

Multiple Revolution Solutions for the Perturbed Lambert Problem using the Method of Particular Solutions and Picard Iteration

Robyn M. Woollands¹ · Julie L. Read¹ ·
Austin B. Probe¹ · John L. Junkins¹

Published online: 10 July 2017

© The Author(s) 2017. This article is an open access publication

Abstract We present a new method for solving the multiple revolution perturbed Lambert problem using the *method of particular solutions* and *modified Chebyshev-Picard iteration*. The *method of particular solutions* differs from the well-known Newton-shooting method in that integration of the state transition matrix (36 additional differential equations) is not required, and instead it makes use of a reference trajectory and a set of n particular solutions. Any numerical integrator can be used for solving two-point boundary problems with the *method of particular solutions*, however we show that using *modified Chebyshev-Picard iteration* affords an avenue for increased efficiency that is not available with other step-by-step integrators. We take advantage of the path approximation nature of modified Chebyshev-Picard iteration (nodes iteratively converge to fixed points in space) and utilize a variable fidelity force model for propagating the reference trajectory. Remarkably, we demonstrate that computing the particular solutions with only low fidelity function evaluations greatly increases the efficiency of the algorithm while maintaining machine precision accuracy. Our study reveals that solving the perturbed Lambert's problem using the *method of particular solutions* with *modified Chebyshev-Picard iteration* is about an order of magnitude faster compared with the classical shooting method and a tenth-twelfth order Runge-Kutta integrator. It is well known that the solution to Lambert's problem over multiple revolutions is not unique and to ensure that all possible solutions are considered we make use of a reliable preexisting Keplerian Lambert solver to *warm start* our perturbed algorithm.

✉ Robyn M. Woollands
robyn.woollands@gmail.com

¹ Department of Aerospace Engineering, Texas A&M University, TAMU 3141, College Station, TX 77843-3141, USA

Keywords Lambert’s problem · Perturbed Lambert’s problem · Method of particular solutions · Picard-Chebyshev

Introduction

Consider a natural second order differential equation of the form:

$$\ddot{\mathbf{r}} = \mathbf{g}(t, \mathbf{r}, \dot{\mathbf{r}}) \quad (1)$$

where \mathbf{r} is an n -vector of position coordinates, $\dot{\mathbf{r}}$ is an n -vector of velocity coordinates. The essence of Lambert’s problem is to find the initial velocity ($\dot{\mathbf{r}}_0(t_0) = \mathbf{v}_0$) such that solving the initial value problem with the initial boundary conditions ($\mathbf{r}_0, \mathbf{v}_0$ at t_0) gives a trajectory that reaches the final position (\mathbf{r}_f) at the given final time (t_f). In the case of Keplerian motion, the solution for the fraction-of-an-orbit case is known to be unique except for the cases that the initial and final vectors ($\mathbf{r}_0, \mathbf{r}_f$) are co-linear [1]. For multi-orbit transfers, the solution is not unique; however, all solutions can be reliably computed except near the co-linear special cases. For the specific case of analytically tractable Keplerian motion, solving the TPBVP can be reduced to finding the roots of a scalar nonlinear algebraic equation, and has been the subject of almost two centuries of refinements culminating in the elegant solution of R. H. Battin [1]. Ochoa and Prussing [2] further refined this work to produce a nice algorithm that finds the multiple solutions to the Keplerian problem. We mention this algorithm in more detail later in the paper.

The classical solution to the generalized (perturbed) Lambert’s problem, for a general right hand side of Eq. 1, involves a Newton iteration method whereby we require the $n \times n$ partial derivatives of the terminal position with respect to the initial velocity [$\partial \mathbf{r}(t_f) / \partial \mathbf{v}(t_0)$] to satisfy the $2n \times 2n$ differential equations (2). [$\partial \mathbf{r}(t_f) / \partial \mathbf{v}(t_0)$] is the final time computation of the upper right $n \times n$ sub-matrix of the state transition matrix $\Phi(t, t_0)$.

$$\dot{\Phi}(t, t_0) = \begin{bmatrix} 0 & I \\ \partial \mathbf{g} / \partial \mathbf{r} & \partial \mathbf{g} / \partial \dot{\mathbf{r}} \end{bmatrix} \Phi(t, t_0), \quad \Phi(t_0, t_0) = I. \quad (2)$$

To implement the classical shooting method these $4n^2$ differential equations (2), coupled to the $2n$ equations of Eq. 1, results in $4n^2 + 2n$ differential equations that must be solved iteratively with the initial velocity updated following each iteration using a multi-dimensional Newton method (shooting method):

$$\dot{\mathbf{r}}_{k+1}(t_0) = \dot{\mathbf{r}}_k(t_0) + \Phi_{21k}^{-1}(t_f, t_0) \{ \mathbf{r}_k(t_f) - \mathbf{r}_f \}; \quad k = 1, 2, \dots \quad (3)$$

The k subscript denotes the solution of Eqs. 1 and 2 with the k^{th} iterative trial value for the unknown initial velocity vector $\dot{\mathbf{r}}_0(t_0) = \mathbf{v}_0$. For sufficiently close starting iteration, one can expect convergence in a single digit number of iterations. However, obtaining a sufficiently close starting estimate is not always easy and that is part of the challenge for each physical problem. This is especially acute for the case of multi-revolution orbit transfers where there are more than one feasible transfer orbit. In many cases, approximate versions of the governing differential equations, for

example Battin's [1] analytical two-body solution of Lambert's problem, are used to initiate iterations for the fully perturbed system. Battin's formulation generates all solutions, including the multiple revolution local roots. Utilizing the most attractive of the Keplerian starting iteratives is virtually always good enough for weakly perturbed orbit mechanics problems. When the perturbations are too large for convergence with the analytical two-body Lambert starting approximation, a homotopic method can frequently be used to sweep a parameter that gradually departs from a known solution to solve the fully perturbed problem at hand. However, in the presence of a general right hand side of Eq. 1, and a poor starting estimate, the problem is more challenging, and each iteration has to pay the overhead associated with computing the partial derivatives by solving Eq. 2.

Two alternative methods for solving the perturbed Lambert's problem are the TPBVP implementation of modified Chebyshev-Picard iteration (MCPI) [3, 4], and the MCPI-TPBVP algorithm regularized using the Kustaanheimo-Stiefel time transformation [5, 6]. The first method converges over a small fraction of an orbit (about a third of an orbit depending on eccentricity) and the second method converges up to about 90% of an orbit. In this paper we present a third method for solving the perturbed Lambert's problem; a local linearity-based approach that converges over multiple revolutions. It is attractive as there is no need to compute a state transition matrix and thereby we avoid solving the $n \times n$ differential equations (2). This alternate shooting technique is known as the *method of particular solutions* (MPS), as developed in by Miele [7]. We combine MPS with the MCPI initial value problem (MCPI-IVP) integration method to solve the perturbed two-body orbital equations of motion accurately and efficiently over multiple revolutions.

The motivation for this work is to respond to the orbital debris problem. There are over 500 USA-launched spent rocket boosters in low Earth orbit, and these are hazardous to operational satellites. Reducing the probability of collision is possible by orbit debris rendezvous, capture and de-orbit missions directed at the most high priority objects. The spent rocket boosters are attractive objects to remove because they all represent potential collisions and removing large objects has been shown to be one of the best ways to mitigate debris growth. Simulating hundreds of thousands of transfer orbits (Lambert problems) accurately and efficiently is essential for aiding in the solution to this debris removal problem. The algorithms we have developed using the methods described above can be used for designing and simulating missions to retrieve these objects, thus reducing the predicted future growth of orbital debris.

Modified Chebyshev-Picard Iteration

Modified Chebyshev Picard Iteration differs from the well-known step integrators such as Gauss-Jackson, Runge-Kutta-Nystrom and ODE45 in that it is a *path approximation* numerical integrator rather than a *step-by-step* integrator. Long state trajectory arcs are approximated continuously and updated at all time instances on each iteration.

The MCPI technique combines Picard iteration with the orthogonal Chebyshev polynomials. Emile Picard observed that any first order differential equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(t, \mathbf{x}(t)), \quad t \in [t_0, t_f], \quad (4)$$

with an initial condition $\mathbf{x}(t_0) = \mathbf{x}_0$ and any integrable right hand side may be rearranged, without approximation, to obtain the following integral equation:

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^t \mathbf{f}(\tau, \mathbf{x}(\tau)) d\tau. \quad (5)$$

A sequence of approximate solutions $\mathbf{x}^i(t)$, ($i = 1, 2, 3, \dots, \infty$), of the true solution $\mathbf{x}(t)$ that satisfies this differential equation may be obtained through Picard iteration using the following Picard sequence of approximate paths $\{\mathbf{x}^0(t), \mathbf{x}^1(t), \dots, \mathbf{x}^{i-1}(t), \mathbf{x}^i(t), \dots\}$:

$$\mathbf{x}^i(t) = \mathbf{x}(t_0) + \int_{t_0}^t \mathbf{f}(\tau, \mathbf{x}^{i-1}(\tau)) d\tau, \quad i = 1, 2, \dots \quad (6)$$

Picard proved that for smooth, differentiable, single-valued nonlinear functions $\mathbf{f}(t, \mathbf{x}(t))$, there is a time interval $|t_f - t_0| < \delta$ and a starting trajectory $\mathbf{x}^0(t)$ satisfying $\|\mathbf{x}^0(t) - \mathbf{x}(t)\|_\infty < \infty$, that for suitable finite bounds (δ, Δ), the Picard sequence of trajectories represents a contraction operator that converges to the unique solution of the initial value problem. The rate of convergence is typically geometric. The guaranteed convergence property sets this method apart from other approaches. The numerical accuracy and efficiency are dominated by the particular process used to carry out the integral; note since the previous ($i - 1$) trajectory approximation is known, the integrand is considered a function of time only. Chebyshev polynomials are used for approximating the integrand in the Picard iteration sequence, and these orthogonal polynomials integrate to produce a Chebyshev series for the integral, including the imposition of initial (or final) boundary conditions.

The original fusion of orthogonal approximation theory and Picard iteration was introduced by Clenshaw and Norton in 1963 [8]. Feagin published his PhD dissertation [9] in 1972 on Picard iteration using Chebyshev approximation. He established the first vector-matrix version of Picard iteration utilising orthogonal basis functions [10]. In 1980 Shaver wrote a related dissertation giving insights on parallel computation using Picard Iteration and Chebyshev approximation [11]. In 1997 Fukushima [12] addressed parallelization of Picard iteration in a particular computer architecture. His results showed that a particular parallel implementation of his algorithm did not give the theoretical speedup he anticipated.

A decade later Bai and Junkins revisited this approach and developed improved algorithms for solving IVPs and TPBVPs [3, 13]. They established new convergence insights and also developed vector-matrix formulations for solving initial and boundary value problems. These are published in Bai's PhD dissertation [4]. Bani Younes and Junkins followed this work with methods to include high order gravity perturbations in order to more accurately represent the motion of satellites orbiting in the vicinity of the Earth [14–16]. Macomber and Junkins developed enhancements that took advantage of the “fixed-point” convergence nature of MCPI and allowed

solutions to the perturbed two-body problem to be computed using variable fidelity force models and radially adaptative gravity approximations. They also made use of *warm* and *hot* starts for solving the perturbed problem [17, 18]. These enhancements resulted in substantial increases in the efficiency of MCPI while maintaining machine precision accuracy.

Method of Particular Solutions

The method of particular solutions is a local-linearity based approach that makes use of a reference trajectory $\mathbf{r}_{ref}(t), \dot{\mathbf{r}}_{ref}(t), \ddot{\mathbf{r}}_{ref}(t)$ and a set of particular solutions. All neighboring solutions of Eq. 1 (reference and particular) can be re-formulated exactly in terms of a departure motion $\Delta\mathbf{r}(t)$ as

$$\mathbf{r}(t) = \mathbf{r}_{ref}(t) + \Delta\mathbf{r}, \quad \dot{\mathbf{r}}(t) = \dot{\mathbf{r}}_{ref}(t) + \Delta\dot{\mathbf{r}}, \quad \ddot{\mathbf{r}}(t) = \ddot{\mathbf{r}}_{ref}(t) + \Delta\ddot{\mathbf{r}}. \tag{7}$$

From Eqs. 7 and 1, we can write the exact departure motion differential equation:

$$\Delta\ddot{\mathbf{r}} = \mathbf{g}(t, \mathbf{r}_{ref}(t) + \Delta\mathbf{r}, \dot{\mathbf{r}}_{ref}(t) + \Delta\dot{\mathbf{r}}) + \mathbf{u} - \ddot{\mathbf{r}}_{ref}(t). \tag{8}$$

Now consider the circumstance that $\ddot{\mathbf{r}}_{ref}(t)$ is a solution of the differential equation which satisfies “good” initial boundary conditions; in this case, the Δ ’s can be expected to be small, $\ddot{\mathbf{r}}_{ref}(t) = \mathbf{g}(t, \mathbf{r}_{ref}(t), \dot{\mathbf{r}}_{ref}(t))$ and to a linear approximation, the exact nonlinear Eq. 8 could be replaced by an approximate linear equation of the form

$$\Delta\ddot{\mathbf{r}} = A\Delta\mathbf{r} + B\Delta\dot{\mathbf{r}} + O(\Delta^2). \tag{9}$$

where A, B are time varying Jacobians of \mathbf{g} with respect to $\mathbf{r}, \dot{\mathbf{r}}$ evaluated along $\mathbf{r}_{ref}(t)$. We can consider the departure motion linear as long as it is to within the accuracy that the linear Eq. 9 approximates the exact departure motion of Eq. 8. Note, the reference trajectory is not generally held invariant, so the initial conditions can be updated to reflect improved knowledge after each successful iteration. Consider the case that the reference motion satisfies the known left position coordinates exactly $\mathbf{r}_{ref}(t_0) = \mathbf{r}_0$, and the initial velocity $\dot{\mathbf{r}}_{ref}(t_0)$ represents the current best estimate of the unknown initial velocity. For this given (or just computed) $\mathbf{r}_{ref}(t)$, consider three variant trajectories obtained by varying the initial velocity by small linearly independent (typically orthogonal) perturbations. Typically, initial perturbations occur in the last three significant digits of the motion; for example if the numerical solution is accurate to 10 digits, the norm of the typical independent initial condition perturbations could be $|\Delta\dot{\mathbf{r}}_j(t_0)| \approx 10^{-7} |\dot{\mathbf{r}}_{ref}(t_0)|$ to obtain the neighboring initial velocities:

$$\mathbf{r}_j(t_0) = \mathbf{r}_{ref}(t_0) = \mathbf{r}_0; \quad \dot{\mathbf{r}}_j(t_0) = \dot{\mathbf{r}}_{ref}(t_0) + \Delta\dot{\mathbf{r}}_j(t_0); \quad j = 1, 2, 3. \tag{10}$$

Next solve the differential Eq. 1 for each of the three particular solutions $\mathbf{r}_j(t)$. Now we can form the exact departure motions

$$\Delta\mathbf{r}_j(t) = \mathbf{r}_j(t) - \mathbf{r}_{ref}(t). \tag{11}$$

These exact departure motions are particular solutions and conjectured to approximately satisfy the linear differential equation in Eq. 9. Since independent velocity

initial conditions were used it is assumed that these trajectories span the space of interest along with all neighboring trajectories of interest that also satisfy the linear departure motion Eq. 9. The linear combination of any particular solution of a linear differential equation satisfies the differential equation as well, and the general solution as a linear combination of three (in general n) departure motions can be written as shown in Eq. 12.

$$\Delta \mathbf{r}(t) \approx \sum_{j=1}^3 \alpha_j \Delta \mathbf{r}_j(t) \Rightarrow \mathbf{r}(t) \approx \mathbf{r}_{ref}(t) + \sum_{j=1}^3 \alpha_j \Delta \mathbf{r}_j(t) \tag{12}$$

If the Δ 's exist rigorously in the linear domain, of course, Eq. 12 would hold with negligible error. Here we consider only 3 variant trajectories because the admissible initial variations are only the unknown initial velocity coordinates. That is, we constrain $\mathbf{r}_j(t_0) = \mathbf{r}_{ref}(t_0)$ and $\Delta \mathbf{r}_j(t_0) = 0$.

Figure 1 shows a conceptual example of the departure motion for the three particular solutions. Notice that $\mathbf{r}_B(t_f)$, the target position, lies at the vertex of the 3-D parallelepiped, which is a scaled (with respect to α) version of the space spanned by $\Delta \mathbf{r}_j$; the $\mathbf{r}_B(t_f)$ is approximated (to within the assumption of linearity) from the linear combination of the three $\alpha_j \Delta \mathbf{r}_j$. For the case shown, all three α_j 's are less than 1, but the general requirement is that the current miss vector $\Delta \mathbf{r}_B = \mathbf{r}_B(t_f) - \mathbf{r}_{ref}(t_f)$ lies in the region approximated by Eq. 9.

Evaluating Eq. 12 at the final time and imposing the desired result that $\mathbf{r}(t_f) = \mathbf{r}_f$, leads to the solution for the coefficients of linear combination

$$\begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{Bmatrix} \approx [\Delta \mathbf{r}_1(t_f) \ \Delta \mathbf{r}_2(t_f) \ \Delta \mathbf{r}_3(t_f)]^{-1} \{ \mathbf{r}_B(t_f) - \mathbf{r}_{ref}(t_f) \} \tag{13}$$

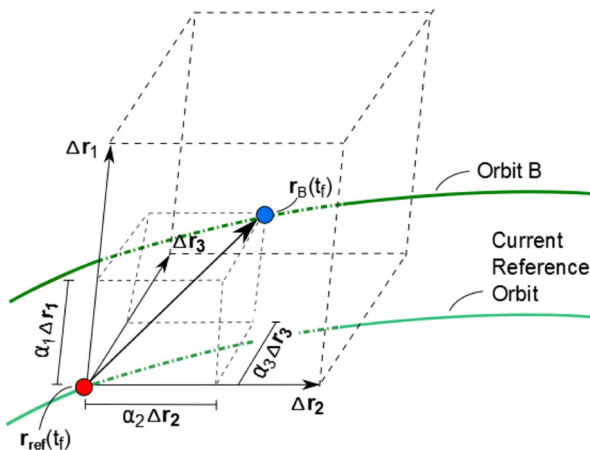


Fig. 1 Departure motion space for the method of particular solutions. $\Delta \mathbf{r}_1$, $\Delta \mathbf{r}_2$ and $\Delta \mathbf{r}_3$ are position vectors from $\mathbf{r}_{ref}(t_f)$ to $\mathbf{r}_1(t_f)$, $\mathbf{r}_2(t_f)$ and $\mathbf{r}_3(t_f)$ respectively

Given the α_i s, we can compute the departure $\Delta \mathbf{r}(t)$ and the time derivatives at any time t .

$$\Delta \mathbf{r}(t_f) = \alpha_1 \Delta \mathbf{r}_1(t_f) + \alpha_2 \Delta \mathbf{r}_2(t_f) + \alpha_3 \Delta \mathbf{r}_3(t_f), \tag{14}$$

$$\Delta \dot{\mathbf{r}}(t_f) = \alpha_1 \Delta \dot{\mathbf{r}}_1(t_f) + \alpha_2 \Delta \dot{\mathbf{r}}_2(t_f) + \alpha_3 \Delta \dot{\mathbf{r}}_3(t_f). \tag{15}$$

The velocity departure equation obviously holds at time t_0 , so the time derivative of Eq. 13, evaluated at time t_0 , allows a new estimate for the initial velocity to be calculated.

$$\dot{\mathbf{r}}_{new}(t_0) = \dot{\mathbf{r}}_{ref}(t_0) + \sum_{j=1}^3 \alpha_j \Delta \dot{\mathbf{r}}_j(t_0). \tag{16}$$

We mention that occasionally the volume shown in Fig. 1 can collapse into a plane, and in extreme cases into a line as a consequence of the particular sensitivities of the local position variations. While rare, these rank deficiencies can be overcome in several ways. The decision to use three orthogonal independent velocity variations at t_0 is heuristically reasonable, but not a constraint. Instead, we can introduce sets of random initial velocity variations. We are also not constrained to use only three particular solutions if a rank deficiency in Eq. 16 is encountered. If $n > 3$ particular solutions are generated, then in lieu of Eq. 13, we can determine the smallest $\Sigma \alpha_j^2$ satisfying

$$\mathbf{r}(t_f) = \mathbf{r}_{ref}(t_f) + \sum_{j=1}^n \alpha_j \Delta \mathbf{r}_j(t_f). \tag{17}$$

Given an invertible matrix in Eq. 13, Eq. 1 can now be re-solved with the reference trajectory’s initial velocity replaced by $\dot{\mathbf{r}}_{new}(t_0)$. The procedure is repeated to compute three reference neighboring trajectories which will result in new α ’s. We iterate for improved α ’s using Eqs. 13 and 16, analogous to Newton’s method, but without the necessity of the partials $\frac{\partial \mathbf{r}}{\partial \mathbf{r}_0}$ (from computing a state transition matrix).

Any numerical integrator can be used for solving two-point boundary problems with MPS, however using MCPI affords an avenue for increased efficiency that is not available with other step-by-step integrators. We take advantage of the path approximation nature of MCPI (that is, nodes iteratively converge to fixed points in space) and utilize a variable fidelity force model for propagating the reference trajectory as described in Macomber’s dissertation [17]. Since the particular solutions lie close to the reference trajectory (within the linear domain), the following approximation for computing the acceleration for the particular solutions is made.

$$\mathbf{a}_{particular} = \mathbf{a}_{particular_{low}} + \Delta \mathbf{a}_{reference}, \tag{18}$$

$$\mathbf{a}_{particular_{low}} = \mathbf{a}_{particular_{(T B + J_2 + J_3 + J_4 + J_5 + J_6)}}, \tag{19}$$

$$\Delta \mathbf{a}_{reference} = \mathbf{a}_{reference_{40 \times 40}} - \mathbf{a}_{reference_{low}}. \tag{20}$$

The particular solutions are then assumed to lie close to the reference trajectory, and their accelerations may be approximated at any node using

$$\mathbf{a}_{approx_{particular}}(\mathbf{r}) = \mathbf{a}_{T B_{particular}}(\mathbf{r}) + \mathbf{a}_{(J_2 + J_3 + J_4 + J_5 + J_6)_{particular}}(\mathbf{r}) + (\mathbf{a}_{full_{particular}}(\mathbf{r}) - \mathbf{a}_{low_{particular}}(\mathbf{r})) + H.O.T. \tag{21}$$

Remarkably, we demonstrate that computing the particular solutions with only low fidelity approximation function evaluations, that is, *two-body plus zonal perturbations* plus the *difference between the full force evaluation and two-body plus zonal perturbations* on the *reference trajectory*, greatly increases the efficiency of the algorithm while maintaining machine precision accuracy. Later we show that solving the perturbed Lambert's problem using MPS with MCPI is about an order of magnitude faster compared with the classical shooting method and a tenth-twelfth order Runga-Kutta (RK(12)10) integrator. The RK(12)10 algorithm was shared by Feagin and Gottlieb [19].

Keplerian Lambert Solver

We now discuss an analytic method for solving the Keplerian TPBVP [2]. For the TPBVP, the initial position vectors (\mathbf{r}_1 and \mathbf{r}_2) and time of flight ($t_{desired}$) are known, but what is unknown is the initial velocity ($\dot{\mathbf{r}}_1$) that is required to reach the final position in the desired time interval. General TPBVP shooting methods require a guess for the initial velocity to propagate the trajectory forward over the desired time interval. The *miss distance* between the current final position and the desired final position is used to refine the initial velocity guess, and the propagation is repeated iteratively until convergence. In contrast, the Keplerian solver used in this paper requires an initial guess for a (semimajor axis), and assuming two-body motion, the resulting trajectory will always terminate at the desired final position. However, the time of arrival is uncertain and it is unlikely that the initial guess for a would allow the spacecraft to arrive at the desired time. The error in the flight time, based on its sensitivity to a , is what must be iterated in order to determine the correct value of a that corresponds to the desired time of flight.

Ochoa and Prussing [2] provides a comprehensive mathematical explanation of Lambert's problem, and develops the following transcendental equation that must be iterated via Newton's method to solve for a corresponding to $t_{desired}$.

$$\sqrt{\mu}t_{des} = a^{3/2} (\alpha - \beta - (\sin(\alpha) - \sin(\beta))), \quad (22)$$

where α and β are given by Eqs. 23 and 24 respectively, and μ is the gravitational parameter. The parameters c and s are the chord and semiperimeter and are defined in Prussing's paper [2].

$$\sin\left(\frac{\alpha}{2}\right) = \left(\frac{s}{2a}\right)^{1/2}. \quad (23)$$

$$\sin\left(\frac{\beta}{2}\right) = \left(\frac{s-c}{2a}\right)^{1/2}. \quad (24)$$

When solving Lambert's problem there exists a minimum semimajor axis ($a_m = \frac{s}{2}$) associated with the minimum energy orbit transfer, and any transfer with a semimajor axis less than a_m will result in an orbit that does not have enough energy to hit the desired target position. Prussing suggests a good initial guess for starting the iterations would be $a = 1.001a_m$, with careful attention (see discussion below and Fig. 5) to ensure that all $2N + 1$ solutions are obtained along all relevant solution branches for the multiple revolution case.

The time of flight associated with this minimum energy transfer is computed as follows:

$$\sqrt{\mu}t_m = \left(\frac{s^3}{8}\right)^{1/2} (\alpha_m - \beta_m + \sin(\beta_m)), \tag{25}$$

where $\alpha_m = \pi$ and $\sin\left(\frac{\beta_m}{2}\right) = \left(\frac{s-c}{s}\right)^{1/2}$. For $0 \leq \theta \leq \pi$, $\beta_m = \beta_{m0}$ and for $\pi \leq \theta \leq 2\pi$, $\beta_m = -\beta_{m0}$, where β_{m0} is the principal value of the \sin^{-1} function. As discussed in Ochoa and Prussing [2], α and β are closely related to eccentric anomaly E_2 and E_1 at times t_1 and t_2 by a constant offset and therefore $\alpha - \beta = E_2 - E_1$, and $\alpha > \beta$. In addition to t_m (transfer time corresponding to minimum energy ellipse) there is also the minimum time in which the elliptic transfer can be made. This is denoted as t_p , or the parabolic transfer time.

$$\sqrt{\mu}t_p = \frac{\sqrt{2}}{3} \left(s^{3/2} - \text{sgn}(\sin(\theta))(s - c)^{3/2}\right), \tag{26}$$

where the *signum function*, which is defined in Eq. 27, takes care of the sign change for transfer angles of $\theta < \pi$ to $\theta > \pi$.

$$\text{sgn}(x) = \begin{cases} 1 & \text{for } x > 0 \\ -1 & \text{for } x < 0 \end{cases}. \tag{27}$$

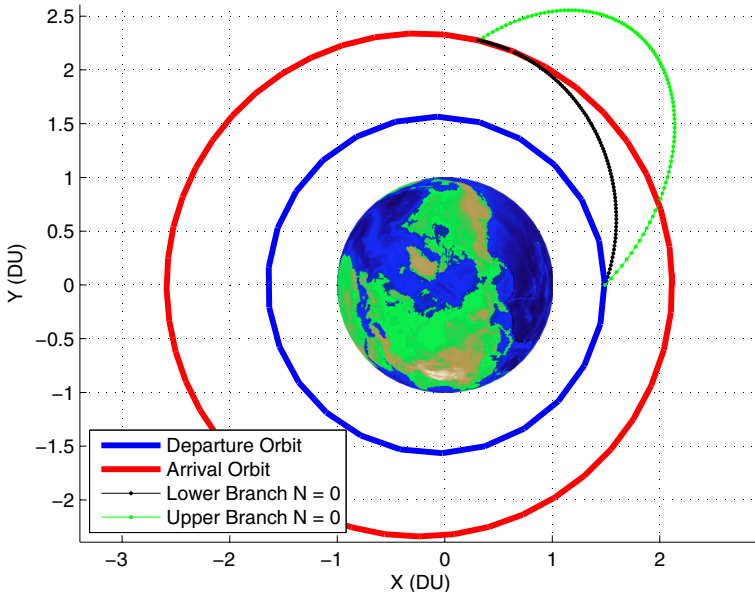


Fig. 2 Two example transfers computed for a transfer maneuver between two low Earth orbits. Note that one distance unit (DU) is one Earth radii

To summarize, Eq. 22 can be used to compute all elliptic transfers in the range $0 \leq \theta \leq 2\pi$ where the values of α and β are determined from the principle values as follows:

$$0 \leq \theta \leq \pi, \quad \beta = \beta_0, \tag{28}$$

$$\pi \leq \theta \leq 2\pi, \quad \beta = -\beta_0, \tag{29}$$

$$t_{des} \leq t_m, \quad \alpha = \alpha_0, \tag{30}$$

$$t_{des} > t_m, \quad \alpha = 2\pi - \alpha_0. \tag{31}$$

To demonstrate the algorithm we show two example transfers that start at the same point on the departure orbit (blue) and terminate at the same point on the arrival orbit (red), but each transfer has a different specified time of flight and a different semimajor axis (Fig. 2). Figure 3 shows a dashed curve that is a Eq. 22 swept for various a values. This figure is easily generated in the forward direction (given r_1 , r_2 , θ or s , and sweeping over values of interest larger than a_m , while computing α and β from Eqs 23 and 24). Note that the two branches that are separated by t_m . The two orbit transfers shown in Fig. 2 lie on different branches of the curve shown in Fig. 3. The solution on the lower branch has a shorter time of flight and was computed with $\alpha = \alpha_0$ (obtained by adopting the principle value from Eq. 23 with $\alpha_0 = 2 \sin^{-1}(\frac{s}{2a}), < \pi$), whereas the solution on the upper branch has a longer time of flight and was computed with $\alpha = 2\pi - \alpha_0$.

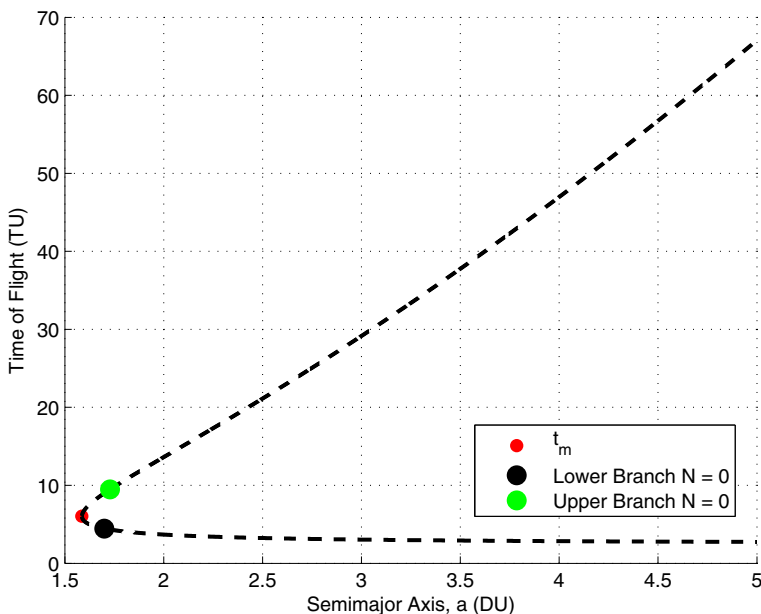


Fig. 3 Transfer time as a function of semimajor axis for transfers between two specific points on two different low Earth orbits. Note that one distance unit (DU) is one Earth radii

Multiple Revolution Solutions

When the desired time of flight is long enough for the transfer orbit to make one or multiple complete revolutions of the focus ($\theta \geq 2\pi$) we find that the solution to Lambert’s problem is no longer unique, and in fact there are $2N + 1$ distinct solutions to the problem (Fig. 4). As before, the semimajor axis is related to the desired time of flight through the following transcendental equation that differs only from Eq. 22 in that there is an additional term of $2N\pi$ added in the parenthesis.

$$\sqrt{\mu}t_{des} = a^{3/2} (2N\pi + \alpha - \beta - (\sin(\alpha) - \sin(\beta))) \tag{32}$$

Figure 5 shows the time of flight as a function of sweeping the semimajor axis over a certain range for different values of N . Note that on each curve for which $N > 0$ there is a minimum possible transfer time (t_{minN}) which is marked by the blue dots. As expected, there is no minimum time-of-flight for the $N = 0$ case because as the time is decreased the orbit will eventually switch from an elliptic transfer to a parabolic transfer.

The value of t_{mN} (33) that corresponds to the minimum energy orbit with semi-major axis ($a_m = \frac{s}{2}$) is marked by the red dots in Fig. 5. This value is critical to the solution process for isolating the multiple roots because it separates each curve into an upper and lower branch, each of which contains a solution.

$$\sqrt{\mu}t_{mN} = \left(\frac{s}{2}\right)^{3/2} ((2N + 1)\pi - \beta_m + \sin(\beta_m)). \tag{33}$$

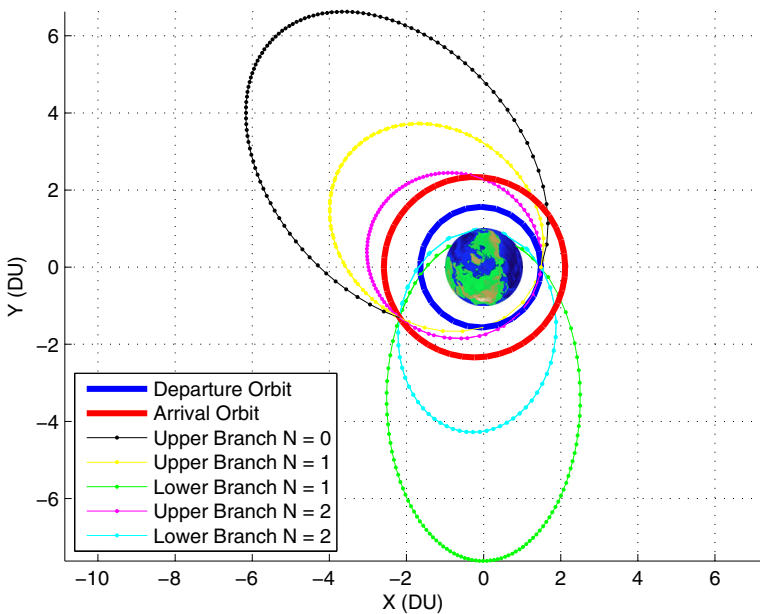


Fig. 4 Multiple solutions for transferring between the same point on the departure orbit (blue) to the same point on the arrival orbit (red), in the same specified time of flight

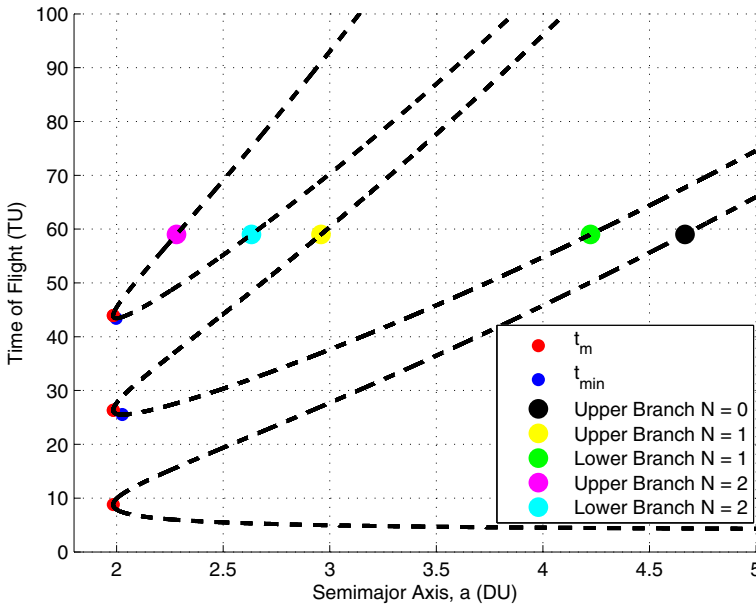


Fig. 5 Time of flight as a function of semimajor axis for transfers between the two orbits shown in Fig. 4. All five orbit transfers occur in the same time of flight but have different values of semimajor axis. The three dashed black curves represent solutions for the $N = 0$ case (lower curve), $N = 1$ (middle curve) and $N = 2$ (upper curve). In this example $N = 2$ and we expect $2N + 1 = 5$ solutions

To start the solution process the value of N_{max} must be determined. This is done by root solving $f(a) = 0$ (34) with different values of $N > 0$. The function $f(a)$ is given by

$$f(a) = (6N\pi + 3(\alpha - \beta) - (\sin(\alpha) - \sin(\beta))) * (\sin(\alpha - \beta) + (\sin(\alpha) - \sin(\beta))) \dots - 8(1 - \cos(\alpha - \beta)), \quad (34)$$

where the derivative, which is required for Newton’s method, is given by

$$f'(a) = \frac{\partial f}{\partial a} = ((6N\pi + 3\xi - \eta)(\cos(\xi) + \cos(\alpha)) \dots + (3 - \cos(\alpha))(\sin(\xi) + \eta) - 8\sin(\xi)) \left(-\frac{1}{a} \tan\left(\frac{\alpha}{2}\right)\right) \dots + ((6N\pi + 3\xi - \eta)(-\cos(\xi) - \cos(\alpha)) \dots + (-3 - \cos(\beta))(\sin(\xi) + \eta) + 8\sin(\xi)) \left(-\frac{1}{a} \tan\left(\frac{\beta}{2}\right)\right), \quad (35)$$

and $\xi \equiv \alpha - \beta$ and $\eta = \sin(\alpha) - \sin(\beta)$. A good initial guess is $a = 1.001a_m$ since the converged value will be bigger than a_m . Once the value of a is found for a particular $N > 0$, t_{minN} can be computed using Eq. 32. If t_{des} is less than say t_{min3} , then $N_{max} = 2$ and there are 5 solutions which must be found. If $t_{des} = t_{minN_{max}}$ then the two solutions on the N_{max} branch are equal and there are a total of $2N_{max}$ unique solutions.

Once the number of solutions is determined, another Newton iteration is required for determining the values of these solutions which are the a 's that correspond to the specified t_{des} . That is, Eq. 36 must be satisfied iteratively, where Δt is the current transfer time estimate on a particular iteration.

$$g(a) = t_{des}(a) - \Delta t = 0. \tag{36}$$

The derivative, which is also required for Newton's method, is given by

$$\frac{\partial \Delta t}{\partial a} = \frac{\frac{1}{2} \left(\frac{a}{\mu} \right)^{\frac{1}{2}}}{\sin(\alpha - \beta) + (\sin(\alpha) - \sin(\beta))} f(a), \tag{37}$$

where the values of α and β are computed using Eqs. 23 and 24 respectively. If $\Delta t \leq t_{mN}$ then $\alpha = \alpha_0$ is used and the solution falls on the lower branch; if $\Delta t > t_{mN}$ then $\alpha = 2\pi - \alpha_0$ and the solution falls on the upper branch. If $t_{minNmax} \leq \Delta t \leq t_{mNmax}$, that is the time of flight on the current iteration is greater than the minimum possible time in which to make the transfer, but smaller than the time of flight corresponding to the minimum energy transfer ellipse, then the situation is a little more challenging as both solutions fall on the lower branch of the curve and thus $\alpha = \alpha_0$ for both solutions. In this case the solutions lie on opposite sides of a_{tmin} , the semimajor axis

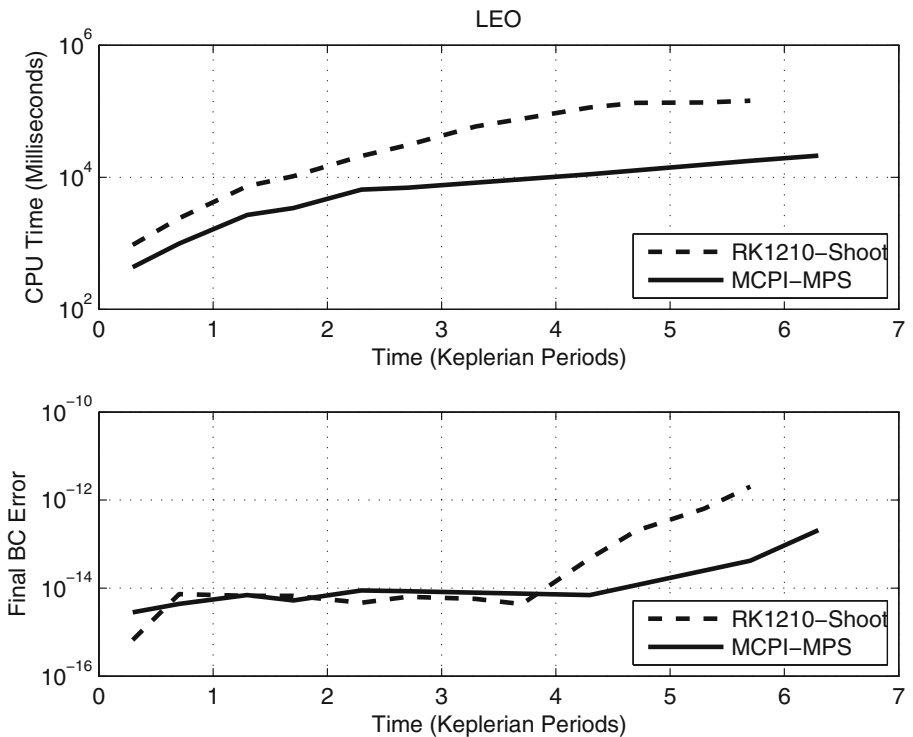


Fig. 6 Accuracy and efficiency comparison for varying time-of-flight transfers along a LEO using MCPI-MPS and Newton-RK(12)10-shooting

associated with the minimum possible transfer time, and this information can be used to pick an appropriate initial guess for a .

Terminal Velocity Vectors

Once the value of a is known the terminal velocity vectors can be computed. These may be written as a set of skewed unit vectors that are co-linear to the local radius and the chord respectively.

$$\mathbf{u}_1 \equiv \frac{\mathbf{r}_1}{r_1}, \tag{38}$$

$$\mathbf{u}_2 \equiv \frac{\mathbf{r}_2}{r_2}. \tag{39}$$

$$\mathbf{u}_c \equiv \frac{(\mathbf{r}_2 - \mathbf{r}_1)}{c}. \tag{40}$$

Prussing and Conway [20] states that the initial velocity vector (\mathbf{v}_1) can be expressed as

$$\mathbf{v}_1 = (B + A)\mathbf{u}_c + (B - A)\mathbf{u}_1 \tag{41}$$

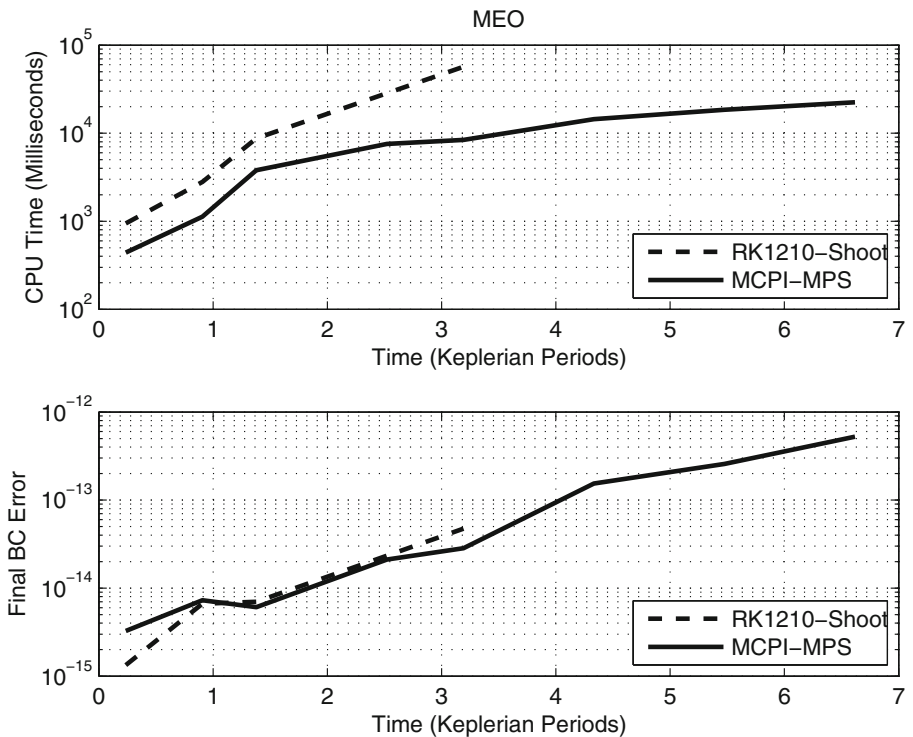


Fig. 7 Accuracy and efficiency comparison for varying time-of-flight transfers along a MEO using MCPI-MPS and Newton-RK(12)10-shooting

where

$$A = \left(\frac{\mu}{4a}\right)^{1/2} \cot\left(\frac{\alpha}{2}\right), \tag{42}$$

$$B = \left(\frac{\mu}{4a}\right)^{1/2} \cot\left(\frac{\beta}{2}\right), \tag{43}$$

and the values of α and β are determined from Eqs. 23 and 24 respectively. The final velocity vector (v_2) is computed as follows:

$$v_2 = (B + A) u_c - (B - A) u_2. \tag{44}$$

Knowing the value for the initial and final velocity on the transfer orbit allows a Δv value for the specific transfer to be computed.

It is important to note that in general, not all the possible solutions are feasible. Some will collide with the Earth (green and cyan in Fig. 4), some exceed escape velocity, and others, near π and multiples of π , are undefined as a result of the plane ambiguity that is associated with a 180° transfer in Lambert’s problem. In addition, if the transfer orbit Δv exceeds an upper limit imposed by the mission then the simulated solution must also be treated as infeasible.

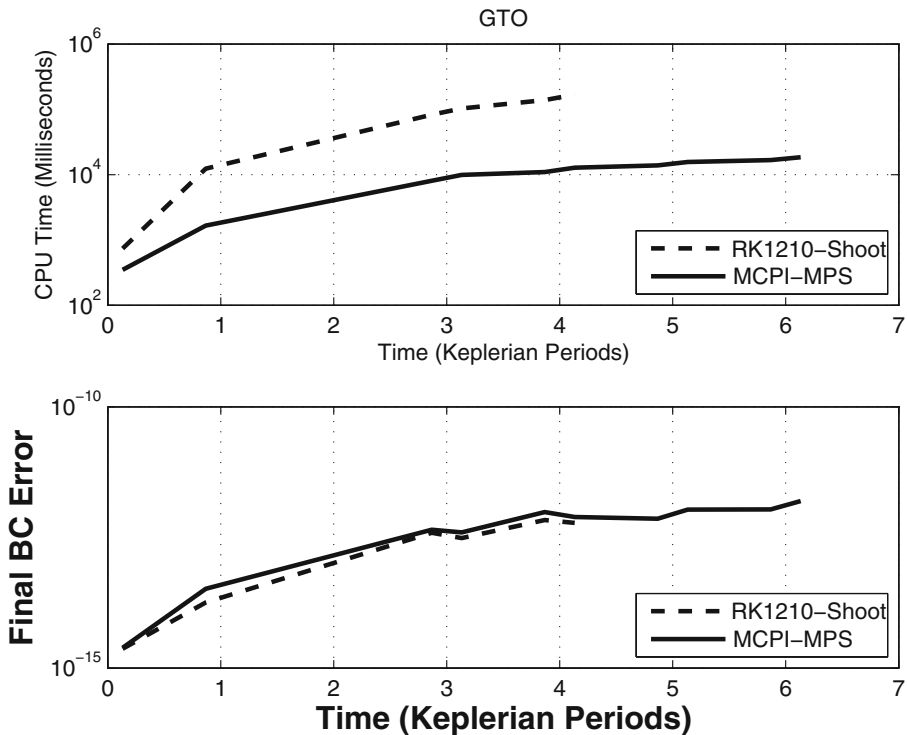


Fig. 8 Accuracy and efficiency comparison for varying time-of-flight transfers along a GTO using MCPI-MPS and Newton-RK(12)10-shooting

Algorithm Efficiency: MPS vs Newton-shooting

To compare the computational efficiency of MCPI-MPS to the Newton-shooting method, where RK(12)10 is used as the integrator, we perform four example orbit transfers spanning low Earth orbit (LEO), medium Earth orbit (MEO), geostationary transfer orbit (GTO), and Molniya or high Earth orbit (HEO). For each test case a two-body initial guess is used to start the computation of the perturbed orbit transfer that considers a (40×40) degree and order EGM2008 gravity model.

Figures 6, 7, 8 and 9 show the computation time in milliseconds (top panel) and the accuracy to which the final boundary conditions are met (bottom panel). These computations were performed in serial on the head node of the LASR Lab SSA computer cluster. Each of the compute nodes has a pair of Intel Xeon 2.6GHz CPU and 64GB of RAM. In all cases the integrators are tuned such that the transfer trajectories preserve the Hamiltonian to machine precision. The Newton-shooting method makes use of a state transition matrix that is populated with the two-body partials. We found that including partials for the full spherical harmonic gravity approximation significantly increased the computation time while providing negligible improvement in the accuracy.

Each plot consists of an average over several runs: 10 runs with an inclination of 0° , 10 runs with an inclination of 10° , and 10 runs with an inclination of 30° . This

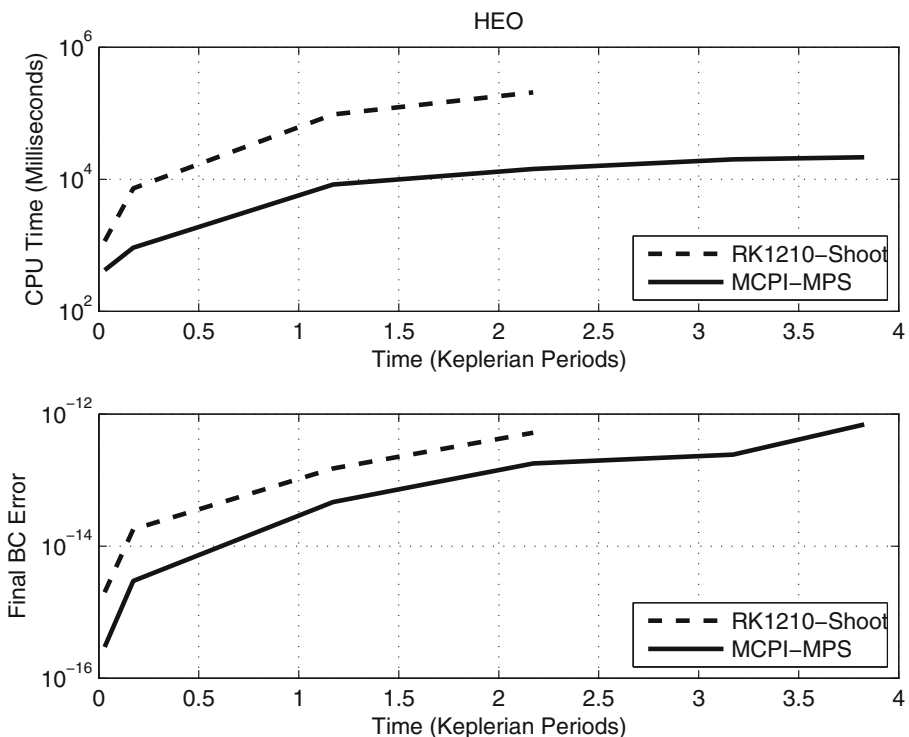


Fig. 9 Accuracy and efficiency comparison for varying time-of-flight transfers along a HEO using MCPI-MPS and Newton-RK(12)10-shooting

allows more of the gravity field to be sampled and results in a more representative comparison of the methods. In all four test cases our MCPI-MPS algorithm outperforms the Newton-RK(12)10-shooting method with regard to efficiency, while we obtain essentially the same (less than half an order of magnitude) or better accuracy.

Conclusion

We presented a new method for solving the multiple revolution perturbed Lambert problem using the *method of particular solutions* and *modified Chebyshev-Picard iteration*. Any numerical integrator could be used for solving two-point boundary problems with the *method of particular solutions*; however, we showed that using *modified Chebyshev-Picard iteration* afforded an avenue for increased efficiency that was not available with other step-by-step integrators. We utilized a variable fidelity force model for propagating the reference trajectory and demonstrated that computing the particular solutions with only low fidelity function evaluations resulted in our algorithm being about an order of magnitude faster compared with the classical shooting method and a tenth-twelfth order Runge-Kutta integrator. It is well known that the solution to Lambert's problem over multiple revolutions is not unique and to ensure that all possible solutions are considered we made use of a reliable preexisting Keplerian Lambert solver to *warm start* our perturbed algorithm.

Acknowledgments We thank our sponsors: AFOSR (Julie Moses and Stacie Williams) and AFRL (Alok Das, et al.) for their support and collaborations under various contracts and grants. We are also grateful to John Prussing for shedding light on the non-uniqueness of solutions to the multiple revolution Lambert problem.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Battin, R.: An introduction to the mathematics and methods of astrodynamics. AIAA Education Series (1999)
2. Ochoa, S., Prussing, J.: Multiple revolution solutions to Lambert's problem. *Adv. Astronaut. Sci.* **79**, 120–1216 (1992)
3. Bai, X., Junkins, J.: Modified Chebyshev-Picard iteration methods for solution of boundary value problems. *Adv. Astronaut. Sci.* **140**, 381–400 (2011)
4. Bai, X.: Modified Chebyshev-Picard Iteration for Solution of Initial Value and Boundary Value Problems. PhD. Dissertation, Texas A&M, College Station, Texas, USA (2010)
5. Woollands, R., Bani Younes, A., Junkins, J.: New solutions for the perturbed Lambert problem using regularization and Picard iteration. *J. Guid. Control. Dyn.* **38**, 1548–1562 (2015)
6. Woollands, R.: Regularization and Computational Methods for Precise Solution of Perturbed Orbit Transfer Problems. PhD. Dissertation, Texas A&M, College Station, Texas, USA (2016)
7. Miele, A., Iyer, R.: General technique for solving nonlinear. Two-point boundary value problems via the method of particular solutions. *J. Optim. Theory Appl.* **5**(5), 392–399 (1970)

8. Clenshaw, C.W., Norton, H.J.: The solution of nonlinear ordinary differential equations in Chebyshev series. *Comput. J.* **6**, 88–92 (1963)
9. Feagin, T.: The Numerical Solution of Two-Point Boundary Value Problems using Chebyshev Polynomial Series. Ph.D. dissertation, University of Texas, Austin, Texas, USA (1972)
10. Feagin, T., Nacozy, P.: Matrix formulation of the Picard method for parallel computation. *Celest. Mech. Dyn. Astron.* **29**, 107–115 (1983)
11. Shaver, J.: Formulation and Evaluation of Parallel Algorithms for the Orbit Determination Problem. Ph.D. dissertation, Department of Aeronautics and Astronautics, MIT, Cambridge MA (1980)
12. Fukushima, T.: Vector integration of dynamical motions by the Picard-Chebyshev method. *Astron. J.* **113**, 2325–2328 (1997)
13. Bai, X., Junkins, J.: Modified Chebyshev-Picard iteration methods for solution of initial value problems. *Adv. Astronaut. Sci.* **139**, 345–362 (2011)
14. Junkins, J., Bani Younes, A., Woollands, R., Bai, X.: Orthogonal approximation in higher dimensions: Applications in astrodynamics, AAS 12-634 JN Juang Astrodynamics Symp (2012)
15. Junkins, J.L., Bani Younes, A., Woollands, R.M., Bai, X.: Efficient and Adaptive Orthogonal Finite Element Representation of the Geopotential. *J. Astronaut. Sci.* January 2017
16. Bani Younes, A.: Orthogonal Polynomial Approximation in Higher Dimensions: Applications in Astrodynamics. Ph.D. dissertation, Texas A&M, College Station, Texas USA (2013)
17. Macomber, B.: Enhancements of Chebyshev-Picard Iteration Efficiency for Generally Perturbed Orbits and Constrained Dynamics Systems. Ph.D. Dissertation, Texas A&M University, College Station, Texas, USA (2015)
18. Macomber, B., Probe, A., Woollands, R., Read, J., Junkins, J.: Enhancements of modified Chebyshev-Picard iteration efficiency for perturbed orbit propagation. *Computational Modelling in Engineering & Sciences* **111**, 29–64 (2016)
19. Feagin, T., Gottlieb, B.: RK(12)10 Numerical Integrator Algorithm Shared by authors (2014)
20. Prussing, J., Conway, B.: *Orbital Mechanics*, 2nd edn. Oxford University Press (2012)