ELSEVIER

Journal of Computational and Applied Mathematics 120 (2000) 27–40

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

Very low-thrust trajectory optimization using a direct SQP method

John T. Betts

Mathematics and Engineering Analysis, Boeing Information and Support Services, P.O. Box 3707, MC 7L-21, Seattle, WA, DC 98124-2207, USA

Received 11 December 1998; received in revised form 6 September 1999

Abstract

The direct transcription or collocation method has demonstrated notable success in the solution of trajectory optimization and optimal control problems. This approach combines a sparse nonlinear programming algorithm with a discretization of the trajectory dynamics. A challenging class of optimization problems occurs when the spacecraft trajectories are characterized by thrust levels that are very low relative to the vehicle weight. Low-thrust trajectories are demanding because realistic forces due to oblateness, aerodynamic drag, and third-body perturbions often dominate the thrust. Furthermore because the thrust is so low, significant changes to the orbits require very long duration trajectories. When a collocation method is applied to a problem of this type, the resulting nonlinear program is very large because the trajectories are long, and very nonlinear because of the perturbing forces. This paper describes the application of the transcription method to the solution of very low-thrust orbit transfers. The vehicle dynamics are defined using a modified set of equinoctial coordinates, and the trajectory modeling is described using these dynamics. A solution is presented for a representative transfer using a spacecraft with a thrust acceleration of approximately 1.25×10^{-7} km/s^2. This transfer requires over 578 revolutions, and leads to a sparse optimization problem with 416 123 variables and 249 674 constraints. Issues related to the numerical conditioning and problem formulation are discussed.

Keywords: Optimal control; Sparse nonlinear programming; Trajectory optimization

1. Introduction

The construction of low earth orbit (LEO) satellite constellations is motivated by applications in the telecommunication industry. Typical constellations require hundreds of small spacecraft which utilize very low-thrust propulsion systems. Deployment of these constellations obviously requires that a trajectory must be constructed for each spacecraft to move from the park orbit to the desired mission orbit. The basic goal of the paper is to determine the optimal thrust magnitude and direction which minimizes the fuel consumed during the transfer between park orbit and mission orbit.

0377-0427/00/$ - see front matter
PII: S0377-0427(00)00301-0

A major attraction of a low-thrust propulsion system is the payload efficiency. In fact for the example problem nearly 97% of the vehicle weight can be inserted into the final mission orbit. On the other hand, because the thrust is very small, the transfer times are very long. Both low-thrust and long transfer times translate into numerical challenges which are addressed in this paper.

In order to illustrate the concepts and motivate the discussion throughout the paper we will focus on a particular (representative) spacecraft and orbit transfer. We consider a circular park orbit at an altitude of 500 km, and inclination of 87.4°. The mission orbit has an apogee altitude of 1400 km, an eccentricity of 0.0011, an argument of perigee of 90° and inclination of 87.4°. For a vehicle with initial weight of 1600 kg, maximum thrust of 0.2 n, and specific impulse 1600 s, the optimal final weight is 1554.29 kg.

2. Equations of motion

2.1. Background

The motion of a body can be described by a system of second-order ordinary differential equations

$$\ddot{\boldsymbol{r}} + \mu \frac{\boldsymbol{r}}{r^3} = \boldsymbol{a}_{\mathrm{d}}, \tag{1}$$

where the radius $r = \|\boldsymbol{r}\|$ is the magnitude of the inertial position vector $\boldsymbol{r}$, and μ is the gravitational constant. In this formulation we define the vector $\boldsymbol{a}_{\mathrm{d}}$ as the *disturbing acceleration*. This representation for the equations of motion is referred to as Gauss' form of the variational equations.

The Gauss' form of the equations of motion isolates the disturbing acceleration from the central force gravitational acceleration. Note that when the disturbing acceleration is zero $\|\boldsymbol{a}_{\mathrm{d}}\| = 0$ the fundamental system (1) is just a two-body problem. The solution of the two-body problem can of course be stated in terms of the constant orbital elements. For low-thrust trajectories this formulation is appealing because we expect $\|\boldsymbol{a}_{\mathrm{d}}\|$ to be "small" and consequently we expect that the solution can be described in terms of "almost constant" orbital elements. In order to exploit the benefits of the variational form of the differential equations (1) it is necessary to transform the cartesian state into an appropriate set of orbit elements. One potential set are the classical elements $(a, e, i, \Omega, \omega, M)$. However, these elements exhibit singularities for $e = 0$, and $i = 0°, 90°$. A set of *equinoctial* orbital elements that avoids the singularities in the classical elements have been described in [1,8,9]. Kechichian developed a particular form of these equations in [11,10,13], and these equations were used to solve a low-thrust earth orbit transfer problem as described in [3]. Unfortunately, this set of equinoctial elements does not accommodate orbits with $e \geqslant 1$. To eliminate this deficiency, a modified set of equinoctial orbit elements is described in [5] based on the work in [12].

2.2. Translational dynamics in modified equinoctial coordinates

The dynamics of the system can be described in terms of the state variables

$$[\boldsymbol{y}, w] = [p, f, g, h, k, L, w] \tag{2}$$

and control variables

$$[\boldsymbol{u}, \tau] = [u_r, u_\theta, u_h, \tau]. \tag{3}$$

Using the modified equinoctial elements the equations of motion for a vehicle with variable thrust can be stated as

$$\dot{\boldsymbol{y}} = \boldsymbol{A}(\boldsymbol{y})\boldsymbol{\Delta} + \boldsymbol{b}, \tag{4}$$

$$\dot{w} = -T[1 + 0.01\tau]/I_{\mathrm{sp}}, \tag{5}$$

$$0 = \|\boldsymbol{u}\| - 1, \tag{6}$$

$$\tau_L \leqslant \tau \leqslant 0. \tag{7}$$

The equinoctial dynamics are defined by the matrix

$$\boldsymbol{A} = \begin{bmatrix} 0 & \dfrac{2p}{q}\sqrt{\dfrac{p}{\mu}} & 0 \\ \sqrt{\dfrac{p}{\mu}}\sin L & \sqrt{\dfrac{p}{\mu}}\dfrac{1}{q}\{(q+1)\cos L + f\} & -\sqrt{\dfrac{p}{\mu}}\dfrac{g}{q}\{h\sin L - k\cos L\} \\ -\sqrt{\dfrac{p}{\mu}}\cos L & \sqrt{\dfrac{p}{\mu}}\dfrac{1}{q}\{(q+1)\sin L + g\} & \sqrt{\dfrac{p}{\mu}}\dfrac{f}{q}\{h\sin L - k\cos L\} \\ 0 & 0 & \sqrt{\dfrac{p}{\mu}}\dfrac{s^2\cos L}{2q} \\ 0 & 0 & \sqrt{\dfrac{p}{\mu}}\dfrac{s^2\sin L}{2q} \\ 0 & 0 & \sqrt{\dfrac{p}{\mu}}\dfrac{1}{q}\{h\sin L - k\cos L\} \end{bmatrix} \tag{8}$$

and the vector

$$\boldsymbol{b}^{\mathrm{T}} = \left[0 \quad 0 \quad 0 \quad 0 \quad 0 \quad \sqrt{\mu p}\left(\frac{q}{p}\right)^2\right], \tag{9}$$

where

$$q = 1 + f\cos L + g\sin L, \tag{10}$$

$$r = \frac{p}{q}, \tag{11}$$

$$\alpha^2 = h^2 - k^2, \tag{12}$$

$$\chi = \sqrt{h^2 + k^2}, \tag{13}$$

$$s^2 = 1 + \chi^2. \tag{14}$$

The equinoctial coordinates $\boldsymbol{y}$ are related to the cartesian state $(\boldsymbol{r}, \boldsymbol{v})$ according to the expressions

$$\boldsymbol{r} = \begin{bmatrix} \frac{r}{s^2}(\cos L + \alpha^2 \cos L + 2hk \sin L) \\ \frac{r}{s^2}(\sin L - \alpha^2 \sin L + 2hk \cos L) \\ \frac{2r}{s^2}(h \sin L - k \cos L) \end{bmatrix}, \tag{15}$$

$$\boldsymbol{v} = \begin{bmatrix} -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}(\sin L + \alpha^2 \sin L - 2hk \cos L + g - 2fhk + \alpha^2 g) \\ -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}(-\cos L + \alpha^2 \cos L + 2hk \sin L - f + 2ghk + \alpha^2 f) \\ \frac{2}{s^2}\sqrt{\frac{\mu}{p}}(h \cos L + k \sin L + fh + gk) \end{bmatrix}. \tag{16}$$

As a result of this transformation the disturbing acceleration vector $\boldsymbol{a}_{\mathrm{d}}$ in (1) is replaced by

$$\boldsymbol{\Delta} = \boldsymbol{\Delta}_{\mathrm{g}} + \boldsymbol{\Delta}_{\mathrm{T}} \tag{17}$$

with a contribution due to oblate earth effects $\boldsymbol{\Delta}_{\mathrm{g}}$ and another caused by thrust $\boldsymbol{\Delta}_{\mathrm{T}}$. The disturbing acceleration is expressed in a rotating radial frame whose principal axes are defined by

$$\boldsymbol{Q} = [\, \boldsymbol{i}_r \;\; \boldsymbol{i}_\theta \;\; \boldsymbol{i}_h \,] = \left[\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|} \;\; \frac{(\boldsymbol{r} \times \boldsymbol{v}) \times \boldsymbol{r}}{\|\boldsymbol{r} \times \boldsymbol{v}\|\|\boldsymbol{r}\|} \;\; \frac{\boldsymbol{r} \times \boldsymbol{v}}{\|\boldsymbol{r} \times \boldsymbol{v}\|} \right]. \tag{18}$$

As stated Eqs. (4)–(7) are perfectly general and describe the motion of a point mass when subject to the disturbing acceleration vector $\boldsymbol{\Delta}$. Notice that when the disturbing acceleration is zero $\boldsymbol{\Delta} = 0$, the first five equations are simply $\dot{p} = \dot{f} = \dot{g} = \dot{h} = \dot{k} = 0$ which implies that the elements are constant. It is important to note that the disturbing acceleration vector can be attributed to any perturbing force(s). A more complete derivation of the equinoctial dynamics can be found in [5].

2.3. Gravitational disturbing acceleration

Oblate gravity models are typically defined in a local horizontal reference frame, that is

$$\boldsymbol{\delta g} = \delta g_n \boldsymbol{i}_n - \delta g_r \boldsymbol{i}_r, \tag{19}$$

where

$$\boldsymbol{i}_n = \frac{\boldsymbol{e}_n - (\boldsymbol{e}_n^{\mathrm{T}} \boldsymbol{i}_r)\boldsymbol{i}_r}{\|\boldsymbol{e}_n - (\boldsymbol{e}_n^{\mathrm{T}} \boldsymbol{i}_r)\boldsymbol{i}_r\|} \tag{20}$$

defines the local North direction with $\boldsymbol{e}_n = (0, 0, 1)$. A reasonably accurate model is obtained if the tesseral harmonics are ignored and only the first four zonal harmonics are included in the geopotential

function. In this case the oblate earth perturbations to the gravitational acceleration are given by

$$\delta g_n = -\frac{\mu \cos\phi}{r^2} \sum_{k=2}^{4} \left(\frac{a_e}{r}\right)^k P'_k J_k, \tag{21}$$

$$\delta g_r = -\frac{\mu}{r^2} \sum_{k=2}^{4} (k+1) \left(\frac{a_e}{r}\right)^k P_k J_k, \tag{22}$$

where ϕ is the geocentric latitude, a_e is the equatorial radius of the earth, $P_k(\sin\phi)$ is the kth-order Legendre polynomial with corresponding derivative P'_k, and the zonal harmonic coefficients are given by J_k.

Finally, to obtain the gravitational perturbations in the rotating radial frame it follows that

$$\boldsymbol{\Delta}_g = \boldsymbol{Q}^T \boldsymbol{\delta g}. \tag{23}$$

2.4. Thrust acceleration — Burn arcs

To this point the discussion has concentrated on incorporating perturbing forces due to oblate earth effects. Of course the second major perturbion is the thrust acceleration defined by

$$\boldsymbol{\Delta}_T = \frac{g_0 T[1 + 0.01\tau]}{w} \boldsymbol{u}, \tag{24}$$

where T is the maximum thrust and $\tau_L \leqslant \tau \leqslant 0$ is a throttle factor. In general the direction of the thrust acceleration vector which is defined by the time varying control vector $\boldsymbol{u}(t) = (u_r, u_\theta, u_h)$, can be chosen arbitrarily as long as the vector has unit length at all points in time, which is achieved using the path constraint (6). The magnitude of the thrust is of course related to the vehicle weight according to (5) where g_0 is the mass to weight conversion factor, and the specific impulse of the motor is denoted by I_{sp}. Defining the thrust direction using the vector $\boldsymbol{u}(t)$ and path constraint $\|\boldsymbol{u}(t)\| = 1$ is particularly well suited for missions that involve steering over large portions of the trajectory, as illustrated in [3], because ambiguities in the pointing direction are avoided. Specifying the thrust direction by two angles (e.g., yaw and pitch) which are treated as control variables is not unique since the angles $\alpha = \alpha_0 \pm 2k\pi$ all yield the *same* direction. In contrast, there is a *unique* set of control variables $\boldsymbol{u}$ corresponding to any thrust direction.

It is important to note that the magnitude of the thrust perturbation $\boldsymbol{\Delta}_T$ is small compared to the oblate earth perturbation $\boldsymbol{\Delta}_g$. For the example problem $\|\boldsymbol{\Delta}_T\| \sim 0.03\|\boldsymbol{\Delta}_g\|$. Thus the *oblate earth perturbation dominates the thrust!* In fact, the thrust is so small that the system is almost uncontrollable.

3. The boundary conditions

The standard approach for defining the boundary conditions of an orbit transfer problem is to specify the final state in terms of the instantaneous or *osculating* orbit elements at the burnout time

t_{bo}. Thus, a typical set of boundary conditions is

$$a = p[1 - f^2 - g^2]^{-1} = \hat{a}, \tag{25}$$

$$e = \sqrt{f^2 + g^2} = \hat{e}, \tag{26}$$

$$i = 2\tan^{-1}(\sqrt{h^2 + k^2}) = \hat{i}, \tag{27}$$

$$\omega = \tan^{-1}(g/f) - \tan^{-1}(k/h) = \hat{\omega}, \tag{28}$$

where (a, e, i, ω) are the classical elements semimajor axis, eccentricity, inclination, and argument of perigee, respectively, and $(\hat{a}, \hat{e}, \hat{i}, \hat{\omega})$ are the desired values. For the example of interest $\hat{a} = (1400 + a_e)/(1+\hat{e})$ (km), $\hat{e}=0.0011$, $\hat{i}=87.4^\circ$, and $\hat{\omega}=90^\circ$. Unfortunately, two factors make imposition of the boundary conditions in this form undesirable. First, the constraints are very nonlinear functions of the variables. Second, oblate earth effects make the osculating elements extremely oscillatory. Both of the factors are aggravated because the thrust is so low that the vehicle is almost uncontrollable.

For an oscillatory element $c(t)$ define the average value

$$d = \frac{1}{P}\int_0^P c(t)\,\mathrm{d}t, \tag{29}$$

where P is the (osculating) orbit period. It is convenient to rewrite this expression as a *normalized average element condition* $(d \neq 0)$

$$1 = \frac{1}{P}\int_0^P \frac{c(t)}{d}\,\mathrm{d}t - s, \tag{30}$$

where the slack variable $s=0$ when the average element is equal to desired value d. Fig. 1 illustrates the oscillatory nature of the elements, plotting the percentage variation relative to the nominal values for motion over one orbital period. Note that the eccentricity changes by over 100% in comparison to the nominal value and this variation is due entirely to oblate earth perturbations.

Because the orbit elements are nonlinear and oscillatory it is necessary to impose the boundary conditions in a more "relaxed" fashion. This can be accomplished by using the relation between classical and equinoctial elements

$$p = a(1 - e^2), \tag{31}$$

$$f = e\cos(\omega + \Omega), \tag{32}$$

$$g = e\sin(\omega + \Omega), \tag{33}$$

$$h = \tan\left(\frac{\mathrm{i}}{2}\right)\cos\Omega, \tag{34}$$

$$k = \tan\left(\frac{\mathrm{i}}{2}\right)\sin\Omega, \tag{35}$$

$$L = \Omega + \omega + v. \tag{36}$$

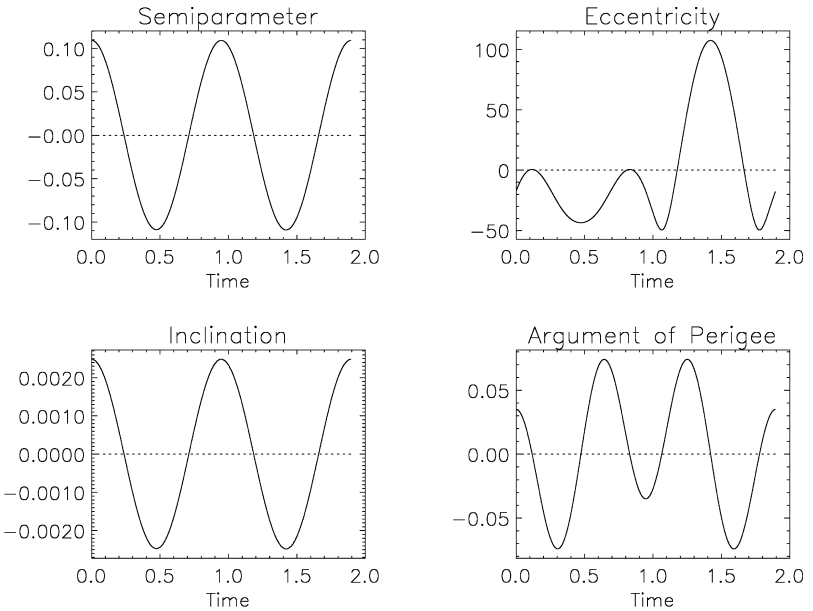


Fig. 1. Element variation over one period in mission orbit.

Since $-1 \leqslant \cos(\omega + \Omega) \leqslant 1$ and $-1 \leqslant \sin(\omega + \Omega) \leqslant 1$ it is clear that we can write

$$-e_u \leqslant f \leqslant e_u,$$

$$-e_u \leqslant g \leqslant e_u,$$

where $e_u = \max|e(t)| \sim 2\hat{e}$ as can be seen from Fig. 1. We can also replace the eccentricity constraint with

$$e_\ell \leqslant \sqrt{f^2 + g^2} \leqslant e_u, \tag{37}$$

where $e_\ell = \min|e(t)| \sim \hat{e}/2$.

In like fashion from (34) and (35) we can derive the bounds

$$-t_u \leqslant h \leqslant t_u,$$

$$-t_u \leqslant k \leqslant t_u,$$

where $t_u = \max|\tan[i(t)/2]| \sim (1 + 2.5 \times 10^{-5})\tan[\hat{i}/2]$. The inclination boundary condition is then expressed as

$$t_\ell \leqslant \sqrt{h^2 + k^2} \leqslant t_u \tag{38}$$

where $t_\ell = \min|\tan[i(t)/2]| \sim (1 - 2.5 \times 10^{-5})\tan[\hat{i}/2]$.

It is demonstrated in [5] that the argument of perigee constraint can be enforced by

$$-\omega_u \leqslant gk + fh \leqslant \omega_u \tag{39}$$

where $\omega_u \sim 0.00075$. Finally, direct examination suggests that

$$p_\ell \leqslant p \leqslant p_u \tag{40}$$

where $p_u = \max|p(t)| \sim (1 + 0.00109)\hat{a}(1 + \hat{e}^2)$ and $p_\ell = \min|p(t)| \sim (1 - 0.00109)\hat{a}(1 + \hat{e}^2)$.

In addition to bounds on the osculating quantities, we can impose conditions on the average elements. From (30) let us define

$$\frac{1}{P}\int_{t_{bo}}^{t_{bo}+P} \frac{p}{\hat{a}(1+\hat{e}^2)}\,dt - s_1 = 1, \tag{41}$$

$$\frac{1}{P}\int_{t_{bo}}^{t_{bo}+P} \frac{\sqrt{f^2+g^2}}{\hat{e}}\,dt - s_2 = 1, \tag{42}$$

$$\frac{1}{P}\int_{t_{bo}}^{t_{bo}+P} \frac{\sqrt{h^2+k^2}}{\tan[\hat{i}/2]}\,dt - s_3 = 1, \tag{43}$$

$$\frac{1}{P}\int_{t_{bo}}^{t_{bo}+P} (gk + fh)\,dt - s_4 = 0. \tag{44}$$

Observe that these conditions have been constructed using the slack variables s_1, s_2, s_3, s_4. Clearly, if they are satisfied and the slacks are zero the average element values equal the desired values. On the other hand, if $s_i \neq 0$ then the slack variables equal the relative deviation between the average and desired orbit elements.

4. Objective function

The problem statement is completed by defining a composite objective function. We will consider two different formulations of the problem. For the first approach *Formulation A* let us minimize the function

$$J_A = -w(t_{bo}) + \int_{t_0}^{t_{bo}} \{\kappa_1\tau^2 + \kappa_2[u_r^2 + (u_\theta - 1)^2 + u_h^2]\}\,dt. \tag{45}$$

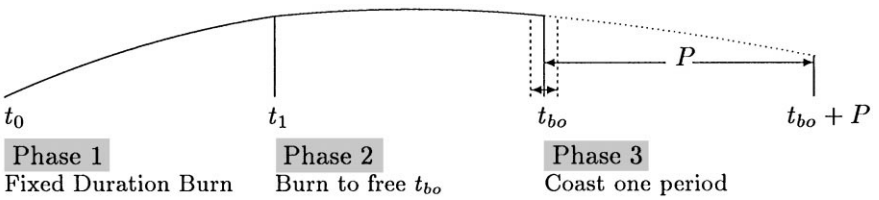
The second approach which we will refer to as *Formulation B* includes an additional penalty term in the objective, namely

$$J_B = J_A + \kappa_3 \sum_{i=1}^{4} s_i^2, \tag{46}$$

where κ_i are positive constants. The primary goal is to perform the orbit transfer with a minimum expenditure of fuel. Mathematically, this goal is expressed as maximizing the final weight, which is equivalent to minimizing the term $-w(t_{bo})$. It is also desirable to keep the thrust nearly constant at the maximum thrust level. This goal is expressed by keeping the value of the throttle parameter small, hence the appearance of the first term in the integral. The second term in the integral defines a "preferred" steering direction. Since the performance modeling does not include fuel consumed by rotational dynamics (i.e., steering), it is important to avoid unnecessary orientation manuevers. In Formulation B, the final term in the objective is introduced to define the preferred mission orbit. In particular, the final term is zero when the average final orbit conditions are equal to the desired values.

5. Complete formulation

The preceding sections have described the major details of the low-thrust trajectory optimization problem. In this section, let us collect the pieces and summarize the complete problem statement. When using Formulation A, the basic mission timeline can be subdivided into two phases, and when using Formulation B three phases are required as shown below. The first phase of fixed duration defines the major portion of the burn. The second phase simulates the remainder of the burn and terminates at the unknown burnout time t_{bo}. In Formulation B, a third phase simulating a coast for one period in the mission orbit terminates at the final time $t_{\mathrm{bo}} + P$.



The dynamics during phases 1 and 2 are defined by the system of differential-algebraic equations (4)–(7). During the coast phase which is simulated in Formulation B, $\Delta_{\mathrm{T}} = 0$ and only the first set of differential equations (4) are needed to model the dynamics. Continuity in the state variables across the phase boundaries is enforced. In addition, the controls are continuous between the first and second phase. Modeling the burn as two phases instead of one is motivated strictly by numerical considerations and this issue will be discussed below.

6. Optimal control algorithm

The method used to solve the optimal control problem is referred to as a *collocation* or direct transcription algorithm [2–5,7], as implemented in the SOCS software [6]. There are three basic operations performed in the method:

Direct transcription: *Transcribe* the optimal control problem into a nonlinear programming (NLP) problem by discretization.

Sparse nonlinear program: Solve the sparse NLP using sequential quadratic programming, i.e.

1. Solve a sparse quadratic program (QP) to estimate the NLP solution.
2. If the solution is acceptable, terminate, otherwise update the NLP solution estimate and solve a new QP subproblem.

Mesh refinement: Assess the accuracy of the approximation (i.e., the finite-dimensional problem), and if necessary refine the discretization, and then repeat the optimization steps.

The basic idea of a transcription method is to replace the optimal control problem by its finite-dimensional counterpart via discretization. All approaches divide the time interval into n_s segments:

$$t_0 < t_1 < t_2 < \cdots < t_{\mathrm{f}} = t_{n_s},$$

where the points are referred to as node, mesh or grid points. Define the number of mesh points as $M \equiv n_s + 1$. Let us introduce the notation $\boldsymbol{y}_k \equiv \boldsymbol{y}(t_k)$ to indicate the value of the state variable

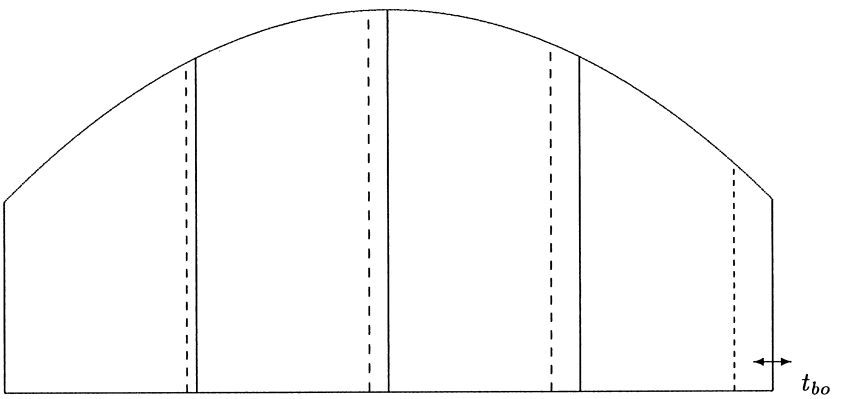
at a grid point. In like fashion denote the control at a grid point by $\boldsymbol{u}_k \equiv \boldsymbol{u}(t_k)$. For the trapezoidal discretization, the NLP variables are

$$\boldsymbol{x} = [\boldsymbol{y}_0, \boldsymbol{u}_0, \boldsymbol{y}_1, \boldsymbol{u}_1, \ldots, \boldsymbol{y}_f, \boldsymbol{u}_f, \boldsymbol{p}, t_0, t_f]^T. \tag{47}$$

The state equations (4)–(5) are approximately satisfied by setting the *defects*

$$\boldsymbol{\zeta}_k = \boldsymbol{y}_k - \boldsymbol{y}_{k-1} - \frac{h_k}{2}[\boldsymbol{f}_k + \boldsymbol{f}_{k-1}] \tag{48}$$

to zero for $k = 1, \ldots, n_s$. The right-hand side of the differential equations (4)–(5) are given by $\boldsymbol{f}_k \equiv \boldsymbol{f}[\boldsymbol{y}(t_k), \boldsymbol{u}(t_k), \boldsymbol{p}, t_k]$. The step size is denoted by $h_k \equiv t_k - t_{k-1} = \sigma_k(t_{bo} - t_0)$, where the constants $0 < \sigma_k < 1$.



When this discretization is applied to the burn phase with a fixed initial time t_0 and free burnout time t_{bo} it is clear that the Jacobian matrix will involve the derivatives $\partial\boldsymbol{\zeta}_k/\partial t_{bo} \neq 0$ for $k = 1, \ldots, M$. In other words, the Jacobian matrix has one dense column corresponding to the variable t_{bo} and changes in this variable affect the entire set of defect constraints. In order to reduce the impact of this variable it is useful to break up the burn into two distinct phases. The first phase is of fixed duration and accounts for most of the burn. In the example, phase one accounts for 569 of the 578 orbital revolutions. The second phase which is of variable duration is used to model the remainder of the burn. This situation is illustrated in Fig. 2.

7. Numerical results

The computational algorithm described was used to solve the stated low-thrust transfer. The results for Formulation A are presented in Figs. 3–5. The numerical behavior of the algorithm is summarized in Table 1. The solution required three mesh refinement iterations with the results of each iteration given in a row of the table. The first discretization used was a trapezoidal rule with $M = 13\,782$ grid points. The resulting sparse nonlinear program had $m = 110\,974$ constraints, and $n = 152\,593$ variables. The solution of the NLP required 12 evaluations of the first derivatives NGC $= 12, 6$ Hessian evaluations NHC $= 6$ and the total number of function evaluations including those needed for finite difference derivative estimates was NFE $= 636$. Because each function evaluation requires

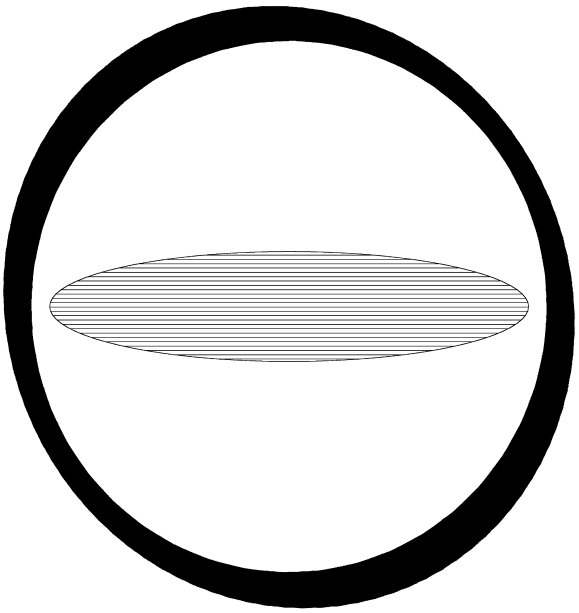


Fig. 2. Optimal trajectory.

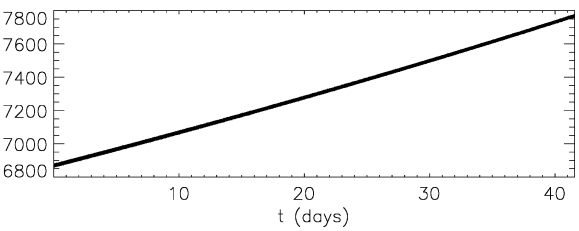


Fig. 3. Semiparameter during transfer.

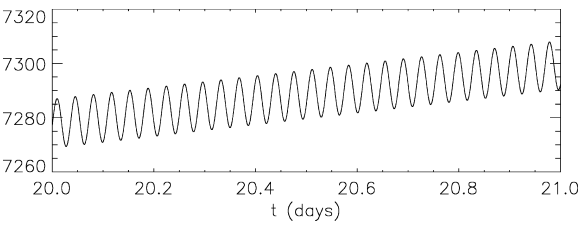


Fig. 4. Semiparameter over one day during transfer.

an evaluation of the right-hand sides of the differential equations the total number of right-hand side evaluations is NRHS $= 8\,822\,592 = 636 \times 13\,872$. The relative error in the solution using this discretization was 0.16×10^{-4} and the NLP was solved in 0.67×10^{3} s on an SGI Origin 2000. After solving the sparse NLP on the first grid the discretization method was changed from Trapezoidal to

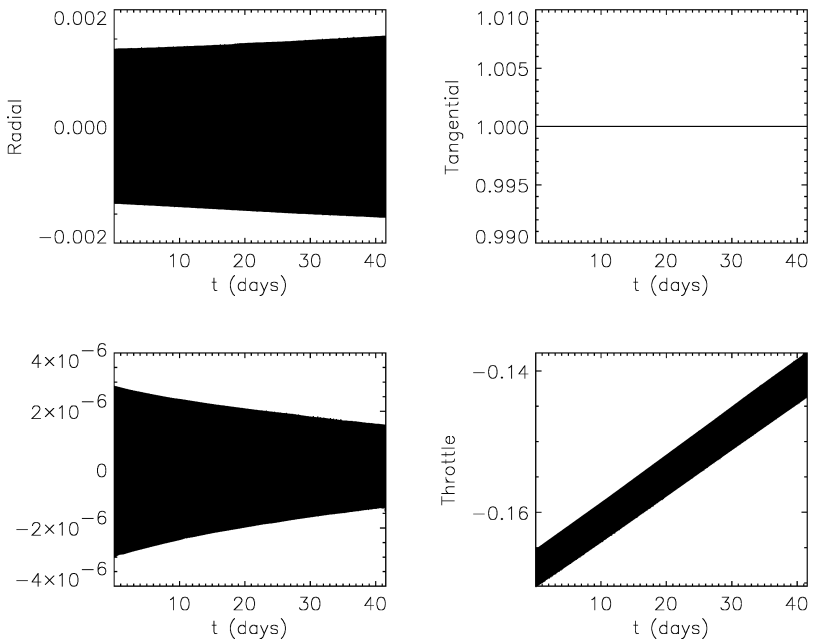


Fig. 5. Optimal control history.

Table 1

GRID	DSC	*M*	*m*	*n*	NGC	NHC	NFE	NRHS	ERRODE	CPU
1	Trp	13 872	110 974	152 593	12	6	636	8 822 592	0.16E−04	0.67E+03
2	H-S	13 872	124 844	208 073	8	5	2168	60 144 656	0.77E−06	0.19E+04
3	H-S	27 742	249 674	416 123	3	1	489	27 130 698	0.48E−07	0.12E+04
Total					23	12	3293	96 097 946		3775.31

Hermite–Simpson thereby increasing the order from two to four. This solution was obtained with an additional 2168 function evaluations in 1900 s and produced as discrete approximation with an accuracy of 0.77×10^{-6}. The second refinement iteration increased the number of grid points to 27 742 resulting in an NLP problem with 249 674 constraints and 416 123 variables. This problem was solved in an addition 1200 s and produced the requested accuracy. The total time required to obtain the complete solution to the problem was 3775.31 s.

Formulation B was also attempted on the same problem and the performance of the algorithm in this case is summarized in Table 2.

Because of the excessive computer time this case was terminated before completing the mesh refinement process. Inspite of the excessive computer run times this case did suggest a number of important points. In order to analyze the behavior of the algorithm it is useful to examine the performance of the underlying sparse quadratic programming algorithm. Table 3 presents two significant pieces of information for both formulations:

Table 2

GRID	DSC	M	m	n	NGC	NHC	NFE	NRHS	ERRODE	CPU
1	Trp	13 920	111 456	153 079	62	51	4821	67 108 320	0.29E−03	1 081 665

Table 3
SQP algorithm performance

Formulation	A	B
Total number of QP subproblems	29	85
Total number of QP iterations	125	263 340

For Formulation A the boundary conditions were relaxed and the optimal solution did not require the throttle limits (7) to become active. Consequently, determining the correct set of active constraints in the quadratic programming algorithm was relatively simple. This in turn lead to reasonable computation times. In contrast, Formulation B requires that the boundary conditions be satisfied more accurately. This forces the throttle limits to become active, and determining the correct active set is very difficult. In particular, note that 263 340 changes were made in the course of solving the NLP, and since each change requires the solution of a linear system this process is extremely expensive. In fact, this illustrates the combinatorial nature of a nonlinear programming problem with many inequality constraints, and suggests that an interior point optimization algorithm may be desirable for this application.

8. Summary and conclusions

This paper describes the application of a direct transcription method to the solution of a very low-thrust orbit transfer problem. Encouraging results were obtained using a formulation with relaxed boundary conditions. In particular, the resulting sparse nonlinear programming problem involved 416 123 variables and 249 674 constraints. The dynamic model incorporated oblate earth perturbations through J_4 and employed modified equinoctial coordinates to achieve the requested accuracy. However, it is also clear that much remains to be done for effective solution of this problem. It seems clear that an interior point sparse nonlinear programming algorithm may be well suited for this application because of the many nonlinear inequality constraints. It also may be necessary to increase the fidelity of the dynamic simulation to include atmospheric drag, as well as more sophisticated gravitational perturbations (e.g., tesseral harmonics, Sun, Moon, Jupiter, etc.). Furthermore, limitations imposed by the vehicle itself such as thruster duty cycles, and shadow limits may need to be incorporated. Finally, it seems clear that the nonlinearities present in the solution may necessitate alternate discretization techniques and/or a method of averaging.

References

[1] R.H. Battin, An Introduction to the Mathematics and Methods of Astrodynamics, AIAA Education Series, Vol. 1633, American Institute of Aeronautics and Astronautics, Inc., Broadway, New York, 1987.

[2] J.T. Betts, Trajectory optimization using sparse sequential quadratic programming, in: R. Bulirsch, A. Miele, J. Stoer, K.H. Well (Eds.), Optimal control, International Series of Numerical Mathematics, Vol. 117, Birkhäuser, Basel, 1993, pp. 115–128.
[3] J.T. Betts, Using sparse nonlinear programming to compute low thrust orbit transfers, J. Astron. Sci. 41 (1993) 349–371.
[4] J.T. Betts, Issues in the direct transcription of optimal control problems to sparse nonlinear programs, in: R. Bulirsch, Dieter Kraft (Eds.), Computational optimal control, International Series of Numerical Mathematics, Vol. 115, Birkhäuser, Basel, 1994, pp. 3–18.
[5] J.T. Betts, Optimal interplanetary orbit transfers by direct transcription, J. Astronaut. Sci. 42 (1994) 247–268.
[6] J.T. Betts, M.J. Carter, W.P. Huffman, Software for nonlinear optimization, Mathematics and Engineering Analysis Library Report MEA-LR-83 R1, Boeing Information and Support Services, The Boeing Company, Seattle, WA 98124-2207, June 1997.
[7] J.T. Betts, W.P. Huffman, Path constrained trajectory optimization using sparse sequential quadratic programming, AIAA J. Guidance Control Dynamics 16 (1993) 59–68.
[8] R.A. Broucke, P.J. Cefola, On equinoctial orbit elements, Celestial Mech. 5 (1972) 303–310.
[9] T. Edelbaum, L. Sackett, H. Malchow, Optimal low thrust geocentric transfer, in: AIAA 10th Electric Propulsion Conference, AIAA 73-1074, Lake Tahoe, NV, October–November 1973.
[10] J.A. Kechichian, Equinoctial orbit elements: application to optimal transfer problems, in: AIAA/AAS Astrodynamics Specialist Conference, AIAA 90-2976, Portland, OR, August 1990.
[11] J.A. Kechichian, Trajectory optimization with a modified set of equinoctial orbit elements, in: AIAA/AAS Astrodynamics Specialist Conference, AAS 91-524, Durango, CO, August 1991.
[12] M. Walker, B. Ireland, J. Owens, A set of modified equinoctial orbit elements, Celestial Mechanics 36 (1985) 409–419.
[13] T. Yee, J.A. Kechichian, On the dynamic modeling in optimal low-thrust orbit transfer, in: AAS/AIAA spaceflight mechanics meeting, AAS 92-177, Colorado Springs, CO, February 1992.