Journal of Guidance, Control, and Dynamics
Vol. 31, No. 3, May–June 2008

Spectral Algorithm for Pseudospectral Methods in Optimal Control

Qi Gong*
University of Texas at San Antonio, San Antonio, Texas 78249
and
Fariba Fahroo† and I. Michael Ross‡
Naval Postgraduate School, Monterey, California 93943

DOI: 10.2514/1.32908

Recent convergence results with pseudospectral methods are exploited to design a robust, multigrid, spectral algorithm for computing optimal controls. The design of the algorithm is based on using the pseudospectral differentiation matrix to locate switches, kinks, corners, and other discontinuities that are typical when solving practical optimal control problems. The concept of pseudospectral knots and Gaussian quadrature rules are used to generate a natural spectral mesh that is dense near the points of interest. Several stopping criteria are developed based on new error-estimation formulas and Jackson's theorem. The sequence is terminated when all of the convergence criteria are satisfied. Numerical examples demonstrate the key concepts proposed in the design of the spectral algorithm. Although a vast number of theoretical and algorithmic issues still remain open, this paper advances pseudospectral methods along several new directions and outlines the current theoretical pitfalls in computation and control.

I. Introduction

OVER the last decade, pseudospectral (PS) methods for optimal control have moved rapidly from theory [1–3] to practice [4–12] to flight application [13,14]. The flight application onboard the International Space Station marks one of the many milestones in the recent developments [15–18] in PS methods for optimal control theory. Such advancements have brought to the forefront many major questions regarding the practice of optimal control theory that were once only in the realm of theory [19,20]. For example, questions of existence, uniqueness, and convergence of a solution, which were once largely a purview of theoreticians, are now major issues in computational optimal control because they address many practical problems such as those described in the following:

1) If a numerical solution to an optimal control problem is not obtained, is it because the problem does not have a solution? Or is the algorithm failing? Or is the computer code incorrect? These questions lie at the intersection of the existence, convergence of the discretization, convergence of the algorithm, and verification and validation (V&V) techniques in computational engineering [15,21–26].

2) If a (computer) solution is obtained, is the solution (in the discretized space) feasible in the realm of optimal control (i.e., in function space)? This is an issue related to discretization error or the convergence of the discretization (which is not the same as the convergence of the algorithm [15,16,18,22,27,28]).

3) If a feasible solution is obtained, is it an optimal solution? This issue is related to the convergence question as well as to V&V techniques in optimal control in terms of the necessary and sufficiency conditions for optimality [15,25,29,30]. These conditions are not the same as the optimality conditions for the discretized problem because, for example, the concept of conjugate points is nonexistent in a discretized problem [25].

The preceding *open questions* articulate the forefront of research in modern optimal control theory and practice. The rapidly narrowing gap between theory and practice allows us to exploit recent existence and convergence results for PS methods [15,16] to lay down the foundations for a mesh-refinement algorithm. The goal of a mesh-refinement algorithm is to design the number of nodes and the distribution of nodes needed for practical convergence. A conceptual algorithm under this framework is to design a sequence of discrete-time problems such that a solution to the final problem is sufficiently close (in some metric) to a solution of the continuous-time optimal control problem. Such algorithmic implementations have been pioneered by Betts et al. [21,26,31] for a large number of Runge–Kutta discretization methods, leading to the professional software SOCS [32]. As noted earlier and further elaborated in Sec. III, because the theory of PS methods is sharply different from Runge–Kutta methods, a new approach to mesh refinement is needed. This paper addresses this major problem.

In a basic PS discretization, Gaussian rules dictate the layout of the mesh. Once the number of nodes is fixed, the distribution of nodes is also fixed. The current practice of PS mesh refinement, as implemented in DIDO [33] and OTIS [34], is performed by manually inspecting the solution and increasing the grid size until the solution is deemed satisfactory. This simple technique leaves the selection of the Gauss–Lobatto grid size completely to the user. Without any guidelines, a good decision on the grid size can be a challenging job. The naïve approach of providing an a priori large number of points as a safety measure is not merely wasteful, it also does not always result in an expected increase in accuracy of the computed solution. The reason is as mentioned in question 1: convergence of the algorithm and convergence of the discretization are not the same thing. In fact, *the interplay between the convergence of algorithm and the convergence of the discretization can lead to a false negative* that the problem posed may not have a solution if we assume the algorithm is not failing. Alternatively, this issue may cast suspicion on theoretical results on convergence if it can be proved that the problem does indeed have a solution. The key to selecting the number of nodes is an error-estimation formula. That is, the size of the grid has to be tied to the accuracy of the solution. To achieve that, we need methods to quantitatively measure the accuracy of the computed solution. Barring known analytic solutions, it is impossible to compute the

Received 18 June 2007; revision received 15 November 2007; accepted for publication 19 November 2007. 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 0731-5090/08 $10.00 in correspondence with the CCC.

*Research Associate, Department of Electrical and Computer Engineering; qi.gong@utsa.edu.

†Professor, Department of Applied Mathematics; ffahroo@nps.edu.

‡Professor, Department of Mechanical and Astronautical Engineering; imross@nps.edu. Associate Fellow AIAA.

exact value of the error; consequently, we must rely on *estimates* on the level of accuracy of the computed solution. In this paper, we exploit the latest theoretical results for PS methods [15,16,18,35,36] and consistent approximation theory [37] to design such error-estimation techniques. By transferring the discrete solution to the spectral domain, we provide several error-estimation methods that can quantitatively measure the level of the accuracy. These techniques greatly facilitate the selection of appropriate grid sizes. Based on these error estimates, we design an algorithm that automatically terminates when the desired estimate of accuracy is achieved.

It is well known that even when the problem data are smooth, the optimal control can be discontinuous in time, as in bang–bang control [29]. For such problems, the PS knotting method [36] has proved to be an effective tool [6,8,9]. By adding knots, one can divide the time interval into smaller segments and therefore change the distribution of PS nodes. Furthermore, Gaussian rules guarantee a dense distribution of the nodes around the knots; hence, the knots should be placed where the solution is subject to sudden changes, as in points of discontinuity. Because the optimal solution is usually not known up front, it is challenging to find these points of interest a priori. Currently, the number and the location of the knots are found by visually inspecting the obtained solution, which is neither efficient nor precise. In this paper, we further the ideas proposed by Ross and Fahroo [36] and Gong and Ross [35] to automatically determine the number and location of the knots needed for segmentation. This technique, together with error-estimation methods, lays down the foundations for a pseudospectral mesh-refinement algorithm that automatically determines the size and the layout of the grid to achieve a desired estimate of accuracy.

II. Problem Formulation

We define a *basic optimal control problem* to find the state-control function pair $t\mapsto(\boldsymbol{x},\boldsymbol{u})\in\mathbb{R}^{N_x}\times\mathbb{R}^{N_u}$ and clock times t_0, and t_f such that

$$(\mathrm{B})\left\{\begin{array}{ll}\text{minimize} & J[\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),t_0,t_f]=E(\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f)\\ & \quad+\int_{t_0}^{t_f}F(\boldsymbol{x}(t),\boldsymbol{u}(t))\,\mathrm{d}t\\ \text{subject to} & \dot{\boldsymbol{x}}(t)=\mathbf{f}(\boldsymbol{x}(t),\boldsymbol{u}(t))\\ & \mathbf{e}(\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f)=\mathbf{0}\\ & \mathbf{h}(\boldsymbol{x}(t),\boldsymbol{u}(t))\le\mathbf{0}\end{array}\right.$$

where $\boldsymbol{x}_0=\boldsymbol{x}(t_0)$, $\boldsymbol{x}_f=\boldsymbol{x}(t_f)$, and all the relevant relationships are assumed to be true for almost all t. It is assumed that all the nonlinear functions (F, E, $\mathbf{f}$, $\mathbf{e}$, and $\mathbf{h}$) are continuously differentiable with respect to their arguments and that their gradients are Lipschitz-continuous over the domain.

A set of necessary conditions for problem B can be obtained from an application of the minimum principle [30]. These conditions can be articulated in terms of a boundary-value problem which is, more formally, a problem of solving a generalized equation [23]. We refer to this process as dualization, and it can be summarized [1] as problem B^λ:

$$(\mathrm{B}^\lambda)\left\{\begin{array}{ll}\text{find} & [\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),t_0,t_f;\boldsymbol{\lambda}(\cdot),\boldsymbol{\mu}(\cdot),\boldsymbol{\nu}]\\ \text{such that} & \dot{\boldsymbol{x}}(t)=\mathbf{f}(\boldsymbol{x}(t),\boldsymbol{u}(t))\\ & \mathbf{e}(\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f)=\mathbf{0}\\ & \mathbf{h}(\boldsymbol{x}(t),\boldsymbol{u}(t))\le\mathbf{0}\\ & \dot{\boldsymbol{\lambda}}(t)=-\frac{\partial\bar{H}[t]}{\partial\boldsymbol{x}}\\ & \frac{\partial\bar{H}}{\partial\boldsymbol{u}}=0\\ & \{\boldsymbol{\lambda}(t_0),\boldsymbol{\lambda}(t_f)\}=\left\{-\frac{\partial\bar{E}}{\partial\boldsymbol{x}_0},\frac{\partial\bar{E}}{\partial\boldsymbol{x}_f}\right\}\\ & \{H[t_0],H[t_f]\}=\left\{\frac{\partial\bar{E}}{\partial t_0},-\frac{\partial\bar{E}}{\partial t_f}\right\}\\ & \mathbf{0}\le\boldsymbol{\mu}(t)\perp-\mathbf{h}(\boldsymbol{x}(t),\boldsymbol{u}(t))\ge\mathbf{0}\end{array}\right.$$

where $\bar{H}$ is the Lagrangian of the Hamiltonian

$$\bar{H}(\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{x},\boldsymbol{u}):=H(\boldsymbol{\lambda},\boldsymbol{x},\boldsymbol{u})+\boldsymbol{\mu}^T\mathbf{h}(\boldsymbol{x},\boldsymbol{u})\tag{1}$$

H is the control Hamiltonian

$$H(\boldsymbol{\lambda},\boldsymbol{x},\boldsymbol{u}):=F(\boldsymbol{x},\boldsymbol{u})+\boldsymbol{\lambda}^T\mathbf{f}(\boldsymbol{x},\boldsymbol{u})\tag{2}$$

$\bar{E}$ is the endpoint Lagrangian

$$\bar{E}(\boldsymbol{\nu},\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f):=E(\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f)+\boldsymbol{\nu}^T\mathbf{e}(\boldsymbol{x}_0,\boldsymbol{x}_f,t_0,t_f)\tag{3}$$

and the shorthand notation $\bar{H}[t_f]$ is used to imply

$$\bar{H}[t_f]\equiv H(\boldsymbol{\lambda}(t_f),\boldsymbol{\mu}(t_f),\boldsymbol{x}(t_f),\boldsymbol{u}(t_f))$$

III. Pseudospectral Methods

Much of the details of PS methods are extensively described elsewhere [1–3,17,36]; here, we briefly summarize the main points for the purposes of relevance and completeness. In addition, we summarize the relationship between PS and spectral methods [38–40] and the differences between the standard convergence theorems and those required to solve optimal control problems.

A. Overview

The goal of a PS method is to solve problem B by discretizing it to problem B^N in a manner that permits the discretization to commute with dualization [25]. This commutation requirement is enunciated as the covector mapping principle [1,15,22,25,41] and is illustrated in Fig. 1. This means that a putative optimal solution of the discretized problem B^N must automatically satisfy the discretized necessary conditions $B^{\lambda N}$. Solving optimal control problems by this approach is far simpler than developing and solving for the necessary conditions, because solving for the necessary conditions requires solving a mixed complementarity problem (see problem B^λ). A covector mapping theorem [1,15] provides the proper connection to commute dualization with discretization.

B. Pseudospectral Discretization

In a standard PS discretization [39,40], we first select $N+1$ cardinal functions ϕ_l ($l=0,1,\ldots,N$) over the time interval $[t_0,t_f]$, such that they satisfy the Kronecker delta condition:

$$\phi_l(t_k)=\delta_{lk}\qquad k=0,1,\ldots,N$$

where the grid points $\pi^N=\{t_0,t_1,\ldots,t_N\}$ are called nodes. The nodes are chosen in a manner consistent with approximation theory (e.g., shifted Gaussian points). When a PS method is applied to optimal control, an application of the covector mapping principle requires that the continuous-time state trajectory be approximated by Nth-order weighted polynomials [17,42]. A choice of Gauss–Lobatto points allows us to choose unit weight functions so that we can write

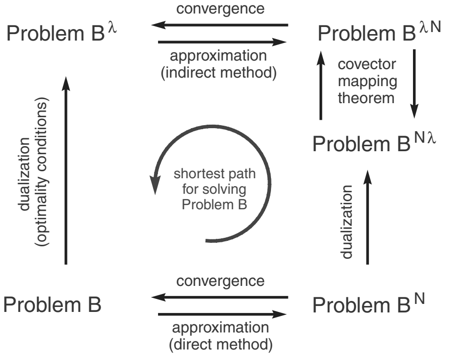

Fig. 1 Illustration of the covector mapping principle (adapted from Ross and Fahroo [1,22]).

$$\boldsymbol{x}(t) \simeq \boldsymbol{x}^N(t) := \sum_{l=0}^{N} \boldsymbol{x}_l \phi_l(t) \tag{4}$$

where $\boldsymbol{x}_l$ are the values of $\boldsymbol{x}^N(t_l)$. The approximation indicated in Eq. (4) is different when the function $\boldsymbol{x}(t)$ is known, in which case we write

$$\boldsymbol{x}(t) \simeq \mathcal{I}_N \boldsymbol{x}(t) := \sum_{l=0}^{N} \boldsymbol{x}(t_l) \phi_l(t) \tag{5}$$

where $\mathcal{I}_N$ is the interpolating operator. Thus, when $\boldsymbol{x}(t)$ is unknown, as is the case in solving optimal control problems, $\boldsymbol{x}_l$ is not necessarily equal to $\boldsymbol{x}(t_l)$; in fact, $\|\boldsymbol{x}(t_l) - \boldsymbol{x}_l\|$ may be quite large (no convergence). Standard convergence theorems in spectral methods [38–40] provide the error estimates between $\boldsymbol{x}(t)$ and $\mathcal{I}_N \boldsymbol{x}(t)$: that is, the error between a known function and its interpolant. Because this error is zero at the node points, these theorems provide the convergence rate for errors in between the node points. *The gap in convergence theorems for solving optimal control problems in which the function $\boldsymbol{x}(t)$ is unknown is an estimate of the error at the node points*, $\|\boldsymbol{x}(t_l) - \boldsymbol{x}_l\|$. Hence, it is the approximation indicated by Eq. (4), rather than Eq. (5), that is crucial for designing a spectral algorithm. The precise nature of the approximation indicated in Eq. (4) was developed recently by Gong et al. [15,18] and Kang et al. [16]. These theoretical results are exploited in this paper to design a practical multigrid algorithm.

One main difference between PS methods and other methods (e.g., Runge–Kutta) is that in PS methods, we seek to approximate the solution rather than the governing equations. Approximation theory then dictates that the cardinal functions be chosen so that they form a basis in some appropriate function space. Thus, operations in the governing equations are applied directly to $\boldsymbol{x}^N(t)$ to generate new, potentially simpler, equations that can be readily solved. In optimal control, operations in the problem setting include differentiation and integration. We approximate the derivative of $\boldsymbol{x}(t)$ by the derivative of its polynomial approximation $\boldsymbol{x}^N(t)$:

$$\dot{\boldsymbol{x}}(t) \simeq \dot{\boldsymbol{x}}^N(t) = \sum_{l=0}^{N} \boldsymbol{x}_l \dot{\phi}_l(t) \tag{6}$$

Thus, the equations of approximation are obtained by projecting the problem equations over the node points; for example,

$$\dot{\boldsymbol{x}}(t_k) \simeq \dot{\boldsymbol{x}}^N(t_k) = \sum_{l=0}^{N} \boldsymbol{x}_l \dot{\phi}_l(t_k) = \mathbf{f}(\boldsymbol{x}_k, \boldsymbol{u}_k)$$

where $\boldsymbol{u}_k$ is the approximation of $\boldsymbol{u}(t_k)$. Note that as part of the recent developments in PS methods, $\boldsymbol{u}(t)$ is not necessarily obtained by a Lagrange interpolant [15,18]. The differentiation matrix

$$D_{kl}^N = \dot{\phi}_l(t_k) \qquad l, k = 0, 1, \ldots, N$$

provides a rapid procedure for evaluating the derivatives at the node points [38,40].

The integration of the cost function is approximated as

$$\int_{t_0}^{t_f} F(\boldsymbol{x}(t), \boldsymbol{u}(t))\,\mathrm{d}t \simeq \sum_{l=0}^{N} F(\boldsymbol{x}_l, \boldsymbol{u}_l) \int_{t_0}^{t_f} \phi_l(t)\,\mathrm{d}t = \sum_{l=0}^{N} F(\boldsymbol{x}_l, \boldsymbol{u}_l) w_l \tag{7}$$

where

$$w_l := \int_{t_0}^{t_f} \phi_l(t)\,\mathrm{d}t$$

forms weights for a discrete 1-form (inner product) that facilitates the commutative operation suggested in Fig. 1.

Pseudospectral methods are far simpler than the traditional Galerkin (spectral) method in which $\boldsymbol{x}^N(t)$ is expanded in terms of noninterpolating orthogonal basis functions $\psi_l(t)$ $(l = 0, 1, \ldots, N)$:

$$\boldsymbol{x}(t) \simeq \boldsymbol{x}^N(t) = \sum_{l=0}^{N} \alpha_l \psi_l(t) \tag{8}$$

where the basis functions are required to automatically satisfy the boundary conditions as well. The spectral coefficients α_l of this generalized Fourier expansion satisfy the condition

$$\alpha_l = \int_{t_0}^{t_f} \boldsymbol{x}^N(t) \psi_l(t)\,\mathrm{d}t \bigg/ \int_{t_0}^{t_f} \psi_l^2(t)\,\mathrm{d}t$$

and the spectral method reduces to finding α_l so that certain residuals are minimal (orthogonal to a certain subspace). Although this method is far more complicated, there is a close relationship between α_l and $\boldsymbol{x}_l$ that can be easily exploited to transform the coefficients from the physical space to the spectral space and vice versa.

As an example of the preceding ideas, in the Legendre PS method, the grid points are the shifted Legendre–Gauss–Lobatto (LGL) points at which the shift is achieved by mapping the physical domain $[t_0, t_f] \ni t$ to a computational domain $[-1, 1] \ni \tau$ by the affine transformation:

$$\tau(t) = \frac{2t - (t_f + t_0)}{(t_f - t_0)}$$

where we have abused notation in using τ to imply both the transformation as well as the transformed variable. The LGL weights and the differentiation matrix are

$$\begin{aligned} w_k &:= \frac{t_f - t_0}{N(N+1)} \frac{1}{[L_N(\tau_k)]^2} \qquad k = 0, 1, \ldots, N \\ D_{kl}^N &:= \frac{2}{t_f - t_0} \begin{cases} \frac{L_N(\tau_k)}{L_N(\tau_l)} \cdot \frac{1}{\tau_k - \tau_l} & k \neq l \\ -\frac{N(N+1)}{4} & k = l = 0 \\ \frac{N(N+1)}{4} & k = l = N \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

where τ_k, $k = 0, 1, \ldots, N$ denote the LGL nodes [38,39] and $L_N(t)$ denotes the Legendre polynomial of order N.

C. Pseudospectral Relaxation

As noted in Sec. I and elsewhere [18], it is critical to guarantee the existence of a solution to the discretized problem if a solution to the original problem (problem B) exists. For this purpose, we introduce the following relaxed discretized problem:

$$\mathrm{B}^N(\varepsilon) \begin{cases} \text{minimize} & J^N([\boldsymbol{x}_k], [\boldsymbol{u}_k], t_0, t_f) = E(\boldsymbol{x}_0, \boldsymbol{x}_N, t_0, t_f) \\ & \quad + \sum_{l=0}^{N} F(\boldsymbol{x}_l, \boldsymbol{u}_l) w_l \\ \text{subject to} & \left\| \mathbf{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) - \sum_{l=0}^{N} D_{kl} \boldsymbol{x}_l \right\|_\infty \leq \varepsilon \\ & \|\mathbf{e}(\boldsymbol{x}_0, \boldsymbol{x}_f, t_0, t_f)\|_\infty \leq \varepsilon \\ & \mathbf{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) \leq \varepsilon \\ & k = 0, 1, \ldots, N \end{cases}$$

where $\varepsilon > 0$ is the relaxation parameter and $\boldsymbol{\epsilon} = [\varepsilon, \ldots, \varepsilon]^T$. For a vector ξ, $\|\xi\|_\infty$ denotes the maximum element of ξ.

Until recently, the theory of PS methods for optimal control was based on setting $\varepsilon = 0$. Because it is impossible to set $\varepsilon = 0$ on a digital computer, the prevailing wisdom was to set it equal to as small a number as possible. The counterexample by Gong et al. [18] shows that setting ε too small may actually result in an infeasible set for the discretized problem. Similar counterexamples have been developed by Mordukhovich [20] for Euler (and hence, Runge–Kutta) methods. Setting ε too large to fix the problem will render the discrete solution infeasible with respect to the original continuous-time problem. Such results have shown that the theory for PS optimal control cannot be directly "lifted" from its corresponding theory employed to solve fluid mechanics problems [38]. In other words, it was necessary to develop a new theory for PS methods to handle the various nuances of optimal control theory. Recent results with PS optimal control [15,16,18] demonstrate a strong relation between the number of nodes and the relaxation parameter ε. The result, posited in the form of the following Lemma, not only closes the gap between theory and practice, it also clarifies a number of apparent anomalies in the practice of PS discretizations.

Lemma 1 (existence)[15]: Given any feasible solution $t \mapsto (x, u)$ of problem B, suppose $x(\cdot) \in W^{m,\infty}$ with $m \geq 2$. Then, for any given $\varepsilon > 0$, there exists a positive integer N_ε such that for any $N > N_\varepsilon$, the feasible set of relaxed discretized problem [problem $B^N(\varepsilon)$] is nonempty.

The definition of Sobolev space $W^{m,\infty}$ can be found in [38]. It is possible to relax this regularity condition [16], but for notional simplicity, we avoid such generalities to focus the attention of this paper to the core algorithmic issues. Lemma 1 theoretically guarantees the well-posedness of PS optimal control methods. It is also a key result that will be used later to construct the spectral algorithm.

Although Lemma 1 is revealing, it does not complete the practical foundation for solving an optimal control problem, because we need a connection between a discretized solution $\boldsymbol{x}_l, \boldsymbol{u}_l$ and the typically unknown optimal solution $\boldsymbol{x}^*(t_l), \boldsymbol{u}^*(t_l)$. This connection, developed by Gong et al. [15,18] and Kang et al. [16], ensures the convergence of $\boldsymbol{x}_l$ to $\boldsymbol{x}^*(t_l)$ in addition to the convergence of the controls and the dual variables under the following assumption (for the purposes of brevity, we state the conditions for only the state and control variables):

Assumption 1: Let $\{(\boldsymbol{x}_l^N, \boldsymbol{u}_l^N), 0 \leq l \leq N\}_{N=N_i}^{\infty}$ be a sequence of optimal solutions to problem B^N and $t \mapsto \{(\boldsymbol{x}^N(t), \boldsymbol{u}^N(t))\}_{N=N_i}^{\infty}$ be their interpolating function sequence. There exist continuous functions $t \mapsto \mathbf{q}(t) \in \mathbb{R}^{N_x}$ and $t \mapsto \boldsymbol{u}^\infty(t) \in \mathbb{R}^{N_u}$ and a constant $\boldsymbol{x}_0^\infty$, such that

$$\lim_{N\to\infty} (\dot{\boldsymbol{x}}^N(t), \boldsymbol{u}^N(t)) = (\mathbf{q}(t), \boldsymbol{u}^\infty(t)) \tag{10}$$

uniformly on $t \in [-1, 1]$, and $\lim_{N\to\infty} \boldsymbol{x}_0^N = \boldsymbol{x}_0^\infty$.

The continuity requirement on $t \mapsto (\mathbf{q}(t), \boldsymbol{u}^\infty(t))$ can be further relaxed (see [16] for details).

Theorem 1 (convergence) [15]: Under Assumption 1, there exists an optimal solution $\boldsymbol{x}^*(\cdot), \boldsymbol{u}^*(\cdot)$ to problem B such that the following limits converge for all $0 \leq k \leq N$:

$$\lim_{N\to\infty} (\boldsymbol{x}_k^N - \boldsymbol{x}^*(t_k)) = 0, \qquad \lim_{N\to\infty} (\boldsymbol{u}_k^N - \boldsymbol{u}^*(t_k)) = 0$$

From this Theorem, it is apparent that if the relaxation parameter ε is chosen to be some fixed constant independent of N, then convergence to the optimal solution is not guaranteed. In other words, if ε is too large, we do not have convergence, and if ε is too small for a given N, we may have an infeasible discretized space (by Lemma 1). Thus, for a successful multigrid algorithm, we must choose the relaxation parameter ε to vary inversely as some power of N. The choice of this variation will be discussed shortly, but it is apparent that our proposed scheme is a spectral analog of the notion of *consistent approximations* discussed in [19,20,37] and recently exploited by Betts et al. [26,31] to illustrate the success and robustness of using multigrid collocation methods.

D. Pseudospectral Knotting Methods

In standard PS methods, Gaussian rules dictate the layout of the mesh. If the solution is piecewise smooth with nonsmooth junction points, it is intuitive that a judicious subdivision of the time interval can be used to exploit the exponential convergence property of the Gaussian layout over carefully chosen subintervals. This is the concept of the PS knots introduced by Ross and Fahroo [36]. The basic idea of the knotting methods is to divide the time interval $[-1, 1]$ into smaller subintervals, renormalize each subinterval to $[-1, 1]$, and then apply PS discretization on each subinterval. Information is exchanged between the subintervals through the double Lobatto points at -1 and 1. The double Lobatto points are called PS knots. The knots can be fixed or free. If the knot is free, it can be treated as an extra decision variable.

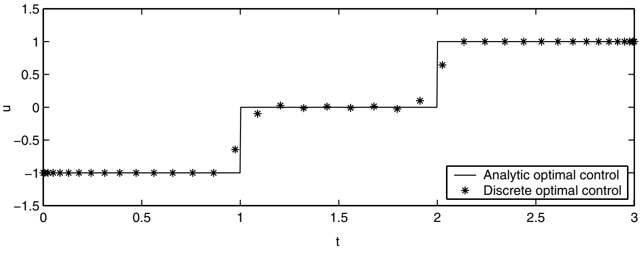

Fig. 2 Unknotted optimal control for problem (11) with 40 nodes.

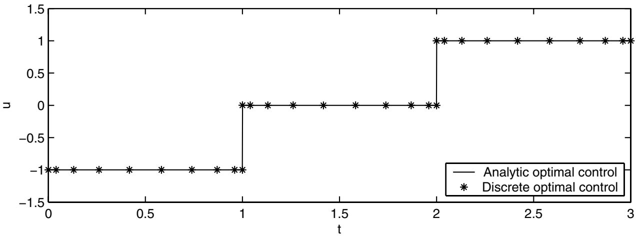

Fig. 3 Knotted optimal control for problem (11) with 30 nodes and knots at $t = 1$ and 2.

Because they provide an efficient way to handle nonsmooth controls or other similar phenomena involving high-frequency components, we use the concept of PS knots as part of our mesh-refinement algorithm. To appreciate a key feature of PS knots, consider the following example:

$$\begin{cases} \text{minimize} & J[x(\cdot), u(\cdot)] = \int_0^3 x(t)\,\mathrm{d}t \\ \text{subject to} & \dot{x} = u \\ & x(0) = 1 \\ & x(3) = 1 \\ & |u| \leq 1 \\ & x \geq 0 \end{cases} \tag{11}$$

The analytic optimal control u^* is

$$u^* = \begin{cases} -1, & t \in [0, 1) \\ 0, & t \in [1, 2] \\ 1, & t \in (2, 3] \end{cases}$$

The discrete optimal control with 40 nodes is shown in Fig. 2. It is clear that the accuracy of the computed solution is diminished at the jump points. Now, choosing knots at $t' = 1$ and $t' = 2$, we resolve problem (11) with 30 nodes (10 on each subinterval). The result is shown in Fig. 3. It is obvious that the 30-node knotted solution is more accurate than the 40-node unknotted solution illustrated in Fig. 2.

One particular advantage of the knotting methods is to control the distribution of the nodes. Note that the distribution of the quadrature nodes is not uniform (see the previous figures). The nodes are much more dense at the two end points. Indeed, the distance between two adjacent nodes around the endpoints converges quadratically to zero. Therefore, placing the knots near those locations at which the control or state trajectories have large variations has the natural effect of a dense distribution of nodes at precisely the locations of interest. This key feature will be used in our mesh-refinement algorithm.

IV. Core Spectral Algorithm

The spectral algorithm proposed in this paper includes two layers. The "inner loop" is to solve the optimization problem $B^N(\varepsilon^N)$ for a fixed number of nodes N and mesh (nodes distribution). The "outer loop" is to generate a sequence of problems $B^{N_i}(\varepsilon^{N_i})$ $(i = 0, 1, \ldots n)$ and corresponding mesh sequence, such that problem $B^{N_n}(\varepsilon^{N_n})$ solves problem B to one or more of some specified tolerance criteria. Before describing this spectral algorithm, we first note that a PS solution can be quickly transformed from the physical space to spectral space. For example, for the Legendre PS method, we can write

$$\boldsymbol{x}^N(t) = \sum_{l=0}^{N} \boldsymbol{x}_l \phi_l(t) = \sum_{j=0}^{N} \mathbf{a}_j \mathsf{L}_j(t) \tag{12}$$

$$\boldsymbol{u}^N(t) = \sum_{l=0}^{N} \boldsymbol{u}_l \phi_l(t) = \sum_{j=0}^{N} \mathbf{b}_j \mathsf{L}_j(t) \tag{13}$$

$$\boldsymbol{\lambda}^N(t) = \sum_{l=0}^{N} \boldsymbol{\lambda}_l \phi_l(t) = \sum_{j=0}^{N} \mathbf{c}_j \mathsf{L}_j(t) \tag{14}$$

$$\boldsymbol{\mu}^N(t) = \sum_{l=0}^{N} \boldsymbol{\mu}_l \phi_l(t) = \sum_{j=0}^{N} \mathbf{d}_j \mathsf{L}_j(t) \tag{15}$$

Using the orthogonal property of the Legendre polynomials, it is easy to show that the spectral coefficients satisfy

$$\mathbf{a}_j = (j + 0.5) \sum_{k=0}^{N} L_j(t_k) w_k \boldsymbol{x}_k \qquad j = 1, \ldots, N-1,$$

$$\mathbf{a}_N \simeq (N + 0.5) \sum_{k=0}^{N} L_N(t_k) w_k \boldsymbol{x}_k$$

The formula for $\mathbf{a}_N$ holds only approximately, due to the property of the Gauss–Lobatto quadrature integration rule. The same formulas apply for $\mathbf{b}_j$, $\mathbf{c}_j$, and $\mathbf{d}_j$. In the proposed spectral algorithm, the coefficients in the spectral space are used to estimate the error and thus serve as one of the indicators of convergence.

Several variants of a spectral algorithm can be designed based on the collection of theoretical results presented in Sec. III. These variants are too numerous to be listed; hence, we begin by presenting only the core algorithm here and we will fill out some key details in the next section.

A. Initialization

From Lemma 1, we first note that the existence of a solution to problem B does not automatically imply the existence of a solution to problem $B^N(\varepsilon)$ for an arbitrary choice of ε and N. Hence, we need to find consistent pairs (ε and N). By Lemma 1, we know that for any given ε, we can always find an N such that the feasible set of problem $B^N(\varepsilon)$ is nonempty. Hence, our initialization step reduces to selecting a pair (ε and N) for which the feasible set is nonempty. To start our algorithm from some arbitrary point (e.g., a poor guess), we adopt an "elastic procedure," defined next.

In the first step, we arbitrarily select a finite grid size sequence:

$$N_I^0 = 20 < N_I^1 = 30 < \cdots < N_I^L = 20 + 10L$$

where the numbers indicate default values implemented in an α version of DIDO [33], and the subscript I denotes that the sequences are only used in the initialization step. Other possible choices of sequences are

$$N_I^0 = 15 < N_I^1 = 20 < \cdots < N_I^L = 15 + 5L$$

and

$$N_I^0 = 8 < N_I^1 = 16 < \cdots < N_I^L = 8 \times 2^L$$

These choices are offered as part of the algorithm parameters similar to those found in any optimal control software package such as SOCS. In the same spirit, we choose a sequence of increasing numbers:

$$\varepsilon_I^0 = 10^{-5} < \varepsilon_I^1 = 10^{-4} < \cdots$$

The initialization procedure is to find a pair (N and ε), such that problem $B^N(\varepsilon)$ is feasible. It is achieved by an elastic procedure as follows: we start the initialization loop by solving problem $B^{N_I^0}(\varepsilon_I^0)$. If a feasible or an optimal solution is obtained, the initialization is declared a success; otherwise, a search for a feasible solution is performed by fixing N_I^0 and increasing the tolerance from

$$\varepsilon_I^0 = 10^{-5} > \varepsilon_I^0 = 10^{-4} > \cdots > \varepsilon_I^M = 10^{-1}$$

We hasten to note that these default numbers are not absolute numbers but are based on normalizations similar to those used in nonlinear programming solvers [43]. If no feasible solution can be found even for the loosest tolerance, the optimal control problem is declared infeasible and the algorithm terminates. A generalized version of the initialization procedure is summarized in the following conceptual algorithm.

Algorithm 1:

1) Set the iteration counters $j = k = 0$.

2) Solving problem $B^{N_I^j}(\varepsilon_I^k)$.

3) If a feasible or an optimal solution is obtained, stop initialization; else proceed to the next step.

4) If $k < M$, set $k = k + 1$, and go to step 2; else proceed to the next step.

5) If $j < L$, set $j = j + 1$, $k = 0$, and go to step 2; else the optimal control problem is declared infeasible and the initialization terminates.

Remark 1: The preceding initialization procedure only demonstrates the basic idea. There are many ways to implement it. Additional implementation details may be found in [44].

B. Optimal Placement and Location of Knots

The elementary observation that the knots must be placed around the points at which the control undergoes a sudden change can be formalized in terms of the time derivative of the control; that is, the knots are placed where the derivative of the control is sufficiently large. This observation can be very easily encoded by way of the PS differentiation matrix D [see Eq. (9)], because we have:

$$\begin{pmatrix} \dot{u}^N(t_0) \\ \vdots \\ \dot{u}^N(t_N) \end{pmatrix} = D \times \begin{pmatrix} u_0 \\ \vdots \\ u_N \end{pmatrix} \tag{16}$$

where t_i are the LGL nodes. This concept is illustrated in Fig. 4 for the unknotted 40-node solution (see Fig. 2). Because the maximum derivatives appear around $t = 1$ and 2, it suggests that we place two knots around $t = 1$ and 2. Thus, the points at which the derivatives are large are the locations of the knots. Obviously, the number of knots is automatically determined by this procedure. Based on these ideas, we propose the following simple algorithm for the placement of the knots:

Algorithm 2:

1) Choose a parameter DU, which is the threshold of the maximum derivative.

2) Given the current mesh $\{t_0, t_1, \ldots, t_N\}$, compute the derivative of the interpolating polynomial $\boldsymbol{u}^N(t)$ at the nodes $t_0, \ldots, t_N$ by formula (16).

3) Choose the knots at the positions at which $\dot{\boldsymbol{u}}^N(t_i) \geq DU$, and denote the knots as $t_1' < \cdots < t_m'$. If there are two or more adjacent

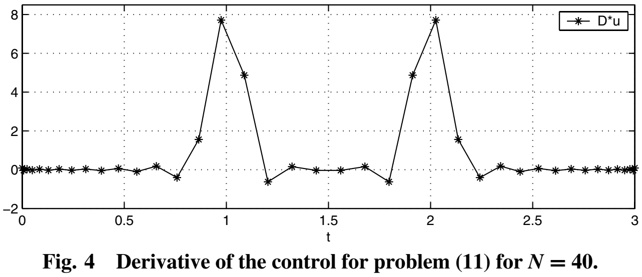

Fig. 4 Derivative of the control for problem (11) for $N = 40$.

nodes such that $\dot{\boldsymbol{u}}^N(t_i) \geq DU$, choose the midpoint of these nodes as the tentative location of the knots.

4) If $\dot{\boldsymbol{u}}^N(t_i) < DU$ for all i, then no knots are needed.

Remark 2: In algorithm 2, for simplicity, we only check the derivative of the control. In practice, one can also check the derivative information of the state variables by the same idea and place the knots where either the state or the control has large variations. Because the state is normally one order smoother than the control, checking the derivative of the control is usually sufficient.

Remark 3: The proposed algorithm provides a simple method to determine the fixed knots. It can be easily modified for free knots [36]. For instance, instead of choosing the knots at the points at which $\dot{\boldsymbol{u}}^N(t_i) \geq DU$, we can free the knots but set up a reasonable bound according to the derivative information.

Remark 4: If knots have already been placed in the current mesh $\{t_0, t_1, \ldots, t_N\}$, formula (16) needs to be updated to be applied on each segment. For instance, suppose there one knot at t_k. Then the node sequences $\{t_0, t_1, \ldots, t_k\}$ and $\{t_k, t_{k+1}, \ldots, t_N\}$ are all shifted LGL quadrature nodes. We then apply formula (16) on both segments $[t_0, t_k]$ and $[t_k, t_N]$ with different differentiation matrices that depend on the number of nodes and the length of the segments [see Eq. (9)].

C. Mesh Generation

Once the knots are determined, the distribution of the nodes in each segment is chosen according to Gaussian rules. Thus, in PS methods, the mesh is completely determined by only three parameters: the number of knots, the location of knots, and the number of nodes in between the knots. Thus, the only remaining problem in constructing a mesh is the number of nodes in between the knots. We distribute the number of nodes according to the length of each segment; that is, the number of nodes on segment $[t'_i, t'_{i+1}]$ is chosen according to

$$N'_i = \text{round}\left(\frac{t'_{i+1} - t'_i}{2} N\right) \qquad i = 0, \ldots, k-1 \tag{17}$$

where round(s) means the closest integer of s. Many simple routines can be used to modify Eq. (17), so that

$$\sum_{i=0}^{k-1} N'_i = N$$

is the total number of nodes.

Remark 5: It is necessary to have a minimum number of nodes over each segment. For the purposes of clarity, we omit such details and implementation procedures; instead, we defer some of the discussion to example problems to demonstrate the concept.

D. Main Algorithm

Before stating the main algorithm, let us introduce a projection operator. It is used to map a solution from one grid to another. Let π^{N_r} denote a PS grid over I so that to each grid, we can define problem $B^{N_r}(\varepsilon_r)$, which is based on an approximation to problem B. Suppose we are given two grids, π^{N_r} and π^{N_n}. Let $(\boldsymbol{X}^{N_r}, \mathbf{U}^{N_r}, \boldsymbol{\Lambda}^{N_r}, \mathbf{M}^{N_r})$ represent the values of functions $\boldsymbol{x}^{N_r}(\cdot)$, $\boldsymbol{u}^{N_r}(\cdot)$, $\boldsymbol{\lambda}^{N_r}(\cdot)$, and $\boldsymbol{\mu}^{N_r}(\cdot)$ over the grid π^{N_r}. Let the operation defined by

$$\boldsymbol{X}^{N_n} := {}^{N_n}\boldsymbol{P}^{N_r}\boldsymbol{X}^{N_r}$$

generate values of the function $\mathcal{I}_{N_r}\boldsymbol{x}^{N_r}(t)$ over the grid π^{N_n}. ${}^{N_n}P^{N_r}$ is a prolongation operator if $N_n > N_r$ and a restriction operator if $N_n < N_r$. If $N_n = N_r$, ${}^{N_n}P^{N_r}$ is an identity operator. The same operation, ${}^{N_n}P^{N_r}$, applies to $(\mathbf{U}^{N_r}, \boldsymbol{\Lambda}^{N_r}, \mathbf{M}^{N_r})$ as well.

To start the main algorithm, first we assume that the initialization procedure has been successful and denote the corresponding number of nodes and tolerance as $\{N_0, \varepsilon_0\}$. Let $\{N_j\}_{j\geq 0}$ be a sequence of increasing integers (i.e., $N_{j+1} > N_j$, which is selected in a manner similar to the initialization step) Likewise, let $\{\varepsilon_k\}_{k\geq 0}$ be a sequence of decreasing positive real numbers [i.e., $\varepsilon_{k+1} < \varepsilon_k$ (see the initialization step for default values)]. The main algorithm can be described as follows:

Algorithm 3:

1) Set the iteration counter $j = k = 0$.

2) From the covector mapping theorem [1,15], obtain the relaxed values of the dual variables $\boldsymbol{\lambda}^{N_j}$, $\boldsymbol{\mu}^{N_j}$, and $\boldsymbol{\nu}^{N_j}$ associated with problem $B^{\lambda N_j}(\varepsilon_k)$.

3) On every interval between two adjacent knots, find the spectral coefficients $\mathbf{a}_l$, $\mathbf{b}_l$, $\mathbf{c}_l$, and $\mathbf{d}_l$.

4) Estimate the error based on the spectral coefficients. If the error is within the tolerance, stop and exit; else proceed to the next step.

5) Use the algorithm in Sec. IV.B to determine the number and the location of the knots.

6) For $N = N_{j+1}$, apply the algorithm in Sec. IV.C to find the new mesh.

7) Use a prolongation operator ${}^{N_{j+1}}P^{N_j}$ to create primal and dual variables over the new grid:

$$\boldsymbol{X}^{N_{j+1}} = {}^{N_{j+1}}\boldsymbol{P}^{N_j}\boldsymbol{X}^{N_j}, \qquad \boldsymbol{U}^{N_{j+1}} = {}^{N_{j+1}}\boldsymbol{P}^{N_j}\boldsymbol{U}^{N_j}$$
$$\boldsymbol{\Lambda}^{N_{j+1}} = {}^{N_{j+1}}\boldsymbol{P}^{N_j}\boldsymbol{\Lambda}^{N_j}, \qquad \mathbf{M}^{N_{j+1}} = {}^{N_{j+1}}\boldsymbol{P}^{N_j}\mathbf{M}^{N_j}$$

8) Set $j = j + 1$, $k = k + 1$.

9) Warm-start the solution procedure to solve problem $B^{N_j}(\varepsilon_k)$ using the dual variables.

10) If problem $B^{N_j}(\varepsilon_k)$ is infeasible, set $k = k - 1$ and go to step 9; else go to step 2.

Remark 6: From steps 2 and 9, it is clear that the covector mapping theorem is a critical and integral part of the spectral algorithm. This step also facilitates a rapid solution to the optimal control problem when integrated with sequential quadratic programming methods that employ the notion of active sets [43].

Remark 7: In step 4 of the main algorithm, we have not yet specified the error-estimation methods and the stopping criteria. Given the importance of this step and the various choices of possible error-estimation criteria, this procedure is described in detail in the next section.

Remark 8: Several variations to the algorithm are possible; for example, if the initial guess for the solution is deemed to be good, then the initialization procedure can be eliminated. Similarly, a warm-start procedure can be activated at the first iteration by way of a flag. For the purpose of brevity, these options are not discussed any further.

V. Error Estimation and Stopping Criteria

Error estimation is the central issue in mesh-refinement algorithms. Any successful mesh-refinement algorithm needs to be terminated when the estimated error is below a specified number. In this section, we derive some error-estimation formulas and their corresponding stopping criteria. The stopping criteria can be used separately or combined with each other to develop several variants of the spectral algorithm.

Theorem 1 shows that if the interpolating polynomials of the discrete optimal solution converge, the limit point must indeed be the optimal solution of the continuous optimal control problem. This result suggests that the accuracy of the computed solution can be captured by the convergence property of the discrete solution sequence. Transferring the discrete solution to its spectral domain provides an accurate and easily implementable process to test the convergence property. Using this notion, we formulate several convergence tests based on the spectral coefficients of the discrete optimal solution. For the sake of simplicity, the following tests are presented for the case when no knots are used. If knots are employed, these tests need to be performed on every segment between two adjacent knots.

A Spectral Coefficient Test

Similar to a Fourier expansion, the Legendre spectral coefficients represent the amplitudes of the signal at different frequencies. If a continuous signal is smooth enough, Jackson's theorem shows that the amplitudes of the high-order frequencies will converge to zero.

Theorem 2 (Jackson's theorem) [45]: Let $h(t)$ be of bounded variation in $[-1, 1]$. Denote $U(h(t))$ as the upper bound of $|h(t)|$ and $V(h(t))$ as the total variation of $h(t)$. Define

$$H(t) = H(-1) + \int_{-1}^{t} h(s)\,\mathrm{d}s$$

then $\{c_n\}_{n=0}^{\infty}$, the sequence of Legendre spectral coefficients of $H(t)$, satisfies the inequality

$$|c_n| < \frac{6}{\sqrt{\pi}}(U(h(t)) + V(h(t)))\frac{1}{n^{3/2}}$$

for all $n \geq 1$.

The result suggests that the last coefficients of the interpolating polynomials of the discrete solution can be good indicators of error. If the last coefficients are small enough, it means that the continuous signals have been well-represented by their interpolating polynomials. Adding more nodes will only capture negligibly-higher-frequency components and will not result in any significant improvement in accuracy. This notion has been validated by Riehl et al. [46] on several problems in aerospace trajectory optimization. Furthermore, as described by Ross et al. [47], it is possible to capture even the high-frequency terms by anti-aliasing the solution through an application of Bellman's principle. Consequently, the following simple stopping criterion can be used in the main algorithm.

Jackson stopping criterion: If $\mathbf{a}_{N_j}$, $\mathbf{b}_{N_j}$, $\mathbf{c}_{N_j}$, and $\mathbf{d}_{N_j}$, defined in Eqs. (12–15), satisfy

$$\|\mathbf{a}_{N_j}\| \leq \delta_x, \qquad \|\mathbf{b}_{N_j}\| \leq \delta_u, \qquad \|\mathbf{c}_{N_j}\| \leq \delta_\lambda, \qquad \|\mathbf{d}_{N_j}\| \leq \delta_\mu$$

then stop. Here δ_x, δ_u, δ_λ, and δ_μ represent the desired accuracy for the states, controls, adjoint covector (costates), and path covector, respectively.

It is extremely important to note that δ_x, δ_u, δ_λ, and δ_μ are not the actual errors between the computed solution and the optimal solution; rather, they are only estimates or indicators of error. Finding true errors for general problems is an impossible task. From an error-analysis perspective, the error estimates are typically overestimates. From an algorithm-design perspective, we merely need good default values. From the underlying assumption in Jackson's theorem, it is apparent that it is not advisable to apply this stopping criterion to a discontinuous control. In such a situation, Jackson's criterion is used on states only, because they are typically continuous even when the control has jumps. Because of these nuances, we first illustrate how to apply the stopping criteria.

Example 1:

Consider the following nonlinear control problem:

$$\begin{cases} \text{minimize} & J[x(\cdot), u(\cdot), x(1)] = 3x(1) + \frac{1}{2}\int_{-1}^{1} u^2(t)\,\mathrm{d}t \\ \text{subject to} & \dot{x} = \exp(x)u \\ & x(-1) = 0 \end{cases} \tag{18}$$

It is quite straightforward to show that this problem has a closed-form solution:

$$x^*(t) = -\,\ell n(t+2), \qquad u^*(t) = -1, \qquad \lambda^*(t) = t + 2$$

Because the default values of DIDO (see Sec. IV) are set for industrial-strength problems, problem (18) converged within a single spectral iteration. This feature illustrates that for certain academic-strength problems, the spectral algorithm may converge within one iteration. Nonetheless, to amplify a few features of our algorithm, we set $\delta_x = 10^{-6}$ and arbitrarily choose

$$N_1 = 5, \qquad \varepsilon^{N_1} = 10^{-6}$$

for our initialization step. A computation of the spectral coefficients based on the first iteration of the spectral algorithm yielded $|a_5| = 1.917 \times 10^{-3} > \delta_x$. This initiated the mesh-refinement segment of our algorithm, resulting in

$$N_2 = N_1 + 5 = 10, \qquad \varepsilon^{N_2} = \varepsilon^{N_1}/10 = 10^{-7}$$

The second iterate yielded $|a_{10}| = 1.835 \times 10^{-6} > \delta_x$. This initiated a third iteration resulting in

$$N_3 = N_2 + 5 = 15, \qquad \varepsilon^{N_3} = 10^{-8}$$

It yielded $|a_{15}| = 1.65031 \times 10^{-9} < \delta_x$ and hence successfully terminated the algorithm.

We now demonstrate the validity of our algorithm by comparing the true errors using the closed-form solution. Table 1 summarizes the results, where

Table 1 Errors norms for Example 1

N	$\|e_x\|_\infty$	$\|e_u\|_\infty$	$\|e_\lambda\|_\infty$
5	2.502e-004	8.423e-003	8.440e-003
10	1.239e-007	1.595e-005	1.617e-005
15	1.597e-009	4.796e-008	5.071e-008

$$\|e_x\|_\infty = \|\boldsymbol{x}^*(t_k) - \boldsymbol{x}_k\|_{L^\infty}$$

$$\|e_u\|_\infty = \|\boldsymbol{u}^*(t_k) - \boldsymbol{u}_k\|_{L^\infty}$$

and

$$\|e_\lambda\|_\infty = \|\lambda^*(t_k) - \lambda_k\|_{L^\infty}$$

It is evident from these error norms that the spectral coefficients serve as good error indicators.

B. L^2 Error Test

As stated in Theorem 1, the convergence of the discrete optimal solution can be guaranteed under Assumption 1. It turns out that Assumption 1 can be numerically verified by examining the convergence of the sequence of spectral coefficients [18]. In the following, we provide a formula to test the Cauchy convergence property of the polynomial sequence in L^2 norm.

Lemma 2: Consider any two polynomial $y^{N_1}(t)$ and $y^{N_2}(t)$ of the order N_1 and N_2, respectively ($N_1 < N_2$). Let a_{1k} ($k = 1, \ldots, N_1$) and a_{2k} ($k = 1, \ldots, N_2$) be their Legendre spectral coefficients; that is,

$$y^{N_1}(t) = \sum_{k=0}^{N_1} a_{1k}L_k(t), \qquad y^{N_2}(t) = \sum_{k=0}^{N_2} a_{2k}L_k(t)$$

Then

$$\int_{-1}^{1} \|y^{N_1}(t) - y^{N_2}(t)\|^2\,\mathrm{d}t = \sum_{k=0}^{N_1}\frac{(a_{1k} - a_{2k})^2}{k + 0.5} + \sum_{k=N_1+1}^{N_2}\frac{a_{2k}^2}{k + 0.5} \tag{19}$$

Proof: By orthogonal property of the Legendre polynomial, we have

$$\int_{-1}^{1} L_i(t)L_j(t)\,\mathrm{d}t = 0, \quad \text{if } i \neq j \quad \text{and} \quad \int_{-1}^{1} L_i^2(t)\,\mathrm{d}t = \frac{1}{i + 0.5}$$

Therefore,

$$
\begin{aligned}
&\int_{-1}^{1} \|y^{N_1}(t) - y^{N_2}(t)\|^2 \,\mathrm{d}t \\
&\quad = \int_{-1}^{1} \left[\sum_{k=0}^{N_1} (a_{1k} - a_{2k}) L_k(t) - \sum_{k=N_1+1}^{N_2} a_{2k} L_k(t) \right]^2 \mathrm{d}t \\
&\quad = \sum_{k=0}^{N_1} (a_{1k} - a_{2k})^2 \int_{-1}^{1} L_k^2(t)\,\mathrm{d}t + \sum_{k=N_1+1}^{N_2} a_{2k}^2 \int_{-1}^{1} L_k^2(t)\,\mathrm{d}t \\
&\quad = \sum_{k=0}^{N_1} \frac{(a_{1k} - a_{2k})^2}{k + 0.5} + \sum_{k=N_1+1}^{N_2} \frac{a_{2k}^2}{k + 0.5}
\end{aligned}
$$

□

Now consider a polynomial sequence $\{y^N(t)\}$. Suppose the sequence converges to a function $y^*(t)$. By the triangular inequality,

$$
\|y^{N_1}(t) - y^{N_2}(t)\|_{L_2} \le \|y^{N_1}(t) - y^*(t)\|_{L_2} + \|y^{N_2}(t) - y^*(t)\|_{L_2} \tag{20}
$$

The left-hand side of Eq. (20) can be evaluated by formula (19); the right-hand side of Eq. (20) is the summation of the L^2 errors. Hence, $\|y^{N_1}(t) - y^{N_2}(t)\|_{L^2}$ can be used as a L^2 error indicator. Based on this concept, we can test the convergence of the interpolating polynomial sequences

$$
\{\boldsymbol{x}^N(t), \boldsymbol{u}^N(t), \boldsymbol{\lambda}^N(t), \boldsymbol{\mu}^N(t)\}
$$

and use the L^2 error norm of the sequences as the stopping criterion as follows:

L^2 stopping criterion: Substitute spectral coefficients at step j ($\{\mathbf{a}_k, \mathbf{b}_k, \mathbf{c}_k, \mathbf{d}_k\}_{k=1}^{k=N_j}$) and the spectral coefficients from the previous step (i.e., $\{\mathbf{a}_k, \mathbf{b}_k, \mathbf{c}_k, \mathbf{d}_k\}_{k=1}^{k=N_{j-1}}$) into formula (19). If the L^2 error norms satisfy

$$
\begin{aligned}
&\|\boldsymbol{x}^{N_j}(t) - \boldsymbol{x}^{N_{j-1}}(t)\|_{L_2} \le \delta_x, \qquad \|\boldsymbol{u}^{N_j}(t) - \boldsymbol{u}^{N_{j-1}}(t)\|_{L_2} \le \delta_u \\
&\|\boldsymbol{\lambda}^{N_j}(t) - \boldsymbol{\lambda}^{N_{j-1}}(t)\|_{L_2} \le \delta_\lambda, \qquad \|\boldsymbol{\mu}^{N_j}(t) - \boldsymbol{\mu}^{N_{j-1}}(t)\|_{L_2} \le \delta_\mu
\end{aligned}
$$

then stop. Note that δ_x, δ_u, δ_λ, and δ_μ are not necessarily the same as those used in the Jackson stopping criterion. This error-estimation method uses all the spectral coefficients from the current and previous runs. Therefore, it is expected to be more accurate than the Jackson stopping criterion, which uses the last spectral coefficient only.

Remark 9: Several other variations of the L^2 stopping criterion may be used. For instance, the L^2 error can be normalized by the L^2 norm of the signal, and the relative error may be used as a stopping criterion. The L^2 norm of a given polynomial signal can be easily calculated by applying Lemma 2 with $y^{N_1}(t)$ set to zero. Furthermore, it is not necessary to check all the variables. As a rule of thumb, if the control is continuous, it is usually sufficient to check the convergence of the control variables. If the control is discontinuous, a more robust approach is to apply the test on the states.

Example 2:

Consider the maximum-energy orbit transfer problem initially proposed and solved by Hermann and Conway [48] and later used by Fahroo and Ross [3] to demonstrate the covector mapping theorem. This problem is posed as follows:

$$
\boldsymbol{x}^T = [r, \theta, v_r, v_t] \qquad \boldsymbol{u} = [\beta]
$$

$$
\left\{
\begin{aligned}
&\text{minimize} && J[\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot), t_f] = -\tfrac{1}{2}[v_r(t_f)^2 + v_t(t_f)^2] + \tfrac{1}{r(t_f)} \\
&\text{subject to} && \dot{r} = v_r \\
& && \dot{\theta} = \tfrac{v_t}{r} \\
& && \dot{v}_r = \tfrac{v_t^2}{r} - \tfrac{1}{r^2} + A\sin(\beta) \\
& && \dot{v}_t = -\tfrac{v_r v_t}{r} + A\cos(\beta) \\
& && (r(0), \theta(0), v_r(0), v_t(0)) = (1.1, 0, 0, \tfrac{1}{\sqrt{1.1}})
\end{aligned}
\right. \tag{21}
$$

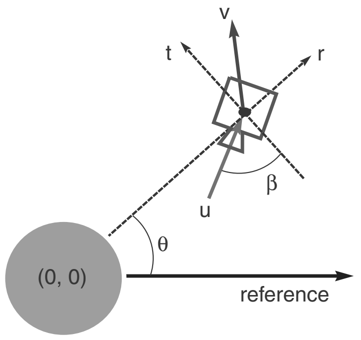

Fig. 5 Schematic of the parameters for the orbit transfer problem.

where r is the radial distance, θ is the true anomaly, v_r is the radial velocity, v_t is the transverse velocity, and β is the thrust steering angle. The final time is $t_f = 50$ and $A = 0.01$ (see Fig. 5).

The initialization step was successfully completed. The results of the spectral algorithm are listed in Table 2, in which $e_\beta(t)$ denotes the difference between two consecutive iterations. After four iterations, the preset tolerance ($\delta_u = 0.001$) is satisfied. The final optimal control and trajectory are shown in Fig. 6.

C. Other Stopping Criteria

It is possible to develop and implement many other stopping criteria. For example, from Theorem 1, the optimality of the discrete solution can be verified by checking the convergence property of the Nth-order interpolating polynomial sequence, which in turn can be numerically verified by the convergence property of the spectral coefficients. Another stopping criterion may be developed by monitoring the changes in the cost function: when the improvements in the cost values are sufficiently small, the iteration may be terminated.

Although it is possible to develop a large number of stopping criteria, the natural question is of which one to choose. To answer this question, it is important to recognize that each of the proposed error-estimation methods and their corresponding stopping criteria have different features. For instance, the cost function typically converges much faster than the state or control variables. Thus, if the algorithm is terminated based on the error of the cost function alone, it is possible that the states or controls may not have yet converged. That is, accuracy of the cost function does not guarantee accuracy in the state and control variables, but the converse is usually true. Furthermore, the state variables in optimal control typically converge one order faster than the control variable. These discussions appear to suggest that it is safest to use a stopping criterion based only on the convergence of the controls. Although this is theoretically sound, it is not entirely practical, because we usually desire controls to converge in the L^∞ norm, and error estimates in this norm are not only difficult to develop, but the number of nodes needed for L^∞ convergence may be excessively large [49]. Thus, a more practical approach to convergence is not to use a single stopping criterion for all problems, but to combine all possible choices to form a criteria bank. When all of the convergence criteria are met, the algorithm terminates. Selecting proper default values for a wide range of problems for the criteria bank is an open problem. Although this research is ongoing, the results so far indicate that the spectral algorithm is a viable approach for PS methods for optimal control.

Table 2 Estimated errors for problem (21) with increasing number of nodes

Iteration	Nodes	Estimated $\|e_\beta(t)\|_{L_2}$	Estimated $\frac{\|e_\beta(t)\|_{L_2}}{\|\beta(t)\|_{L_2}}$
2	40	2.300×10^{-2}	1.577×10^{-1}
3	50	3.345×10^{-3}	2.294×10^{-2}
4	60	2.976×10^{-4}	2.042×10^{-3}
5	70	2.490×10^{-5}	1.707×10^{-4}

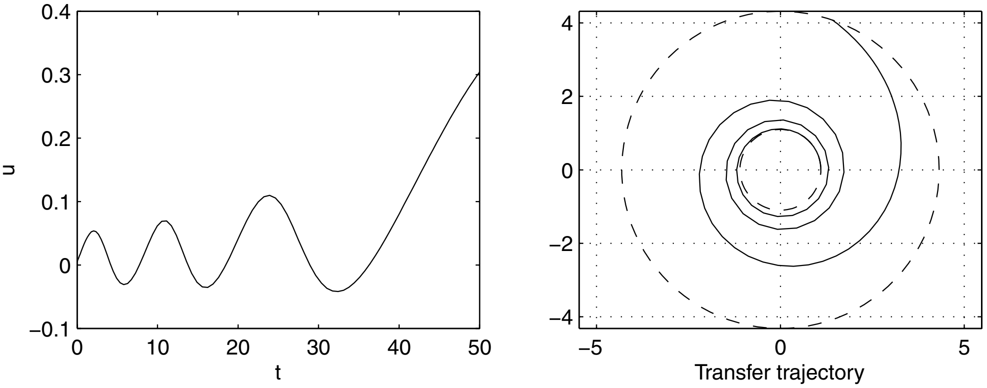

Fig. 6 Solution to problem (21) after the spectral mesh-refinement algorithm.

VI. Optimal Attitude Control of NPSAT1 Spacecraft

NPSAT1 is a multipurpose small satellite being built at the Naval Postgraduate School, and it is scheduled to be launched in 2008. It is currently in its assembly stage. One of the purposes of NPSAT1 is to serve as a testbed for advanced control algorithms. The spacecraft uses magnetic torque rods for attitude control (see Fig. 7). Because this is a highly nonlinear problem with no analytic solution for optimal control, it provides us with a practical means to test many of the ideas proposed in the previous section. Choosing the standard quaternion and body rates as the state variables, the dynamic equations of motion for NPSAT1 are given by [50]

$$\dot{q}_1(t)=\tfrac{1}{2}[\omega_x(t)q_4(t)-\omega_y(t)q_3(t)+\omega_z(t)q_2(t)+\omega_0 q_3(t)]$$

$$\dot{q}_2(t)=\tfrac{1}{2}[\omega_x(t)q_3(t)+\omega_y(t)q_4(t)-\omega_z(t)q_1(t)+\omega_0 q_4(t)]$$

$$\dot{q}_3(t)=\tfrac{1}{2}[-\omega_x(t)q_2(t)+\omega_y(t)q_1(t)+\omega_z(t)q_4(t)-\omega_0 q_1(t)]$$

$$\dot{q}_4(t)=\tfrac{1}{2}[-\omega_x(t)q_1(t)-\omega_y(t)q_2(t)-\omega_z(t)q_3(t)-\omega_0 q_2(t)]$$

$$\dot{\omega}_x(t)=\frac{I_2-I_3}{I_1}\left[\omega_y(t)\omega_z(t)-3\frac{\mu}{r_0^3}C_{23}(q(t))C_{33}(q(t))\right]+\frac{1}{I_1}[B_z(q(t),t)u_2(t)-B_y(q(t),t)u_3(t)]$$

$$\dot{\omega}_y(t)=\frac{I_3-I_1}{I_2}\left[\omega_x(t)\omega_z(t)-3\frac{\mu}{r_0^3}C_{13}(q(t))C_{33}(q(t))\right]+\frac{1}{I_2}[B_x(q(t),t)u_3(t)-B_z(q(t),t)u_1(t)]$$

$$\dot{\omega}_z(t)=\frac{I_1-I_2}{I_3}\left[\omega_x(t)\omega_y(t)-3\frac{\mu}{r_0^3}C_{13}(q(t))C_{23}(q(t))\right]+\frac{1}{I_3}[B_y(q(t),t)u_1(t)-B_x(q(t),t)u_2(t)]$$

where $\omega_0 = 0.00108$ rad/s is the angular velocity of the orbit with respect to the inertial frame; $(I_1, I_2, I_3) = (5, 5.1, 2)$ kg · m^2 are the principal moments of inertia of NPSAT1; $\mu = 3.98601\times 10^{14}$ m^3/s^2 is Earth's gravitational constant; $r_0 = 6938$ km is the distance from the mass center of NPSAT1 to the center of the Earth; and $C_{ij}(q)$ denotes the quaternion-parameterized ijth element of the matrix

$$C(q)=\begin{bmatrix} q_1^2-q_2^2-q_3^2+q_4^2, & 2(q_1q_2+q_3q_4), & 2(q_1q_3-q_2q_4) \\ 2(q_1q_2-q_3q_4), & q_2^2-q_1^2-q_3^2+q_4^2, & 2(q_2q_3+q_1q_4) \\ 2(q_1q_3+q_2q_4), & 2(q_2q_3-q_1q_4), & q_3^2-q_1^2-q_2^2+q_4^2 \end{bmatrix}\in SO(3)$$

$(B_x(q,t), B_y(q,t), B_z(q,t))$ are the components of the Earth's magnetic field in the body frame

$$[B_x(q,t), B_y(q,t), B_z(q,t)]^T = C(q)[B_1(t), B_2(t), B_3(t)]^T$$

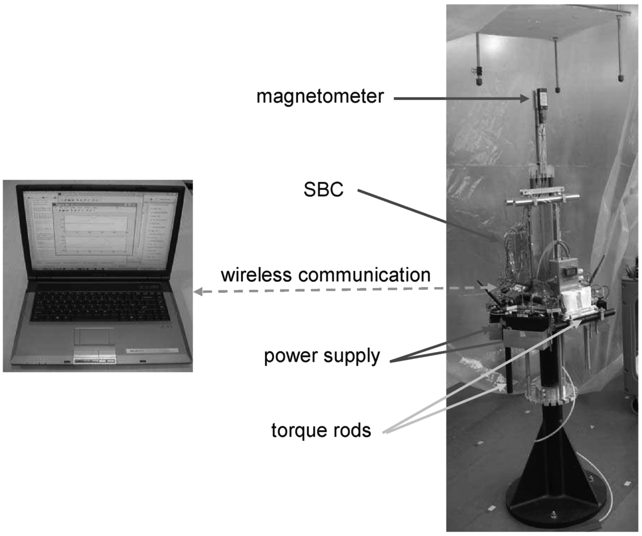

Fig. 7 NPSAT1 undergoing inertia tests at the Naval Postgraduate School.

$(B_1(t), B_2(t), B_3(t))$ are the time-varying components of the Earth's magnetic field in the orbit frame:

$$B_1=\frac{M_e}{r_0^3}[\cos(\omega_0 t)[\cos(\epsilon)\sin(i)-\sin(\epsilon)\cos(i)\cos(\omega_e t)]-\sin(\omega_0 t)\sin(\epsilon)\sin(\omega_e t)]$$

$$B_2=-\frac{M_e}{r_0^3}[\cos(\epsilon)\cos(i)+\sin(\epsilon)\sin(i)\cos(\omega_e t)]$$

$$B_3=\frac{2M_e}{r_0^3}[\sin(\omega_0 t)[\cos(\epsilon)\sin(i)-\sin(\epsilon)\cos(i)\cos(\omega_e t)]+2\cos(\omega_0 t)\sin(\epsilon)\sin(\omega_e t)]$$

where $M_e = 7.943\times 10^{15}$ Wb · m is the magnetic dipole moment of the Earth, $\epsilon = 11.7$ deg is the magnetic dipole tilt, i is the orbit inclination of NPSAT1, and $\omega_e = 7.29\times 10^{-5}$ rad/s is the spin rate of the Earth (see [50] for further details). The controls $(u_1, u_2, u_3)\in\mathbb{R}^3$ are the dipole moments on NPSAT1 that are bounded by $|u_i|\le 30$ A · m^2 ($i = 1, 2, 3$). Clearly, the dynamics of NPSAT1 are quite complex, with substantial nonlinearities. Note also that the system is not autonomous. Furthermore, the fact that the quaternions must lie on S^3 is given by the state variable constraint:

$$q_1^2(t)+q_2^2(t)+q_3^2(t)+q_4^2(t)=1$$

Thus, the NPSAT1 control system contains both state and control constraints.

A benchmark set of endpoint conditions for NPSAT1 is given by [50]

$$[q(t_0), \omega(t_0)] = [0, 0, 0, 1, 0, -0.0011, 0]$$

$$[q(t_f), \omega(t_f)] = [\sin(\phi/2), 0, 0, \cos(\phi/2), 0, 7.725 \times 10^{-4}, 7.725 \times 10^{-4}]$$

where $\phi = 135$ deg is the principal rotation angle. These endpoint conditions represent a horizon-to-horizon scan. The objective is to find the control that minimizes the slew transfer time.

A. Performance Specifications for the Spectral Algorithm

Because of the nature of minimum time control, it is reasonable to assume that the controls are discontinuous. Therefore, knots are expected to be added. Denote the state variable as

$$\boldsymbol{x} = (x_1, \ldots, x_7) = (q_1, \ldots, q_4, \omega_x, \omega_y, \omega_z)$$

Denote the number of segments as s and the number of nodes on each segment as N_i' $(i = 1, \ldots, s)$.

Denote the spectral coefficients of the state variable $x_r(t)$ on segment i as $\{a^r_{i,j}\}_{j=1,\ldots,N_i'}$. Then, on segment i, the normalized last coefficients for the state variables (averaged over the dimension of $\boldsymbol{x}$) are

$$e_i \triangleq \frac{1}{7}\sum_{r=1}^{7}\left(\frac{|a^r_{i,N_i'}|}{\max\{|a^r_{i,j}|, j = 1, \ldots, N_i'\}}\right) \qquad i = 1, \ldots, s$$

Using

$$E = \max\{e_i, i = 1, \ldots, s\}$$

as the first error indicator, we set up the desired accuracy as 10^{-5}.

Define the normalized optimal cost improvement as

$$\Delta J \triangleq \left|\frac{J^{N_j} - J^{N_{j-1}}}{J^{N_j}}\right|$$

If the improvement of the optimal cost at each iteration is less than 1%, the algorithm stops.

B. Discussion of the Results

The initialization was successful with the following parameters:

$$N = 50, \qquad J = 272.3948 \text{ s}, \qquad E = 4.4254 \times 10^{-4}$$

Following the initiation of the algorithm in Sec. IV.B, the locations of the knots were found as

$$t' = (14, 42, 59, 66, 145, 157, 166, 200, 250)$$

The results from this algorithm are illustrated in Fig. 8. The spectral mesh-generation algorithm (Sec. IV.C) results in a node distribution given by

$$N' = [10, 20, 12, 5, 56, 9, 6, 24, 36, 16]$$

and initiates a the second iteration that generates error indicators

$$E = 3.64 \times 10^{-5}, \qquad \Delta J = 0.505\%$$

Although the performance specification for J is met, it is clear that E is still larger than the desired value. This initiates a third iteration that results in the node distribution given by

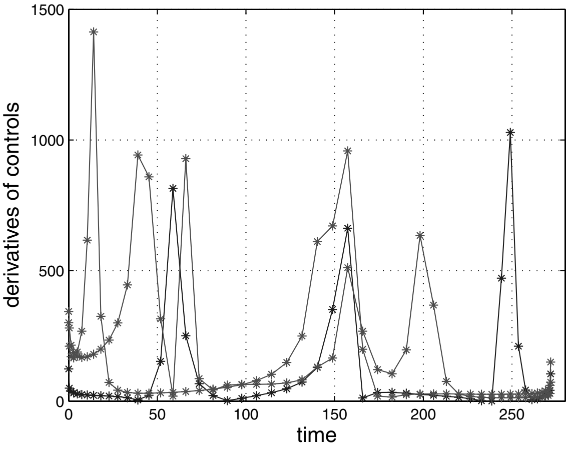

Fig. 8 Illustrating the determination of the location and number of knots.

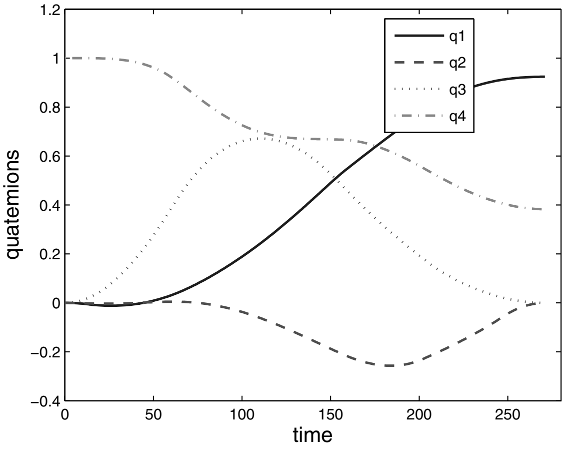

Fig. 10 Candidate optimal quaternions from the spectral algorithm.

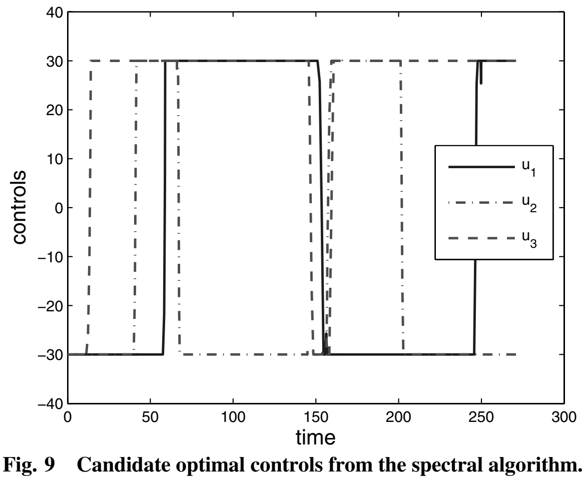

Fig. 9 Candidate optimal controls from the spectral algorithm.

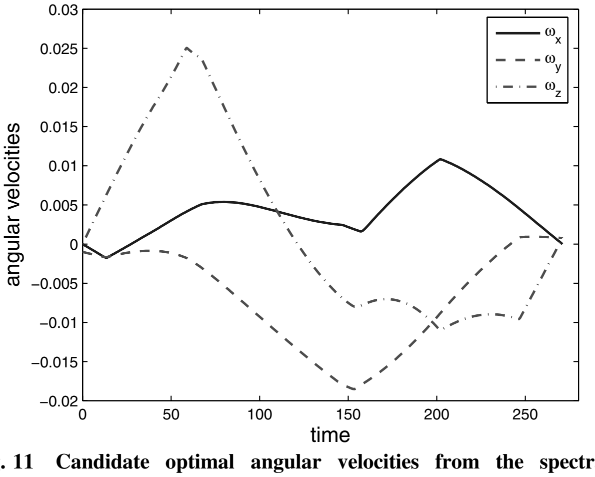

Fig. 11 Candidate optimal angular velocities from the spectral algorithm.

$$N' = [15, 30, 18, 8, 84, 14, 9, 36, 54, 24]$$

At the completion of the third iteration, we get $E = 5.29 \times 10^{-7}$ and the algorithm terminates. The final cost is $J = 270.9590$ s and the optimal control and state trajectories are shown in Figs. 9–11. These results have the same structure as those obtained by Fleming [50], who also verified the extremality of the results by way of the standard Pontryagin tests. For the sake of brevity, we omit a discussion of these tests because they are extensively discussed by Fleming.

VII. Conclusions

The rapid success of pseudospectral (PS) methods has lead to higher demands on their performance. A long-standing issue in PS methods for optimal control has been the absence of a viable algorithm for escaping Gaussian rules, which provided a predetermined node distribution while maintaining spectral accuracy. Although the pseudospectral knotting method provided an early first step along this direction, algorithmic constructions were far from complete. Theoretical advances in PS methods were necessary to advance the concept of knots. This paper has shown how recent developments in PS methods can be exploited to design a spectral algorithm for PS methods.

Given that mesh-refinement algorithms are some of the most challenging problems in computational optimal control, there is no doubt that they will continue to be so for a very long time. Although the spectral algorithm proposed in this paper solves a few long-standing problems in PS methods, it has also exposed a number of open issues in theory, computation, and algorithm design. There is no doubt that these issues will occupy researchers for a very long time. Nonetheless, it is easy to conclude that this paper has shown, for the first time, how to integrate theoretical advances in PS methods toward the construction of a heretofore-absent algorithm for spectral mesh refinement.

Acknowledgments

We gratefully acknowledge the generous funding provided by various agencies in support of this research. The research of Q. Gong was supported by the Naval Postgraduate School under grant N00244-07-1-0008. The research of I. M. Ross was supported in part by the Secretary of the U.S. Air Force, NASA, and the U.S. Air Force Office of Scientific Research under grant F1ATA0-60-6-2G002. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Air Force Office of Scientific Research or the U.S. Government.

References

[1] Ross, I. M., and Fahroo, F., *Legendre Pseudospectral Approximations of Optimal Control Problems*, Lecture Notes in Control and Information Sciences, Vol. 295, Springer–Verlag, New York, 2003, pp. 327–342.

[2] Elnagar, G., 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.
doi:10.1109/9.467672

[3] Fahroo, 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.

[4] Hawkins, A. M., Fill, T. R., Proulx, R. J., and Feron, E. M., "Constrained Trajectory Optimization for Lunar Landing," AAS Spaceflight Mechanics Meeting, Tampa, FL, American Astronautical Society Paper 06-153, Jan. 2006.

[5] Infeld, S. I., and Murray, W., "Optimization of Stationkeeping for a Libration Point Mission," AAS Spaceflight Mechanics Meeting, Maui, HI, American Astronautical Society Paper 04-150, Feb. 2004.

[6] Lu, P., Sun, H., and Tsai, B., "Closed-Loop Endoatmospheric Ascent Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 283–294.

[7] Paris, S. W., Riehl, J. P., and Sjaw, W. K., "Enhanced Procedures for Direct Trajectory Optimization Using Nonlinear Programming and Implicit Integration," AIAA/AAS Astodynamics Specialist Conference and Exhibit, Keystone, CO, AIAA Paper 2006-6309, 21–24 Aug. 2006.

[8] Pietz, J., and Bedrossian, N., "Momentum Dumping Using Only CMGs," 2003 AIAA Guidance, Navigation, and Control Conference, Austin, TX, AIAA, 2003.

[9] Rea, J., "Launch Vehicle Trajectory Optimization Using a Legendre Pseudospectral Method," Proceedings of the AIAA Guidance, Navigation and Control Conference, Austin, TX, AIAA Paper 2003-5640, Aug. 2003.

[10] Stanton, S., Proulx, R., and D'Souza, C., "Optimal Orbit Transfer Using a Legendre Pseudospectral Method," AAS/AIAA Astrodynamics Specialist Conference, Big Sky, MT, American Astronautical Society Paper 03-574, 3–7 Aug. 2003.

[11] Williams, P., "Application of Pseudospectral Methods for Receding Horizon Control," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 2., 2004, pp. 310–314.

[12] Yan, H., and Alfriend, K. T., "Three-Axis Magnetic Attitude Control Using Pseudospectral Control Law in Eccentric Orbits," AAS Spaceflight Mechanics Meeting, Tampa, FL, American Astronautical Society Paper 06-103, Jan. 2006.

[13] Kang, W., and Bedrossian, N., "Pseudospectral Optimal Control Theory Makes Debut Flight," *SIAM News*, Vol. 40, No. 7, Sept. 2007.

[14] Bedrossian, N., Bhatt, S., Lammers, M., Nguyen, L., and Zhang, Y., "First Ever Flight Demonstration of Zero Propellant Maneuver Attitude Control Concept," AIAA Guidance, Navigation, and Control Conference, AIAA Paper 2007-6734, 2007.

[15] Gong, Q., Ross, I. M., Kang, W., and Fahroo, F., "Connections Between the Covector Mapping Theorem and Convergence of Pseudospectral Methods for Optimal Control," *Computational Optimization and Applications* (to be published).

[16] Kang, W., Gong, Q., Ross, I. M., and Fahroo, F., "On the Convergence of Nonlinear Optimal Control Using Pseudospectral Methods for Feedback Linearizable Control Systems," *International Journal of Robust and Nonlinear Control*, Vol. 17, No. 14, 2007, pp. 1251–1277.
doi:10.1002/rnc.1166

[17] Fahroo, F., and Ross, I. M., "On Discrete-Time Optimality Conditions for Pseudospectral Methods," AIAA/AAS Astrodynamics Conference, Keystone, CO, AIAA Paper 2006-6304, Aug. 2006.

[18] Gong, Q., Kang, W., and Ross, I. M., "A Pseudospectral Method for the Optimal Control of Constrained Feedback Linearizable Systems," *IEEE Transactions on Automatic Control*, Vol. 51, No. 7, July 2006, pp. 1115–1129.
doi:10.1109/TAC.2006.878570

[19] Mordukhovich, B. S., *Variational Analysis and Generalized Differentiation, 1: Basic Theory*, Fundamental Principles of Mathematical Sciences Series, Vol. 330, Springer, Berlin, 2005.

[20] Mordukhovich, B. S., *Variational Analysis and Generalized Differentiation, 2: Applications*, Fundamental Principles of Mathematical Sciences Series, Vol. 331, Springer, Berlin, 2005.

[21] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[22] Ross, I. M., and Fahroo, F., "A Perspective on Methods for Trajectory Optimization," AIAA/AAS Astrodynamics Conference, Monterey, CA AIAA Paper 2002-4727, Aug. 2002.

[23] Hager, W. W., "Numerical Analysis in Optimal Control," *International Series of Numerical Mathematics*, edited by K.-H. Hoffmann, I. Lasiecka, G. Leugering, J. Sprekels, and F. Troeltzsch, , Vol. 139, Birkhäuser, Basel, Switzerland, 2001, pp. 83–93.

[24] Hager, W. W., "Runge–Kutta Methods in Optimal Control and the Transformed Adjoint System," *Numerische Mathematik*, Vol. 87, No. 2, 2000, pp. 247–282.
doi:10.1007/s002110000178

[25] Ross, I. M., "A Roadmap for Optimal Control: The Right Way to Commute," *Annals of the New York Academy of Sciences*, Vol. 1065, New York Academy of Sciences, New York, Jan. 2006.

[26] Betts, J. T., Biehn, N., and Campbell, S. L., "Convergence of Nonconvergent IRK Discretizations of Optimal Control Problems with State Inequality Constraints," *SIAM Journal on Scientific Computing*, Vol. 23, No. 6, 2002, pp. 1981–2007.
doi:10.1137/S1064827500383044

[27] Dontchev, A. L., and Hager, W. W., "The Euler Approximation in State Constrained Optimal Control," *Mathematics of Computation*, Vol. 70, No. 233, 2001, pp. 173–203.
doi:10.1090/S0025-5718-00-01184-4

[28] Dontchev, A. L., Hager, W. W., and Veliov, V. M., "Second-Order Runge–Kutta Approximations in Control Constrained Optimal

Control," *SIAM Journal on Numerical Analysis*, Vol. 38, No. 1, 2000, pp. 202–226.
doi:10.1137/S0036142999351765

[29] Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mischenko, E. F., *The Mathematical Theory of Optimal Processes*, Wiley-Interscience, New York, 1962.

[30] Vinter, R. B., *Optimal Control*, Birkhäuser, Boston, 2000.

[31] Betts, J. T., *Practical Methods for Optimal Control Using Nonlinear Programming*, SIAM, Philadelphia, 2001.

[32] Betts, J. T., and Huffman, W. P., "Sparse Optimal Control Software SOCS," The Boeing Company, Rept. MEA-LR-085, Seattle, WA, July 1997.

[33] Ross, I. M., *A Beginner's Guide to DIDO: A MATLAB Application Package for Solving Optimal Control Problems*, Elissar Publications, Monterey, CA, 2007.

[34] Paris, S. W., and Hargraves, C. R., *OTIS 4.0 Manual*, *Boeing Space and Defense Group*, Seattle, WA, 2006.

[35] Gong, Q., and Ross, I. M., "Autonomous Pseudospectral Knotting Methods for Space Mission Optimization," American Astronautical Society Paper 06-151.

[36] Ross, I. M., and Fahroo, F., "Pseudospectral Knotting Methods for Solving Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 3, 2004, pp. 397–405.

[37] Polak, E., *Optimization: Algorithms and Consistent Approximations*, Springer–Verlag, Heidelberg, 1997.

[38] Canuto, C., Hussaini, M. Y., Quarteroni, A., and Zang, T. A., *Spectral Methods in Fluid Dynamics*, Springer–Verlag, New York, 1988.

[39] Boyd, J. P., *Chebyshev and Fourier Spectral Methods*, 2nd ed, Dover, Mineola, N. Y., 1999.

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

[41] Ross, I. M., "A Historical Introduction to the Covector Mapping Principle," *Advances in the Astronautical Sciences*, Vol. 123, Univelt, San Diego, CA, 2006, pp. 1257–1278.

[42] Fahroo, F., and Ross, I. M., "Pseudospectral Methods for Infinite Horizon Optimal Control Problems," *Journal of Guidance, Control, and Dynamics* (to be published).

[43] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Journal on Optimization*, Vol. 12, No. 4, 2002, pp. 979–1006.
doi:10.1137/S1052623499350013

[44] Ross, I. M., Fahroo, F., and Gong, Q., "A Spectral Algorithm for Pseudospectral Methods in Optimal Control," *Proceedings of the 10th International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA)*, International Institute of Informatics and Systemics , Orlando, FL, 21–25 July 2004, pp. 104–109.

[45] Sansone, G., *Orthogonal Functions*, Robert E. Krieger Publishing, Huntington, NY, 1977.

[46] Riehl, J. P., Paris, S. W., and Sjaw, W. K., "Comparison of Implicit Integration Methods for Solving Aerospace Trajectory Optimization Problems," AIAA/AAS Astrodynamics Specialists Conference and Exhibit, Keystone, CO, AIAA Paper 2006-6033, 21–24 Aug. 2006.

[47] Ross, I. M., Gong, Q., and Sekhavat, P., "Low-Thrust, High-Accuracy Trajectory Optimization," *Journal of Guidance, Control, and Dynamics* (to be published).

[48] Herman, A. L., 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.

[49] Ross, I. M., "Space Trajectory Optimization and L^1-Optimal Control Problems," *Modern Astrodynamics*, edited by P. Gurfil, Elsevier, St. Louis, MO, 2006, pp. 155–188, Chap. 6.

[50] Fleming, A., "Real-Time Optimal Slew Maneuver Design and Control," M.E. Thesis, U.S. Naval Postgraduate School, Monterey, CA, Dec. 2004.