OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. 2016; **00**:1–10
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/oca

Combined Homotopy and Neighboring Extremal Optimal Control

Rohit Gupta[1,∗,†], Anthony M. Bloch[2] and Ilya V. Kolmanovsky[1]

[1] *Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109-2140, USA*
[2] *Department of Mathematics, University of Michigan, Ann Arbor, MI 48109-1043, USA*

SUMMARY

This paper presents a new approach to trajectory optimization for nonlinear systems. The method exploits homotopy between a linear system and a nonlinear system and neighboring extremal optimal control, in combination with few iterations of a convergent optimizer at each step, to iteratively update the trajectory as the homotopy parameter changes. To illustrate the proposed method, a numerical example of a three dimensional orbit transfer problem for a spacecraft is presented. Copyright © 2016 John Wiley & Sons, Ltd.

KEY WORDS: iterative/numerical methods; optimal control

1. INTRODUCTION

For most optimal control problems in engineering applications, it is difficult to obtain analytical or closed form solutions using Pontryagin's maximum principle or dynamic programming. Consequently, iterative/numerical methods are utilized for solving such optimal control problems (OCPs) (see [1], [2]).

In this paper we propose a new approach to trajectory optimization of a nonlinear system with a given cost functional. The method exploits the idea of homotopy (see, e.g., [3]) to continuously deform the trajectory from that of a linear system to that of a nonlinear system, and it uses neighboring extremal optimal control (NEOC) to predict the optimal solution as the homotopy parameter changes. Note that the method presented here is different from [4] as we, additionally, exploit the idea of NEOC. The main motivation for our approach is that it is easier to solve OCPs for linear systems than for nonlinear systems. Once we obtain the optimal control for the linear system, the control is iteratively updated using NEOC theory, combined with only a few iterations of a convergent optimizer at each step. We note that while the homotopy method is used in many practical trajectory optimization methods, e.g., in aerospace applications (see [5], [6]), its use is limited to systems with contractible state space, i.e., state space with a trivial fundamental group, such as $\mathbb{R}^n$. We will briefly discuss the homotopy and NEOC next. In what follows, we will suppress the explicit dependence of the state, costate and control trajectories on time unless otherwise necessary.

∗Correspondence to: Rohit Gupta, Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109-2140, USA.
†E-mail: rohitgpt@umich.edu

2. HOMOTOPY

Homotopy is a topological concept (see, e.g., [7]), which can be used, typically in combination with another optimization method, to solve OCPs. The basic idea is to start out with a simpler problem, whose solution is easy to compute, and then gradually evolve the solution to the solution of the harder problem by changing the homotopy parameter. Consider an OCP, where the objective is to minimize a cost functional given by

$$\min_{u(.)} J = K(x(T)) + \int_0^T L(x(t), u(t))dt \tag{1}$$

subject to

$$\dot{x}(t) = f(x(t), u(t)), \;\; x(0) = x_0, \tag{2}$$

where $x(.) \in AC([0,T], \mathbb{R}^n)$, $u(.) \in L^\infty([0,T], \mathbb{R}^m)$, $K : \mathbb{R}^n \to \mathbb{R}$, $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ satisfy appropriate differentiability assumptions. Suppose the OCP (1)-(2) is difficult to solve with the dynamic constraint given by the model $\dot{x}(t) = f(x(t), u(t))$ but is easier to solve with the dynamic constraint given by the model $\dot{x}(t) = g(x(t), u(t))$ (e.g., $g(x(t), u(t)) = Ax(t) + Bu(t) + d$), where $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ also satisfies appropriate differentiability assumptions. Then by creating a homotopy given by

$$\dot{x}(t) = \lambda f(x(t), u(t)) + (1 - \lambda) g(x(t), u(t)), \tag{3}$$

where $\lambda \in [0, 1]$ is the homotopy parameter and under appropriate assumptions, we can solve the original OCP (1)-(2) by changing λ from 0 to 1 and re-using the solution from the previous homotopy step as an initial guess for the solution at the next homotopy step. For the background on homotopy methods see [4], [8]. The survey paper [9] discusses continuation methods and their application to OCPs. For the use of homotopy method in OCPs see also [10], [11], [12], [13], [14].

3. NEIGHBORING EXTREMAL OPTIMAL CONTROL

Consider a parameter dependent OCP, where the objective is to minimize a cost functional given by

$$\min_{u(.)} J = K(x(T), p) + \int_0^T L(x(t), u(t), p)dt \tag{4}$$

subject to

$$\dot{x}(t) = f(x(t), u(t), p), \;\; x(0) = x_0, \tag{5}$$

where $x(.) \in AC([0,T], \mathbb{R}^n)$, $u(.) \in L^\infty([0,T], \mathbb{R}^m)$, $p \in \mathbb{R}^l$ is a parameter, $K : \mathbb{R}^n \times \mathbb{R}^l \to \mathbb{R}$, $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l \to \mathbb{R}$ and $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l \to \mathbb{R}^n$ are functions of class C^2. Let (x_p^*, u_p^*) be a solution for the OCP (4)-(5), where $u_p^*(t)$ denotes the optimal control, which satisfies the Lagrange multiplier rule in a normal form (see, e.g., [15]). Let Ψ_p^* be the solution corresponding to $(x, u) = (x_p^*, u_p^*)$ of the following costate equation

$$\dot{\Psi} = -H_x(x, u, \Psi, p), \;\; \Psi(T) = K_x(x(T), p),$$

where $\Psi(.) \in AC([0,T], \mathbb{R}^n)$, H is the Hamiltonian and $H(x, u, \Psi, p) := L(x, u, p) + \Psi^T f(x, u, p)$. Altogether, (x_p^*, u_p^*, Ψ_p^*) satisfy the following necessary conditions for optimality

$$\dot{x}(t) = f(x(t), u(t), p), \;\; x(0) = x_0, \tag{6}$$

$$\dot{\Psi}(t) = -H_x(x(t), u(t), \Psi(t), p), \;\; \Psi(T) = K_x(x(T), p), \tag{7}$$

DOI: 10.1002/oca

$$0 = H_u(x(t), u(t), \Psi(t), p). \tag{8}$$

Suppose there is a small variation in the initial condition and/or the parameter, and we would like to update the optimal control. Instead of solving the original OCP again, we employ a first order approximation of the necessary conditions for optimality around the nominal trajectory. This approximation is given by (see, e.g., [16], [17], [18], [19])

$$\delta\dot{x}(t) = \frac{\partial f}{\partial x}\delta x(t) + \frac{\partial f}{\partial u}\delta u(t) + \frac{\partial f}{\partial p}\delta p, \;\; \delta x(0) = \delta x_0, \tag{9}$$

$$\delta\dot{\Psi}(t) = -H_{xx}\delta x(t) - H_{xu}\delta u(t) - H_{x\Psi}\delta\Psi(t) - H_{xp}\delta p, \;\; \delta\Psi(T) = K_{xx}\delta x(T) + K_{xp}\delta p, \tag{10}$$

$$0 = H_{ux}\delta x(t) + H_{uu}\delta u(t) + H_{u\Psi}\delta\Psi(t) + H_{up}\delta p. \tag{11}$$

Under the the second order sufficient optimality condition (see, e.g., [17], [19]), (9)-(11) represents the optimality condition for the following OCP (see, e.g., [16], [17], [18], [19])

$$\min_{\delta u(.)} \delta^2 J = \frac{1}{2}\begin{bmatrix}\delta x(T)\\ \delta p\end{bmatrix}^T \begin{bmatrix}K_{xx}(T) & K_{xp}(T)\\ K_{px}(T) & 0\end{bmatrix}\begin{bmatrix}\delta x(T)\\ \delta p\end{bmatrix} + \frac{1}{2}\int_0^T \left[\begin{bmatrix}\delta x(t)\\ \delta u(t)\\ \delta p\end{bmatrix}^T \begin{bmatrix}H_{xx}(t) & H_{xu}(t) & H_{xp}(t)\\ H_{ux}(t) & H_{uu}(t) & H_{up}(t)\\ H_{px}(t) & H_{pu}(t) & 0\end{bmatrix}\begin{bmatrix}\delta x(t)\\ \delta u(t)\\ \delta p\end{bmatrix}\right] dt \tag{12}$$

subject to the perturbed dynamics

$$\delta\dot{x}(t) = \frac{\partial f}{\partial x}\delta x(t) + \frac{\partial f}{\partial u}\delta u(t) + \frac{\partial f}{\partial p}\delta p, \;\; \delta x(0) = \delta x_0, \tag{13}$$

where the matrices in the cost functional (12) and the Jacobian matrices in the dynamic constraint (13) are evaluated at the nominal trajectories. The optimal control for the OCP (12)-(13) is given by

$$\delta u^*(t) = -H_{uu}^{-1}(t)\left[H_{ux}(t)\delta x(t) + f_u^T(t)\delta\Psi(t) + H_{up}(t)\delta p\right], \tag{14}$$

where all partial derivative matrices are evaluated at the nominal trajectories and $\delta\Psi(t)$ is a perturbation from $\Psi^*(t)$, ultimately expressible in terms of $\delta x(t)$ and δp.

The updated control is now calculated as the sum of $u^*(t)$ and $\delta u^*(t)$ and can be used directly or to warm start an optimizer for parameter $p + \delta p$. This is the basic idea behind NEOC. For a detailed description of NEOC see [16]. For a mathematically rigorous introduction to NEOC see [20].

Remark 1
The OCP (12)-(13) is known as the accessory minimum problem in the calculus of variations (see, e.g., [21]). If there is no variation in the initial condition, i.e., the initial condition remains fixed, then $\delta x(0) = 0$ and similarly, if there is no variation in the parameter, i.e., the parameter remains fixed, then $\delta p = 0$. Note that it is also possible to obtain the solution in the conventional NEOC setting (see, e.g., [16]), by adding p as a state, with $\dot{p} = 0$.

For $(x_p^*(t), u_p^*(t))$ to be a strong local minimizer for the OCP (4)-(5), the second order sufficient condition (strengthened Legendre-Clebsch condition) requires that $H_{uu}(t) \succ 0$, for a.e. $t \in [0, T]$ and conjugate points for the OCP (12)-(13) must not exist (Jacobi condition) (see, e.g., [20]). An indicator for the existence of conjugate points is that the Riccati equation associated with the OCP (12)-(13) has a finite escape time (see, e.g., [20]). Existence of a solution of the Riccati equation associated with the OCP (12)-(13) over the interval $[0, T]$ is enough to rule out the existence of conjugate points. For a modern exposition on conjugate points see [20], [22]. For more on conjugate points for OCPs see [11], [16], [23], [24], [25], [26], [27], [28], [29].

We will now discuss the proposed method that combines the ideas of homotopy and NEOC.

DOI: 10.1002/oca

4. METHOD DESCRIPTION

Consider a linear system and a nonlinear system given as

$$\dot{x} = Ax + Bu + d, \ \ x(0) = x_0, \tag{15}$$
$$y = Cx, \tag{16}$$
$$\dot{x} = f(x, u), \ \ x(0) = x_0, \tag{17}$$

where $x(.) \in AC([0,T], \mathbb{R}^n)$, $u(.) \in L^\infty([0,T], \mathbb{R}^m)$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{q\times n}$, $d \in \mathbb{R}^n$ and $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a function of class C^2. Create a homotopy between the linear system and the nonlinear system by

$$\dot{x} = \lambda f(x, u) + (1 - \lambda)(Ax + Bu + d) =: F(x, u, \lambda), \tag{18}$$

where $\lambda \in [0, 1]$. Note that the linear system (15) can be defined as the linearization of the nonlinear system (17) at a selected steady-state operating point (x_{op}, u_{op}), with $d = f(x_{op}, u_{op}) - Ax_{op} - Bu_{op}$. Consider a class of problems with a quadratic type cost defined over a finite horizon given by

$$J = \frac{1}{2}e^T(T)K_f e(T) + \frac{1}{2}\int_0^T [e^T(t)Qe(t) + u^T(t)Ru(t)]dt, \tag{19}$$

where K_f, $Q \succeq 0$, $R \succ 0$ and $e(t) = y(t) - y_d(t)$, with $y_d(t)$ being the desired trajectory.

Remark 2
While we introduce our ideas in the context of a specific OCP with cost functional (19), many generalizations are possible. For instance, a minimum time problem can be handled using the given approach by rescaling time and introducing final time as an additional variable to be optimized. Note that for a minimum time problem, the optimal control is usually discontinuous (at least for control affine systems with a box constraint on u) and for the proposed approach to be used practically, the cost should be "regularized" with a small control-dependent term to make the optimal control continuous (see, e.g., [30], [31]). The case when the homotopy parameter enters the cost or the cost is not quadratic can be handled as well. However, simplifications do occur in the case of quadratic costs as is apparent from the next section.

4.1. ALGORITHM

The proposed algorithm is based on applying neighboring extremal updates to predict the optimal control trajectory as $p = \lambda$ changes. Note the superscripts in the following discussion represent the iteration number.

Step 1: Start with $k = 0$ and set $\lambda^{(0)} = 0$. Solve the OCP with the cost functional (19) subject to the dynamic constraint (18). The solution to this OCP is given by

$$u^{*(0)} = -R^{-1}B^T Px^{(0)} + R^{-1}B^T r_1, \tag{20}$$

where P and r_1 are the solutions of the differential equations

$$-\dot{P} = A^T P + PA - PBR^{-1}B^T P + C^T QC, \ \ P(T) = C^T K_f C, \tag{21}$$
$$-\dot{r}_1 = (A - BR^{-1}B^T P)^T r_1 - Pd + C^T Qy_d, \ \ r_1(T) = C^T K_f y_d(T). \tag{22}$$

Note that (21) is a Riccati differential equation that does not depend on y_d and is solved backwards in time and (22) is a linear differential equation which is also solved backwards in time. Obtain $x^*_{\lambda^{(0)}}$ from $\dot{x}^{(0)} = F(x^{(0)}, u^{*(0)}, \lambda^{(0)}) = Ax^{(0)} + Bu^{*(0)}$ and $u^*_{\lambda^{(0)}}$ from (20).

Step 2: Set $k = k + 1$ and $\lambda^{(k)} = \lambda^{(k-1)} + \delta\lambda^{(k)}$, where $\delta\lambda^{(k)} > 0$ is small and solve the OCP given below

$$\min_{\delta u^{(k)}(.)} \delta^2 J^{(k)} = \frac{1}{2}\begin{bmatrix}\delta x^{(k)}(T)\\ \delta\lambda^{(k)}\end{bmatrix}^T \begin{bmatrix}C^T K_f C & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\delta x^{(k)}(T)\\ \delta\lambda^{(k)}\end{bmatrix} +$$

DOI: 10.1002/oca

$$\frac{1}{2}\int_0^T \left[\begin{bmatrix} \delta x^{(k)}(t) \\ \delta u^{(k)}(t) \\ \delta\lambda^{(k)} \end{bmatrix}^T \begin{bmatrix} H_{xx}^{(k)}(t) & H_{xu}^{(k)}(t) & H_{x\lambda}^{(k)}(t) \\ H_{ux}^{(k)}(t) & H_{uu}^{(k)}(t) & H_{u\lambda}^{(k)}(t) \\ H_{\lambda x}^{(k)}(t) & H_{\lambda u}^{(k)}(t) & 0 \end{bmatrix} \begin{bmatrix} \delta x^{(k)}(t) \\ \delta u^{(k)}(t) \\ \delta\lambda^{(k)} \end{bmatrix} \right] dt \tag{23}$$

subject to the perturbed dynamics

$$\delta\dot{x}^{(k)}(t) = A^{(k)}(t)\delta x^{(k)}(t) + B^{(k)}(t)\delta u^{(k)}(t) + G^{(k)}(t)\delta\lambda^{(k)}, \ \ \delta x^{(k)}(0) = 0, \tag{24}$$

where

$$\begin{aligned}
H_{xx}^{(k)}(t) &= \frac{\partial}{\partial x}\frac{\partial H}{\partial x}\bigg|_{(x^*_{\lambda^{(k-1)}}(t), u^*_{\lambda^{(k-1)}}(t), \lambda^{(k-1)})}, \\
H_{xu}^{(k)}(t) &= \frac{\partial}{\partial u}\frac{\partial H}{\partial x}\bigg|_{(x^*_{\lambda^{(k-1)}}(t), u^*_{\lambda^{(k-1)}}(t), \lambda^{(k-1)})}, \\
&\ \ \vdots \\
A^{(k)}(t) &= \frac{\partial F}{\partial x}\bigg|_{(x^*_{\lambda^{(k-1)}}(t), u^*_{\lambda^{(k-1)}}(t), \lambda^{(k-1)})}, \\
B^{(k)}(t) &= \frac{\partial F}{\partial u}\bigg|_{(x^*_{\lambda^{(k-1)}}(t), u^*_{\lambda^{(k-1)}}(t), \lambda^{(k-1)})}, \\
G^{(k)}(t) &= \frac{\partial F}{\partial \lambda}\bigg|_{(x^*_{\lambda^{(k-1)}}(t), u^*_{\lambda^{(k-1)}}(t), \lambda^{(k-1)})},
\end{aligned}$$

with $H(x,u,\Psi,\lambda) := \frac{1}{2}[(Cx-y_d)^T Q(Cx-y_d) + u^T Ru] + \Psi^T F(x,u,\lambda)$. The solution to the OCP (23)-(24) is given by (see, e.g., [16])

$$\delta u^{*(k)} = -H_{uu}^{-1(k)}(t)\left[H_{ux}^{(k)}(t)\delta x^{(k)} + B^{T(k)}(t)\delta\Psi^{(k)} + H_{u\lambda}^{(k)}(t)\delta\lambda^{(k)}\right], \tag{25}$$

where $\delta\Psi^{(k)} = S^{(k)}\delta x^{(k)} - r_2^{(k)}$, $S^{(k)}$ and $r_2^{(k)}$ are the solutions of the differential equations

$$-\dot{S}^{(k)} = \tilde{A}^{T(k)}(t)S^{(k)} + S^{(k)}\tilde{A}^{(k)}(t) - S^{(k)}\tilde{B}^{(k)}(t)S^{(k)} + \tilde{C}^{(k)}(t), \ \ S^{(k)}(T) = C^T K_f C, \tag{26}$$

$$-\dot{r}_2^{(k)} = (\tilde{A}^{T(k)}(t) - S^{(k)}\tilde{B}^{(k)}(t))r_2^{(k)} - (S^{(k)}\tilde{D}_1^{(k)}(t) + \tilde{D}_2^{(k)}(t))\delta\lambda^{(k)}, \ \ r_2^{(k)}(T) = 0, \tag{27}$$

where

$$\begin{aligned}
\tilde{A}^{(k)}(t) &= A^{(k)}(t) - B^{(k)}(t)H_{uu}^{-1(k)}(t)H_{ux}^{(k)}(t), \\
\tilde{B}^{(k)}(t) &= B^{(k)}(t)H_{uu}^{-1(k)}(t)B^{T(k)}(t), \\
\tilde{C}^{(k)}(t) &= H_{xx}^{(k)}(t) - H_{xu}^{(k)}(t)H_{uu}^{-1(k)}(t)H_{ux}^{(k)}(t), \\
\tilde{D}_1^{(k)}(t) &= G^{(k)}(t) - B^{(k)}(t)H_{uu}^{-1(k)}(t)H_{u\lambda}^{(k)}(t), \\
\tilde{D}_2^{(k)}(t) &= H_{x\lambda}^{(k)}(t) - H_{xu}^{(k)}(t)H_{uu}^{-1(k)}(t)H_{u\lambda}^{(k)}(t).
\end{aligned}$$

Obtain $\delta x^*_{\delta\lambda^{(k)}}$ from (24), $\delta u^*_{\delta\lambda^{(k)}}$ from (25) and $\delta\Psi^*_{\delta\lambda^{(k)}} = S^{(k)}\delta x^*_{\delta\lambda^{(k)}} - r_2^{(k)}$. Calculate $x^*_{\lambda^{(k)}} = x^*_{\lambda^{(k-1)}} + \delta x^*_{\delta\lambda^{(k)}}$, $u^*_{\lambda^{(k)}} = u^*_{\lambda^{(k-1)}} + \delta u^*_{\delta\lambda^{(k)}}$ and $\Psi^*_{\lambda^{(k)}} = \Psi^*_{\lambda^{(k-1)}} + \delta\Psi^*_{\delta\lambda^{(k)}}$.

Step 3: Repeat **Step 2** until $\lambda^{(k)} = 1$.

Following the above steps, we can obtain a sub-optimal control for a nonlinear system with a given cost functional. Note that special methods exist for solving the differential equations (26)-(27) efficiently (see, e.g., [32]). We consider a numerical example in the next section.

DOI: 10.1002/oca

Remark 3

Note that a sub-optimal control provides performance close to the optimal control, where the closeness of sub-optimal control to the optimal control performance can be controlled by controlling the rate of change of the homotopy parameter. The proposed algorithm can also be extended (under appropriate assumptions see, e.g., [17], [18], [19]) to OCPs with control input/state constraints. An alternative way to extend the proposed algorithm to OCPs with control input/state constraints is by using the penalty function approach. Moreover, the weighting factor multiplying the penalty function could be treated as an additional parameter in applying neighboring extremal predictions, so as to avoid the problem of ill-conditioning caused by starting directly with a very high value of the weighting factor.

Recall that an indicator for the existence of conjugate points is that (26) has a finite escape time. We will now give three sufficient conditions for the nonexistence of conjugate points, if the optimal control is obtained at each iteration of the proposed algorithm.

Proposition 1

Assume that $\begin{bmatrix}\tilde{C}^{(k-1)}(t) & \tilde{A}^{T(k-1)}(t)\\ \tilde{A}^{(k-1)}(t) & -\tilde{B}^{(k-1)}(t)\end{bmatrix} \succeq \begin{bmatrix}\tilde{C}^{(k)}(t) & \tilde{A}^{T(k)}(t)\\ \tilde{A}^{(k)}(t) & -\tilde{B}^{(k)}(t)\end{bmatrix}$, $H_{uu}^{(k-1)}(t) \succeq 0$ and $H_{uu}^{(k)}(t) \succeq 0$, for a.e. $t \in [0,T]$ and for $k \in \mathbb{Z}_+$, then $S^{(k-1)}(t) \succeq S^{(k)}(t)$ on the interval $[0,T]$. Moreover, if there exists a solution $S^{(k-1)}(t)$ for (26) on the interval $[0,T]$, then there exists a solution $S^{(k)}(t)$ for (26) on the interval $[0,T]$.

Proof

It is easy to verify that $\tilde{A}^{(k-1)}(t)$, $\tilde{A}^{(k)}(t)$, $\tilde{B}^{(k-1)}(t)$, $\tilde{B}^{(k)}(t)$, $\tilde{C}^{(k-1)}(t)$ and $\tilde{C}^{(k)}(t)$ are integrable on the interval $[0,T]$. It follows from Theorem 4.1.4 of [33] that $S^{(k-1)}(t) \succeq S^{(k)}(t)$ on the interval $[0,T]$. It is also easy to verify that $\tilde{B}^{(k-1)}(t) = \tilde{B}^{T(k-1)}(t) \succeq 0$, $\tilde{B}^{(k)}(t) \succeq 0$, $\tilde{C}^{(k-1)}(t) = \tilde{C}^{T(k-1)}(t)$ and $S^{(k-1)}(t) = S^{T(k-1)}(t)$ on the interval $[0,T]$. It follows from Theorem 5.7 of [34] that there exists a solution $S^{(k)}(t)$ for (26) on the interval $[0,T]$. □

Proposition 2

Assume that $\tilde{C}^{(k-1)}(t) \succeq 0$ and $H_{uu}^{(k-1)}(t) \succeq 0$, for a.e. $t \in [0,T]$ and for $k \in \mathbb{Z}_+$, then there exists a solution $S^{(k-1)}(t)$ for (26) on the interval $[0,T]$.

Proof

It is easy to verify that $\tilde{A}^{(k-1)}(t)$, $\tilde{B}^{(k-1)}(t)$ and $\tilde{C}^{(k-1)}(t)$ are integrable on the interval $[0,T]$. It is also easy to verify that $\tilde{B}^{(k-1)}(t) \succeq 0$ on the interval $[0,T]$. It follows from Theorem 4.1.6 of [33] that there exists a solution $S^{(k-1)}(t)$ for (26) on the interval $[0,T]$. □

Proposition 3

Assume that $H_{uu}^{(k-1)}(t) \succeq 0$, for a.e. $t \in [0,T]$ and for $k \in \mathbb{Z}_+$. In addition, assume that there exists $\bar{S}^{(k-1)}(.) \in AC([0,T], \mathbb{R}^{n\times n})$ on the interval $[0,T]$ such that

$$0 \succeq \dot{\bar{S}}^{(k-1)} + \tilde{A}^{T(k-1)}(t)\bar{S}^{(k-1)} + \bar{S}^{(k-1)}\tilde{A}^{(k-1)}(t) - \bar{S}^{(k-1)}\tilde{B}^{(k-1)}(t)\bar{S}^{(k-1)} + \tilde{C}^{(k-1)}(t),$$

for a.e. $t \in [0,T]$ and $\bar{S}^{(k-1)}(T) \succeq C^T K_f C$, then there exists a solution $S^{(k-1)}(t)$ for (26) on the interval $[0,T]$ and $\bar{S}^{(k-1)}(t) \succeq S^{(k-1)}(t)$ on the interval $[0,T]$.

Proof

It is easy to verify that $\tilde{A}^{(k-1)}(t)$, $\tilde{B}^{(k-1)}(t)$ and $\tilde{C}^{(k-1)}(t)$ are integrable on the interval $[0,T]$. It is also easy to verify that $\tilde{B}^{(k-1)}(t) = \tilde{B}^{T(k-1)}(t) \succeq 0$, $\tilde{C}^{(k-1)}(t) = \tilde{C}^{T(k-1)}(t)$ and $S^{(k-1)}(t) = S^{T(k-1)}(t)$ on the interval $[0,T]$. It follows from Theorem 5.8 of [34] that there exists a solution $S^{(k-1)}(t)$ for (26) on the interval $[0,T]$ and $\bar{S}^{(k-1)}(t) \succeq S^{(k-1)}(t)$ on the interval $[0,T]$. □

Remark 4

Note that the proposed algorithm only gives a prediction step and not a correction step. To improve the solution, a prediction step can be augmented by a correction step that can be implemented by a few iterations of a convergent optimizer.

DOI: 10.1002/oca

We will now present a numerical example.

5. NUMERICAL EXAMPLE

To illustrate our combined homotopy and NEOC method, we consider a three dimensional orbit transfer problem for a spacecraft from an initial circular orbit of radius R_i (km) to a final circular orbit of radius R_f (km) (see, e.g., [12]). The OCP is given below

$$\min_{u(.)} J = \frac{1}{2}(x(T)-x_d)^T K_f (x(T)-x_d) + \frac{1}{2}\int_0^{14000} u^T(t)u(t)dt \tag{28}$$

subject to

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\\ \dot{x}_4(t)\\ \dot{x}_5(t)\\ \dot{x}_6(t)\end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_1(t)x_4^2(t)\cos^2(x_5(t)) + x_1(t)x_6^2(t) - \dfrac{\mu}{x_1^2(t)} + u_1(t) \\ x_4(t) \\ -\dfrac{2x_2(t)x_4(t)}{x_1(t)} + 2x_4(t)x_6(t)\tan(x_5(t)) + \dfrac{u_2(t)}{x_1(t)\cos(x_5(t))} \\ x_6(t) \\ -\dfrac{2x_2(t)x_6(t)}{x_1(t)} - x_4^2(t)\sin(x_5(t))\cos(x_5(t)) + \dfrac{u_3(t)}{x_1(t)} \end{bmatrix}, \tag{29}$$

$$u^T(t)u(t) \le 10^{-8}, \tag{30}$$

where

$$\begin{aligned} K_f &= \mathrm{diag}(10^{-4},1,1,1,1,1),\\ x(0) = x_0 &= \left[R_e + R_i \;\; 0 \;\; 0 \;\; \sqrt{\frac{\mu}{(R_e+R_i)^3}} \;\; 0 \;\; 0\right]^T,\\ x_d &= \left[R_e + R_f \;\; 0 \;\; \frac{17\pi}{4} \;\; \sqrt{\frac{\mu}{(R_e+R_f)^3\cos^2\left(\frac{5\pi}{180}\right)}} \;\; \frac{5\pi}{180} \;\; 0\right]^T. \end{aligned}$$

In (29), $x_1 = r$ (km) (radius of orbit), $x_2 = \dot{r}$ (km/sec), $x_3 = \theta$ (rad) (azimuth angle), $x_4 = \dot{\theta}$ (rad/sec), $x_5 = \phi$ (rad) (elevation angle), $x_6 = \dot{\phi}$ (rad/sec), $u_1 = a_r$ (km/sec^2) (acceleration in the r direction), $u_2 = a_\theta$ (km/sec^2) (acceleration in the θ direction), $u_3 = a_\phi$ (km/sec^2) (acceleration in the ϕ direction), $R_e = 6378$ (km) (radius of earth) and $\mu = 398600.4$ (km^3/sec^2) (gravitational parameter).

Instead of solving the OCP (28)-(30), we use the penalty function approach and solve the OCP given below

$$\min_{u(.)} J = \frac{1}{2}(x(T)-x_d)^T K_f (x(T)-x_d) + \frac{1}{2}\int_0^{14000} [u^T(t)u(t) + \nu\Phi(h(u(t)))]dt \tag{31}$$

DOI: 10.1002/oca

subject to

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\\ \dot{x}_4(t)\\ \dot{x}_5(t)\\ \dot{x}_6(t)\end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_1(t)x_4^2(t)\cos^2(x_5(t)) + x_1(t)x_6^2(t) - \dfrac{\mu}{x_1^2(t)} + u_1(t) \\ x_4(t) \\ -\dfrac{2x_2(t)x_4(t)}{x_1(t)} + 2x_4(t)x_6(t)\tan(x_5(t)) + \dfrac{u_2(t)}{x_1(t)\cos(x_5(t))} \\ x_6(t) \\ -\dfrac{2x_2(t)x_6(t)}{x_1(t)} - x_4^2(t)\sin(x_5(t))\cos(x_5(t)) + \dfrac{u_3(t)}{x_1(t)} \end{bmatrix}, \tag{32}$$

where $h(u) = u^T u - 10^{-8}$, $(\Phi \circ h)(.) = \max\{0, h(.)\}^4$ is by choice a differentiable penalty function and $\nu \in \mathbb{R}_+$ is the weighting factor.

We consider a linear system given by $\dot{x} = Ax + Bu + d$, $x(0) = x_0$, which is obtained by the linearization of (29) at a selected steady-state operating point $x_{op} = x_0$ and $u_{op} = [0\ \ 0\ \ 0]^T$. We create a homotopy between the nonlinear system and the linear system and use the indirect single shooting method as a solver for the OCP with the cost functional (31) at each homotopy iteration. The indirect single shooting method converts the OCP into a root finding problem and solves for the initial values of the costate variables.

To demonstrate the advantages of the combined homotopy and NEOC method, two cases are considered. In the first case, we set the initial guess for the initial value of the costate variables for the next iteration to be equal to the optimal value of the costate variables obtained from the previous iteration (in the subsequent figure, we call this case as “without proposed method”). In the second case, we use the combined homotopy and NEOC method discussed in the previous section to set the initial guess for the initial value of the costate variables for the next iteration (in the subsequent figure, we call this case as “with proposed method”). Note that [12] uses (3) to solve OCPs but does not use neighboring extremal updates to predict the change in the initial value of the costate variables. The Matlab function `fsolve.m` has been used to solve the root finding problem, the weighting factor is $\nu = 10^{30}$ and λ has been varied from 0 to 1 in increments of 0.1.

Figures 1(a)-(f) show the trajectory for the states of the nonlinear system, along with trajectories for some values of λ, with $R_i = 600$ (km) and $R_f = 2000$ (km). Figures 1(g)-(i) show the control inputs to the nonlinear system, along with trajectories for some values of λ. Figure 1(j) shows the control input constraint as λ varies from 0 to 1. Figure 1(k) shows the total cost for the nonlinear system as λ varies from 0 to 1. Figure 1(l) shows the spacecraft maneuver from an initial circular orbit of radius $R_i = 600$ (km) to a final circular orbit of radius $R_f = 2000$ (km). Figure 1(m) shows the total number of function evaluations of `fsolve.m` for different spacecraft maneuvers, for the two cases described above. Figure 1(n) shows the total number of iterations of `fsolve.m` for different spacecraft maneuvers, for the two cases described above. From Figures 1(m)-(n), one can see that the second case described above needs fewer function evaluations and iterations of `fsolve.m`.

6. CONCLUSIONS AND FUTURE WORK

The proposed method is based on the approach of combined use of homotopy and NEOC, which to the authors’ knowledge has not been reported in the previous literature. This approach was illustrated using a numerical example, which suggested benefits of the combined application of these techniques in terms of reducing the number of function evaluations and iterations. In the future, we intend to investigate the use of this method for more complicated control input/state constrained OCPs and for other real world applications.

DOI: 10.1002/oca

ACKNOWLEDGMENTS

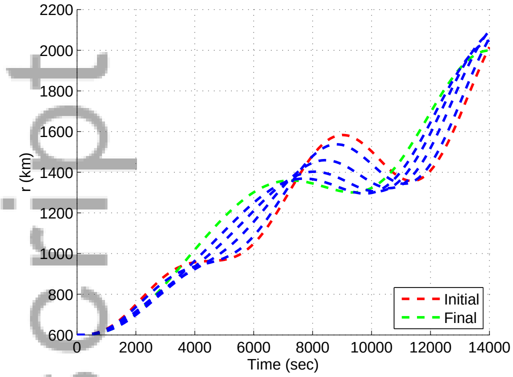
The research of Anthony M. Bloch was supported by NSF grants DMS-0907949, DMS-1207693, INSPIRE-1343720 and the Simons Foundation. The research of Ilya V. Kolmanovsky was supported by NSF grants CMMI-1130160, CNS-1544844 and ECCS-1404814.

REFERENCES

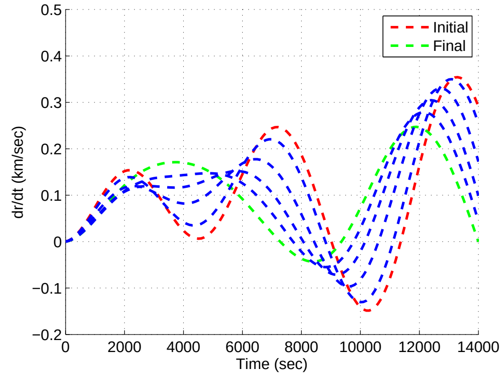
1. Betts JT. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics* 1998; **21**(2):193–207.
2. Rao AV. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences* 2009; **135**(1):497–528.
3. Ben-Asher JZ. *Optimal control theory with aerospace applications*. American Institute of Aeronautics and Astronautics, 2010.
4. Hauser J, Meyer DG. Trajectory morphing for nonlinear systems. *Proceedings of American Control Conference*, 1998; 2065–2070.
5. Graichen K, Petit N. A continuation approach to state and adjoint calculation in optimal control applied to the reentry problem. *Proceedings of IFAC World Congress*, 2008; 14 307–14 312.
6. Olympio JT. A second-order gradient solver using a homotopy method for space trajectory problems. *Proceedings of AIAA/AAS Astrodynamics Specialist Conference*, 2010.
7. Hatcher A. *Algebraic Topology*. Cambridge University Press, 2002.
8. Allgower EL, Georg K. *Numerical continuation methods: an introduction*. Springer Science & Business Media, 1990.
9. Trélat E. Optimal control and applications to aerospace: some results and challenges. *Journal of Optimization Theory and Applications* 2012; **154**(3):713–758.
10. Bonilla J, Diehl M, Logist F, De Moor B, Van Impe JF. A convexity-based homotopy method for nonlinear optimization in model predictive control. *Optimal Control Applications and Methods* 2010; **31**(5):393–414.
11. Caillau JB, Cots O, Gergaud J. Differential continuation for regular optimal control problems. *Optimization Methods and Software* 2012; **27**(2):177–196.
12. Kim M. Continuous low-thrust trajectory optimization: techniques and applications. PhD Thesis, Virginia Polytechnic Institute and State University 2005.
13. Rostalski P, Fotiou IA, Bates DJ, Beccuti AG, Morari M. Numerical algebraic geometry for optimal control applications. *SIAM Journal on Optimization* 2011; **21**(2):417–437.
14. Zhulin SS. Homotopy method for finding extremals in optimal control problems. *Differential Equations* 2007; **43**(11):1495–1504.
15. Bloch AM. *Nonholonomic mechanics and control*. Springer Science & Business Media, 2015.
16. Bryson AE. *Applied optimal control: optimization, estimation and control*. CRC Press, 1975.
17. Dontchev AL, Hager WW. Lipschitzian stability in nonlinear control and optimization. *SIAM Journal on Control and Optimization* 1993; **31**(3):569–603.
18. Dontchev AL, Hager WW, Poore AB, Yang B. Optimality, stability, and convergence in nonlinear control. *Applied Mathematics and Optimization* 1995; **31**(3):297–326.
19. Dontchev AL, Hager WW. Lipschitzian stability for state constrained nonlinear optimal control. *SIAM Journal on Control and Optimization* 1998; **36**(2):698–718.
20. Schättler H, Ledzewicz U. *Geometric optimal control: theory, methods and examples*. Springer Science & Business Media, 2012.
21. Speyer JL, Jacobson DH. *Primer on optimal control theory*. SIAM, 2010.
22. Agrachev AA, Sachkov YL. *Control theory from the geometric viewpoint*. Springer Science & Business Media, 2004.
23. Breakwell JV, Yu-Chi H. On the conjugate point condition for the control problem. *International Journal of Engineering Science* 1965; **2**(6):565–579.
24. Caroff N, Frankowska H. Conjugate points and shocks in nonlinear optimal control. *Transactions of the American Mathematical Society* 1996; **348**(8):3133–3153.
25. Loewen PD, Zheng H. Generalized conjugate points for optimal control problems. *Nonlinear Analysis: Theory, Methods & Applications* 1994; **22**(6):771–791.
26. Mereau PM, Powers WF. Conjugate point properties for linear quadratic problems. *Journal of Mathematical Analysis and Applications* 1976; **55**(2):418–433.
27. Zeidan V, Zezza P. The conjugate point condition for smooth control sets. *Journal of Mathematical Analysis and Applications* 1988; **132**(2):572–589.
28. Zeidan V, Zezza P. Conjugate points and optimal control: counterexamples. *IEEE Transactions on Automatic Control* 1989; **34**(2):254–255.
29. Zeidan V. The riccati equation for optimal control problems with mixed state-control constraints: necessity and sufficiency. *SIAM Journal on Control and Optimization* 1994; **32**(5):1297–1321.
30. Bertrand R, Epenoy R. New smoothing techniques for solving bang-bang optimal control problemsnumerical results and statistical interpretation. *Optimal Control Applications and Methods* 2002; **23**(4):171–197.
31. Silva C, Trélat E. Smooth regularization of bang-bang optimal control problems. *IEEE Transactions on Automatic Control* 2010; **55**(11):2488–2499.
32. Davison EJ, Maki MC. The numerical solution of the matrix riccati differential equation. *IEEE Transactions on Automatic Control* 1973; **18**(1):71–73.

DOI: 10.1002/oca

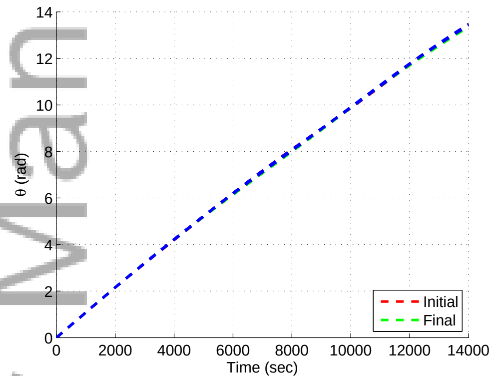
33. Abou-Kandil H, Freiling G, Ionescu V, Jank G. *Matrix Riccati equations in control and systems theory*. Birkhäuser, 2012.
34. Frankowska H. Value function in optimal control 2001.



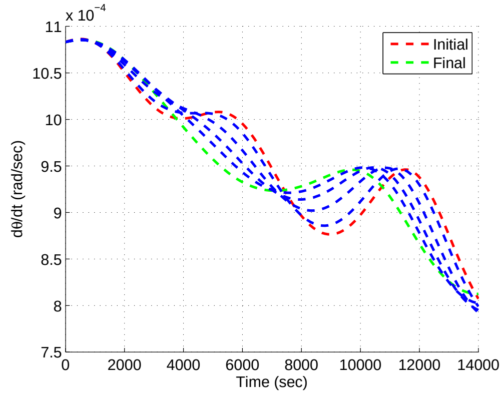
(a) Trajectory of r.



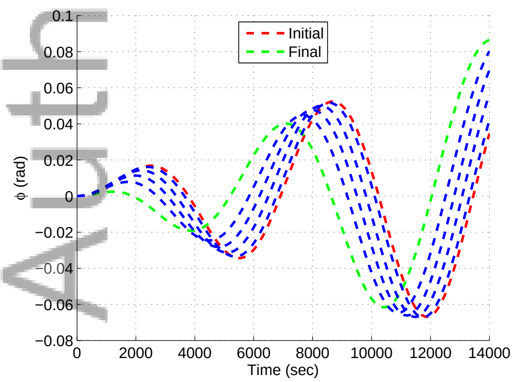
(b) Trajectory of $\dot{r}$.



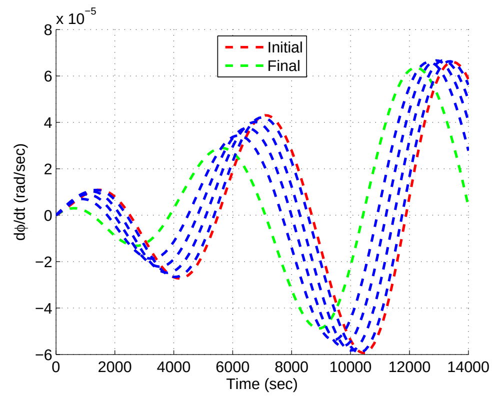
(c) Trajectory of θ.



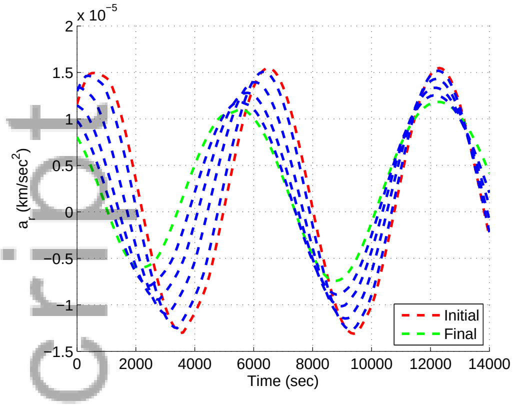
(d) Trajectory of $\dot{\theta}$.



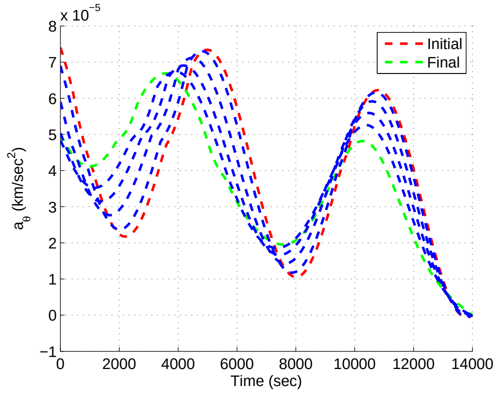
(e) Trajectory of ϕ.



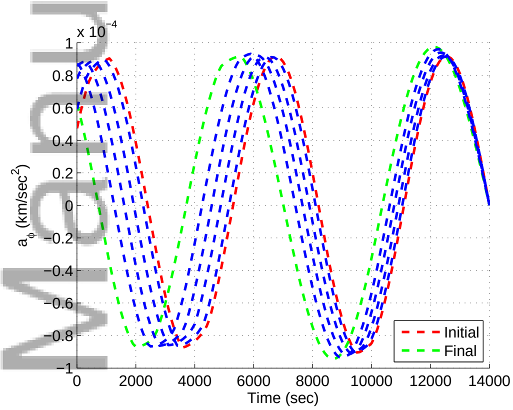
(f) Trajectory of $\dot{\phi}$.



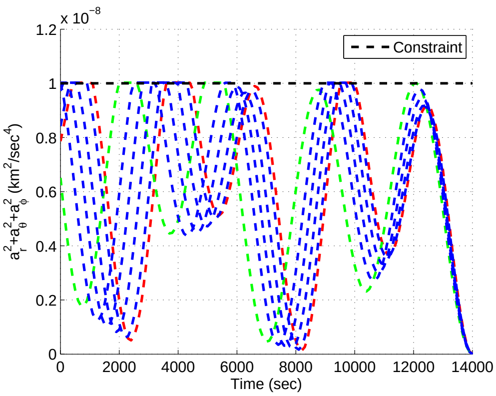
(g) Control Input a_r.



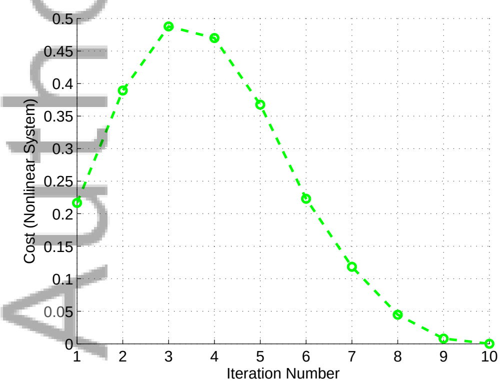
(h) Control Input a_θ.



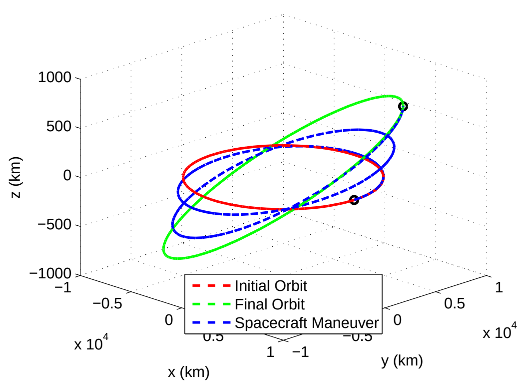
(i) Control Input a_ϕ.



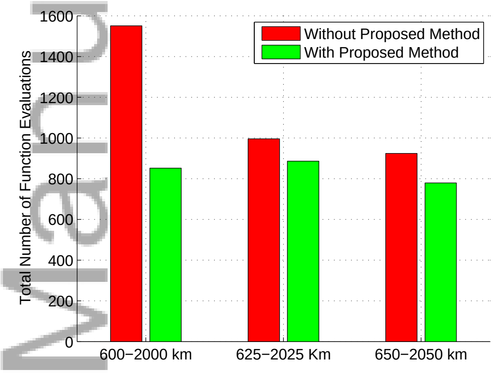
(j) Control Input Constraint.



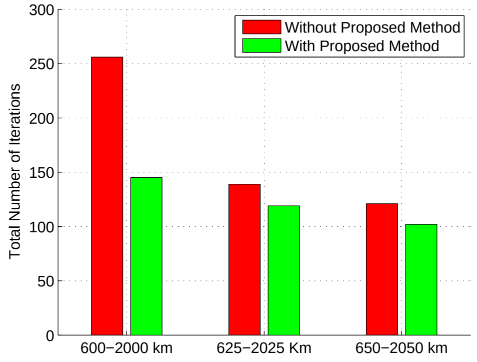
(k) Total Cost for the Nonlinear System.



(l) Spacecraft Maneuver.



(m) Total Number of Function Evaluations.



(n) Total Number of Iterations.

Figure 1: Results.