

Direct Trajectory Optimization by a Chebyshev Pseudospectral Method

Fariba Fahroo* and I. Michael Ross†

Naval Postgraduate School, Monterey, California 93943

We present a Chebyshev pseudospectral method for directly solving a generic Bolza optimal control problem with state and control constraints. This method employs N th-degree Lagrange polynomial approximations for the state and control variables with the values of these variables at the Chebyshev–Gauss–Lobatto (CGL) points as the expansion coefficients. This process yields a nonlinear programming problem (NLP) with the state and control values at the CGL points as unknown NLP parameters. Numerical examples demonstrate that this method yields more accurate results than those obtained from the traditional collocation methods.

I. Introduction

IN recent years, direct solution methods have been used extensively in a variety of trajectory optimization problems.^{1,2} Their advantage over indirect methods, which rely on solving the necessary conditions derived from the Pontryagin et al. minimum principle,³ is that direct method algorithms have better convergence properties. In addition, because the necessary conditions do not have to be derived, the direct methods can be quickly used to solve a number of practical trajectory optimization problems.

Direct methods are based on discretizing the control time history and/or state variable time history at the nodes of discretization, transforming the optimal control problem to a nonlinear programming problem (NLP), and then solving the resulting NLP. In direct collocation schemes, which discretize both the control and state variables, the unknowns are the values of the controls and the states at these nodes. The cost function and the state equations can be expressed in terms of these values. An interpolation scheme is used to obtain the time histories of both the control and the state variables. In the traditional collocation schemes, piecewise-continuous polynomials such as linear or cubic splines are used as the interpolating polynomials over each time segment.^{4,5} Satisfaction of the state differential equations is achieved by using some form of integration scheme over each subinterval, which results in formulation of defects. One of the popular schemes is the Hermite–Simpson rule, which amounts to Simpson’s quadrature rule over the subintervals (see Ref. 4). Recently, Gauss–Lobatto quadrature rules such as trapezoidal, Simpson’s, or higher-order rules with Jacobi polynomials as the interpolant have also been used for collocation (see Refs. 6 and 7).

In literature, the choice of polynomial space for expansion of state and control variables is not limited to piecewise-continuous polynomials and in fact globally orthogonal polynomials such as Legendre and Chebyshev polynomials or Lagrange interpolating polynomials have been used for approximating the control and state variables in nonlinear optimal control problems.^{8–11} In general, there are two ways to construct a polynomial approximation to the solution $y(t)$: One is to use an interpolating polynomial between the values $y(t_j)$ at some points t_j . The other is to use a series expansion in terms of orthogonal polynomials. In Refs. 10 and 11, the latter idea is pursued, and both the states and the control variables are expanded in terms of Chebyshev polynomials with unknown generalized Fourier coefficients.

When the properties of the Chebyshev polynomials are used, the state equations, performance index, and the boundary conditions are converted to algebraic or transcendental equations in terms of these unknown coefficients. In this manner, conversion of the original optimal control problem to an NLP is achieved.

Recently, Elnagar et al.⁸ and Fahroo and Ross^{9,12,13} have used the idea of expansion of the state and control variables in terms of Lagrange polynomials at suitably chosen points, the Legendre–Gauss–Lobatto (LGL) nodes. This idea is the basis for pseudospectral methods, which are used extensively for solving fluid dynamics problems.^{14,15} In approximating optimal control problems, with this choice of node points and properties of the Lagrange polynomials, the state equations and the possible state and control constraints are readily transformed to algebraic equations in terms of values of the state and control variables at the nodes. Differential constraints are imposed only at the LGL points by way of a differentiation operator (matrix). In this sense, the pseudospectral methods are different from other collocation schemes,^{1,2,4} where higher-order numerical integration techniques are used to approximate the differential constraints not only at the nodes but at points in between (midpoints, for example, as in Hermite–Simpson).

In this paper, we present a Chebyshev pseudospectral method and consider differential constraints given in terms of controlled differential inclusions, differential algebraic equations (DAEs), and ordinary differential equations (ODEs). Our method applies without change to all of these types of constraints and, thus, encompasses a wider range of problems than the usual collocation methods. Note that, while revising this paper, the parallel work of Elnagar and Razzaghi¹⁶ came to our attention. We pay homage to their work and cite some key differences. The authors of Ref. 16 also propose a Chebyshev pseudospectral method for solving a variety of optimal control problems. One major difference between their work and ours is the way the integral cost function is approximated. They employ a cell-averaging technique, whereas we use the Clenshaw–Curtis quadrature scheme (see Refs. 17 and 18) for discretizing the cost function. The focus of their paper is mostly on handling the discontinuous functions, which indeed establishes and validates the power and range of applicability of pseudospectral methods to optimal control problems with discontinuous controls. In our effort to show the power of pseudospectral methods, we apply it to a more general class of problems including DAEs and differential inclusions. Thus, in this sense, our paper is complementary to that of Ref. 16. It should be noted that an earlier version of this present paper appeared in Ref. 19.

One notable advantage of using Chebyshev pseudospectral method over other direct collocation methods is the high degree of accuracy that pseudospectral approximations offer.¹⁷ One reason for this is the choice of node points as the Chebyshev–Gauss–Lobatto (CGL) points that result in interpolating polynomials that are closest to the optimal polynomial in the max-norm approximation of any function. Also the derivatives of these interpolating polynomials can be calculated exactly at these nodes by a differentiation matrix.

Received 24 July 2000; revision received 17 May 2001; accepted for publication 22 May 2001. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/02 \$10.00 in correspondence with the CCC.

*Associate Professor, Department of Mathematics, Code MA/FF; ffahroo@nps.navy.mil. Senior Member AIAA.

†Associate Professor, Department of Aeronautics and Astronautics, Code AA/Ro; imross@nps.navy.mil. Associate Fellow AIAA.

Aside from their popularity in engineering applications, Chebyshev polynomials have the added computational advantage in that their corresponding CGL points can be evaluated in closed form. Thus, the Chebyshev pseudospectral method offers the possibility of rapid trajectory optimization because it does not require the use of advanced numerical linear algebra techniques that are necessary for the calculation of LGL points and weights.²⁰ In the three numerical examples presented in this paper, the results clearly show that Chebyshev polynomials are very effective in direct optimization techniques and offer superior results to those of the existing collocation methods.

II. Problem Formulation

Consider the following optimal control problem. Determine the control function $\mathbf{u}(\cdot)$ and the corresponding state trajectory $\mathbf{x}(\cdot)$ that minimize the Bolza cost function:

$$\mathcal{J}[\mathbf{x}(\cdot), \mathbf{u}(\cdot), \tau_f] = \mathcal{M}[\mathbf{x}(\tau_f), \tau_f] + \int_{\tau_0}^{\tau_f} \mathcal{L}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau] d\tau \quad (1)$$

where $\tau \in R$, $\mathbf{x} \in R^n$, $\mathbf{u} \in R^m$, $\mathcal{M}: R^n \times R \rightarrow R$, and $\mathcal{L}: R^n \times R^m \times R \rightarrow R$. We assume the dynamic constraints to be given by the following system of controlled differential inclusions:

$$\mathbf{f}_l \leq \dot{\mathbf{x}}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau), \tau \leq \mathbf{f}_u, \quad \tau \in [\tau_0, \tau_f] \quad (2)$$

where $\mathbf{f}: R^n \times R^n \times R^m \times R \rightarrow R^n$. This formulation generalizes and unifies much of the previous apparently distinct formulations.^{7,21} For example, if we set the lower and upper bounds equal to zero, we get a controlled DAE,

$$\mathbf{f}[\dot{\mathbf{x}}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau), \tau] = \mathbf{0}, \quad \tau \in [\tau_0, \tau_f] \quad (3)$$

If, in addition, we assume that the Jacobian $\partial \mathbf{f} / \partial \dot{\mathbf{x}}$ is nonsingular, then Eq. (3) may be reduced to a controlled ODE,

$$\dot{\mathbf{x}}(\tau) = \mathbf{f}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau], \quad \tau \in [\tau_0, \tau_f] \quad (4)$$

Furthermore, for $\mathbf{u} \in \mathcal{U}$, the following hodograph transformation may be used to transform the controlled ODE to an ordinary differential inclusion:

$$\dot{\mathbf{x}}(\tau) \in \{\mathbf{y}(\tau) \mid \mathbf{y}(\tau) = \mathbf{f}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau], \mathbf{u} \in \mathcal{U}\}, \quad \tau \in [\tau_0, \tau_f] \quad (5)$$

We elaborate on these formulations further in the numerical section. In addition, we assume the boundary conditions to be given by the relations

$$\psi_l \leq \psi[\mathbf{x}(\tau_0), \mathbf{x}(\tau_f), (\tau_f - \tau_0)] \leq \psi_u \quad (6)$$

where $\psi: R^n \times R^n \times R \rightarrow R^p$ and ψ_l and $\psi_u \in R^p$ are constant vectors representing the lower and upper bounds of these inequalities. Possible state and control constraints are formulated as mixed constraints,

$$\mathbf{g}_l \leq \mathbf{g}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau] \leq \mathbf{g}_u, \quad \mathbf{g}: R^n \times R^m \times R \rightarrow R^r \quad (7)$$

where \mathbf{g}_l and $\mathbf{g}_u \in R^r$ denote the lower and upper bounds of \mathbf{g} . Clearly, an equality constraint may be obtained by simply setting the lower and upper bounds to be equal. Thus, this formulation is not only more general, but is also more elegant than the one that separately identifies equality and inequality constraints on the states and control variables.

In the numerical approximation of the optimal control problem, because the node points (CGL) lie in the computational interval $[-1, 1]$, the problem is transformed to this interval by the linear transformation for $t \in [t_0, t_N] = [-1, 1]$:

$$\tau = [(\tau_f - \tau_0)t + (\tau_f + \tau_0)]/2 \quad (8)$$

resulting in the following reformulation of Eqs. (1), (2), (6), and (7):

$$\mathcal{J}[\mathbf{x}(\cdot), \mathbf{u}(\cdot), \tau_f] = \mathcal{M}[\mathbf{x}(1), \tau_f] + \left(\frac{\tau_f - \tau_0}{2} \right) \int_{-1}^1 \mathcal{L}[\mathbf{x}(t), \mathbf{u}(t), \tau(t)] dt \quad (9)$$

$$\mathbf{f}_l \leq \mathbf{f} \left[\frac{2}{\tau_f - \tau_0} \dot{\mathbf{x}}(t), \mathbf{x}(t), \mathbf{u}(t), \tau(t) \right] \leq \mathbf{f}_u \quad (10)$$

$$\psi_l \leq \psi[\mathbf{x}(-1), \mathbf{x}(1), \tau_f - \tau_0] \leq \psi_u \quad (11)$$

$$\mathbf{g}_l \leq \mathbf{g}[\mathbf{u}(t), \mathbf{x}(t), \tau(t)] \leq \mathbf{g}_u \quad (12)$$

To reflect transformation of the time domain, one must use different symbols. However, for the purpose of brevity we retain these symbols. Thus, in this context, one must view $\mathbf{x}(t)$, for example, as

$$\mathbf{x}[\tau(t)] = \mathbf{x}\{[(\tau_f - \tau_0)t + (\tau_f + \tau_0)]/2\}$$

III. Chebyshev Pseudospectral Method

The Chebyshev pseudospectral method is one special case of a more general class of spectral methods.¹⁵ The basic formulation of these methods involves two essential steps: One is to choose a finite-dimensional space (usually a polynomial space) from which an approximation to the solution of the differential equation is made. The other step is to choose a projection operator, which imposes the differential equation in the finite-dimensional space. One important feature of spectral methods, which distinguishes them from finite element or finite difference methods, is that the underlying polynomial space is spanned by orthogonal polynomials that are infinitely differentiable global functions. Among examples of these orthogonal polynomials are Legendre and Chebyshev polynomials, which are orthogonal on the interval $[-1, 1]$, with respect to an appropriate weight function [$w(t) = 1$ for Legendre polynomials, and $w(t) = 1/\sqrt{1-t^2}$ for Chebyshev polynomials of the first kind]. In pseudospectral methods, the functions are expanded in terms of interpolating polynomials so that the expansion coefficients are the values of the function at the node points. Because an arbitrary choice of node points can give very poor results in interpolation, different Gauss quadrature points are chosen to give the best accuracy in interpolation of a function. The derivatives of these interpolating polynomials at these node points are given exactly by a differentiation matrix. These two important aspects (choice of interpolating polynomials as the trial functions and Gauss quadrature points) of pseudospectral methods separate them from the other collocation methods. In the Chebyshev pseudospectral method, the interpolation points are given in closed form,

$$t_k = \cos(\pi k/N), \quad k = 0, \dots, N \quad (13)$$

These points lie in the interval $[-1, 1]$ and are the extrema of the N th-order Chebyshev polynomial $T_N(t)$. The j th-order Chebyshev polynomial is expressed by

$$T_j(t) = \cos(j \cos^{-1} t), \quad j = 0, \dots, N \quad (14)$$

which yields

$$T_j(t_k) = \cos(\pi k j/N)$$

Thus, $T_0(t) = 1$, $T_1(t) = t$, $T_2(t) = 2t^2 - 1$, $T_3(t) = 4t^3 - 3t$, and so on. These node points have the distribution property that they cluster around the endpoints of the interval. This clustering results in avoidance of the Runge phenomenon, which is the divergence at the endpoints of interpolation on equispaced interpolation points. In fact, it has been shown that interpolation at the CGL nodes gives the closest result to the optimal polynomial approximation to a given function. This slight error from optimality is given by the Lebesgue constant, which is the smallest for the CGL nodes distribution (see Ref. 17).

For approximating the continuous equations, we seek polynomial approximations of the form

$$\mathbf{x}^N(t) = \sum_{j=0}^N \mathbf{x}_j \phi_j(t) \quad (15)$$

$$\mathbf{u}^N(t) = \sum_{j=0}^N \mathbf{u}_j \phi_j(t) \quad (16)$$

where, for $j = 0, 1, \dots, N$,

$$\phi_j(t) = \frac{(-1)^{j+1} (1-t^2) \dot{T}_N(t)}{N^2 c_j (t-t_j)}$$

are the Lagrange interpolating polynomials of order N (see the Appendix for a derivation), with

$$c_j = \begin{cases} 2, & j = 0, N \\ 1, & 1 \leq j \leq N-1 \end{cases}$$

From its interpolating property, the Lagrange polynomials satisfy the condition

$$\phi_j(t_k) = \delta_{jk} = \begin{cases} 1, & \text{if } j = k \\ 0, & \text{if } j \neq k \end{cases}$$

Hence, it follows that

$$\mathbf{x}^N(t_k) = \mathbf{x}_k, \quad \mathbf{u}^N(t_k) = \mathbf{u}_k \quad (17)$$

In other words, the node points are the interpolating points. To express the derivative $\dot{\mathbf{x}}^N(t)$ in terms of $\mathbf{x}^N(t)$ at the node points t_k , we differentiate Eq. (15), which results in a matrix multiplication of the following form^{14,15}:

$$\mathbf{d}_k = \dot{\mathbf{x}}^N(t_k) = \sum_{j=0}^N \mathbf{x}_j \dot{\phi}_j(t_k) = \sum_{j=0}^N D_{kj} \mathbf{x}_j \quad (18)$$

where D_{kj} are entries of the $(N+1) \times (N+1)$ differentiation matrix \mathbf{D}

$$\mathbf{D} := [D_{kj}] := \begin{cases} (c_k/c_j)[(-1)^{j+k}/(t_k-t_j)], & j \neq k \\ -t_k/2(1-t_k^2), & 1 \leq j = k \leq N-1 \\ (2N^2+1)/6, & j = k = 0 \\ -(2N^2+1)/6, & j = k = N \end{cases} \quad (19)$$

Multiplication by this matrix, transforms a vector of the state variables at the CGL points to the vector of approximate derivatives at these points.

A. Modification of the Chebyshev Differentiation Matrix

The definitions of the CGL points and the given differentiation matrix are the standard ones in the literature on spectral methods.^{15,17} However, in our implementation of this method for discretizing optimal control problems, it is more convenient to modify these definitions. The CGL points as defined by Eq. (13) are from 1 to -1 , that is, the initial point is 1 and the final point is -1 . For trajectory optimization problems, it is more convenient to take the initial point at -1 and the final point at 1. This sorting modifies the definition of the differentiation matrix in the following sense: Using the definition of Eq. (19), we will show that the new matrix with the sorted points, $\tilde{\mathbf{D}}$ is of the opposite sign of the matrix with the unsorted points, \mathbf{D} . To see this, note that in the definition of matrix \mathbf{D} , the points $t_0 = 1, t_1, \dots, t_{N-1}, t_N = -1$ are used, and for matrix $\tilde{\mathbf{D}}$, the sorted points $\tilde{t}_0 = -1, \dots, \tilde{t}_N = 1$, are used, that is, $t_{N-k} = \tilde{t}_k, k = 0, \dots, N$. For the CGL points, we have the following symmetry relationship:

$$t_k = -t_{N-k} = -\tilde{t}_k \quad (20)$$

which yields

$$t_k - t_j = -(t_{N-k} - t_{N-j}) = -(\tilde{t}_k - \tilde{t}_j)$$

For the elements D_{00} and D_{NN} , it can be easily shown that

$$\tilde{D}_{00} = D_{NN} = -D_{00}, \quad \tilde{D}_{NN} = D_{00} = -D_{NN}$$

For the other diagonal elements, $D_{kk} = -t_k/2(1-t_k^2)$, from Eq. (20) we have

$$D_{kk} = \frac{t_{N-k}}{2(1-t_{N-k}^2)} = \frac{\tilde{t}_k}{2(1-\tilde{t}_k^2)} = -\tilde{D}_{kk} \quad (21)$$

For the off-diagonal elements, from Eq. (20) we can easily see that

$$\begin{aligned} D_{kj} &= \frac{c_k (-1)^{j+k}}{c_j (t_k - t_j)} = \left(\frac{c_k}{c_j} \right) \frac{(-1)^{j+k}}{(-t_{N-k}) - (-t_{N-j})} \\ &= \left(\frac{c_k}{c_j} \right) \frac{(-1)^{j+k}}{-(\tilde{t}_k - \tilde{t}_j)} = -\tilde{D}_{kj} \end{aligned} \quad (22)$$

Therefore, we have

$$\tilde{\mathbf{D}}_{kj} = -D_{kj} \quad (23)$$

Additionally, the original matrix \mathbf{D} has another symmetry property, which can be easily shown by the preceding relationships²²:

$$D_{kj} = -D_{N-k, N-j} \quad (24)$$

Using Eqs. (23) and (24) and denoting the values of the approximate states at the sorted CGL points by

$$\mathbf{x}_{N-k} = \mathbf{x}^N(t_{N-k}) = \tilde{\mathbf{x}}^N(\tilde{t}_k) = \tilde{\mathbf{x}}_k$$

we have

$$\dot{\mathbf{x}}^N(t_k) = \sum_{j=0}^N D_{kj} \mathbf{x}^N(t_j) = \sum_{j=0}^N \tilde{D}_{N-k, N-j} \tilde{\mathbf{x}}^N(\tilde{t}_{N-j}) = \dot{\tilde{\mathbf{x}}}^N(\tilde{t}_{N-k}) \quad (25)$$

or more succinctly, we have

$$\mathbf{d}_k = \dot{\mathbf{x}}^N(t_k) = \dot{\tilde{\mathbf{x}}}^N(\tilde{t}_{N-k}) \quad (26)$$

This result shows that the sorting of the CGL points does not affect the underlying definitions and formulation of the pseudospectral method. One can now proceed with the formulation of the NLP with the sorted CGL points. In the formulation to come, we drop the tilde notation with the understanding that the sorted points are used as the node points in the differentiation matrix and elsewhere.

B. NLP Formulation

The state equations and the initial and terminal state conditions are discretized by first substituting Eqs. (15), (16), and (18) in Eq. (10) and collocating at the CGL nodes t_k . The state equations are transformed into the following algebraic inequalities:

$$\mathbf{f}_l \leq \mathbf{f}[2/(\tau_f - \tau_0) \mathbf{d}_k, \mathbf{x}_k, \mathbf{u}_k, \tau_k] \leq \mathbf{f}_u, \quad k = 0, \dots, N$$

where \mathbf{d}_k is as defined in Eq. (26). The initial and terminal state conditions are given by

$$\psi_l \leq \psi[\mathbf{x}_0, \mathbf{x}_N, (\tau_f - \tau_0)] \leq \psi_u$$

The control inequality constraints are approximated by

$$\mathbf{g}_l \leq \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) \leq \mathbf{g}_u$$

Next, the cost function in Eq. (9) is discretized: We use the Clenshaw-Curtis quadrature scheme (see Ref. 17, chapter 12) to discretize the integral part of the cost function to a finite sum. The idea is based on finding optimal weights w_k for a given set of CGL points t_k such that the approximation

$$\int_{-1}^1 p(t) dt = \sum_{k=0}^N p(t_k) w_k, \quad p(t) \in \mathcal{P}_N$$

for \mathcal{P}_N , the space of polynomials of degree $\leq N$, is exact. For N even, the weights are^{17,18}

$$w_0 = w_N = \frac{1}{N^2 - 1} \quad (27)$$

$$w_s = w_{N-s} = \frac{4}{N} \sum_{j=0}^{N/2''} \frac{1}{1-4j^2} \cos \frac{2\pi js}{N}, \quad s = 1, \dots, \frac{N}{2} \quad (28)$$

For N odd, the weights are given by¹⁷

$$w_0 = w_N = \frac{1}{N^2} \quad (29)$$

$$w_s = w_{N-s} = \frac{4}{N} \sum_{j=0}^{(N-1)/2''} \frac{1}{1-4j^2} \cos \frac{2\pi js}{N} \quad (30)$$

$$s = 1, \dots, \frac{N-1}{2}$$

In the preceding summations, the double prime means that the first and the last elements have to be halved.

Therefore, the cost function in terms of the coefficients

$$\mathbf{X} = (\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N), \quad \mathbf{U} = (\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_N)$$

can be discretized as

$$\mathcal{J}(\mathbf{X}, \mathbf{U}, \tau_f) \approx \mathcal{M}(\mathbf{x}_N, \tau_f) + \frac{\tau_f - \tau_0}{2} \sum_{k=0}^N \mathcal{L}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) w_k \quad (31)$$

Thus, the optimal control problem in Eqs. (9–12) is approximated by the following nonlinear optimization problem: Find coefficients

$$\mathbf{X} = (\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N), \quad \mathbf{U} = (\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_N)$$

and possibly the final time τ_f to minimize

$$\mathcal{J}^N(\mathbf{X}, \mathbf{U}, \tau_f) = \mathcal{M}(\mathbf{x}_N, \tau_f) + \frac{\tau_f - \tau_0}{2} \sum_{k=0}^N \mathcal{L}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) w_k \quad (32)$$

subject to

$$\mathbf{f}_l \leq \mathbf{f}[2/(\tau_f - \tau_0)\mathbf{d}_k, \mathbf{x}_k, \mathbf{u}_k, \tau_k] \leq \mathbf{f}_u, \quad k = 0, \dots, N \quad (33)$$

$$\mathbf{g}_l \leq \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) \leq \mathbf{g}_u, \quad k = 0, \dots, N \quad (34)$$

$$\psi_l \leq \psi[\mathbf{x}_0, \mathbf{x}_N, (\tau_f - \tau_0)] \leq \psi_u \quad (35)$$

From the preceding equations, one can see the simplicity of the method, which retains much of the structure of the continuous problem.

IV. Numerical Examples

In this section, we present three numerical examples that are intended to illustrate separate points. In the first example, we choose the benchmark brachistochrone problem discussed in Ref. 7 and demonstrate the accuracy of our technique. In the second example, we reformulate the brachistochrone problem as a DAE and show how our method remains unchanged. In the third and final problem, we solve the moon-landing problem²³ formulated as a differential inclusion by a hodograph transformation.

A. Example 1: Brachistochrone as an ODE

In this well-known problem, the control problem is formulated as finding the shape of a wire so that a bead sliding on the wire will reach a given horizontal displacement in minimum time. No frictional forces are considered, and the gravity force is uniform. The problem is then to minimize τ_f subject to the equations

$$\dot{x} = \sqrt{2gy} \cos \theta \quad (36)$$

$$\dot{y} = \sqrt{2gy} \sin \theta \quad (37)$$

with boundary conditions

$$x(0) = y(0) = 0, \quad x(\tau_f) = 0.5 \quad (38)$$

The control, angle θ , is the slope of the wire as a function of time.

Table 1 Comparison of cost functions for various methods and formulations for $\hat{N} = 11$

Method	J_i	$ J_i - J_{\text{ana}} $	N_p
Analytic solution	1.253314		
Collocation (Simpson)	1.253005	0.000309	34
Collocation (fifth-degree GL)	1.253183	0.000131	84
Pseudospectral (CGL)	1.253309	0.000005	34
Pseudospectral (DAE)	1.253366	0.000052	23

Let X , Y , and Θ represent the solution vector to this problem at the CGL nodes. Then the NLP for this problem is simply to minimize τ_f subject to the equality constraints

$$(2/\tau_f)D * X - \sqrt{2gY} \cos \Theta = \mathbf{0} \quad (39)$$

$$(2/\tau_f)D * Y - \sqrt{2gY} \sin \Theta = \mathbf{0} \quad (40)$$

and

$$X(0) = Y(0) = 0, \quad X(N) - 0.5 = 0$$

All operations in the preceding equations are array operations, that is, term-by-term, whereas the asterisk is used to denote matrix multiplication. With this notation, the striking resemblance of the NLP to the original problem is immediately apparent. The analytic solutions to this problem are the equations of a cycloid,

$$x(\tau) = (g\tau_f/\pi)\{\tau - (\tau_f/\pi) \sin[\pi(1 - \tau/\tau_f)]\} \quad (41)$$

$$y(\tau) = (2g\tau_f^2/\pi^2) \cos^2[(\pi/2)(1 - \tau/\tau_f)] \quad (42)$$

The optimal control (angle) is given by the following expression:

$$\theta(\tau) = (\pi/2)(1 - \tau/\tau_f) \quad (43)$$

From $x(\tau_f) = 0.5$, Eq. (41), and $g = 1$, the minimum time calculated to six decimal places is $\tau_f = \sqrt{(0.5\pi)} = 1.253314$.

The Chebyshev pseudospectral method was used for discretization of the problem and NPSOL²⁴ was utilized as the NLP solver. The implementation was performed in MATLAB[®] via the use of MEX files. In Table 1, we compare the minimum cost function obtained from our method to the ones from the Simpson collocation method and a fifth-degree Gauss–Lobatto method reported in Ref. 7. All reported results are for $\hat{N} = 11$ nodes, where $\hat{N} = N + 1$. The variable N_p in the last column of Table 1 stands for the total number of NLP variables, which includes the free final time as well. Note that for the fifth-degree Gauss–Lobatto (GL) collocation method (done the traditional way), the number of NLP variables are significantly higher than the pseudospectral method for the same number of nodes. In any case, it is evident that even for a low number degree of discretization (and number of NLP variables), the Chebyshev pseudospectral method gives results superior to either of the other collocation methods.

In Figs. 1 and 2, we show the time histories of the state and control variables against the analytic solutions for $\hat{N} = 11$. The solid lines represent the analytic solutions and * is for the solutions from the numerical method. The graphs clearly demonstrate the accuracy of the results from the Chebyshev pseudospectral method. Note that from Ref. 7 the collocation solutions are generally accurate to $\mathcal{O}(10^{-4})$, which is about the same order of accuracy for the Chebyshev pseudospectral method for this number of nodes. However, the order of accuracy in the control for the other collocation methods is about $\mathcal{O}(10^{-2})$ [with a maximum pointwise error at the first node of about $\mathcal{O}(10^{-1})$], whereas for our method the maximum pointwise error is of order $\mathcal{O}(10^{-3})$.

B. Example 2: Brachistochrone as a DAE

Here we eliminate the control variable θ and rewrite Eqs. (36) and (37) in terms of a single fully implicit DAE [see Eq. (3)]:

$$\dot{x}^2 + \dot{y}^2 = 2gy \quad (44)$$

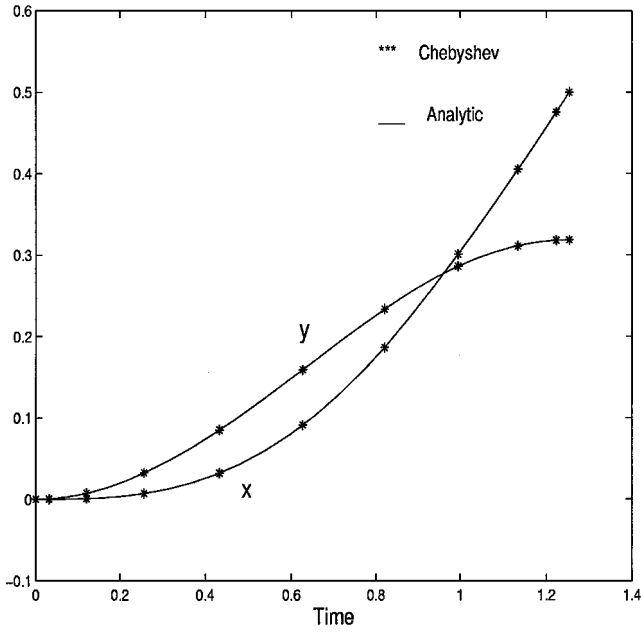


Fig. 1 Time history of x and y for the brachistochrone problem.

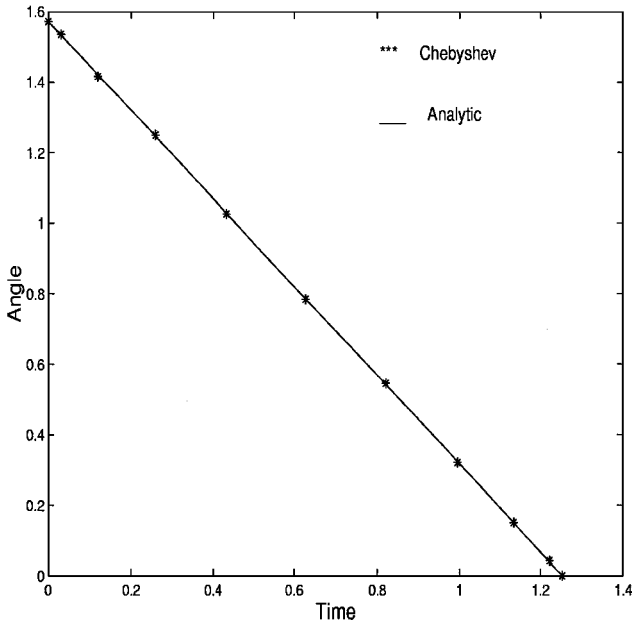


Fig. 2 Time history of control θ for the brachistochrone problem.

Thus, there is a reduction in the number of NLP variables by \hat{N} , the number of CGL points. The DAE constraint is discretized at the CGL points resulting in

$$(4/\tau_f^2)(D * X)^2 + (4/\tau_f^2)(D * Y)^2 - 2gY = \mathbf{0} \quad (45)$$

From Table 1, we see that, although the degree of accuracy for the cost function is not as high as the original formulation, it is still better than the results from the other collocation methods. The accuracy of the solutions is still of the order of $\mathcal{O}(10^{-4})$. Therefore, the pseudospectral method can handle formulation of the problem as an DAE without sacrificing accuracy.

C. Example 3: Moon-Landing Problem as a Differential Inclusion

The control problem is formulated as maximizing the final mass and, hence, minimizing

$$J = -m(\tau_f) \quad (46)$$

subject to the equations of motion

$$\frac{dh}{d\tau} = v \quad (47)$$

$$\frac{dv}{d\tau} = -g + \frac{T}{m} \quad (48)$$

$$\frac{dm}{d\tau} = -\frac{T}{I_{sp}g} \quad (49)$$

where the state variables h , v , and m are altitude, speed, and mass, respectively. The control is provided by the thrust T , which is bounded by

$$0 \leq T \leq T_{\max}$$

The other parameters in the problem are g , the gravity of moon (or any planet without an atmosphere), and I_{sp} , the specific impulse of the propellant. Given any set of initial conditions h_0 , v_0 , and m_0 , the normalized parameters for the problem were arbitrarily chosen as

$$\begin{aligned} T_{\max}/m_0g &= 1.1, & I_{sp}g/v_0 &= 1, & h(0)/h_0 &= 0.5 \\ v(0)/v_0 &= -0.05, & m(0)/m_0 &= 1 \end{aligned}$$

Therefore, we have the following normalized initial conditions:

$$h(0) = 0.5, \quad v(0) = -0.05, \quad m(0) = 1.0 \quad (50)$$

For soft landing, we must have

$$h(\tau_f) = 0, \quad v(\tau_f) = 0 \quad (51)$$

In addition, for a physically meaningful trajectory, we must have

$$m(\tau_f) > 0 \quad (52)$$

The discrete differential inclusion version of these equations for the variables $H = [h(t_0), h(t_1), \dots, h(t_N)]^T$, $V = [v(t_0), v(t_1), \dots, v(t_N)]^T$, and $M = [m(t_0), m(t_1), \dots, m(t_N)]^T$ can be formulated as

$$(2/\tau_f)D * H - V = \mathbf{0} \quad (53)$$

$$M[(2/\tau_f)D * V + g] + I_{sp}g[(2/\tau_f)D * M] = \mathbf{0} \quad (54)$$

$$\mathbf{0} \leq -I_{sp}g[(2/\tau_f)D * M] \leq T_{\max} \quad (55)$$

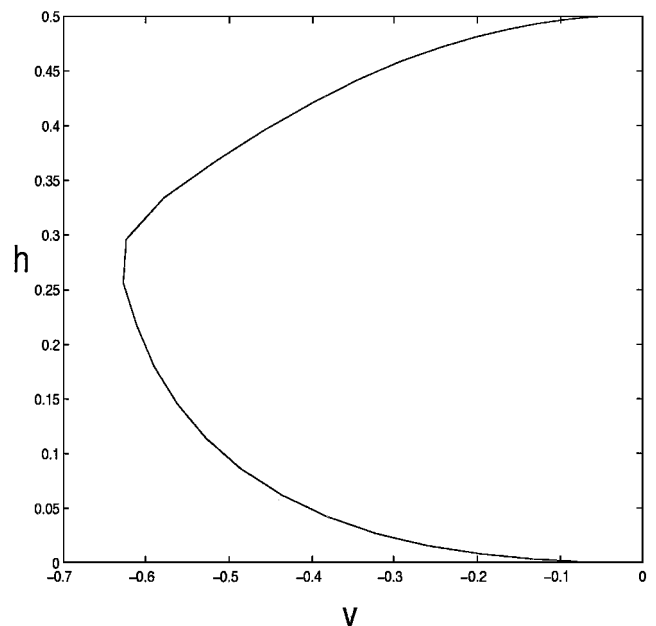


Fig. 3 Phase portrait of h vs v for the moon-landing problem.

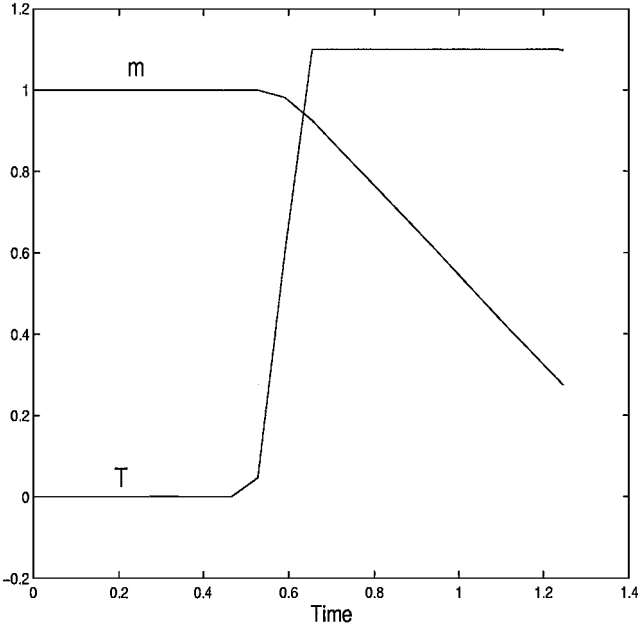


Fig. 4 Time history of m and control T for the moon-landing problem.

Figure 3 shows the phase portrait projected into the $v-h$ space for 32 CGL points. This problem has at most one switch in the control variable (Ref. 22), and the results are shown in Fig. 4. The control was calculated from Eq. (49). It is clear that the method has adequately captured the optimal bang-bang structure of the control with one switch.

V. Conclusions

The simplicity, efficiency, and versatility of the Chebyshev pseudospectral method allows one to perform rapid and accurate trajectory optimization. A low degree of discretization appears to be sufficient to generate good results. Because the CGL points and weights can be quickly evaluated, it suggests that the technique has a high potential for use in optimal guidance algorithms that require a corrective maneuver from the perturbed trajectory. In any case, reference optimal paths can be easily generated, and the numerical examples indicate that the converged solutions are indeed optimal. Further tests and analysis are necessary to investigate the stability and accuracy of the method.

Appendix: Derivation of Lagrange Polynomials for CGL Nodes

We derive the following expression for the Lagrange polynomials of order N interpolating a function at the CGL nodes:

$$\phi_k(t) = \frac{(-1)^{k+1} (1-t^2) \dot{T}_N(t)}{N^2 c_k (t-t_k)}$$

where

$$c_k = \begin{cases} 2, & k = 0, N \\ 1, & 1 \leq k \leq N-1 \end{cases} \tag{A1}$$

$T_N(t)$ is the Chebyshev polynomial of degree N , and t_k are the CGL points over the interval $[-1, 1]$:

$$t_k = \cos(\pi k/N), \quad k = 0, \dots, N$$

Differentiation is denoted by a prime.

The general expression for a Lagrange interpolating polynomial at N distinct nodes t_k is

$$\phi_k(t) = \prod_{\substack{m=0 \\ m \neq k}}^N \frac{(t-t_m)}{(t_k-t_m)}, \quad k = 0, \dots, N$$

Defining

$$w(t) = \prod_{m=0}^N (t-t_m)$$

we easily obtain

$$\begin{aligned} w'(t_k) &= (t_k-t_0)(t_k-t_1), \dots, (t_k-t_{k-1})(t_k-t_{k+1}), \dots, (t_k-t_N) \\ &= \prod_{\substack{m=0 \\ m \neq k}}^N (t_k-t_m) \end{aligned}$$

Therefore, we have

$$\phi_k(t) = \frac{w(t)}{(t-t_k) w'(t_k)}$$

From the definition of the CGL points that are the extrema of T_N or the zeros of $T'_N(t)$, and $t_0 = 1, t_N = -1$, we can write

$$T'_N(t) = (t-t_1)(t-t_2), \dots, (t-t_{N-1})$$

$$\begin{aligned} w(t) &= \prod_{m=0}^N (t-t_m) = (t-t_0)(t-t_1), \dots, (t-t_{N-1})(t-t_N) \\ &= (t^2-1)T'_N(t) \end{aligned}$$

For $w'(t_k)$, we use that the Chebyshev polynomials are the eigenfunctions of

$$[(1-t^2)T''_N] - tT'_N + N^2T_N(t) = 0$$

From this expression and the equation for $w(t)$, we have

$$w'(t) = [-(1-t^2)T'_N]' = N^2T_N(t) + tT'_N \tag{A2}$$

Evaluating the preceding expression at $t = t_k, k \neq 0, N$, and using that $T'_N(t_k) = 0$, for $k = 1, \dots, N-1$, and $T_N(t_k) = \cos(\pi k N/N) = (-1)^k, k = 1, \dots, N-1$, we have

$$w'(t_k) = (-1)^k N^2, \quad k = 1, \dots, N-1$$

For $k = 0, N$, we use the following identities:

$$\begin{aligned} T_N(t_0 = 1) &= 1, & T'_N(1) &= N^2 \\ T_N(t_N = -1) &= (-1)^N, & T'_N(-1) &= (-1)^N N^2 \end{aligned}$$

which give us from Eq. (A2)

$$w'(1) = 2N^2, \quad w'(-1) = 2(-1)^N N^2$$

Finally, we have

$$\begin{aligned} \phi_k(t) &= \frac{w(t)}{t-t_k} \frac{1}{w'(t_k)} = \frac{(t^2-1)T'_N(t)}{(t-t_k)} \frac{(-1)^k}{N^2 c_k} \\ &= \frac{(1-t^2)T'_N(t)}{(t-t_k)} \frac{(-1)^{k+1}}{N^2 c_k} \end{aligned}$$

where c_k is as defined in Eq. (A1).

References

- 1 Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193-207.
- 2 Hull, D. G., "Conversion of Optimal Control Problems into Parameter Optimization Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 57-60.
- 3 Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mischenko, E. F., *The Mathematical Theory of Optimal Processes*, Interscience, New York, 1962, pp. 1-114.
- 4 Hargraves, C. R., and Paris, S. W., "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 4, 1987, pp. 338-342.

Downloaded by BIBLIOTHEK DER TU MUENCHEN on September 30, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2014-4862

⁵von Stryk, O., "Numerical Solution of Optimal Control Problems by Direct Collocation," *Optimal Control*, edited by R. Bulirsch, A. Miele, and J. Stoer, Birkhäuser Verlag, Basel, Switzerland, 1993, pp. 129–143.

⁶Herman, A. H., and Conway, B. A., "Direct Optimization Using Collocation Based on High-Order Gauss-Lobatto Quadrature Rules," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 3, 1996, pp. 592–599.

⁷Conway, B. A., and Larson, K. M., "Collocation Versus Differential Inclusion in Direct Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 5, 1998, pp. 780–785.

⁸Elnagar, J., Kazemi, M. A., and Razzaghi, M., "The Pseudospectral Legendre Method for Discretizing Optimal Control Problems," *IEEE Transactions on Automatic Control*, Vol. 40, No. 10, 1995, pp. 1793–1796.

⁹Fahroo, F., and Ross, I. M., "Costate Estimation by a Legendre Pseudospectral Method," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 2, 2001, pp. 270–277.

¹⁰Vlassenbroeck, J., and Van Doreen, R., "A Chebyshev Technique for Solving Nonlinear Optimal Control Problems," *IEEE Transaction on Automatic Control*, Vol. 33, No. 4, 1988, pp. 333–340.

¹¹Vlassenbroeck, J., "A Chebyshev Polynomial Method for Optimal Control with State Constraints," *Automatica*, Vol. 24, No. 4, 1988, pp. 499–506.

¹²Fahroo, F., and Ross, I. M., "Second Look at Approximating Differential Inclusions," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1, 2001, pp. 131–133.

¹³Fahroo, F., and Ross, I. M., "A Spectral Patching Method for Direct Trajectory Optimization," American Astronautical Society, Paper AAS 00-260, 20–21 March 2000; also *Journal of Astronautical Sciences*, Vol. 48, No. 2/3, 2000, pp. 269–286.

¹⁴Canuto, C., Hussaini, M. Y., Quarteroni, A., and Zang, T. A., *Spectral*

Methods in Fluid Dynamics, Springer-Verlag, New York, 1988.

¹⁵Gottlieb, D., Hussaini, M. Y., and Orszag, S. A., "Theory and Applications of Spectral Methods," *Spectral Methods for PDE's*, edited by R. G. Voigt, D. Gottlieb, and M. Y. Hussaini, Society for Industrial and Applied Mathematics, Philadelphia, 1984, pp. 1–54.

¹⁶Elnagar, G., and Kazemi, M. A., "Pseudospectral Chebyshev Optimal Control of Constrained Nonlinear Dynamical Systems," *Computational Optimization and Applications*, Vol. 11, 1998, pp. 195–217.

¹⁷Trefethen, L. N., *Spectral Methods in MATLAB*, Society for Industrial and Applied Mathematics, Philadelphia, 2000.

¹⁸Davis, P. J., and Rabinowitz, P., *Methods of Numerical Integration*, Academic, New York, 1977, p. 68.

¹⁹Fahroo, F., and Ross, I. M., "Direct Trajectory Optimization by Chebyshev Pseudospectral Method," *Proceedings of the American Control Conference*, Inst. of Electrical and Electronics Engineers, New York, 2000, pp. 3860–3864.

²⁰Golub, G. H., "Some Modified Matrix Eigenvalue Problems," *SIAM Review*, Vol. 15, No. 2, 1973, pp. 318–334.

²¹Brenan, K. E., "Differential-Algebraic Equations Issues in the Direct Transcription of Path Constrained Optimal Control Problems," *Annals of Numerical Mathematics*, Vol. 1, 1994, pp. 247–263.

²²Solomonoff, A., "A Fast Algorithm for Spectral Differentiation," *Journal of Computational Physics*, Vol. 98, 1992, pp. 174–177.

²³Knowles, G., *An Introduction to Applied Optimal Control*, Academic, New York, 1981, pp. 50–55.

²⁴Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. A., "User's Guide to NPSOL 5.0: A Fortran Package for Nonlinear Programming," Stanford Optimization Lab., Technical Rept. SOL 86-1, Stanford Univ., Stanford, CA, July 1998.