2002-08-15

# Real-Time Computation of Neighboring Optimal Control Laws

Yan, Hui

The American Institute of Aeronautics and Astronautics (AIAA)

http://hdl.handle.net/10945/29679

AIAA Guidance, Navigation, and Control Conference and Exhibit
5-8 August 2002, Monterey, California
AIAA 2002-4657

# Real-Time Computation of Neighboring Optimal Control Laws

Hui Yan,* Fariba Fahroo† and I. Michael Ross ‡
*Naval Postgraduate School, Monterey, CA 93943*

**Feedback solutions to the neighboring optimal control problem are typically obtained by solving the matrix Riccati differential equation. In this paper, we propose a new approach based on solving linear algebraic equations in real-time. Our method is based on a pseudospectral discretization of the linear time-varying boundary value problem that arises from an application of the Minimum Principle. We show how feedback control laws can be computed without any explicit integration, construction of transition matrices or solving the matrix Riccati differential equation. This facilitates a real-time implementation of the scheme and the design of predictive guidance and control laws. A numerical example of a low thrust orbit transfer problem shows the effectiveness of the method for neighboring optimal guidance.**



## 1 Introduction

AMONG different feedback control schemes to handle deviations from nominal trajectories, neighboring optimal control (NOC) laws have been used effectively in a variety of guidance problems. Since the initial presentations by Bryson and others,[1–4] NOC laws have been used for the guidance of low-thrust trajectories[5] and advanced launch systems.[6] More recently,[7] NOC has been used to handle parameter changes in the dynamic model.

The basic idea in neighboring optimal control is to control deviations about a nominal optimal trajectory by minimizing a second order expansion of the performance index, subject to first-order expansions of the appropriate differential and terminal constraints around the nominal trajectory. From this approach, a linear state perturbation feedback control law with *time-varying gains* results. The feedback gain matrix is typically computed by using a backward sweep method which involves solving a matrix Riccati differential equation (RDE).[4] One standard practice is to solve the RDE off-line and store the gain matrices at a number of points along the nominal trajectory, and then interpolate the gains. This and other gain-scheduling techniques are manpower intensive and highly time consuming.[8] Another issue is the fact that the gain matrices can go to infinity when at least one *hard* constraint is imposed on one of the state variables, or if the final time is free. Some solutions to these problems have been proposed in Refs. [3, 9] using time-to go guidance and min-distance guidance, respectively, and in Ref. 5 these two techniques were compared by an application to a low-thrust orbit transfer problem.

Motivated by the higher demands of future astro-

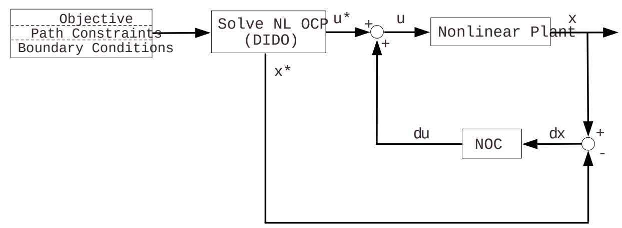


**Fig. 1 A 2-DOF Nonlinear Optimal Control System Architecture**

nautical systems, recently Lu[8] proposed the use of a receding-horizon guidance system for precision entry guidance. Implementing this type of a model-predictive guidance law requires an on-line (i.e. real-time) solution to the Riccati differential equation. To circumvent the issues inherent in solving RDEs, Lu[10] proposes a method to solve the linear-quadratic problem directly. Based on Simpson-trapezoid approximations for the integral and Euler-type approximations for the derivatives, he approximates the problem to a quadratic programming (QP) problem. The QP can be solved analytically thus yielding approximate control laws. But in his method, finding higher order control laws for step-by-step replacements for states can be too tedious.

In recent years, pseudospectral methods have emerged as a new way of rapidly solving computational optimal control problems.[11–15] When applied to linear time-varying (LTV) systems with quadratic cost, the method reduces the linear two-point-boundary-value problem to a linear system of algebraic equations. Thus, approximate analytical feedback laws can be easily and accurately obtained without solving RDEs.[16] Thus our method can be implemented in real-time. When combined with a feedforward outer-loop that uses the same pseudospectral method for nonlinear optimal control problems (NL OCP), an integrated two degree-of-freedom control system can be designed (see Figure 1). The design of the outer-loop is discussed in our companion paper, Ref. 17. The focus of this paper is the design of the inner-loop.

*Research Associate, Department of Mathematics; currently at Texas A&M. Member, AIAA. hyan@neo.tamu.edu
†Associate Professor, Department of Mathematics. Senior Member, AIAA. ffahroo@nps.navy.mil
‡Associate Professor, Department of Aeronautics and Astronautics. Associate Fellow, AIAA. imross@nps.navy.mil

The design of the inner-loop extends our prior method for solving soft terminal controllers for fixed horizons. Neighboring optimal control demands that both *hard* and *soft* boundary conditions be considered. While hard boundary conditions create additional problems in solving RDEs, it is not a stumbling block for our method. In the following sections, we outline our approach and present a numerical example.

## 2 Nominal Optimal Trajectories

The problem is to choose a control function that minimizes the performance index

$$J = \phi[\mathbf{x}(\tau_f), \tau_f] \tag{1}$$

subject to the state equations

$$\dot{\mathbf{x}} = \mathbf{f}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau_f] \tag{2}$$

and the boundary conditions

$$\boldsymbol{\varphi}[\mathbf{x}(\tau_f), \tau_f] = \mathbf{0} \tag{3}$$

where $\mathbf{x}$ is the state vector, and $\mathbf{u}$ is the control vector. The initial conditions $\mathbf{x}(\tau_0) = \mathbf{x}_0$ are given at the fixed initial time $\tau_0$ . Define the Hamiltonian

$$H = \boldsymbol{\lambda}^T \mathbf{f}[\mathbf{x}(\tau), \mathbf{u}(\tau), \tau_f] \tag{4}$$

and the auxiliary function

$$\Phi = \phi + \boldsymbol{\nu}^T \boldsymbol{\varphi} \tag{5}$$

where $\boldsymbol{\lambda}$ is the adjoint variable and $\boldsymbol{\nu}$ is a constant Lagrange multiplier. The necessary conditions for optimal control are given by the adjoint equations

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \mathbf{x}} \tag{6}$$

and the control optimality condition

$$\frac{\partial H}{\partial \mathbf{u}} = \mathbf{0} \tag{7}$$

subject to the transversality condition

$$\boldsymbol{\lambda}(\tau_f) = -\frac{\partial \Phi}{\partial \mathbf{x}}|_{\tau=\tau_f} \tag{8}$$

When the final time $\tau_f$ is free, we have the additional transversality condition

$$(\frac{\partial \Phi}{\partial \tau} + H)|_{\tau=\tau_f} = 0 \tag{9}$$

From Eqs. (3) and (8), we have

$$\boldsymbol{\psi}[\mathbf{x}(\tau_f), \boldsymbol{\lambda}(\tau_f)] = \mathbf{0} \tag{10}$$

where $\boldsymbol{\psi}$ is a $n \times 1$ vector. In all the equations, vector-valued variables are in boldface, and the partials of the scalar and vector-values quantities are gradient or Jacobian of appropriate dimensions. From these equations, it is clear that the necessary optimality conditions for the optimal control problem result in a two-point-boundary-value problem (TPBVP).

## 3 Neighboring Optimal Control

In practice the actual trajectory flown by a vehicle is never precisely the same as the pre-calculated nominal trajectory. This deviation can be attributed to the systematic and random errors in the dynamic model as well as measurement errors. One way to handle deviations from a nominal trajectory is to re-compute new optimal trajectories in flight as deviations are sensed. But this approach can prove computationally expensive and intensive. A less demanding approach is the neighboring optimal control which involves solving the corresponding linear time-varying system with a linear quadratic cost function (see Figure 1). A neighboring optimal control problem is formulated as follows:[4] Consider small perturbations from the nominal trajectory produced by small perturbations in the initial state, $\boldsymbol{\delta}\mathbf{x}(\tau_0)$ and in the terminal conditions, $\boldsymbol{\delta\psi}$. Such perturbations will give rise to perturbations $\boldsymbol{\delta}\mathbf{x}, \boldsymbol{\delta\lambda}, \boldsymbol{\delta}\mathbf{u}$ that are governed by equations that are obtained by linearizing Eqs.(2-10) around the nominal trajectory

$$\begin{aligned}
\boldsymbol{\delta}\dot{\mathbf{x}} &= \mathbf{f}_{\mathbf{x}}\boldsymbol{\delta}\mathbf{x} + \mathbf{f}_{\mathbf{u}}\boldsymbol{\delta}\mathbf{u} && (11)\\
\boldsymbol{\delta}\dot{\boldsymbol{\lambda}} &= -H_{\mathbf{xx}}\boldsymbol{\delta}\mathbf{x} - \mathbf{f}_{\mathbf{x}}^T\boldsymbol{\delta\lambda} - H_{\mathbf{xu}}\boldsymbol{\delta}\mathbf{u} && (12)\\
\mathbf{0} &= H_{\mathbf{ux}}\boldsymbol{\delta}\mathbf{x} + \mathbf{f}_{\mathbf{u}}^T\boldsymbol{\delta\lambda} + H_{\mathbf{uu}}\boldsymbol{\delta}\mathbf{u} && (13)\\
\boldsymbol{\delta}\mathbf{x}(\tau_0) &= \boldsymbol{\delta}\mathbf{x}_0 && (14)\\
\mathbf{0} &= [\boldsymbol{\psi}_{\mathbf{x}}\boldsymbol{\delta}\mathbf{x} + \boldsymbol{\psi}_{\boldsymbol{\lambda}}\boldsymbol{\delta\lambda}]|_{\tau=\tau_f} && (15)
\end{aligned}$$

Eqs. (11-15) represent a linear TPBVP since the coefficients are evaluated on the nominal trajectory. When $H_{\mathbf{uu}}$ is nonsingular for $\tau_0 \leq \tau \leq \tau_f$, we may solve Eq. (13) for $\boldsymbol{\delta}\mathbf{u}(\tau)$ in terms of $\boldsymbol{\delta\lambda}(\tau)$ and $\boldsymbol{\delta}\mathbf{x}(\tau)$ :

$$\boldsymbol{\delta}\mathbf{u}(\tau) = -H_{\mathbf{uu}}^{-1}(H_{\mathbf{ux}}\boldsymbol{\delta}\mathbf{x}(\tau) + \mathbf{f}_{\mathbf{u}}^T\boldsymbol{\delta\lambda}(\tau)) \tag{16}$$

Substituting Eq. (16) into Eqs. (11-12), we have

$$\begin{aligned}
\boldsymbol{\delta}\dot{\mathbf{x}} &= A(\tau)\boldsymbol{\delta}\mathbf{x} + B(\tau)\boldsymbol{\delta\lambda} && (17)\\
\boldsymbol{\delta}\dot{\boldsymbol{\lambda}} &= -Q(\tau)\boldsymbol{\delta}\mathbf{x} - A^T(\tau)\boldsymbol{\delta\lambda} && (18)
\end{aligned}$$

where

$$\begin{aligned}
A(\tau) &= \mathbf{f}_{\mathbf{x}} - \mathbf{f}_{\mathbf{u}}H_{\mathbf{uu}}^{-1}H_{\mathbf{ux}} && (19)\\
B(\tau) &= -\mathbf{f}_{\mathbf{u}}H_{\mathbf{uu}}^{-1}f_{\mathbf{u}}^T && (20)\\
Q(\tau) &= H_{\mathbf{xx}} - H_{\mathbf{xu}}H_{\mathbf{uu}}^{-1}H_{\mathbf{ux}} && (21)
\end{aligned}$$

Equations (17-18) subject to the initial conditions (14) and final time conditions (15) form a linear two-point value problem for $\boldsymbol{\delta}\mathbf{x}(\tau)$ and $\boldsymbol{\delta\lambda}(\tau)$.

## 4 Discretization of the NOC Problem

The traditional techniques in solving the two-point boundary value problem posed above are either using transition matrices or the sweep method. In the transition matrices method, the gain matrix can become infinite as $\tau \to \tau_f$, and in the sweep method the Riccati solution can become infinite when the final time is free. Although various solutions and approaches to resolve these issues have been proposed, they are, at best, computationally expensive or at worst unstable.

The approach in this paper is to first discretize the linear boundary value problem using a Legendre pseudospectral method. It will be shown that the problem is discretized to a coupled set of linear algebraic equations. These equations can then be efficiently and accurately solved using linear algebra techniques. The ultimate goal is to derive a feedback law that maps the initial states to the control, and our method presents this law without any explicit integration or use of Riccati solutions and in the process avoids the pitfalls associated with these techniques.

### 4.1 Legendre Pseudospectral Approximations

The basic idea of this method is to seek approximations for the state, costate and control functions in terms of their values at some carefully chosen node points. The approximation space is typically polynomial and the node points belong to the class of Gauss quadrature points which yield *optimal* results in polynomial interpolation and approximation of integrals. In Legendre pseudospectral method, the node points are Legendre-Gauss-Lobatto (LGL) points which have fixed end-points at $-1$ and 1, and the interior points are the extrema of the Legendre polynomials which are orthogonal on the interval $[-1, 1]$, with respect to the weight function $w(t) = 1$. The approximation polynomials are the interpolating polynomials with the values of the functions as the coefficients of expansions. Since the node points lie in the computational interval $[-1, 1]$, before performing any approximation, the problem is transformed to this interval by the linear transformation for

$$t \in [t_0, t_N] = [-1, 1] : \tau = \frac{(\tau_f - \tau_0)t + (\tau_f + \tau_0)}{2} \tag{22}$$

It follows that Eqs. (17-18) and Eqs.(14) and (15) can be transformed to

$$\boldsymbol{\delta}\dot{\mathbf{x}}(t) = \frac{\tau_f - \tau_0}{2}\left[A(t)\boldsymbol{\delta}\mathbf{x}(t) + B(\tau)\boldsymbol{\delta\lambda}(t)\right] \tag{23}$$

$$\boldsymbol{\delta}\dot{\boldsymbol{\lambda}} = -\frac{\tau_f - \tau_0}{2}\left[Q(t)\boldsymbol{\delta}\mathbf{x}(t) + A^T(t)\boldsymbol{\delta\lambda}(t)\right] \tag{24}$$

$$\boldsymbol{\delta}\mathbf{x}(-1) = \boldsymbol{\delta}\mathbf{x}_0 \tag{25}$$

$$\boldsymbol{\psi}_{\mathbf{x}}\boldsymbol{\delta}\mathbf{x}(1) + \boldsymbol{\psi}_{\boldsymbol{\lambda}}\boldsymbol{\delta\lambda}(1) = \mathbf{0} \tag{26}$$

In the equations above all the functions and functional such as $\delta\mathbf{x}(t)$ are actually $\delta\mathbf{x}(\tau(t))$ where $\tau(t)$ is given by (22).

Let $L_N(t)$ be the Legendre polynomial of degree $N$ on the interval $[-1, 1]$. In the Legendre pseudospectral approximation[11]-[18] of Eqs.(23)-(26), we use the LGL points $t_l, l = 0, \ldots, N$ which are given by

$$t_0 = -1, \quad t_N = 1$$

and for $1 \leq l \leq N-1$, $t_l$ are the zeros of $\dot{L}_N$, the derivative of the Legendre polynomial, $L_N$.

We start by approximating the continuous state and control variables by $N$th degree polynomials of the form

$$\boldsymbol{\delta}\mathbf{x}(t) \approx \boldsymbol{\delta}\mathbf{x}^N(t) = \sum_{l=0}^{N} \boldsymbol{\delta}\mathbf{x}_l\phi_l(t) \tag{27}$$

$$\boldsymbol{\delta\lambda}(t) \approx \boldsymbol{\delta\lambda}^N(t) = \sum_{l=0}^{N} \boldsymbol{\delta\lambda}_l\phi_l(t) \tag{28}$$

where, for $l = 0, 1, ..., N$

$$\phi_l(t) = \frac{1}{N(N+1)L_N(t_l)}\frac{(t^2-1)\dot{L}_N(t)}{t - t_l} \tag{29}$$

are the Lagrange polynomials of order $N$ which interpolate the functions at the LGL points. The interpolating polynomials satisfy the condition,

$$\phi_l(t_k) = \delta_{lk} = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{if } l \neq k \end{cases}$$

It follows that for collocation at the LGL points

$$\boldsymbol{\delta}\mathbf{x}^N(t_l) = \tilde{\mathbf{x}}_l, \;\; \boldsymbol{\delta\lambda}^N(t_l) = \tilde{\boldsymbol{\lambda}}_l \tag{30}$$

To carryout the approximation of the state equations, we impose the condition that the approximations above satisfy the differential equations at the node points. To express the derivative $\boldsymbol{\delta}\dot{\mathbf{x}}^N(t)$ in terms of $\boldsymbol{\delta}\mathbf{x}^N(t)$ at the points $t_k$, we differentiate (27) and evaluate the result at $t_k$ to obtain a matrix multiplication of the following form:

$$\boldsymbol{\delta}\dot{\mathbf{x}}^N(t_k) = \sum_{l=0}^{N}\tilde{\mathbf{x}}_l\dot{\phi}_l(t_k) = \sum_{l=0}^{N} D_{kl}\tilde{\mathbf{x}}_l \tag{31}$$

For LGL points $D_{kl} = \dot{\phi}_l(t_k)$ are the entries of the $(N+1)\times(N+1)$ differentiation matrix $\mathbf{D}$

$$\mathbf{D} := [D_{kl}] := \begin{cases} \frac{L_N(t_k)}{L_N(t_l)}\cdot\frac{1}{t_k - t_l} & k \neq l \\ -\frac{N(N+1)}{4} & k = l = 0 \\ \frac{N(N+1)}{4} & k = l = N \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

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

In any event, the state and costate equations are transformed into the following algebraic equations for $k = 0, \ldots, N$,

$$\sum_{l=0}^{N} D_{kl}\tilde{\mathbf{x}}_l - \frac{\tau_f - \tau_0}{2}(A_k\tilde{\mathbf{x}}_k + B_k\tilde{\boldsymbol{\lambda}}_k) = \mathbf{0} \tag{33}$$

$$\sum_{l=0}^{N} D_{kl}\tilde{\boldsymbol{\lambda}}_l + \frac{\tau_f - \tau_0}{2}(Q_k\tilde{\mathbf{x}}_k + A_k^T\tilde{\boldsymbol{\lambda}}_k) = \mathbf{0} \tag{34}$$

where for a generic matrix $A(t)$, the notation $A_k$ denotes $A(t_k)$. Also, the boldface $\mathbf{0}$ represents the zero vector of appropriate dimension. Writing these equations in block matrix notation, for

$$\mathbf{X} = [\tilde{\mathbf{x}}_0^T, \tilde{\mathbf{x}}_1^T, \ldots, \tilde{\mathbf{x}}_N^T]^T,\ \mathbf{\Lambda} = [\tilde{\boldsymbol{\lambda}}_0^T, \tilde{\boldsymbol{\lambda}}_1^T, \ldots, \tilde{\boldsymbol{\lambda}}_N^T]^T \quad (35)$$

we have

$$\tilde{A}_- \mathbf{X} - \frac{\tau_f - \tau_0}{2} \tilde{G} \mathbf{\Lambda} = \mathbf{0} \quad (36)$$

$$\frac{\tau_f - \tau_0}{2} \tilde{Q} \mathbf{X} + \tilde{A}_+ \mathbf{\Lambda} = \mathbf{0} \quad (37)$$

Where $\tilde{A}_-, \tilde{A}_+, \tilde{G}, \tilde{Q}$ are $[n(N+1) \times n(N+1)]$ matrices whose $(ij)$th blocks are $n \times n$ matrices of the following form

$$[\tilde{A}_-]_{ij} = \begin{cases} D_{ij} I_n, & i \neq j \\ D_{ii} I_n - \frac{\tau_f - \tau_0}{2} A_i & i = j \end{cases}$$

$$[\tilde{A}_+]_{ij} = \begin{cases} D_{ij} I_n, & i \neq j \\ D_{ii} I_n + \frac{\tau_f - \tau_0}{2} A_i^T & i = j \end{cases}$$

$$[\tilde{G}]_{ij} = \begin{cases} 0_n, & i \neq j \\ B_i, & i = j \end{cases}$$

$$[\tilde{Q}]_{ij} = \begin{cases} 0_n, & i \neq j \\ Q_i & i = j \end{cases}$$

In the above, $I_n$ and $0_n$ are the $n \times n$ identity and zero matrices, respectively. The initial and final conditions are

$$\tilde{\mathbf{x}}_0 = \boldsymbol{\delta}\mathbf{x}_0 \quad (38)$$

$$\boldsymbol{\psi}_{\mathbf{x}} \tilde{\mathbf{x}}_N + \boldsymbol{\psi}_{\boldsymbol{\lambda}} \tilde{\boldsymbol{\lambda}}_N = \mathbf{0} \quad (39)$$

The goal is to solve Eqs.(36) and (37) subject to the transversality conditions Eqs. (38) and (39). Therefore, first we write the equations for the values of the state and costate vectors at the nodes, $\mathbf{X}$ and $\mathbf{\Lambda}$, in block matrix form

$$\begin{bmatrix} \tilde{A}_- & -\frac{\tau_f - \tau_0}{2} \tilde{G} \\ \frac{\tau_f - \tau_0}{2} \tilde{Q} & \tilde{A}_+ \\ \tilde{P} & \tilde{I} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{\Lambda} \end{bmatrix} \equiv V\mathbf{Z} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (40)$$

In these equations $\mathbf{Z}^T = [\mathbf{X}^T, \mathbf{\Lambda}^T]$ and $\tilde{P}$ and $\tilde{I}$ are the following $n \times n(N+1)$ matrices

$$\tilde{P} = [0_n, \ldots, 0_n, \boldsymbol{\psi}_{\mathbf{x}}] \quad (41)$$

$$\tilde{I} = [0_n, \ldots, 0_n, \boldsymbol{\psi}_{\boldsymbol{\lambda}}] \quad (42)$$

The matrix $V$ in Eq.(40) is of dimension $n(2N+3) \times 2n(N+1)$. We partition $V$ as $V = [V_0 \quad V_e]$ such that

$$V_0 \tilde{\mathbf{x}}_0 + V_e \mathbf{Z}_e = \mathbf{0} \quad (43)$$

where vector $\mathbf{Z}_e$ is of dimension $n(2N+1) \times 1$ and is defined as

$$\mathbf{Z}_e = [\tilde{\mathbf{x}}_1^T, \tilde{\mathbf{x}}_2^T, \ldots, \tilde{\mathbf{x}}_N^T, \tilde{\boldsymbol{\lambda}}_0^T, \ldots, \tilde{\boldsymbol{\lambda}}_N^T]^T \quad (44)$$

Thus, $V_0$ and $V_e$ are $[n(2N+3) \times n]$, $[n(2N+3) \times n(2N+1)]$ block matrices of $V$, respectively. We can solve Eq. (43) for $\mathbf{Z}_e$ as

$$\mathbf{Z}_e = -V_e \backslash V_0 \tilde{\mathbf{x}}_0 = W \tilde{\mathbf{x}}_0 \quad (45)$$

where, the \ operator denotes the least-squares solution in MATLAB. As indicated in Eq. (45), $W \equiv -V_e \backslash V_0$ is a matrix of dimension $(2nN+n) \times n$. Since $\mathbf{Z} = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{Z}_e \end{bmatrix}$ we get

$$\mathbf{Z} = \begin{bmatrix} \mathbf{X} \\ \mathbf{\Lambda} \end{bmatrix} = \begin{bmatrix} I_n \\ W \end{bmatrix} \tilde{\mathbf{x}}_0 \equiv \begin{bmatrix} W_1 \\ W_2 \end{bmatrix} \tilde{\mathbf{x}}_0$$

where $W_1$ and $W_2$ are partitions of the $[I_n \quad W]$ matrix, each of dimension $n(N+1) \times n$ so that we have,

$$\tilde{\mathbf{x}}_k = W_{1k} \tilde{\mathbf{x}}_0 \quad (46)$$

$$\tilde{\boldsymbol{\lambda}}_k = W_{2k} \tilde{\mathbf{x}}_0 \quad (47)$$

Where $W_{1k}$ and $W_{2k}$ are partitions of dimension $n \times n$ of the matrices $W_1$ and $W_2$, respectively. The subscript $k$ refers to the $k$th LGL point. To derive a discretization for the optimal control law, Eq. (16) can be evaluated at the shifted LGL points, $\tau_k$

$$\boldsymbol{\delta}\mathbf{u}(\tau_k) = -H_{uu}^{-1} (H_{ux} \boldsymbol{\delta}\mathbf{x}(\tau_k) + \mathbf{f}_u^T \boldsymbol{\delta}\boldsymbol{\lambda}(\tau_k)) \quad (48)$$

Substituting Eqs.(46-47) into Eq.(48) and using the notation in Eq.(30) we have

$$\boldsymbol{\delta}\mathbf{u}(\tau_k) = -H_{uu}^{-1} \left( H_{ux} W_{1k} + \mathbf{f}_u^T W_{2k} \right) \boldsymbol{\delta}\mathbf{x}_0 \quad (49)$$

The values of the control at instants of time between the LGL points can be obtained by interpolation.

Replacing $\tau_0$ by the current time, $\tau$, we obtain a sampled data feedback law. Thus we can form a predictive controller for a fixed- horizon where the horizon is the time-to-go. Since these controllers are obtained without any explicit integration or construction of transition matrices, it is apparent that they can be computed in real-time.

The following example of low-thrust guidance illustrates the technique and numerically demonstrates the accuracy and stability of our approach. See Ref. 17 for a discussion of the outer-loop for this same problem.

## 5 Example: Low-Thrust Guidance

The example is the well known Earth-Mars transfer from Ref. 4. Consider the problem of determining the optimal trajectory and thrust direction history to transfer a rocket from an initial circular orbit to the largest circular orbit. The variables are: $r$ the radial distance, $v_r$ the radial component of velocity, $v_t$

the tangential component of velocity, $m$ the mass of spacecraft, $\dot{m}$ the constant fuel consumption rate, $\mu$ the gravitational constant and $\epsilon$ , the thrust steering angle measured from the local horizontal. The control problem is formulated as finding $\epsilon(\tau)$ to maximize $r(\tau_f)$ subject to the equations of motion

$$\frac{dr}{d\tau} = v_r, \tag{50}$$

$$\frac{dv_r}{d\tau} = \frac{v_t^2}{r} - \frac{\mu}{r^2} + A(\tau)\sin\epsilon, \tag{51}$$

$$\frac{dv_t}{d\tau} = -\frac{v_r v_t}{r} + A(\tau)\cos\epsilon, \tag{52}$$

where

$$A(\tau) = \frac{T}{m_0 - |\dot{m}|\tau}. \tag{53}$$

The boundary conditions are

$$r(0) = 1.0LU, \quad v_r(0) = 0, \tag{54}$$

$$v_t(0) = 1.0\,LU/TU \tag{55}$$

$$v_r(\tau_f) = 0, \quad v_t(\tau_f) - \sqrt{\frac{\mu}{r(\tau_f)}} = 0. \tag{56}$$

The normalized constants in this problem are: $\mu = 1.0$, $m_0 = 1.0$, $T = 0.1405$, $\tau_f = 3.32$ and $|\dot{m}| = 0.0749$. . Let $\mathbf{x} = [r, v_r, v_t]^T$ be the state vector and $u = \epsilon$. The difference between the actual state and control variables from their nominal values due to perturbations in the initial states is denoted by $\boldsymbol{\delta}\mathbf{x}(\tau)$ and $\delta u(\tau)$. The time-to-go guidance scheme for the neighboring optimal control can be outlined as follows: First we calculate a nominal optimal trajectory $\mathbf{x}^*$ and optimal control $u^*(\tau)$ using optimal control theory and shooting methods to maximize $r(\tau_f)$ subject to Eqs.(50-56). We define this optimal trajectory as the reference trajectory. We can easily compute the Jacobian matrices in Eq. (49) at each $\tau$. Then, we set the initial perturbation $\boldsymbol{\delta}\mathbf{x}_0$. The actual (perturbed) trajectory is then controlled by $u = \delta u + u^*$ with the nonlinear dynamics governed by Eqs.(50-52), where the asterisk denotes the reference value. The next perturbations $\boldsymbol{\delta}\mathbf{x}$ are generated from $\boldsymbol{\delta}\mathbf{x} = \mathbf{x} - \mathbf{x}^*$, where $\mathbf{x}$ is the state response from system Eqs. (50-52) with $u = \delta u + u^*$. In other words, $\boldsymbol{\delta}\mathbf{x}$ is not generated from Eqs. (11-12) and the feedback control law Eq.(49) is used only for the control law. We repeat this procedure to the final time.

In Fig. 2 , the optimal nominal control and states calculated using a shooting method are shown . The accuracy for the shooting method is of order of $10^{-12}$. We set the number of LGL points at 64 and the sampling time interval at 0.02 seconds. The initial perturbations are shown in Table 1, which are equivalent to those in Ref. 5. Figures 3-6 depict the feedback laws and the variations of the states compared with the nominal values. The control $\delta u \to 0$ as $\tau \to \tau_f$ as shown in Fig. 3. Notice the example includes a hard

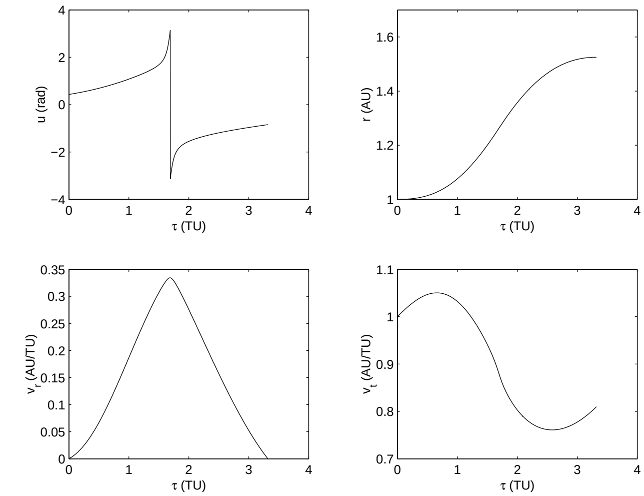


**Fig. 2 The nominal control and states**

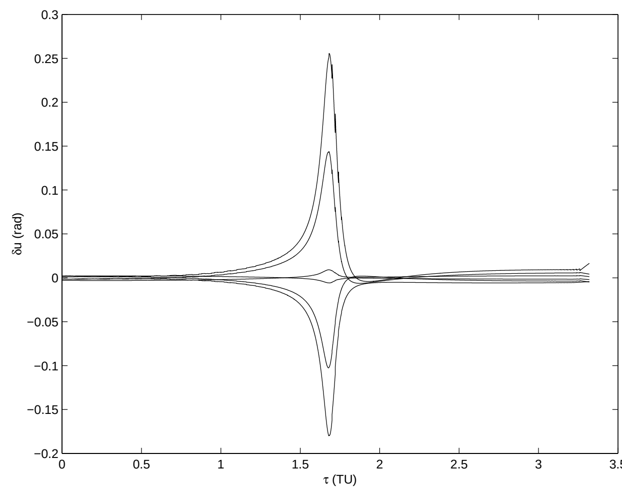


**Fig. 3 The feedback control law error**

terminal constraint for $v_r$ and soft constraints for $r$ and $v_t$ in Eq. (56). So, $\delta r$ and $\delta v_t$ do not approach zero but $\delta v_r \to 0$ as $\tau \to \tau_f$ due to the hard constraint as illustrated in Figs. 4-6.

**Table 1 Initial Perturbations**

| Cases | $\boldsymbol{\delta}\mathbf{x}_0$ |
|---|---|
| 1 | (0.0033, 0, 0) |
| 2 | (−0.0023, 0, 0) |
| 3 | (0, 0.0033, 0) |
| 4 | (0, −0.0023, 0) |
| 5 | (0, 0 , 0.0033) |
| 6 | (0, 0, −0.0023) |

To verify the accuracy for the neighboring optimal control using the spectral method, the reference trajectory is recomputed with the perturbed initial conditions, $\mathbf{x}_0^* + \boldsymbol{\delta}\mathbf{x}_0$. Define the states obtained by re-optimizations as $\mathbf{x}_{reo}$. We illustrate the error between the actual states (computed using the perturbed control $u^* + \delta u$) and the re-optimized states in Figs. 7-9.

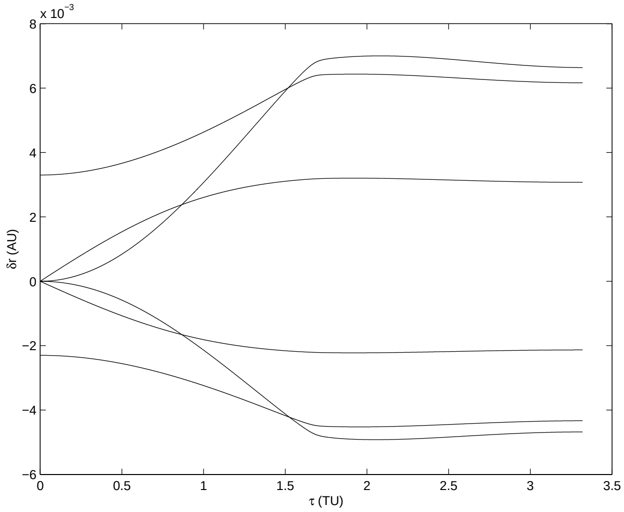


**Fig. 4 Deviations for radial distance**

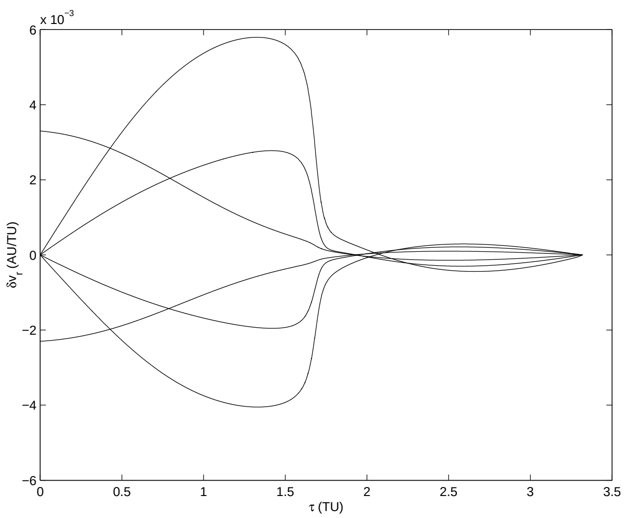


**Fig. 5 Deviations for the radial component of velocity**

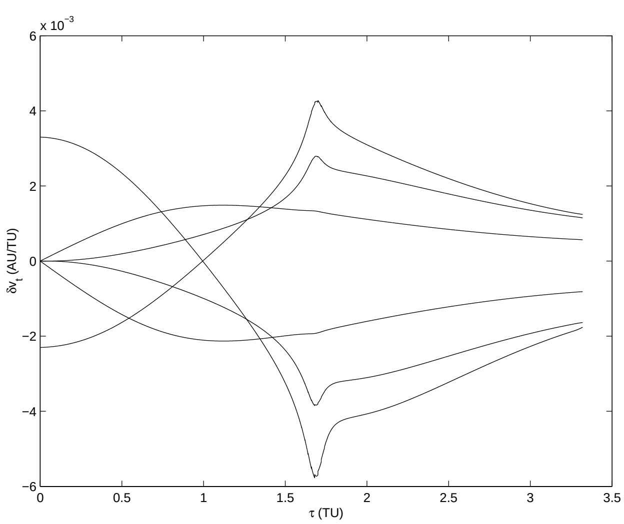


**Fig. 6 Deviations for the tangential component of velocity**

**Table 2 Final REO Errors**

| Cases | $\boldsymbol{\delta} r_{reo}$ | $\boldsymbol{\delta} v_{r_{reo}}$ | $\boldsymbol{\delta} v_{t_{reo}}$ |
|---|---|---|---|
| 1 | -5.08e-6 | -1.44e-7 | -3.56e-6 |
| 2 | -2.64e-6 | -2.96e-8 | -1.65e-6 |
| 3 | -3.57e-6 | -3.77e-9 | -1.44e-7 |
| 4 | -1.79e-6 | 3.14e-8 | 3.87e-8 |
| 5 | -4.00e-6 | -1.66e-6 | -5.13e-6 |
| 6 | -1.66e-6 | -3.29e-8 | -2.34e6 |

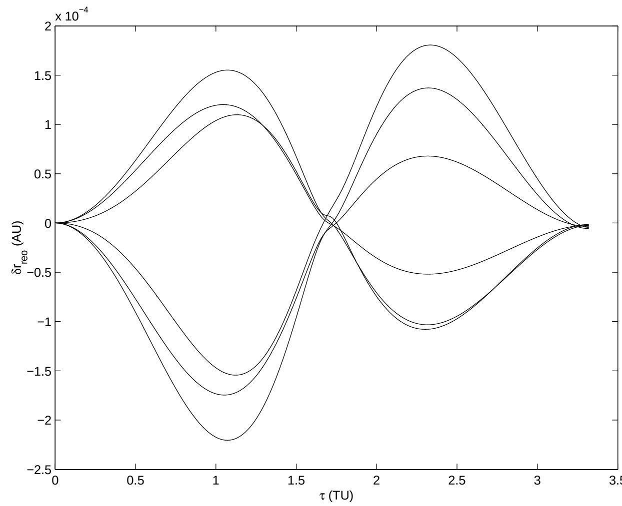


**Fig. 7 Re-optimized deviations for radial distance**

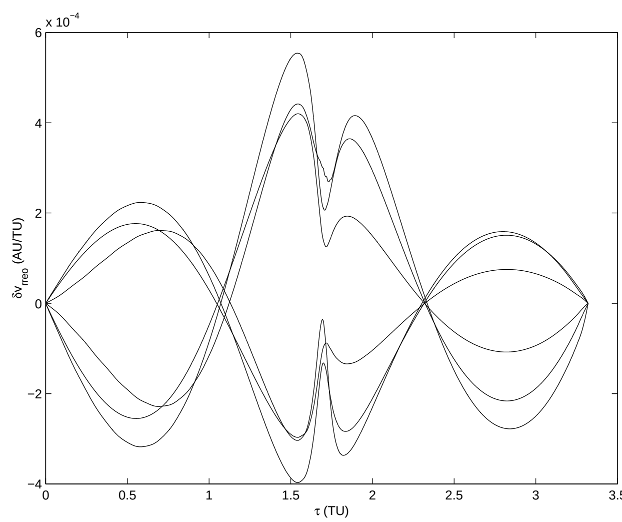


**Fig. 8 Re-optimized deviations for radial velocity**

From the figures, one can see the deviations are very close to zero at the final time. In Table 2, the states' error at final time for different initial perturbations are presented. The results illustrate that actual final states are in excellent agreement with the re-optimized final states which indicate accuracy and effectiveness of our technique in solving and applying neighboring optimal control laws.

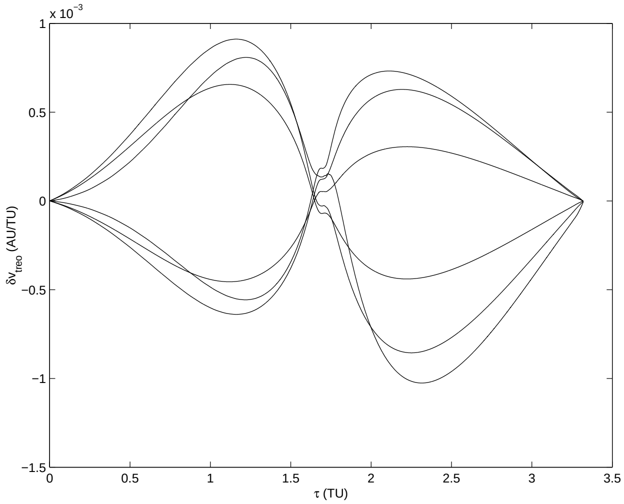


**Fig. 9 Re-optimized deviations for tangential velocity**

## 6 Conclusions

Pseudospectral methods can be effectively used in an integrated fashion for the design of a high-performance two degree-of-freedom (DOF) guidance and control system. The proposed "spectral method" uses the same discretization for the designs of the outer- and inner-loops. The outer-loop employs a direct method while the inner-loop uses a modified indirect method. Speed, efficiency and robustness of the inner-loop is obtained by a predictive control method that circumvents the real-time-solving of a matrix Riccati differential equation. Since the outer-loop manages "slow" dynamics, an online "concurrent" parameter identification model can be easily incorporated to facilitate a 2-DOF adaptive system. It is apparent that this research has opened many new vistas in the design of guidance and control systems.

## References

[1]Breakwell, J.V., Speyer, J.L., and Bryson, A.E. Jr., "Optimization and Control of Nonlinear Systems Using the Second Variation," *SIAM Journal on Control,* Vol. 1, No. 2, 1963, pp. 193–223.

[2]Kelley, H.J., "An Optimal Guidance Approximation Theory," *IEEE Transactions on Automatic Control,* Vol. AC-9, Oct. 1964, pp. 375–380.

[3]Speyer, J.L. and Bryson, A.E. Jr., "A Neighboring Optimum Feedback Control Scheme Based on Time-to-Go with Application to Re-Entry Flight Paths," *AIAA Journal,* Vol. 6, May 1968, pp. 769–776.

[4]Bryson, A.E., Jr., and Ho., Y.C., *Applied Optimal Control,* Hemisphere New York, 1975.

[5]Bauer, T.P., Wood, L.J., and Caughey, T.K., "Gain Indexing Schemes for Low-thrust Perturbation Guidance," *Journal of Guidance, Control, and Dynamics,* Vol. 6, No. 6, 1983, pp. 518–525.

[6]Seywald, H., and Cliff, E.M., " Neighboring Optimal Control Based Feedback Law for the Advanced Launch System," *Journal of Guidance, Control, and Dynamics,* Vol. 17, No. 6, 1994, pp.1154–1162.

[7]Jardin, M. R., and Bryson, A.E., " Neighboring Optimal Aircraft Guidance in Winds," *Journal of Guidance, Control, and Dynamics,* Vol. 24, No. 4, 2001, pp.710–715.

[8]Lu., P, "Regulation About Time-Varying Trajectories: Precision Entry Guidance Illustrated," *Journal of Guidance, Control, and Dynamics,* Vol. 22, No. 6, 1999, pp. 784–790.

[9]Powers, W. F., "A Method for Comparing Trajectories in Optimal Linear Perturbation Guidance Schemes," *AIAA Journal,* Vol. 6, Dec 1968, pp. 2451–2452.

[10]Lu., P, "Approximate Nonlinear Receding-Horizon Control Laws in Closed Form," *International Journal of Control,* Vol. 71, No. 1, 1998, pp. 19–34.

[11]Elnagar, 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.

[12]Elnagar, J. and Razzaghi, M., "A Collocation Type Method for Linear Quadratic Optimal Control Problems," *Optimal Control Applications and Methods,* Vol. 18, 1997, pp. 227–235.

[13]Fahroo, 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.

[14]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.

[15]Ross, I. M. and Fahroo, F., "A Direct Method for Solving Nonsmooth Optimal Control Problems," *Proceedings of the 2002 World Congress of The International Federation on Automatic Control*, Barcelona, Spain, July 2002.

[16]Yan, H., Fahroo, F., and Ross, I.M., "Optimal Feedback Control laws by Pseudospectral Approximations," *Proceedings of the 2001 American Control Conference,* Arlington, VA, June 2001.

[17]Strizzi, J., Ross, I. M. and Fahroo, F. "Towards Real-Time Computation of Optimal Controls for Nonlinear Systems," *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* Monterey, CA, August 2002, AIAA Paper No. 2002-4945.

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