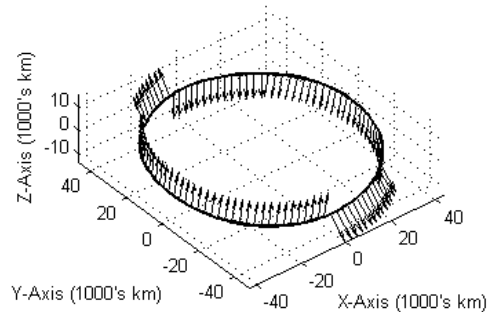


Figure 3: Acceleration Vectors

#### 4.1.1 Comparison to Numerical Solution Based on Edelbaum's Method

The solution obtained by the multi-impulse optimization method is compared to an implementation of Edelbaum's method and a trial copy of GPOPS-II MATLAB software. GPOPS-II is general-purpose software designed to solve multiple-phase optimal control problems using variable-order Gaussian quadrature collocation methods [Patterson and Rao (2014)]. The low-thrust orbit transfer example that



was provided with a copy of GPOPS-II (developed from a published optimization example [Betts (2001)]) was modified to minimize time of flight instead of fuel. Table 2 shows the results for the Edelbaum Transfer, the Multi-Impulse Transfer, and the GPOPS-II transfer case. In this case, the optimization routine minimizes

Table 2: Final Orbital Elements and Flight Times for GEO-GEO Transfer.

Final Conditions	Edelbaum	Multi-Impulse Transfer	GPOPS-II
$a_f$	42014.065	42000.000	42000.000
$e_f$	0.0260	0.00	0.000
$i_f$	0.667°	0.000°	0.000°
$\Omega_f$	0.000°	0.000°	0.000°
$\omega_f$	0.000°	0.000°	0.000°
$M_f$	124.390°	124.390°	356.036°
$\Delta V_{tot}$	841.94 m/s	841.83 m/s	848.25 m/s
$t_f$	84193.883 sec	84183.1071 sec	84824.731 sec

the total time of flight required for the transfer instead of the  $\Delta V$ . Theoretically, the minimum transfer time should correspond to the minimum  $\Delta V$  for the low continuous thrust case. Multiplying the total transfer time by the acceleration should produce the  $\Delta V$  associated with the transfer. As observed from the table, the solution will approach that of the constant acceleration case as the total number of impulses increase.

#### 4.1.2 Numerical Solution Using Additional Perturbations

For the  $J_2$  perturbation force model as described in Eq. 32, the Multi-Impulse Transfer solution is obtained by running SNOPT, shown in Table 3. Table 4 shows the run-time performance results associated with different numerical propagation techniques within the nonlinear constraint routine. It can be seen that the implementation with MPCM results in a significant improvement in performance for both the two-body and  $J_2$  perturbations as compared to the implementations with Adams-Bashforth-Moulton and Gauss-Jackson 8<sup>th</sup> order methods. In Figs. 4–6, the thrust vectors associated with the transfer for the two-body solution are shown. Note that with a minor perturbation such as  $J_2$ , these vectors may show large changes in direction, suggesting that this optimization problem is not well posed.

## 4.2 LEO to GEO Minimum Time Low Thrust Transfer

This section addresses the solution of a LEO to GEO transfer scenario based on Kechichian [1997a]. The initial conditions for this scenario are  $a_0 = 7000$  km,

Table 3: Final Orbital Elements and Flight Times for GEO-GEO Transfer with  $J_2$  Effects.

Final Conditions	Two-Body	Multi-Impulse ( $J_2$ )
$a_f$	42000.000 km	42000.000 km
$e_f$	0.000	0.000
$i_f$	0.000°	0.000°
$\Omega_f$	0.000°	0.000°
$\omega_f$	0.000°	0.000°
$M_f$	124.390°	124.390°
$\Delta V_{tot}$	841.83 m/s	840.35 m/s
$t_f$	84183.1071 sec	84035.0476 sec

Table 4: SNOPT Run Time Performance of GJ8, ODE113, and MPCM (GEO-GEO).

SNOPT Run Time Performance	GJ8	ODE113	MPCM	Analytical
Two-Body Force Model	275 sec	255 sec	53 sec	14 sec
$J_2$ Force Model	602 sec	246 sec	65 sec	N/A

$e_0 = 0.00$ ,  $i_0 = 28.5^\circ$ ,  $\Omega_0 = 0.00^\circ$ ,  $\omega_0 = 0.00^\circ$ ,  $M_0 = -220^\circ$ , with a constant acceleration  $\bar{a}_t = 9.8 \times 10^{-5}$  km/s<sup>2</sup>. And the final conditions of the target GEO are  $a_f = 42000$  km,  $e_f = 0.001$ ,  $i_f = 1.00^\circ$ ,  $\Omega_f = 0.00^\circ$ ,  $\omega_f = 0.00^\circ$ ,  $M_f = -43.779715^\circ$ , and  $t_f = 58624.094$  sec. This equates to  $\Delta V_{tot} = \bar{a}_t t_f = 5.745161$  km/s. Note that Kechichian allowed the final mean anomaly to be free in order to optimize where in the GEO orbit the satellite may enter. In this paper, it was chosen to fix this entry point at his optimal two-body solution of  $43.7797^\circ$ . This allows for a full rendezvous from one fixed inertial position to another while minimizing time of flight and subsequently  $\Delta V$ , assuming that the final implementation of thrust will be constant and continuous. A comparison of convergence performance versus total number of impulses is provided in Table 5. While fewer impulses may appear

Table 5: Number of Impulses vs Total  $\Delta V$ , Time of Flight, and Feasibility.

Impulses	Total $\Delta V$	Time of Flight	Feasibility	Optimality	Maximum $\eta_i$
24	5.7445 km/s	58617.85 sec	2.20E-11	3.00E-07	1.0000
25	5.7431 km/s	58603.21 sec	4.60E-10	1.60E-06	1.0000
30	5.7346 km/s	58516.19 sec	1.70E-09	1.20E-06	1.0000
34	5.7353 km/s	58523.61 sec	8.00E-10	1.30E-06	1.0000
38	5.7369 km/s	58540.17 sec	1.70E-09	8.70E-07	1.0000

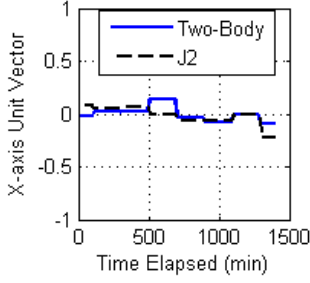


Figure 4: X-axis

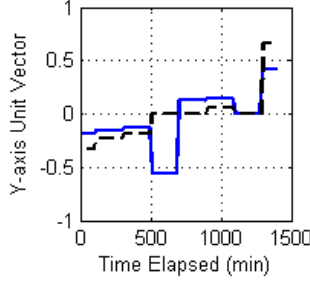


Figure 5: Y-axis

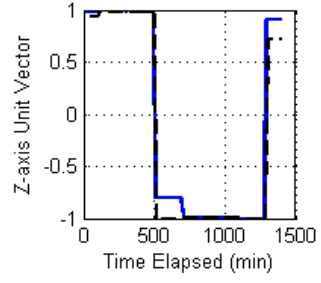


Figure 6: Z-axis

to be attractive for this optimization procedure, the resulting trajectory is highly eccentric and is not easily converted to a continuous thrust solution. 30 impulses were chosen for the optimization process as it meets the required feasibility and accuracy.

#### 4.2.1 Comparison to Kechichian Solution

With the Lagrange multipliers specified by Kechichian as input into the low thrust orbit propagation IVP (see Table 6), this trajectory was discretized into 30 nodes which were then optimized through SNOPT. The  $\Delta V$  was computed as 5.743 km/s. The minimized flight time was reported as 58624.093 seconds. Table 7 compares the final position of the satellite after the low thrust transfer. Fig. 7 depicts a three dimensional view of the full orbit transfer.

Table 6: Initial Costate Values Corresponding to the LEO-GEO Transfer Problem Outlined by Kechichian [1997a] and reprinted in Chobotov [1996].

Costate	Initial Guess
$(\lambda_a)_0$	1.260484756 s/km
$(\lambda_h)_0$	$3.865626962 \times 10^2$ s
$(\lambda_k)_0$	$-9.388262635 \times 10^3$ s
$(\lambda_p)_0$	$-2.277132367 \times 10^3$ s
$(\lambda_q)_0$	$-1.743027218 \times 10^4$ s
$(\lambda_\lambda)_0$	$5.155487187 \times 10^2$ s/rad

Table 7: Comparison of Final Orbital Elements.

Orbital Element	Kechichian	Multi-Impulse	GPOPS-II
$a_f$	42,000.007 km	42000.000 km	42000.000 km
$e_f$	$1.00022 \times 10^{-3}$	$1.000000 \times 10^{-3}$	$1.000000 \times 10^{-3}$
$i_f$	$1.000012^\circ$	$1.00000^\circ$	$1.00000^\circ$
$\Omega_f$	$359.999963^\circ$	$359.999994^\circ$	$359.999863^\circ$
$\omega_f$	$1.966524^\circ \times 10^{-2}$	$1.67^\circ \times 10^{-4}$	$4.445109^\circ$
$M_f$	$43.779715^\circ$	$43.858910^\circ$	$39.5072109^\circ$
$\Delta V_{tot}$	5.75 km/s	5.74 km/s	5.75 km/s
$t_f$	58624.09 sec	58624.09 sec	58622.90 sec

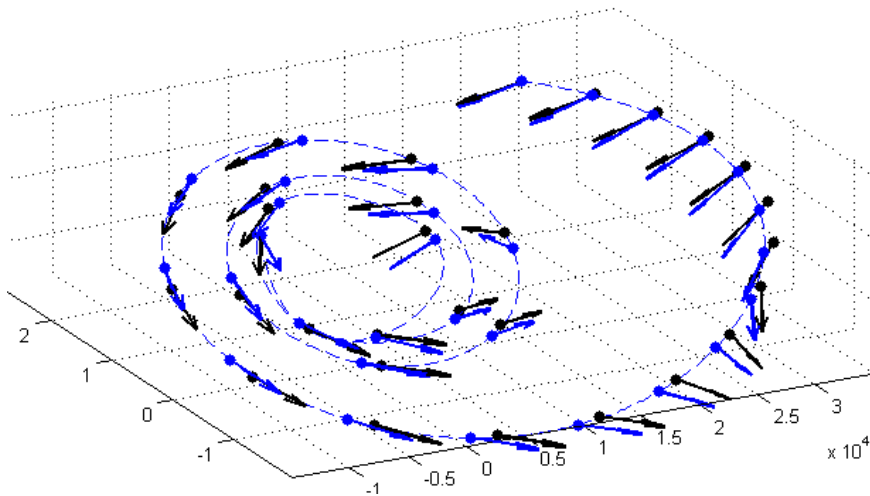


Figure 7: 30-Node ECI Thrust Vector Comparison Multiple Impulse vs Costates (Blue: Multi-Impulse Method; Black: Costate Method)

The unit vectors of the acceleration in the ECI reference frame are compared at each node along the transfer trajectory in Figs. 8–10. Note that the black dot represents the solution found from optimizing the costates while the blue line is the thrust vector at each discrete point as found from running the multiple impulse method.

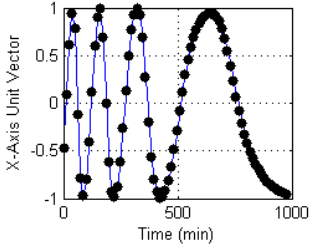


Figure 8: x-axis

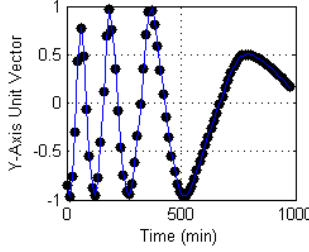


Figure 9: y-axis

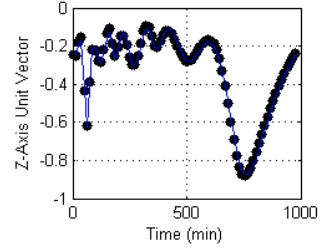


Figure 10: z-axis

4.2.2 Numerical Solution Using Additional Perturbations

For the  $J_2$  perturbation force model as described in Eq. 32, Table 8 shows the Multi-Impulse Transfer solution obtained by running SNOPT.

Table 8: Final Orbital Elements and Flight Times for LEO-GEO Transfer with  $J_2$  Effects.

Final Conditions	Two-Body	Multi-Impulse ( $J_2$ )
$a_f$	42000.000 km	42000.00 km
$e_f$	0.001	0.001
$i_f$	1.000°	1.000°
$\Omega_f$	0.000°	0.000°
$\omega_f$	0.000°	0.000°
$M_f$	43.859°	43.859°
$\Delta V_{tot}$	5.74 km/s	5.73 km/s
$t_f$	58624.093 sec	58519.690 sec

Table 9: SNOPT Run Time Performance of GJ8, ODE113, and MPCM (LEO-GEO).

SNOPT Run Time Performance	GJ8	ODE113	MPCM	Analytical
Two-Body Force Model	5460 sec	3633 sec	777 sec	557 sec
$J_2$ Force Model	10214 sec	6986 sec	1677 sec	N/A

Table 9 shows the comparison of the run times of the multi-impulse low thrust optimizations with implementation of MPCM, ODE113 and GJ8. The results show a significant improvement in the performance of the MPCM as compared to the other two solvers.

## 5 Discussion

This study presents the computational run-time performance gain of using the MPCM to solve discretized constant low thrust orbit transfer optimization problems in conjunction with a sparse sequential quadratic programming (SQP). Previous work in relation to the MPCM has shown applications for Lambert targeting problems, two-body coplanar low thrust trajectory optimization, and Station-Keeping of Translunar Halo Orbits [Bai and Junkins (2011b); Bai and Junkins (2012b)].

In regard to performance on numerical orbit propagation problems, previous research has shown that when distributed over many GPUs, the MPCM can reduce the computational time between one and two orders of magnitude over other conventional sequential ODE solvers such as Runge–Kutta 4(5) and Runge–Kutta–Nystrom 12(10) [Bai and Junkins (2012a)]. Additionally, the work by Koblick and Shankar (2014) has shown that a parallel MPCM implementation across just five CPU cores can reduce the computational time of orbit propagation to below the performance of the fastest sequential numerical ODE solvers (e.g., Runge–Kutta 4(5), Gauss Jackson 8, and Dormand and Prince (8,7)) [Koblick and Shankar (2014)]. Fukushima, after testing a vectorized version of the Picard–Chebyshev method on a vectorized computer, theorized that the performance improvement of solving perturbed orbit propagation problems can be between two and three orders of magnitude [Fukushima (1999)]. This suggests it is likely that the run-time performance of this low thrust discretization method, when using the MPCM, can benefit significantly when switching to a parallel implementation (especially when the force model is increased in its complexity).

Direct optimization, for moderate dimensional parametric representation of the unknown control function, is widely known to efficiently lead to good sub-optimal results, and the dimensionality can be frequently increased to refine the solution without becoming numerically intractable. Excellent approximations of the optimal trajectory can be obtained by using a direct approach, and this approximate truth is confirmed by the results obtained in this paper.

If we upgrade our numerical force models to include multi-body perturbations, this will allow the multi-impulse method to be extended to interplanetary missions where, traditionally, patched conics and multi-segmented optimization would need to be performed. There are some unique characteristics to consider when using a direct optimization method which discretizes the problem in order to optimize an orbit transfer:

- *Very good initial guess of trajectory required* — because there are many segments, it may take a large number of function evaluations (if feasible) for a nonlinear solver to converge to an optimal transfer as the search space is

significantly large ( $N \times 3 + 1$ ) dimensions.

- *Segments must be a fraction of each orbit revolution* — as shown in the thrust restriction identified in Eq. 52, the fewer number of segments used, the faster the optimization (less optimization variables reduce the dimensionality of the problem), but the solution will approach a high thrust transfer and will no longer be similar to that of a low thrust trajectory.
- *Real time trajectory controller extension* — with a finite number of impulses, the solution is not directly translatable to instantaneous thrust vectors with respect to time. The trajectory must be modeled with some type of trajectory controller, such as a predictive method.

## 6 Conclusions and Future Work

This paper presents a multi-impulse low thrust trajectory optimization method that utilizes the Modified Picard–Chebyshev Method to improve the run-time performance of the nonlinear optimization routine even with the inclusion of perturbation terms in the gravitation force model. The method shows a significant improvement in the run-time performance as compared to ODE113 and GJ8. The multi-impulse method with the implementation of MPCM was also compared to well known solutions and tools indicating similar accuracies. Future efforts will be directed towards parallelizing the Modified Picard–Chebyshev Method when called from a nonlinear optimization solver in order to further increase the performance benefits of using a perturbative approach to solving numerical propagation problems. This will be followed by an evaluation of existing approaches to convert a multi-segmented solution into a trajectory control algorithm which may be used on-board a spacecraft once an optimal transfer has been found. This multi-impulse discretization method could also be applied to the  $N$ -body problem, specifically for interplanetary low thrust rendezvous, low thrust gravity assists (LTGA), or multiple-LTGA problems. One approach used to solve the rendezvous and LTGA problems exploits the natural dynamics of  $N$ -body models in the formation of an initial guess trajectory. LTGA consist of a set of patched controlled planar Circularly Restricted  $N$ -Body Problems (CRNBP) which analytically generate low energy initial guess formulations (e.g., invariant manifolds). This initial guess is then fed to a traditional  $N$ -body optimizer to compute a complete trajectory, inclusive of planetary inclinations and eccentricities. This approach could extend the multi-impulse discretization method by using the CRNBP as an initial approximation in place of the Edelbaum or the Euler–Lagrange indirect method. Combining systems of CRNBPs as the initial guess to the multi-impulse discretization method could yield a robust strategy for computing complex interplanetary transfer trajectories.

**Acknowledgement:** The authors would like to thank Phillip Gill from the University of California, San Diego for providing a copy of the Sparse Nonlinear OPTimizer (SNOPT) software which was used during this study. They would also like to thank Anil Rao from the University of Florida for providing a trial copy of GPOPS-II which was used for comparison results. Also, the authors thank Professor John Junkins from Texas A&M University for his review and feedback of this paper.

## References

- Bai, X.** (2010): *Modified Chebyshev-Picard Iteration Methods For Solution Of Initial Value And Boundary Value Problems*. PhD thesis, Texas A&M University, 2010.
- Bai, X.; Junkins, J.** (2011a): Modified chebyshev-picard iteration methods for orbit propagation. *The Journal of the Astronautical Sciences*, vol. 58, no. 4, pp. 583–613.
- Bai, X.; Junkins, J.** (2011b): Modified chebyshev-picard iteration methods for solution of boundary value problems. *Journal of the Astronautical Sciences*, vol. 58, no. 4, pp. 615–642.
- Bai, X.; Junkins, J.** (2012a): Modified chebyshev-picard iteration methods for solution of initial value problems. *Journal of the Astronautical Sciences*, vol. 59, no. 1, pp. 327–351.
- Bai, X.; Junkins, J.** (2012b): Modified chebyshev-picard iteration methods for station-keeping of translunar halo orbits. *Mathematical Problems in Engineering*, vol. 2012, pp. 1–18.
- Battin, R.** (1999): *An Introduction To The Mathematics And Methods Of Astrodynamics*. AIAA Education Series.
- Berry, M.; Healy, L.** (2004): Implementation of Gauss-Jackson integration for orbit propagation. *Journal of the Astronautical Sciences*, vol. 52, pp. 331–357.
- Betts, J.** (2000): Very low-thrust trajectory optimization using a direct sqp method. *Journal of Computational and Applied Mathematics*, vol. 120, no. 1, pp. 27–40.
- Betts, J.** (2001): *Practical Methods For Optimal Control Using Nonlinear Programming*. SIAM, first edition.
- Bradley, B.** (2012): Numerical integration in astrodynamics (initial value problem). [http://ccar.colorado.edu/asen5519/imd/documents/Integration\\_Lecture\\_Slides.pdf](http://ccar.colorado.edu/asen5519/imd/documents/Integration_Lecture_Slides.pdf), 2012.



- Broucke, R.; Cefola, P.** (1972): On the equinoctial orbit elements. *Celestial Mechanics*, vol. 5, no. 3, pp. 303–310.
- Bryson, A.; Ho, Y.-C.** (1975): *Applied Optimal Control: Optimization, Estimation and Control*. CRC Press, first edition.
- Chobotov, V.** (1996): *Orbital Mechanics*. AIAA Education Series, second edition.
- Der, G. J.** (2011): The superior lambert algorithm. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, HI.
- Eagle, D.** (2013): Low thrust transfer between non-coplanar circular orbits. Online, 2013.
- Edelbaum, T.** (1961): Propulsion requirements for controllable satellites. *American Rocket Society Journal*, vol. 31, pp. 1079–1089.
- Falck, R.; Dankanich, J.** (2012): Optimization of low-thrust spiral trajectories by collocation. In *Astrodynamics Specialist Conference*, no. 2012-4423. AIAA/AAS.
- Feagin, T.; Nacozy, P.** (1983): Matrix formulation of the picard method for parallel computation. *Celestial mechanics*, vol. 29, no. 2, pp. 107–115.
- Fukushima, T.** (1997): Vector integration of dynamical motions by the picard-chebyshev method. *The Astronomical Journal*, vol. 113, no. 6, pp. 2325–2328.
- Fukushima, T.** (1999): Parallel/vector integration methods for dynamical astronomy. *Celestial Mechanics and Dynamical Astronomy*, vol. 73, no. 1, pp. 231–241.
- Gauss, K.** (1857): *Theoria Motus (in Latin) Theory of Motion of the Heavenly Bodies about Sun in a Conic Section*. Little, Brown and Company.
- Geering, H.** (2007): *Optimal Control With Engineering Applications*. Springer, first edition.
- Gill, P.; Murray, W.; Saunders, M.** (2002): Snopt: An sqp algorithm for large-scale constrained optimization. *Journal on Optimization*, vol. 12, no. 4, pp. 979–1006.
- Gill, P.; Murray, W.; Saunders, M.** (2006): Users guide for snopt version 7: Software for large-scale nonlinear programming, 2006.
- Godal, T.** (1961): Method for determining the initial velocity vector corresponding to a given time of free flight transfer between given points in a simple gravitational field. *Astronautik*, vol. 2, pp. 183–186.
- Graham, K.; Rao, A.** (2015): Minimum-time trajectory optimization of multiple revolution low-thrust earth-orbit transfers. *Journal of Spacecraft and Rockets*, vol. 52, no. 3, pp. 711–727.

**Keaton, P.** (2002): Low-thrust rocket trajectories. Technical Report LA-10625-MS, Los Alamos National Laboratory, Los Alamos, NM, 2002.

**Kechichian, J.** (1997a): Optimal low-earth-orbit-geostationary-earth-orbit intermediate acceleration orbit transfer. *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 4, pp. 803–811.

**Kechichian, J.** (1997b): Reformulation of edelbaum’s low-thrust transfer problem using optimal control theory. *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 5, pp. 988–994.

**Kechichian, J.** (1997c): Mechanics of trajectory optimization using nonsingular variational equations in polar coordinates. *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 4, pp. 812–818.

**Kechichian, J.** (2007): The streamlined and complete set of the nonsingular  $j_2$ -perturbed dynamic and adjoint equations for trajectory optimization in terms of eccentric longitude. *Journal of the Astronautical Sciences*, vol. 55, no. 3, pp. 325–348.

**Kim, M.** (2005): *Continuous Low-Thrust Trajectory Optimization: Techniques and Application*. PhD thesis, Virginia Polytechnic Institute and State University, 2005.

**Koblick, D.** (2012): Parallel high-precision orbit propagation using the modified picard-chebyshev method. Master’s thesis, California State University, Long Beach, 2012.

**Koblick, D.; Poole, M.; Shankar, P.** (2012): Parallel high-precision orbit propagation using the modified picard-chebyshev method. In *International Mechanical Engineering Congress and Exposition*, pp. 587–605, Houston, TX. ASME.

**Koblick, D.; Shankar, P.** (2014): Evaluation of the modified picard-chebyshev method for high-precision orbit propagation. *Journal of Aerospace Engineering*, vol. 28, no. 5.

**Lancaster, E.; Blanchard, R.** (1969): A unified form of lambert’s theorem. Technical Report TN D-5368, NASA, 1969.

**Larson, W.; Wertz, J.** (1999): *Space Mission Analysis And Design*. Space Technology Library. Microcosm, third edition.

**Lewis, F.; Vrabie, D.; Syrmos, V.** (2012): *Optimal Control*. Wiley, Hoboken, NJ, third edition.

**Neta, B.** (1998): Parallel version of special perturbations orbit propagator. Technical report, Naval Postgraduate School, Monterey, CA, 1998.

- Patterson, M.; Rao, A.** (2014): Gpops-ii: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. *Transactions on Mathematical Software (TOMS)*, vol. 41, no. 1, pp. 1–39.
- Polsgrove, T.; Kos, L.; Hopkins, R.; Crane, T.** (2006): Comparison of performance predictions for new low-thrust trajectory tools. In *Astrodynamics Specialist Conference and Exhibit*, Keystone, CO. AIAA/AAS.
- Schaub, H.; Junkins, J.** (2003): *Analytical Mechanics of Space Systems*. AIAA Education Series, second edition.
- Sun, F.-T.** (1979): On the minimum time trajectory and multiple solutions of lambert’s problem. In *American Institute of Aeronautics and Astronautics Conference*, Provincetown, MA. AIAA.
- Thorne, J.** (1990): Series reversion/inversion of lambert’s time function. In *Astrodynamics Conference*. AIAA.
- Ulybyshev, Y.** (2009): Spacecraft trajectory optimization based on discrete sets of pseudoimpulses. *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 4, pp. 1209–1217.
- Vallado, D.; McClain, W.** (2001): *Fundamentals of Astrodynamics and Applications*. Space Technology Library. Springer, second edition.
- Von Stryk, O.; Bulirsch, R.** (1992): Direct and indirect methods for trajectory optimization. *Annals of operations research*, vol. 37, no. 1, pp. 357–373.
- Wächter, A.; Biegler, L.** (2006): On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, vol. 106, no. 1, pp. 25–57.
- Waltz, R.; Morales, J.-L.; Nocedal, J.; Orban, D.** (2006): An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Mathematical Programming*, vol. 107, no. 3, pp. 391–408.
- Xiao-yong, J.; Yi-jun, L.; Hong-bo, Z.; Guo-jian, T.** (2013): A multi-impulse extended method for low-thrust trajectory optimization. In *Recent Advances in Space Technologies (RAST), 2013 6th International Conference on*, pp. 365–368. IEEE.
- Yang, G.** (2009): Direct optimization of low-thrust many-revolution earth-orbit transfers. *Chinese Journal of Aeronautics*, vol. 22, no. 4, pp. 426–433.

