



Method for solving bang-bang and singular optimal control problems using adaptive Radau collocation

Elisha R. Pager¹ · Anil V. Rao¹

Received: 7 September 2021 / Accepted: 17 January 2022 / Published online: 29 January 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

A method is developed for solving bang-bang and singular optimal control problems using adaptive Legendre–Gauss–Radau collocation. The method is divided into several parts. First, a structure detection method is developed that identifies switch times in the control and analyzes the corresponding switching function for segments where the solution is either bang-bang or singular. Second, after the structure has been detected, the domain is decomposed into multiple domains such that the multiple-domain formulation includes additional decision variables that represent the switch times in the optimal control. In domains classified as bang-bang, the control is set to either its upper or lower limit. In domains identified as singular, the objective function is augmented with a regularization term to avoid the singular arc. An iterative procedure is then developed for singular domains to obtain a control that lies in close proximity to the singular control. The method is demonstrated on four examples, three of which have either a bang-bang and/or singular optimal control while the fourth has a smooth and nonsingular optimal control. The results demonstrate that the method of this paper provides accurate solutions to problems whose solutions are either bang-bang or singular when compared against previously developed mesh refinement methods that are not tailored for solving nonsmooth and/or singular optimal control problems, and produces results that are equivalent to those obtained using previously developed mesh refinement methods for optimal control problems whose solutions are smooth.

Keywords Singular optimal control · Bang-bang optimal control · Regularization · Gaussian quadrature collocation

✉ Anil V. Rao
anilvrao@ufl.edu

Elisha R. Pager
epager@ufl.edu

¹ Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250, USA

1 Introduction

Optimal control problems arise in many engineering applications due to the need to optimize the performance of a controlled dynamical system. In general, optimal control problems do not have analytic solutions and must be solved numerically. A key challenge in solving an optimal control problem numerically arises due to the fact that most optimal control problems are subject to constraints on the system. These constraints often take the form of path constraints where limits are imposed on functions of either the control and/or the state. Constrained optimal control problems often have nonsmooth solutions, where the nonsmoothness arises in the forms of instantaneous switches in the control or switches between activity and inactivity in the state path constraints. Moreover, many constrained optimal control problems have solutions that lie on one or more singular arcs. The existence of a singular arc makes solving constrained optimal control problems even more challenging because Pontryagin's minimum principle (that is, the first and second-order optimality conditions) fail to yield a complete solution along the singular arc. As a result, when applying a computational method to a problem whose solution lies on a singular arc, standard methods produce nonsensical results. This research is motivated by the importance of solving optimal control problems whose solutions are nonsmooth and singular.

Numerical methods for optimal control fall into two broad categories: indirect methods and direct methods. In an indirect method, the first-order variational optimality conditions are derived, and the optimal control problem is converted to a Hamiltonian boundary-value problem (HBVP). The HBVP is then solved numerically using a differential-algebraic equation solver. In a direct method, the state and control are approximated, and the optimal control problem is transcribed into a finite-dimensional nonlinear programming problem (NLP) [1]. The NLP is then solved numerically using well-developed software such as *SNOPT* [2] or *IPOPT* [3].

Over the past two decades, a particular class of direct methods, called direct collocation methods, has been used extensively for solving continuous optimal control problems. A direct collocation method is an implicit simulation method where the state and control are parameterized, and the constraints in the continuous optimal control problem are enforced at a specially chosen set of collocation points. In more recent years, a great deal of research has been carried out in the area of *Gaussian quadrature orthogonal collocation methods* [4–7]. In a Gaussian quadrature collocation method, the state is approximated using a basis of Lagrange polynomials where the support points of the polynomials are chosen to be the points associated with a Gaussian quadrature. The most well developed Gaussian quadrature methods employ either Legendre–Gauss (LG) points [4, 8], Legendre–Gauss–Radau (LGR) points [5–7, 9, 10], or Legendre–Gauss–Lobatto (LGL) points [11]. More recently, a convergence theory has been developed to show that under certain assumptions of smoothness and coercivity, an *hp*-Gaussian quadrature method employing either LG or LGR collocation points will converge to a local minimizer of the optimal control problem at an exponential rate

[12–15]. For these reasons, direct methods should be explored in the effort to develop novel computational methods for singular optimal control.

Computational issues arise when a solution to an optimal control problem is either nonsmooth or singular. The difficulty with such optimal control problems is twofold. First, the precise locations of any discontinuities and the structure of the control must be identified. Static mesh refinement methods that employ Gaussian quadrature have been developed recently as an initial attempt to locate (approximately) discontinuities in the solution [16–18]. Even more recently, the idea of using variable mesh refinement methods and structure detection methods has been developed [19–23]. Unlike static mesh refinement methods such as those found in Refs. [10, 24, 25], variable mesh refinement methods work by including parameters in the optimization that define the location of the discontinuities. The methods in Refs. [19–21] use the Lagrange multipliers to detect the switch point locations in the control structure and then place variable mesh points to represent the switch times in the NLP, while Refs. [22, 23] use the switching function and a sensitivity analysis to place moving finite elements at the switch point locations. Furthermore, Ref. [26] describes a mesh refinement method for solving bang-bang optimal control problems based on the switching function associated with the Hamiltonian. More recently, in Ref. [27] a switch point algorithm was developed for optimizing over the locations of switch points in a nonsmooth control solution, but a priori knowledge of the switch points existence is required. Finally, methods that utilize structure detection on a static mesh are described in Refs. [28, 29].

The second difficulty in solving optimal control problems with nonsmooth or singular solutions arises when the optimal control is singular. Several approaches have been developed for solving singular optimal control problems using both indirect and direct methods. A majority of these methods typically employ either a regularization approach or use of the optimality conditions with an indirect method to solve for the singular control (see Refs. [30, 31]). A regularization method transforms the singular control problem into a series of nonsingular problems by minimizing the sum of the original objective and a regularization term, where the regularization term is a quadratic function of the control. Regularization approaches have been implemented using dynamic programming, indirect methods, direct methods, and nested indirect/direct approaches as described in Refs. [32–35]. More recent implementations of regularization based techniques include the uniform trigonometrization method (UTM) developed in [36], the use of a continuation method of a regularized term in [37], and the total variation based regularization approach in [27]. Aside from regularization based approaches, research has also been conducted on the use of low-order representations of the control including straight line, monotonic, and nonmonotonic function approximations to reduce the oscillations and numerical challenges observed with singular arcs [38, 39].

Motivated by the prevalence of bang-bang and singular arcs in optimal control solutions and the need for a general method that can handle such problems, this paper describes a new method for detecting and solving optimal control problems whose solutions are bang-bang and singular. The method described in this paper consists of the following parts. First, a multiple-domain reformulation of the LGR collocation method is developed that enables partitioning the problem into segments

that are categorized as either regular, bang-bang, or singular. This multiple-domain LGR collocation reformulation partitions the time horizon into domains such that additional decision variables are introduced that correspond to the endpoints of each newly created domain. Second, a method is developed that detects the structure of the optimal control (that is, the method detects segments where the solution is either regular, bang-bang, or singular). Third, a regularization method (inspired by the approach used in Ref. [32]) is employed to solve for the control in any domain where the control is categorized as singular. The method presented in [32] implements a regularization procedure over the entire time domain using differential dynamic programming, whereas in this current work a similar regularization procedure is implemented only in the intervals denoted as singular. This difference in implementation results in a more accurate solution of the control. It is noted that, because bang-bang and singular optimal control problems frequently have Hamiltonians that are affine in the control, the structure detection method developed in this paper is designed to work on those components of the control that appear linearly in the Hamiltonian. Moreover, switch times in the control (which then lead to the partitioning into multiple domains) are identified using the jump function mesh refinement method described in Ref. [18]. Using the aforementioned structure detection method, the control in the newly created domains is then either left free (in the case of a regular domain), set equal to one of its limits (in the case of a bang-bang domain), or is determined via the aforementioned regularization method (in the case of a singular domain).

The contributions of this work are as follows. First, using jump function approximations provide an accurate way to determine the number of discontinuities along with accurate estimates of the locations of these discontinuities. Second, using the switching function and the Hamiltonian enables determining those intervals where the control is either bang-bang or singular. Third, the method automatically partitions the solution into domains based on the results of the structure detection method. Fourth, the method does not require any a priori knowledge of the structure in the optimal control or whether the optimal control is bang-bang or singular. Fifth, in this paper the use of regularization methods is extended to direct collocation methods. In particular, the multiple-domain partition of the solution obtained from the structure detection method enables regularizing only over those domains where the control is singular. Consequently, within a singular domain the regularization leads to a control that lies in close proximity to the singular control while simultaneously eliminating the need to derive the singular control conditions (where deriving such conditions may prove to be intractable depending upon the problem). The performance of the method developed in this paper is demonstrated on four examples. The optimal control for each of the first three of these examples is either bang-bang and/or singular, while the optimal control for the fourth example is smooth. The numerical results obtained of the first three examples demonstrate that the method of this paper produces significantly more accurate results when compared against mesh refinement methods that are not developed for solving optimal control problems whose solutions are nonsmooth or singular. Finally, the numerical results of the fourth example demonstrate the method of this paper correctly identifies when a solution is smooth and applies only static mesh refinement in order

to obtain a solution. As a result, results comparable to those obtained using previously developed mesh refinement methods are obtained.

The remainder of the paper is organized as follows. Section 2 introduces the Bolza optimal control problem and the necessary conditions for optimality. Section 3 describes the multiple-domain Legendre–Gauss–Radau collocation used to transcribe the multiple-domain Bolza optimal control problem. Section 4 provides a brief overview of optimal control problems whose solutions are nonsmooth. Section 5 details the method for solving bang-bang and singular optimal control problems. Section 6 provides numerical solutions obtained by demonstrating the method on four examples. Section 7 describes limitations of the method. Finally, Sect. 8 provides conclusions on this research.

2 Bolza optimal control problem

Without loss of generality, consider the following single-phase optimal control problem in Bolza form defined on the time horizon $t \in [t_0, t_f]$. Determine the state $\mathbf{x}(t) \in \mathbb{R}^{n_x}$, the control $\mathbf{u}(t) \in \mathbb{R}^{n_u}$, and the terminal time $t_f \in \mathbb{R}$ that minimize the objective functional

$$\mathcal{J} = \mathcal{M}(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), t) dt, \quad (1)$$

subject to the dynamic constraints

$$\frac{d\mathbf{x}(t)}{dt} \equiv \dot{\mathbf{x}}(t) = \mathbf{a}(\mathbf{x}(t), \mathbf{u}(t), t), \quad (2)$$

the control constraints

$$\mathbf{u}_{\min} \leq \mathbf{u}(t) \leq \mathbf{u}_{\max}, \quad (3)$$

and the boundary conditions

$$\mathbf{b}(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = \mathbf{0}, \quad (4)$$

where the functions $\mathcal{M}$, $\mathcal{L}$, $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ are defined by the mappings

$$\begin{aligned} \mathcal{M} &: \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R} \rightarrow \mathbb{R}, \\ \mathcal{L} &: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R} \rightarrow \mathbb{R}, \\ \mathbf{a} &: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x}, \\ \mathbf{b} &: \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R} \rightarrow \mathbb{R}^{n_b}. \end{aligned}$$

The Bolza optimal control problem given in Eqs. (1)–(4) gives rise to the following first-order calculus of variations [40–42] conditions:

$$\dot{\mathbf{x}}(t) = \left[\frac{\partial \mathcal{H}}{\partial \lambda} \right]^\top = \mathcal{H}_\lambda^\top, \quad (5)$$

$$\dot{\lambda}(t) = - \left[\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \right]^\top = -\mathcal{H}_{\mathbf{x}}^\top, \quad (6)$$

$$\mathbf{0} = \frac{\partial \mathcal{H}}{\partial \mathbf{u}} = \mathcal{H}_{\mathbf{u}}, \quad (7)$$

where $\lambda(t) \in \mathbb{R}^{n_x}$ is the costate and

$$\mathcal{H}(\mathbf{x}(t), \mathbf{u}(t), \lambda(t), t) = \mathcal{L}(\mathbf{x}(t), \mathbf{u}(t), t) + \lambda^\top(t) \mathbf{a}(\mathbf{x}(t), \mathbf{u}(t), t), \quad (8)$$

is the augmented Hamiltonian. Finally, the transversality conditions are given by

$$\lambda(t_0) = - \frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_0)} + \mathbf{v}^\top \frac{\partial \mathbf{b}}{\partial \mathbf{x}(t_0)}, \quad \lambda(t_f) = \frac{\partial \mathcal{M}}{\partial \mathbf{x}(t_f)} - \mathbf{v}^\top \frac{\partial \mathbf{b}}{\partial \mathbf{x}(t_f)} \quad (9)$$

$$\mathcal{H}(t_0) = \frac{\partial \mathcal{M}}{\partial t_0} - \mathbf{v}^\top \frac{\partial \mathbf{b}}{\partial t_0}, \quad \mathcal{H}(t_f) = - \frac{\partial \mathcal{M}}{\partial t_f} + \mathbf{v}^\top \frac{\partial \mathbf{b}}{\partial t_f} \quad (10)$$

where $\mathbf{v}$ is the Lagrange multiplier associated with the boundary conditions. Equations (5) and (6) form what is classically known as a *Hamiltonian system* [40, 41]. The conditions in Eqs. (9) and (10) are called *transversality conditions* [40–42] on the boundary values of the costate. For some problems, the control cannot be uniquely determined, either implicitly or explicitly, from the optimality conditions given in Eqs. (5)–(10). In such cases, the weak form of Pontryagin's minimum principle can be used which solves for the permissible control that minimizes the Hamiltonian in Eq. (8). If $\mathbf{u} \in [u_{\min}, u_{\max}]$ is the set of permissible controls, then Pontryagin's minimum principle states that the optimal control, $\mathbf{u}$, satisfies the condition

$$\mathcal{H}(\mathbf{x}^*(t), \mathbf{u}^*, \lambda^*(t), t) \leq \mathcal{H}(\mathbf{x}^*(t), \mathbf{u}, \lambda^*(t), t), \quad \mathbf{u} \in [u_{\min}, u_{\max}]. \quad (11)$$

The Hamiltonian system, together with the original boundary conditions and the costate transversality conditions, forms a *Hamiltonian boundary-value problem* (HBVP) [40–42]. Any solution $(\mathbf{x}^*(t), \mathbf{u}^*, \lambda^*(t), \mathbf{v}^*)$ to the HBVP is called an *extremal* solution.

3 Multiple-domain Legendre–Gauss–Radau collocation

In this paper, the previously developed *hp*-adaptive Legendre–Gauss–Radau (LGR) collocation method [5–7, 9, 10] is used to approximate the optimal control problem (where the term LGR collocation will be used from this point onwards to mean *hp*-adaptive Legendre–Gauss–Radau collocation). LGR collocation is used because it has been shown to converge at an exponential rate to a local solution of the optimal control problem for problems where the solution is smooth [12–15]. The focus of this paper, however, is on solving optimal control problems whose solutions are nonsmooth and/or singular. As a result, modifications to the standard LGR formulation are made.

Specifically, a multiple-domain reformulation of LGR collocation is developed as described in the remainder of this section.

The multiple-domain formulation of LGR collocation divides the domain $t \in [t_0, t_f]$ into distinct partitions such that the endpoints of each partition are decision variables. The division into domains is obtained using a structure decomposition method as described in Sect. 5.1. The continuous-time Bolza optimal control problem described in Eqs. (1)–(4) is discretized using collocation at the Legendre–Gauss–Radau (LGR) points [5–7, 9]. The time horizon $t \in [t_0, t_f]$ may be divided into D time domains, $\mathcal{P}_d = [t_s^{[d-1]}, t_s^{[d]}] \subseteq [t_0, t_f]$, $d \in \{1, \dots, D\}$, such that

$$\bigcup_{d=1}^D \mathcal{P}_d = [t_0, t_f], \quad \bigcap_{d=1}^D \mathcal{P}_d = \{t_s^{[1]}, \dots, t_s^{[D-1]}\}, \quad (12)$$

where $t_s^{[d]}$, $d \in \{1, \dots, D-1\}$ are the domain interface variables of the problem, $t_s^{[0]} = t_0$, and $t_s^{[D]} = t_f$. Thus, in the case where $D = 1$ the phase consists of only a single domain $\mathcal{P}_1 = [t_0, t_f]$ and $\{t_s^{[1]}, \dots, t_s^{[D-1]}\} = \emptyset$.

$$\begin{aligned} t &= \frac{t_s^{[d]} - t_s^{[d-1]}}{2} \tau + \frac{t_s^{[d]} + t_s^{[d-1]}}{2}, \\ \tau &= 2 \frac{t - t_s^{[d-1]}}{t_s^{[d]} - t_s^{[d-1]}} - 1. \end{aligned} \quad (13)$$

The interval $\tau \in [-1, +1]$ for each domain $\mathcal{P}_d$ is then divided into K mesh intervals, $\mathcal{I}_k = [T_{k-1}, T_k] \subseteq [-1, +1]$, $k \in \{1, \dots, K\}$ such that

$$\bigcup_{k=1}^K \mathcal{I}_k = [-1, +1], \quad \bigcap_{k=1}^K \mathcal{I}_k = \{T_1, \dots, T_{K-1}\}, \quad (14)$$

and $-1 = T_0 < T_1 < \dots < T_{K-1} < T_K = +1$. For each mesh interval, the LGR points used for collocation are defined in the domain of $[T_{k-1}, T_k]$ for $k \in \{1, \dots, K\}$. The state of the continuous optimal control problem is then approximated in mesh interval $\mathcal{I}_k$, $k \in \{1, \dots, K\}$, as

$$\mathbf{x}^{(k)}(\tau) \approx \mathbf{X}^{(k)}(\tau) = \sum_{j=1}^{N_k+1} \mathbf{X}_j^{(k)} \ell_j^{(k)}(\tau), \quad \ell_j^{(k)}(\tau) = \prod_{\substack{l=1 \\ l \neq j}}^{N_k+1} \frac{\tau - \tau_l^{(k)}}{\tau_j^{(k)} - \tau_l^{(k)}}, \quad (15)$$

where $\ell_j^{(k)}(\tau)$ for $j \in \{1, \dots, N_k+1\}$ is a basis of Lagrange polynomials on $\mathcal{I}_k$, $(\tau_1^{(k)}, \dots, \tau_{N_k}^{(k)})$ are the set of N_k Legendre–Gauss–Radau (LGR) collocation points in the interval $[T_{k-1}, T_k]$, $\tau_{N_k+1}^{(k)} = T_k$ is a non-collocated support point, and $\mathbf{X}_j^{(k)} \equiv \mathbf{X}^{(k)}(\tau_j^{(k)})$. Differentiating $\mathbf{X}^{(k)}(\tau)$ in Eq. (15) with respect to τ gives

$$\frac{d\mathbf{X}^{(k)}(\tau)}{d\tau} = \sum_{j=1}^{N_k+1} \mathbf{X}_j^{(k)} \frac{d\ell_j^{(k)}(\tau)}{d\tau}. \quad (16)$$

The dynamics are then approximated at the N_k LGR points in mesh interval $k \in \{1, \dots, K\}$ as

$$\sum_{j=1}^{N_k+1} D_{lj}^{(k)} \mathbf{X}_j^{(k)} - \frac{t_f - t_0}{2} \mathbf{a} \left(\mathbf{X}_l^{(k)}, \mathbf{U}_l^{(k)}, t(\tau_l^{(k)}, t_0, t_f) \right) = \mathbf{0}, \quad l \in \{1, \dots, N_k\}, \quad (17)$$

where

$$D_{lj}^{(k)} = \frac{d\ell_j^{(k)}(\tau_l^{(k)})}{d\tau}, \quad l \in \{1, \dots, N_k\}, j \in \{1, \dots, N_k + 1\},$$

are the elements of the $N_k \times (N_k + 1)$ Legendre–Gauss–Radau differentiation matrix in mesh interval $\mathcal{I}_k$, $k \in \{1, \dots, K\}$, and $\mathbf{U}_l^{(k)}$ is the approximation of the control at the l^{th} collocation point in mesh interval $\mathcal{I}_k$. The time variables t_0 and t_f in Eq. (17) represent the initial and final domain interface variables, $t_s^{[d-1]}$ and $t_s^{[d]}$, on the domain $\mathcal{P}_d$. It is noted that continuity in the state and time between mesh intervals $\mathcal{I}_{k-1}$ and $\mathcal{I}_k$, $k \in \{1, \dots, K\}$, is enforced by using the same variables to represent $\mathbf{X}_{N_{k-1}+1}^{(k-1)}$ and $\mathbf{X}_1^{(k)}$, while continuity in the state between the domains $\mathcal{P}_{d-1}$ and $\mathcal{P}_d$, $d \in \{2, \dots, D\}$, is achieved by using the same variables to represent $\mathbf{X}_{N_{d-1}+1}^{[d-1]}$ and $\mathbf{X}_1^{[d]}$ where the superscript $[d]$ is used to denote the d^{th} time domain, $\mathbf{X}_j^{[d]}$ denotes the value of the state approximation at the j^{th} discretization point in the time domain $\mathcal{P}_d$, and $N^{[d]}$ is the total number of collocation points used in time domain $\mathcal{P}_d$ computed by

$$N^{[d]} = \sum_{k=1}^{K^{[d]}} N_k^{[d]}. \quad (18)$$

The Legendre–Gauss–Radau approximation of the multiple-domain optimal control problem results in the following nonlinear programming problem (NLP). Minimize the objective function

$$\mathcal{J} = \mathcal{M}(\mathbf{X}_1^{[1]}, t_0, \mathbf{X}_{N^{[D]}+1}^{[D]}, t_f) + \sum_{d=1}^D \frac{t_s^{[d]} - t_s^{[d-1]}}{2} [\mathbf{w}^{[d]}]^\top \mathbf{L}^{[d]}, \quad (19)$$

subject to the collocated dynamic constraints

$$\Delta^{[d]} = \mathbf{D}^{[d]} \mathbf{X}^{[d]} - \frac{t_s^{[d]} - t_s^{[d-1]}}{2} \mathbf{A}^{[d]} = \mathbf{0}, \quad d \in \{1, \dots, D\}, \quad (20)$$

the control constraints

$$\mathbf{u}_{\min} \leq \mathbf{U}_j^{[d]} \leq \mathbf{u}_{\max}, \quad j \in \{1, \dots, N^{[d]}\}, d \in \{1, \dots, D\}, \quad (21)$$

the boundary conditions

$$\mathbf{b}(\mathbf{X}_1^{[1]}, t_0, \mathbf{X}_{N^{[D]}+1}^{[D]}, t_f) = \mathbf{0}, \quad (22)$$

and the continuity constraints

$$\mathbf{X}_{N^{[d-1]}+1}^{[d-1]} = \mathbf{X}_1^{[d]}, \quad d \in \{2, \dots, D\}, \quad (23)$$

noting that Eq. (23) is implicitly satisfied by employing the same variable in the NLP for $\mathbf{X}_{N^{[d-1]}+1}^{[d-1]}$ and $\mathbf{X}_1^{[d]}$. The matrices in Eqs. (19)–(20) are defined as follows

$$\mathbf{A}^{[d]} = \begin{bmatrix} \mathbf{a}(\mathbf{X}_1^{[d]}, \mathbf{U}_1^{[d]}, t_1^{[d]}) \\ \vdots \\ \mathbf{a}(\mathbf{X}_{N^{[d]}}^{[d]}, \mathbf{U}_{N^{[d]}}^{[d]}, t_{N^{[d]}}^{[d]}) \end{bmatrix} \in \mathbb{R}^{N^{[d]} \times n_x}, \quad (24)$$

$$\mathbf{L}^{[d]} = \begin{bmatrix} \mathcal{L}(\mathbf{X}_1^{[d]}, \mathbf{U}_1^{[d]}, t_1^{[d]}) \\ \vdots \\ \mathcal{L}(\mathbf{X}_{N^{[d]}}^{[d]}, \mathbf{U}_{N^{[d]}}^{[d]}, t_{N^{[d]}}^{[d]}) \end{bmatrix} \in \mathbb{R}^{N^{[d]} \times 1}, \quad (25)$$

$\mathbf{D}^{[d]} \in \mathbb{R}^{N^{[d]} \times [N^{[d]}+1]}$ is the LGR differentiation matrix in time domain $\mathcal{P}_d$, $d \in \{1, \dots, D\}$, and $\mathbf{w}^{[d]} \in \mathbb{R}^{N^{[d]} \times 1}$ are the LGR weights at each node in time domain $\mathcal{P}_d$, $d \in \{1, \dots, D\}$. It is noted that $\mathbf{a} \in \mathbb{R}^{1 \times n_x}$ and $\mathcal{L} \in \mathbb{R}^{1 \times 1}$ correspond, respectively, to the vector fields that define the right-hand side of the dynamics and the integrand of the optimal control problem. Additionally, the state matrix, $\mathbf{X}^{[d]} \in \mathbb{R}^{[N^{[d]}+1] \times n_x}$, and the control matrix, $\mathbf{U}^{[d]} \in \mathbb{R}^{N^{[d]} \times n_u}$, in time domain $\mathcal{P}_d$, $d \in \{1, \dots, D\}$, are formed as

$$\mathbf{X}^{[d]} = \begin{bmatrix} \mathbf{X}_1^{[d]} \\ \vdots \\ \mathbf{X}_{N^{[d]}+1}^{[d]} \end{bmatrix} \text{ and } \mathbf{U}^{[d]} = \begin{bmatrix} \mathbf{U}_1^{[d]} \\ \vdots \\ \mathbf{U}_{N^{[d]}}^{[d]} \end{bmatrix}, \quad (26)$$

respectively, where n_u is the number of control components and n_x is the number of state components in the problem.

3.1 Costate estimation

Estimates of the costate may be obtained at each of the discretization points in the time domain $\mathcal{P}_d$, $d \in \{1, \dots, D\}$ using the transformation [5–7],

$$\begin{aligned} \lambda^{[d]} &= (\mathbf{W}^{[d]})^{-1} \mathbf{\Lambda}^{[d]}, \\ \lambda_{N^{[d]}+1}^{[d]} &= (\mathbf{D}_{N^{[d]}+1}^{[d]})^T \mathbf{\Lambda}^{[d]}, \end{aligned} \quad (27)$$

where $\lambda^{[d]} \in \mathbb{R}^{N^{[d]} \times n_x}$ is a matrix of the costate estimates at the collocation points in time domain $\mathcal{P}_d$, $\mathbf{W}^{[d]} = \text{diag}(\mathbf{w}^{[d]})$ is a diagonal matrix of the LGR weights at the collocation points in time domain $\mathcal{P}_d$, $\mathbf{\Lambda}^{[d]} \in \mathbb{R}^{N^{[d]} \times n_x}$ is a matrix of the NLP multipliers obtained from the NLP solver corresponding to the defect constraints at the collocation points in time domain $\mathcal{P}_d$, $\lambda_{N^{[d]}+1}^{[d]} \in \mathbb{R}^{1 \times n_x}$ is a row vector of the costate estimates at the non-collocated end point in time domain $\mathcal{P}_d$, and $\mathbf{D}_{N^{[d]}+1}^{[d]} \in \mathbb{R}^{N^{[d]} \times 1}$ is the last column of the LGR differentiation matrix in time domain $\mathcal{P}_d$.

The aforementioned multiple-domain LGR formulation is summarized as follows. First, a single *phase* problem on $t \in [t_0, t_f]$ is divided into D domains, $\mathcal{P}_d = [t_s^{[d-1]}, t_s^{[d]}]$, $d \in \{1, \dots, D\}$. Each of the D domains are then mapped to the interval $\tau^{[d]} \in [-1, +1]$, $d \in \{1, \dots, D\}$. The interval $\tau^{[d]} \in [-1, +1]$, $d \in \{1, \dots, D\}$ for each domain is then divided into K *mesh intervals*, $\mathcal{I}_k = [T_{k-1}, T_k] \subseteq [-1, +1]$, $k \in \{1, \dots, K\}$. Finally, the intersection of each domain is determined by the *domain interface variables*, $t_s^{[d]}$, $d \in \{1, \dots, D-1\}$.

4 Nonsmooth and singular optimal control

The term nonsmooth is used to denote the optimal control as displaying both non-smooth and singular behavior. By definition a singular arc occurs when Eq. (11) fails to uniquely describe an optimal control; for example, the set of minimizers in Eq. (11) form an interval which contains the optimal control [42]. In this case, knowledge of the interval does not uniquely describe the optimal control itself. This phenomena can occur in many situations but is most common when the dynamics are linear in the control and the control is bounded, or the Hamiltonian is not an explicit function of time. It should be noted that singular arcs can also occur in other situations, but in order to provide structure to the method developed in this paper, only problems that fall into the aforementioned categories will be considered.

For simplicity, in the discussion that follows it is assumed that the control is a scalar (that is, $u(t) \in \mathbb{R}$). Note, however, that without loss of generality the discussion below can be extended to multiple control components, and the use of the method developed in this paper on problems with multiple control components is demonstrated in the examples provided in Sect. 6. Suppose the optimal control problem described in Eqs. (1)–(4) is nonsmooth and singular as defined by the assumptions mentioned previously. The dynamics can now be rewritten in the affine form as

$$\dot{\mathbf{x}}(t) = \mathbf{a}(\mathbf{x}(t), u(t)) = \mathbf{g}(\mathbf{x}(t)) + \mathbf{h}(\mathbf{x}(t))u(t), \quad (28)$$

where $\mathbf{g}(\mathbf{x}(t))$ and $\mathbf{h}(\mathbf{x}(t))$ are not functions of the control. The Hamiltonian from Eq. (8) is redefined as

$$\mathcal{H}(\mathbf{x}(t), \lambda(t), u(t), t) = \mathbf{f}(\mathbf{x}(t), \lambda(t)) + \phi^\top(\mathbf{x}(t), \lambda(t))u(t), \quad (29)$$

where $\mathbf{f}(\mathbf{x}(t), \lambda(t))$ and $\phi(\mathbf{x}(t), \lambda(t))$ are the components of the Hamiltonian that are not a function of the control, and mixed state and control path constraints are not considered. If the following holds along an optimal control

$$\frac{\partial \mathcal{H}}{\partial u} = \mathcal{H}_u = \phi(\mathbf{x}(t), \lambda(t)) = 0, \quad (30)$$

then the minimizing control in Eq. (11) is the interval $[u_{\min}, u_{\max}]$; thus Eq. (11) only implies that an optimal control is feasible. Note that the control does not appear in Eq. (30) because the Hamiltonian is linear in the control. A singular arc is characterized as $\mathcal{H}_u = 0$ and $\mathcal{H}_{uu}$ is singular everywhere on the arc. When this occurs, the reduced Hessian matrix associated with the corresponding NLP that arises from the

direct transcription method of Sect. 3 is ill-conditioned such that the projected Hessian matrix is not positive definite. This leads to poor conditioning in the control profiles which often presents itself in the form of oscillations, or chattering behavior, in the control solution.

The sign and value of $\phi(\mathbf{x}(t), \lambda(t))$ (where ϕ is called the switching function) determines if the control is called a *bang-bang* control or a *singular* control. The weak form of PMP [42] is used in the case of nonsmooth control and the minimization of the Hamiltonian leads to the following piecewise-continuous control, u , that is dependent on the switching function as follows

$$u^* = \arg \min_{u \in [u_{\min}, u_{\max}]} \mathcal{H} = \begin{cases} u_{\min}, & \phi(\mathbf{x}(t), \lambda(t)) > 0, \\ u_s, & \phi(\mathbf{x}(t), \lambda(t)) = 0, \\ u_{\max}, & \phi(\mathbf{x}(t), \lambda(t)) < 0, \end{cases} \quad (31)$$

where the sign of the switching function, $\phi(\mathbf{x}(t), \lambda(t))$, is determined by the state and the costate and u_s lies in the closed interval $[u_{\min}, u_{\max}]$. As the switching function $\phi(\mathbf{x}(t), \lambda(t))$ changes sign, the control coincides with the sign changes by switching between its maximum and minimum values. Any time interval over which $\phi(\mathbf{x}(t), \lambda(t))$ is zero is referred to as a singular arc and any control in the admissible control set will minimize the Hamiltonian. Furthermore, switching between nonsingular and singular arcs give rise to discontinuities on the state and control profiles, and the location of these transition points are referred to as *switch times*. These discontinuities defined by the switch times create numerical issues, whereas, the singular control suffers from non-uniqueness issues that occur when the control is free to lie between its upper and lower bounds and is not defined by the optimality conditions.

The singular control is obtained implicitly from the switching function. Specifically, ϕ is differentiated repeatedly until the control u explicitly appears [43]. Therefore, u^* can be solved for by

$$\frac{d^{(2r)}}{dt^{(2r)}} \phi = 0, \quad (r = 0, 1, 2, \dots), \quad (32)$$

where $2r$ is the minimum number of differentiations of ϕ required to obtain the corresponding control u_s . For u to be optimal over a singular arc, the number of differentiations $2r$ must be even [42, 43]. Furthermore, the generalized Legendre–Clebsch condition [42, 44, 45]

$$(-1)^r \frac{\partial}{\partial u} \left[\frac{d^{2r}}{dt^{(2r)}} \phi \right] \geq 0, \quad (r = 0, 1, 2, \dots), \quad (33)$$

must hold over the duration of a singular arc. While in some problems of interest it is possible to use Eq. (32) to determine a condition for the singular control, in many cases it is unable to produce the singular control (for example, if the order of the singular arc is infinite). Even in cases where the singular control could be determined from Eq. (32), taking derivatives higher than second-order is not easy to implement

numerically or analytically. As mentioned previously, the singular control might be a function of both the state and the costate. If this is the case then a direct collocation method could not be utilized to determine the optimal trajectory. In this paper a method is developed that can be fully automated for detecting and accurately approximating the solution of bang-bang and singular optimal control problems.

5 Method for bang-bang & singular optimal control problems

In this section the method for solving bang-bang and singular optimal control (BBSOC) problems is developed. The method consists of two stages. The first stage of the method described in Sect. 5.1 details the detection of the control structure and the decomposition of the optimal control problem into a multiple-domain optimal control problem dictated by the discontinuities identified that are represented as domain interface variables. These domain interface variables are then treated as additional decision variables in the nonlinear programming problem (NLP). The first stage is only implemented on the first mesh iteration. The second stage described in Sect. 5.2 describes the new constraints that are added to the NLP depending on the structure detection's classification of a domain as being bang-bang or singular in order to constrain the modified optimal control problem correctly. The constraints and methods applied in each type of domain are provided in Sect. 5.2. The second stage initiates the iterative procedures in the proposed method described in Sect. 5.3.

5.1 Structure detection and decomposition

Assume now that the optimal control problem formulated in Sect. 2 under the assumptions of Sect. 4 has been transcribed into a NLP using multiple-domain LGR collocation developed in Sect. 3 with $D = 1$ (that is, a single domain is used). The solution obtained from the NLP then leads to estimates of the state, the control, and the costate as given in Eqs. (26) and (27), respectively. Assume further that the mesh refinement accuracy tolerance is not satisfied. As a result, mesh refinement is required which simultaneously enables the decomposition of the problem into domains that are either bang-bang, singular, or regular. This decomposition is obtained using structure detection as described now.

Structure detection locates discontinuities identified on the initial mesh and then uses the locations of the discontinuities to determine the classification of the interval formed by two adjacent discontinuities. In this work, only control discontinuities are considered because only problems where the Hamiltonian is linear in control are analyzed. Their locations are estimated using jump function approximations [18] and then the intervals formed by each discontinuity are analyzed using the switching function. Structure detection begins by applying the method of Sect. 5.1.1 to identify and estimate the locations of any control discontinuities. After discontinuity locations have been estimated, the method of Sect. 5.1.2 takes the estimated discontinuity locations and determines the classification of the domain as bang-bang or singular. The structure

detection and decomposition process only occurs once on the initial solution. Detailed descriptions of the structure detection process are described next.

5.1.1 Identification of control switch times

Discontinuities in each component of the control are identified using jump function approximations of the control solution as shown in Ref. [18]. In particular, the method given in Ref. [18] is employed here because it is effective for estimating locations of nonsmoothness in the optimal control. A brief overview of the process given in Ref. [18] is provided here for completeness. For further details related to jump function approximations for detecting nonsmoothness in an optimal control, see Ref. [18].

First, a jump function is defined as follows. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be an arbitrary function defined on the interval $t \in [t_0, t_f]$. The jump function of $f(t)$, denoted $[f](t)$, is defined as $[f](t) = f(t^+) - f(t^-)$ where $f(t^+)$ and $f(t^-)$ are the right-hand and left-hand limits of $f(t)$, $t \in [t_0, t_f]$. The jump function is zero across intervals where $f(t)$ is continuous and takes on the value of the jump in $f(t)$ at those locations where $f(t)$ is discontinuous. According to Refs. [18, 46] the jump function of a function $f(t)$ is approximated by

$$L_m f(t) = \frac{1}{q_m(t)} \sum_{t_j \in \mathcal{S}_t} c_j(t) f(t_j) \approx [f](t), \quad (34)$$

where $q_m(t)$ is defined by

$$q_m(t) = \sum_{t_j \in \mathcal{S}_t^+} c_j(t), \quad (35)$$

$c_j(t)$ is defined by

$$c_j(t) = \frac{m!}{\prod_{\substack{i=1 \\ i \neq j}}^{m+1} (t_j - t_i)}, \quad (36)$$

and m specifies the order of the approximation. Higher order approximations converge to the jump function faster outside the neighborhood of discontinuities but have oscillatory behavior in the vicinity of discontinuities. The oscillations are reduced via the **minmod** function, defined here as

$$MM(L_{\mathcal{M}} f(t)) = \begin{cases} \min_{m \in \mathcal{M}} L_m f(t), & L_m f(t) > 0 \quad \forall m \in \mathcal{M}, \\ \max_{m \in \mathcal{M}} L_m f(t), & L_m f(t) < 0 \quad \forall m \in \mathcal{M}, \\ 0 & \text{otherwise,} \end{cases} \quad (37)$$

where $\mathcal{M} \subset \mathbb{N}^+$ is a finite set of choices of the approximation order m .

Suppose an initial control solution is obtained. The control solution, $U(\tau_j^{(k)}), j = \{1, \dots, N_k\}; k = \{1, \dots, K\}$, is normalized to the interval $[0, 1)$ by the transformation

$$u(\tau_j^{(k)}) = \frac{U(\tau_j^{(k)}) - u_{\min}}{1 + u_{\max} - u_{\min}}, \quad (38)$$

where $u_{\min}$ and $u_{\max}$ are the minimum and maximum values of the control. Together, the normalized control solution and the corresponding collocation points, $\tau_j^{(k)}$ on $[-1, 1)$ of the initial mesh, are applied to Eqs. (34)–(37) to produce a jump function approximation for the normalized control. The jump function approximation is then evaluated at the points $\tau_{j+\frac{1}{2}}^{(k)} = \frac{1}{2}(\tau_j^{(k)} + \tau_{j+1}^{(k)})$, $j = \{1, \dots, N_k\}$; $k = \{1, \dots, K\}$. Let the evaluation of the jump function approximation at $\tau_{j+\frac{1}{2}}^{(k)}$ be denoted by $MM(\tau_{j+\frac{1}{2}}^{(k)})$. The method detects a discontinuity at the location, $\tau_{j+\frac{1}{2}}^{(k)}$, if the following condition is satisfied:

$$\left| MM(\tau_{j+\frac{1}{2}}^{(k)}) \right| \geq \eta. \quad (39)$$

The identified discontinuities are referred to as b_i , $i = \{1, \dots, n_d\}$ where n_d is the total number of identified control discontinuities. It is noted that $\eta \in [0, 1)$ in Eq. (39) is a user-specified threshold that specifies the relative size of jumps that are detected (where the likelihood of jumps being detected in the control decreases as η increases). The default value of η is set to 0.1 for the examples considered in this paper.

Bounds on the discontinuity locations are now defined. Consider for some $j = \{1, \dots, N_k\}$ and $k = \{1, \dots, K\}$ that Eq. (39) is satisfied, indicating that a discontinuity is present somewhere on the mesh interval $\tau \in [\tau_j^{(k)}, \tau_{j+1}^{(k)}]$. To account for the uncertainty incurred by using the numerical solution as a sample for the jump function approximation, a safety factor, $\mu \geq 1$ is introduced to extend the bounds estimated for the discontinuity. This safety factor provides a larger threshold to adequately capture the potential search space of the estimated switch times. Let $[b_i^-, b_i^+]$, $i = \{1, \dots, n_d\}$ be the lower and upper bounds on the locations of discontinuities in the control (that is, any discontinuity is bounded to lie on the interval $[b_i^-, b_i^+]$). The estimates of these bounds are defined as

$$\left. \begin{aligned} b_i^- &= \tau_{j+\frac{1}{2}}^{(k)} - \mu \left(\tau_{j+\frac{1}{2}}^{(k)} - \tau_j^{(k)} \right), & i &= \{1, \dots, n_d\}, \\ & & j &= \{1, \dots, N_k\}, \\ b_i^+ &= \tau_{j+\frac{1}{2}}^{(k)} + \mu \left(\tau_{j+1}^{(k)} - \tau_{j+\frac{1}{2}}^{(k)} \right), & k &= \{1, \dots, K\}. \end{aligned} \right\} \quad (40)$$

Larger values of μ are more desirable as it is more likely that the discontinuity b_i will lie in the interval $[b_i^-, b_i^+]$, $i = \{1, \dots, n_d\}$. Furthermore, the default value of μ is set to 1.5 for the examples considered in this paper.

5.1.2 Identification of bang-bang and singular domains

Identification of the domains is employed to determine the structure of the control using the discontinuities identified from the jump function approximations. Specifically, the control solution is inspected to determine if any bang-bang or singular arcs exist. The first and second derivatives of the Hamiltonian with respect to control are computed and used to detect if the Hamiltonian is linear in the control by assessing if the second derivatives are zero. If the Hamiltonian is affine in control, the first derivatives are computed and represent the switching function of the system. If the Hamiltonian is not linear in the control the structure detection process is finished and smooth mesh refinement (see Sect. 5.2.4) can be performed.

Suppose the initial solution contains the following newly identified discontinuities b_i and corresponding bounds $[b_i^-, b_i^+]$, $i = \{1, \dots, n_d\}$. The solution is divided into intervals starting with the initial time, the discontinuity locations, and ending with the final time, $\{-1, b_1\}, [b_1, b_2], \dots, [b_i, b_{i+1}], [b_{i+1}, +1]\}$, $i = \{1, \dots, n_d\}$. Next, the Hamiltonian in Eq. (29) is computed using the initial solution

$$\mathcal{H} = \mathbf{f}(\mathbf{X}(\tau), \lambda(\tau)) + \phi^\top(\mathbf{X}(\tau), \lambda(\tau))U(\tau). \quad (41)$$

Note that the costates are also obtained when solving the NLP that results from multiple-domain LGR collocation (see Sect. 3.1). The first and second derivatives with respect to the control are computed using the already computed derivatives required by the NLP solver

$$\frac{\partial \mathcal{H}}{\partial U} = \phi(\mathbf{X}(\tau), \lambda(\tau), t(\tau, t_0, t_f)), \quad (42)$$

$$\frac{\partial^2 \mathcal{H}}{\partial U^2} = \Phi(\mathbf{X}(\tau), \lambda(\tau), t(\tau, t_0, t_f)). \quad (43)$$

First, the values of Eq. (43) must be zero. If this condition is satisfied then Eq. (42) is analyzed as follows. A bang-bang interval in the control structure will occur when the switching function ϕ changes sign. The sign of the switching function ϕ is checked in each interval $[b_i, b_{i+1}]$, $i = \{1, \dots, n_d\}$. If the sign in the interval is positive, the control is constrained to its minimum value. If the sign in the interval is negative, the control is constrained to its maximum value. Additional details on these constraints are discussed in Sect. 5.2.1.

A singular interval in the control structure will occur when the switching function ϕ is zero at every point in the interval $[b_i, b_{i+1}]$. Assessing if the switching function ϕ is zero over the current interval is nontrivial. Due to the NLP being ill-conditioned when a singular arc is present, the estimated solution over a singular interval will suffer from larger numerical error. The user-defined zero threshold becomes critical in detecting the presence of a singular arc because the switching function will never be exactly zero. This threshold is heavily influenced by the coarseness of the initial mesh and the accuracy of the detected discontinuity locations. Once a singular arc has been detected, a regularization method is employed as described in Sects. 5.2.2 and 5.2.3. If a scenario occurs where the entire control is singular on $[t_0, t_f]$, then

no discontinuities will be detected. In this situation, the identification procedure is applied over the entire time domain so that the singular arc can be identified.

5.1.3 Structure decomposition

Assuming the methods of Sects. 5.1.1 and 5.1.2 have identified discontinuities and intervals that are bang-bang or singular, the initial mesh is now decomposed into the multiple-domain structure. Once acquired, the detected structure of the nonsmooth control is used to introduce the appropriate number of domain interface variables, $t_s^{[d]}$, $d = \{1, \dots, D-1\}$, to be solved for on subsequent mesh iterations, where the initial guess for each variable is the estimated discontinuity location b_i , $i = \{1, \dots, n_d\}$ that was found using the method in Sect. 5.1.1. The domain interface variables are included in the NLP by adding them as additional decision variables that define the new domains, $\mathcal{P}_d = [t_s^{[d-1]}, t_s^{[d]}]$, $d = \{1, \dots, D\}$. Specifically, the domain interface variables are employed by dividing the time horizon $t \in [t_0, t_f]$ of the original optimal control problem into D domains as described in Sect. 3.

Next, bounds on the domain interface variables are enforced to prevent the collapse or overlap of domains. The bounds provide an additional constraint on the domain interface variables. The upper and lower bounds on each domain interface variable are determined by taking the discontinuity bounds found in Sect. 5.1.1 and transforming them to the time interval $t \in [t_0, t_f]$ using the transformation in Eq. (13). Thus, the bounds $[b_i^-, b_i^+]$, $i = \{1, \dots, n_d\}$ are transformed to $[t_l^{[d-1]}, t_u^{[d]}]$, $d = \{1, \dots, D-1\}$.

This approach to structure decomposition partitions the entire problem domain into multiple domains of the form described in Sect. 3 such that the switch times are represented by the strategically placed domain interface variables $t_s^{[d]}$, $d \in \{1, \dots, D-1\}$. A schematic for the process of decomposing the nonsmooth control structure into a multiple-domain formulation with domain interface variables is shown in Fig. 1. Additionally, the form of the control in each domain is classified as either bang-bang, singular, or regular. In the next section, the constraints and refinement strategies required by each type of domain are discussed.

5.2 Domain constraints and refinement

Now that structure detection and decomposition has taken place by the methods of Sect. 5.1, additional constraints are required to properly constrain the multiple-domain optimal control problem. Recall that there are three types of domain classifications: bang-bang, singular, and regular. Each domain type requires its own set of constraints and refinement methods that are detailed in the following sections.

5.2.1 Bang-bang domain constraints

Suppose the problem has been partitioned into D domains based on the results of structure detection, and it has been determined that B domains are bang-bang by the method of Sect. 5.1.2, where $B \leq D$. The value of the switching function ϕ

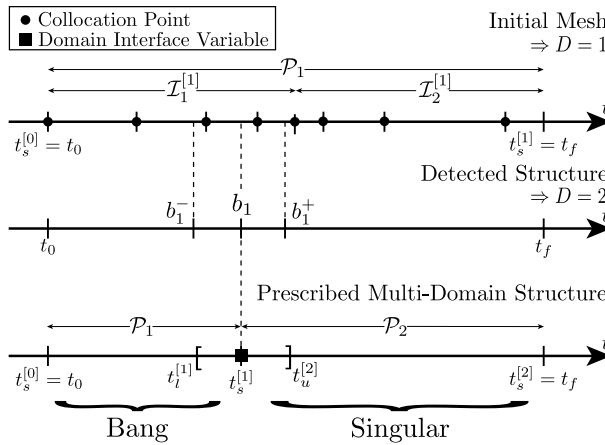


Fig. 1 Schematic of process for decomposing the nonsmooth optimal control problem into D domains where the $D - 1$ domain interface variables are included as optimization variables to determine the optimal switch times in the control

in the current domain of interest $\mathcal{P}_d$ is used to determine the value of the control over that domain. Recall, that the switching function assigns the value of the control according to

$$\begin{aligned} u^{[d]}(\tau) &= u_{\min}, & \phi^{[d]}(\mathbf{X}, \lambda, t(\tau, t_0, t_f)) &> 0, \\ u^{[d]}(\tau) &= u_{\max}, & \phi^{[d]}(\mathbf{X}, \lambda, t(\tau, t_0, t_f)) &< 0. \end{aligned} \quad (44)$$

The control is then constrained to its corresponding maximum or minimum value in the resulting multiple-domain optimal control problem. The bang-bang control is now appropriately constrained over its domain and the corresponding domain interface variables can be optimized to the optimal switch time locations.

5.2.2 Regularization of a singular domain

Assume now that the entire domain $[t_0, t_f]$ of the optimal control problem has been partitioned into D domains using the structure detection method as described in Sect. 5.1. Assume further that, using the procedure given in Sect. 5.1.2, $\mathcal{S}$ of these D domains are classified as singular (where $\mathcal{S} \leq D$ and $\mathcal{S} + \mathcal{B} \leq D$) such that $\{s_1, \dots, s_{\mathcal{S}}\} \subseteq \{1, \dots, D\}$ are the indices corresponding to the singular domains. The singular domains are then defined, respectively, on the intervals $[t_s^{[s_d-1]}, t_s^{[s_d]}] \subseteq [t_0, t_f]$, $d = \{1, \dots, \mathcal{S}\}$.

In any domain that is classified as singular, the following iterative regularization method is employed. First, the objective functional in the singular domain is augmented with the regularization term

$$\delta_{s_d} = \frac{\epsilon}{2} \int_{t_{s_d-1}}^{t_{s_d}} (u(t) - \alpha_p(t))^2 dt, \quad d = \{1, \dots, S\}, p = \{1, 2, \dots\} \quad (45)$$

where $u(t)$ is the optimal control to be determined when solving the problem, $\alpha_p(t) \in \mathbb{R}$ is a known function that changes with each iteration of the regularization method (see Sect. 5.2.3), and

$$\frac{\epsilon}{2} (u(t) - \alpha_p(t))^2 \quad (46)$$

is the integrand of Eq. (45). Furthermore, ϵ is a user-defined weighting parameter that is chosen based on the particular problem under consideration. Augmenting the Hamiltonian with the term in Eq. (46) results in a Hamiltonian that is quadratic in the control. Consequently, the optimal control problem, that would be singular without the inclusion of the term in Eq. (46), becomes regular (nonsingular). While in principle ϵ can be any positive value, it must be sufficiently large to eliminate the indeterminacy of determining the optimal control on the singular arc, but it must be sufficiently small so that the resulting optimal control is in close proximity to the true singular optimal control.

5.2.3 Iterative procedure for determining singular control

Next, it is important to understand the source of the function $\alpha_p(t)$. Singular domain refinement is employed in an iterative fashion where p is the iteration number of the singular domain refinement procedure. For $p = 1$, $\alpha_p(t)$ is set to zero. Then, for $p > 1$, $\alpha_p(t)$ is obtained using a continuous piecewise cubic approximation of the control obtained from the solution of the NLP on iteration $p - 1$ and the approximation for $\alpha_p(t)$ is obtained by interpolation using a piecewise cubic polynomial with the properties defined in Ref. [47].

The augmented multiple-domain optimal control problem that is solved by the regularization method is then stated as follows. First, for any iteration $p \geq 1$ the objective functional to be minimized includes the terms δ_{s_d} , $\{d = 1, \dots, S\}$, and is defined as

$$\mathcal{J}_a = \mathcal{J} + \sum_{d=1}^S \delta_{s_d}. \quad (47)$$

Furthermore, the constraints include the dynamic constraints, the boundary conditions, and the path constraints given, respectively, in Eqs. (2)–(4). The process for updating the iteration of the regularization method is as follows. First, after solving the NLP arising from LGR collocation with $D = 1$, the structure detection and decomposition method of Sect. 5.1 is employed for one mesh iteration. On the second mesh iteration (that is, $M = 2$) the value of p is set to unity (that is, $p \equiv 1$) and the regularization term of Eq. (45) is augmented to the objective functional in any domain classified as singular with $\alpha_p(t) \equiv 0$ and the resulting NLP is then solved. Then, from the third mesh iteration onwards (that is, $M > 2$), the value of p is incremented as $p \rightarrow p + 1$ and the value of $\alpha_p(t)$ is determined based on the

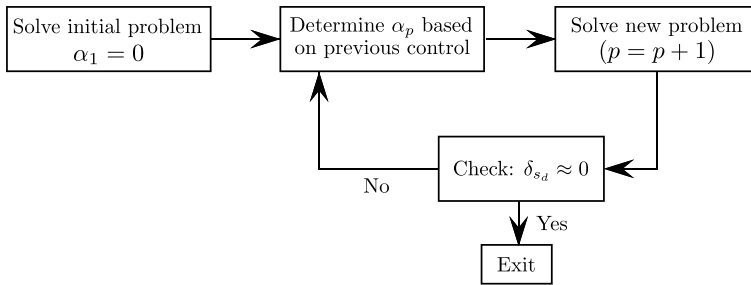


Fig. 2 Description of the regularization method during mesh refinement iteration, M

aforementioned continuous piecewise cubic approximation from the discrete control solution obtained on iteration p . The process of refining the mesh and updating the function $\alpha_p(t)$ is then repeated until the regularization term δ lies below a user-specified tolerance σ . The regularization procedure is independent of regular mesh refinement meaning that refinement of the mesh always occurs on each iteration and the updates to $\alpha_p(t)$ occur independently from the mesh refinement actions. Figure 2 provides a schematic of the regularization method that is enforced over each mesh refinement iteration in a singular domain.

5.2.4 Regular domain refinement

Suppose that a particular domain $\mathcal{P}_d$ has been determined to be regular, that is, the domain $\mathcal{P}_d$ is categorized neither as bang-bang nor singular. Although the domain $\mathcal{P}_d$ is categorized as regular, this domain may still require mesh refinement. In particular, mesh refinement of the domain $\mathcal{P}_d$ will be required if the maximum relative error on the domain $\mathcal{P}_d$ exceeds the relative error tolerance. Several methods have been previously developed for smooth mesh refinement [10, 16, 24, 25]. In this research, domains where the solution is smooth are refined using the method developed in Ref. [10]. The mesh refinement method of Ref. [10] occurs on every iteration until the user-specified mesh error tolerance is met and is independent of the status of the regularization procedure in Section 5.2.2. It is noted that Ref. [10] also includes a method for the computation of a relative error estimate. The reader is referred to Ref. [10] for a more detailed explanation of how both smooth mesh refinement operates and an estimate of the relative error on a mesh.

5.3 Procedure for solving bang-bang and singular optimal control problems

An overview of the proposed method for bang-bang and singular optimal control problems is shown below. The mesh refinement iteration is denoted by M and is incremented by one with each loop of the method. The regularization refinement iteration is denoted by p and is incremented by one on each mesh iteration as required by the regularization method. This method terminates when two requirements are met. First, the regularization term (45) must be within a user specified

tolerance σ from zero, or if Eq. (45) remains identical in value over three consecutive iterations. Second, the mesh error tolerance, e , must be satisfied on each mesh interval or if M reaches a prescribed limit, $M_{\max}$. The method is executed as follows:

Method for Solving Bang-Bang & Singular Optimal Control Problems

Step 1: Set $M = 0$ and specify initial mesh. All mesh intervals form a single domain.

Step 2: Solve NLP of Section 3 on mesh M .

Step 3: Compute relative error e on current mesh M .

Step 4: If $M = 1$, employ structure detection and decomposition in Section 5.1.

- (a): Determine the number of switch times, n_d , using the methods of Section 5.1.1.
- (b): Classify the intervals as bang-bang or singular using the method of 5.1.2.
- (c): Assign domain interface variables by method of Section 5.1.3.
- (d): Partition time horizon into domains by method of Section 5.1.3.
- (e): Perform domain refinement according to Section 5.2.
 - (i): Enforce control constraints in each bang-bang domain as in Section 5.2.1.
 - (ii): Employ regularization method in singular domains according to Section 5.2.2.
 - (iii): Apply mesh refinement in regular domains as in Section 5.2.4.

Step 5: If $n_d = 0$ or $M > 1$, apply smooth mesh refinement and proceed to **Step 7**.

Step 6: If $(M > 1, \delta \leq \sigma, \text{ and } e_{\max} \leq e)$ or $(M > M_{\max})$, then quit. Otherwise:

- (a): Apply iterative procedure of Section 5.2.3.
- (b): Increment $M \rightarrow M + 1$, $p \rightarrow p + 1$ and return to **Step 2**

Step 7: Increment $M \rightarrow M + 1$ and return to **Step 2**.

6 Examples

In this section, the aforementioned BBSOC method described in Sect. 5 is studied on three nontrivial bang-bang and singular optimal control problems. Each of the three problems demonstrates the methods abilities to solve different types of nonsmooth optimal control problems including a purely bang-bang control, a

bang-singular-bang control with no analytic solution, and a bang-singular control where the analytic solution exists. A fourth example with a smooth optimal solution is also solved using the method of Sect. 5 and compared against a previously developed mesh refinement method. In addition, the following values of design parameters in the method are chosen for all examples: $\eta = 0.1$ and $\mu = 1.5$ while the value of the regularization parameter ϵ is different for each example and ranges between unity and 10^{-8} (depending upon the problem being solved).

All results obtained using the BBSOC method are compared with either an analytic solution (if an analytic solutions exists), a numerical solution obtained without enforcing the known structure of the optimal control, or a highly accurate numerical solution obtained by enforcing the known structure of the optimal control. Any numerical solution obtained by enforcing the known structure of the optimal control is referred to as a *baseline* solution. All numerical solutions, other than those obtained using the BBSOC method, are obtained using the MATLAB® optimal control software GPOPS – II as described in Ref. [48], and GPOPS – II is referred to from this point forth as the *hp*-LGR method. For any results obtained where the known structure of the optimal control is not enforced, the problem is formulated as a single-phase optimal control problem for use with the *hp*-LGR method. For any results obtained where the known structure of the optimal control is enforced, the problem is formulated as a multiple-phase optimal control problem for use with *hp*-LGR method, and the control in each phase is either free (if the control is regular), is set to either its known lower or upper limit (if the control is bang-bang), or is determined by enforcing the singular arc optimality conditions (if the control is singular), and the switch times (which are the endpoints of each phase) are determined as part of the optimization. Finally, for all numerical results obtained, the accuracy of the solution is improved using the mesh refinement method of Ref. [10] where an error analysis is performed using the error estimate described in Ref. [10]. For completeness, Table 1 provides a table with the nomenclature that identifies the various methods being compared in this section.

All results are obtained using MATLAB® and the nonlinear program developed in Sect. 3 is solved using IPOPT [3] in full-Newton mode. The NLP solver tolerance is set to 10^{-8} and the mesh refinement tolerance for smooth mesh refinement [10] is 10^{-6} . All first and second derivatives are supplied to IPOPT using the automatic differentiation software ADiGator [49]. In each example, the initial mesh consists of ten uniformly spaced mesh intervals and four collocation points per mesh interval, and the initial guess for all examples is a straight line for variables with boundary

Table 1 Nomenclature of the various methods being compared in Sect. 6

Method Name	Meaning
BBSOC	Method Developed in This Paper
<i>hp</i> -LGR	One-Phase Implementation of GPOPS – II Without Enforcement of Control Structure
Baseline	Multiple-Phase Implementation of GPOPS – II With Enforcement of Control Structure

conditions at both endpoints and is a constant for variables with boundary conditions at only one endpoint. Finally, all computations were performed on a 2.9 GHz 6-Core Intel Core i9 MacBook Pro running Mac OS Big Sur Version 11.3 with 32 GB 2400 MHz DDR4 of RAM, using MATLAB version R2019b (build 9.7.0.1190202) and all computation (CPU) times are in reference to this aforementioned machine.

6.1 Example 1: Robot arm problem

Consider the following problem where the goal is to reorient a robotic arm in minimum time [50]:

$$\begin{aligned} &\text{minimize} && \mathcal{J} = t_f, \\ &\text{subject to} && \begin{cases} \dot{y}_1(t) = y_2(t), & y_1(0) = 9/2, \quad y_1(t_f) = 9/2, \\ \dot{y}_2(t) = u_1(t)/L, & y_2(0) = 0, \quad y_2(t_f) = 0, \\ \dot{y}_3(t) = y_4(t), & y_3(0) = 0, \quad y_3(t_f) = 2\pi/3, \\ \dot{y}_4(t) = u_2(t)/I_\theta, & y_4(0) = 0, \quad y_4(t_f) = 0, \\ \dot{y}_5(t) = y_6(t), & y_5(0) = \pi/4, \quad y_5(t_f) = \pi/4, \\ \dot{y}_6(t) = u_3(t)/I_\phi, & y_6(0) = 0, \quad y_6(t_f) = 0, \\ -1 \leq u_i(t) \leq 1, & (i = 1, 2, 3), \end{cases} \end{aligned} \quad (48)$$

where t_f is free, $I_\phi = \frac{1}{3}((L - y_1(t))^3 + y_1^3(t))$, $I_\theta = I_\phi \sin^2(y_5(t))$, and $L = 5$. The robot arm problem has a bang-bang structure for all three components of the control and contains a total of five switch points in the control. A baseline solution is obtained for comparison and the results are provided in Table 2.

The control solution obtained from solving this problem using the BBSOC and the *hp*-LGR method are shown in Fig. 3. Observing the controls in Fig. 3a–c, it is seen that the five switch times are identified and the controls are constrained to their corresponding boundaries after just one iteration of the BBSOC method. In contrast, the *hp*-LGR method does not correctly identify all five discontinuities and as a result obtains incorrect approximations of the controls. In particular, the *hp*-LGR method attempts to place more collocation points in the neighborhood of the switch times (thus increasing the size of the NLP) while still not correctly identifying the exact locations of the switches in the control. Table 2 shows the numerical values of the switch times obtained from the BBSOC method, *hp*-LGR method, and the baseline solutions. It is seen that the BBSOC method produces highly accurate approximations of both the switch times and objective and is in excellent agreement with the baseline solution. On the other hand, the *hp*-LGR method (which is not designed specifically for solving

Table 2 Comparison of computational results for Example 1

	$t_s^{[1]}$	$t_s^{[2]}$	$t_s^{[3]}$	$t_s^{[4]}$	$t_s^{[5]}$	$\mathcal{J}$	CPU [s]
BBSOC	2.285228	2.796043	4.570456	6.344869	6.855684	9.140912	0.63
<i>hp</i> -LGR	2.249813	2.795031	4.518293	6.330691	6.820305	9.140984	3.25
Baseline	2.285228	2.796043	4.570456	6.344869	6.855684	9.140912	1.59

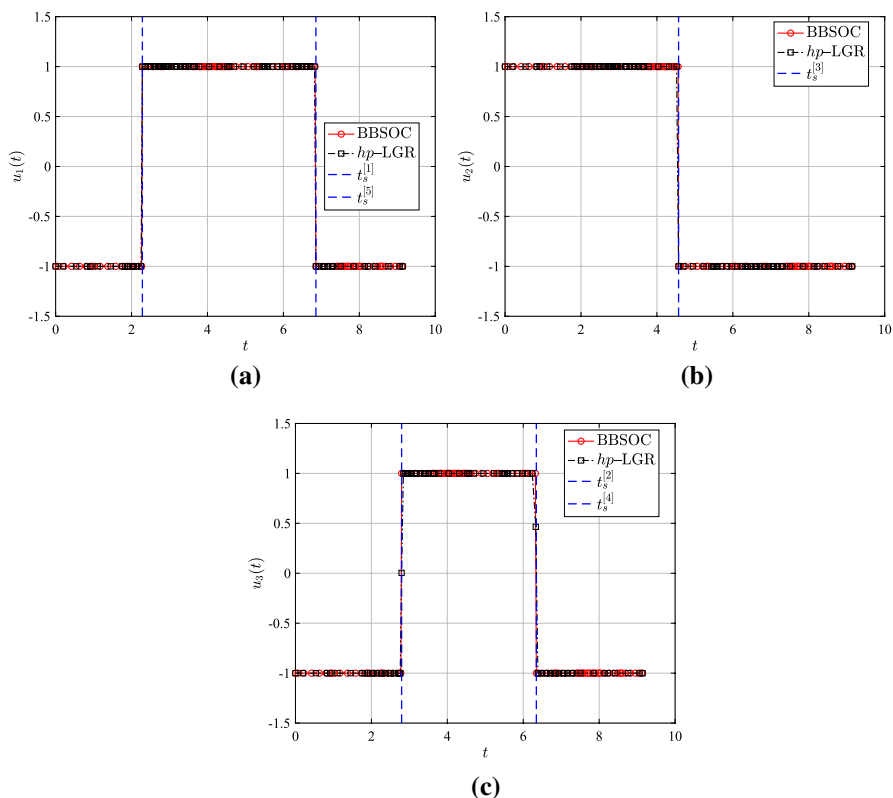


Fig. 3 Control component solutions for Example 1 illustrate the discrepancies between the solutions obtained by the BBSOC and hp -LGR methods

problems with nonsmooth solutions) attains a less accurate solution when compared with the BBSOC method. In addition to accuracy, Table 2 provides a comparison of the CPU time required for each approach. In particular, it is seen from Table 2 that the BBSOC method converges to the optimal solution more efficiently compared with either the hp -LGR method or the baseline solution.

6.2 Example 2: Goddard rocket problem

Consider the following optimal control problem [42]:

$$\begin{aligned}
 &\text{minimize} && \mathcal{J} = -h(t_f), \\
 &\text{subject to} && \begin{cases} \dot{h}(t) = v(t), & h(0) = 0, \ h(t_f) = \text{Free}, \\ \dot{v}(t) = \frac{T(t)-D}{m(t)} - g, & v(0) = 0, \ v(t_f) = \text{Free}, \\ \dot{m}(t) = -\frac{T(t)}{c}, & m(0) = 3, \ m(t_f) = 1, \\ 0 \leq T(t) \leq T_{\max}, \end{cases} \quad (49)
 \end{aligned}$$

Table 3 Comparison of computational results for Example 2

	$t_s^{[1]}$	$t_s^{[2]}$	t_f	$\mathcal{J}^e$	δ	ϵ	p	CPU [s]
BBSOC	13.751266	21.987362	42.887912	-18550.87185	7.86×10^{-8}	10^{-6}	3	2.86
<i>hp</i> -LGR	13.541287	22.213037	42.887750	-18550.87039	—	—	—	2.24
Baseline	13.751270	21.987363	42.887912	-18550.87186	—	—	—	2.57

where h is the altitude, v is the velocity, m is the mass, T is the thrust (and is the control), $D = D_0 v^2(t) \exp(-h(t)/H)$, and the final time is free. Further details on the model and the parameters can be found in Ref. [42]. In particular, the control T has a bang-singular-bang structure and differentiating $\mathcal{H}_u$ twice with respect to time leads to the following singular control law [1]:

$$T_{\text{sing}}(t) = D + m(t)g + \left[\frac{c^2(1 + \frac{v(t)}{c})}{Hg} - 1 - \frac{2c}{v(t)} \right] \left[\frac{m(t)g}{1 + \frac{4c}{v(t)} + \frac{2c^2}{v^2(t)}} \right]. \quad (50)$$

The singular arc condition in Eq. (50) is used to obtain a baseline solution and the results are included in Table 3 for comparison.

The problem in Eq. (49) is solved using the BBSOC and *hp*-LGR methods. Both of these control solutions are compared in Fig. 4a and parameters related to the regularization method are provided in Table 3. Figure 4a demonstrates that the regularization method has converged to the correct singular control while also correctly identifying the switch times. In contrast, the *hp*-LGR method obtains a solution that exhibits oscillations at the switch times defining the singular arc. These oscillations are also seen in the initial iterations of the regularization method, but are removed by the third iteration as seen in Fig. 4b.

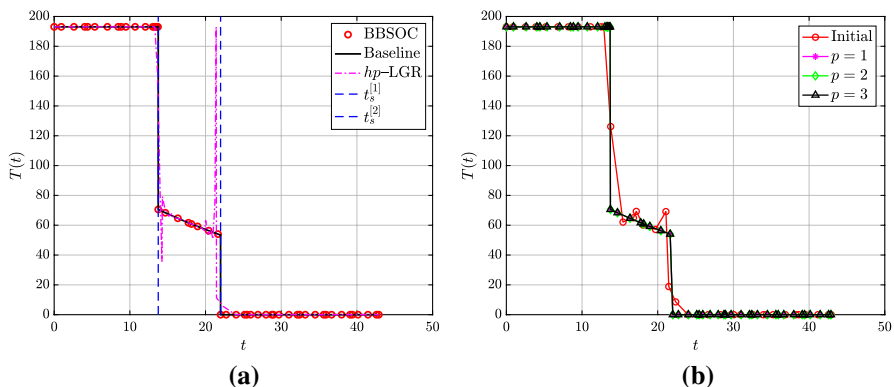


Fig. 4 Control solution for Example 2 in Fig. 4a illustrates the numerical discrepancies between the solutions obtained by the BBSOC method, *hp*-LGR method, and the baseline method. Figure 4b shows the control history of the regularization procedure for obtaining the singular control

Next, Table 3 compares the solution obtained using the BBSOC method, *hp*-LGR method, and the baseline solution. These results further validate the BBSOC methods ability to identify the optimal switch times, final time, and cost. The BBSOC results are also in close agreement with the baseline solution while the *hp*-LGR results are not. The computation times for all three methods are also compared in Table 3 and all three methods have very similar CPU times with the BBSOC method taking the longest; however, it produces a far more accurate solution and achieves the same accuracy as the baseline solution without any a priori knowledge of the singular problem.

It is well known that the singular surface in the optimal control problem of Eq. (49) can be reduced to lie in the state space (that is, the singular surface is a three-dimensional surface in the space defined by h , v , and m). In particular, the singular surface is defined as

$$m = \frac{D_0 v^2 \exp(-h/H)(1 + v/c)}{g}. \quad (51)$$

Figure 5 shows the singular surface defined by Eq. (51) and the state solution obtained using the BBSOC method, *hp*-LGR method, and baseline solution. It is observed in Fig. 5b that the segment of the *hp*-LGR solution that corresponds to the singular interval does not lie in close proximity to the singular surface. On the other hand, the portion of the trajectory corresponding to the singular interval using either the BBSOC or the baseline solution does lie on the singular surface.

6.3 Example 3: Jacobson's problem

Consider the following problem posed in [32]:

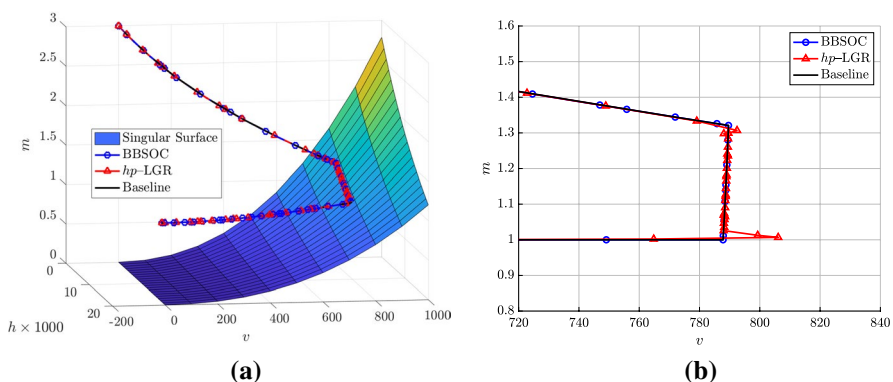


Fig. 5 The singular surface for Example 2 is presented with the corresponding trajectory obtained using the BBSOC method, *hp*-LGR method, and the baseline solution. The numerical error accrued by the *hp*-LGR solution is shown in a zoomed in 2D plot of the trajectory in Fig. 5b

Table 4 Comparison of computational results for Example 3

	$t_s^{[1]}$	$\mathcal{J}^*$	δ	ϵ	p	CPU [s]
BBSOC	1.41376404	0.37699193	6.31×10^{-14}	10^{-8}	4	2.02
<i>hp</i> -LGR	1.41194167	0.37699503	—	—	—	2.13
Analytic	1.41376409	—	—	—	—	—

$$\begin{aligned}
 &\text{minimize} \quad \mathcal{J} = \frac{1}{2} \int_0^{t_f} (x_1^2(t) + x_2^2(t)) dt, \\
 &\text{subject to} \quad \begin{cases} \dot{x}_1(t) = x_2(t), & x_1(0) = 0, \quad x_1(t_f) = \text{Free}, \\ \dot{x}_2(t) = u(t), & x_2(0) = 1, \quad x_2(t_f) = \text{Free}, \\ -1 \leq u(t) \leq 1, \end{cases} \quad (52)
 \end{aligned}$$

where $t_f = 5$. The optimal control problem of Eq. (52) has an analytic solution. Consequently, the analytic solution can be used to assess the accuracy of the BBSOC method. Further details on the derivation of the analytic solution to the example in Eq. (52) can be found in [27]. The singular control is $u_{\text{sing}}^*(t) = x_1(t)$, $t \geq \approx 1.41376409$ and the analytic switch time $t_s^{[1]} = 1.41376409$ is shown in Table 4.

Figure 6 shows the control solutions obtained when solving the optimal control problem of Eq. (52) using both the BBSOC and *hp*-LGR methods. Using the BBSOC method, it is seen that a high-accuracy approximation of both the switch time and the singular control are obtained after two iterations of the BBSOC method. In contrast, the *hp*-LGR solution exhibits fluctuations in the neighborhood of the switch time. Next, Fig. 6b shows the history of the control obtained on each iteration of the regularization method that is part of the BBSOC method. As already indicated in Fig. 6 and further emphasized by Fig. 6b, these fluctuations present in the *hp*-LGR solution are eliminated because of the inclusion of the regularization term in the objective functional.

Next, Table 4 provides a comparison of the switch time and objective values obtained using both the BBSOC and *hp*-LGR methods alongside the analytic solution

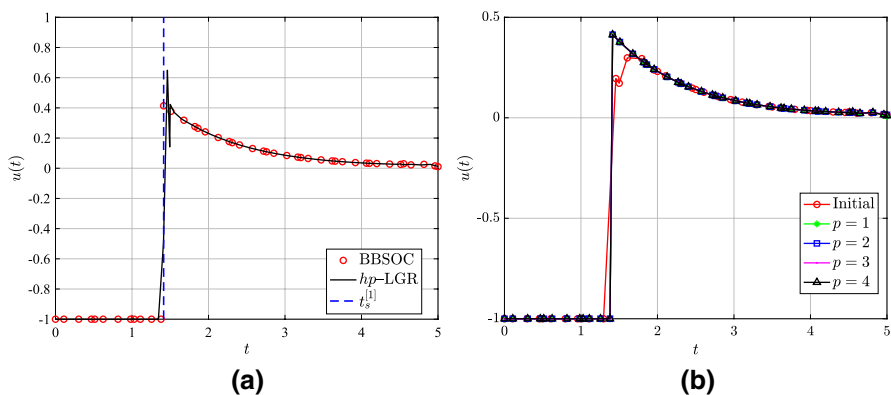


Fig. 6 Control solution for Example 3 illustrating the numerical discrepancies between the solutions obtained by the BBSOC and *hp*-LGR methods. Figure 6b shows the control history of the regularization procedure for obtaining the singular control

(where it is noted that Table 4 also provides the parameter ϵ used in the regularization term of Eq. (45) along with the value δ obtained for the regularization term itself). It is seen that the value of the analytic switch time is in close agreement with the switch time obtained from the BBSOC method. Next, the computation times for both methods are also compared in Table 4. It is observed that the BBSOC method is more computationally efficient when compared with the *hp*-LGR method while simultaneously producing a more accurate solution.

6.4 Example 4: Entry vehicle crossrange maximization (smooth solution)

Consider now the crossrange maximization of an entry vehicle problem given in Ref. [1] and whose formulation is taken from Ref. [48]. It is known that the solution to this optimal control problem is smooth. For problems whose solutions are smooth, the BBSOC method should not identify any discontinuities or singular intervals. Consequently, the method should not divide the entire domain into multiple domains nor should it apply regularization. In this section the entry vehicle problem is solved using the BBSOC method. While the details of this problem are omitted here for brevity, further details of the maximum crossrange entry vehicle problem can be found in Refs. [1, 48].

The results of applying the BBSOC method to the aforementioned maximum crossrange entry vehicle problem are provided in Table 5, where M denotes the number of mesh refinement iterations and K denotes the total number of collocation points. In particular, it is seen from Table 5 that the BBSOC method obtains the same solution as the optimal control software in Ref. [48] while taking an average of 0.6 s more CPU time to solve the problem. In particular, the BBSOC method does not identify any discontinuities nor does it identify any intervals as singular. The results obtained for this example demonstrate the ability of the BBSOC method to identify correctly the fact that the optimal control for the maximum crossrange entry vehicle problem is smooth and that the BBSOC method applies only static mesh refinement (see Sect. 5.2.4) as necessary.

7 Limitations of the BBSOC method

As with any computational method, the method of this paper has limitations. The first limitation is that the approach depends upon the quality of the initial mesh that is supplied. In particular, the solution on the initial mesh is used to ascertain the structure of the solution to the optimal control problem. As a result, if the solution structure determined by the structure decomposition method described in Sect. 5.1.3

Table 5 Comparison of computational results for Example 4

	$\mathcal{J}^*$	M	K	CPU [s]
BBSOC	-0.5963	4	105	1.81
<i>hp</i> -LGR	-0.5963	4	105	1.17

is not representative of the optimal solution structure, the method may not produce a sufficiently accurate solution.

Next, the regularization method described in Sect. 5.2.2 is dependent upon the value of the regularization parameter ϵ as shown in Eq. (45). If ϵ is chosen to be too large, then the control obtained in a domain categorized as singular may not be a sufficiently accurate approximation of the singular control. On the other hand, if ϵ is chosen to be too small, then the approximation of the singular control may be noisy (which would be as if no regularization was performed). Next, if the structure detection method produces initial estimates of the switch times that do not lie within a reasonable proximity of the actual switch times or incorrectly identifies the actual switch times, then the structure obtained may be quite different from the optimal control structure. The detection of the control structure can be adjusted by changing the values of the parameters η and μ , as discussed in Sect. 5.1. While the default values of the parameters given in Sect. 5.1 worked for the problems discussed in this paper, more complex problems may require more testing from a user standpoint.

Finally, it is noted that the method of this paper is limited to problems where the control appears linearly in the Hamiltonian, that is, this paper does not consider problems where the solution is bang-bang and/or singular and the Hamiltonian is not linear in the control. Furthermore, this method does not consider the inclusion of state or mixed state-input constraints.

8 Conclusions

A method has been described for solving bang-bang and singular optimal control problems using adaptive Legendre–Gauss–Radau (LGR) collocation. First, the standard single-domain LGR collocation method is modified to be formulated as a multiple-domain method where the endpoints of each domain are variables in the optimal control problem. Next, a structure detection method is developed that identifies intervals of the original domain as either regular, bang-bang, or singular. This structure detection method is developed based on the sign of the switching function for problems where the control appears linearly in the optimal control Hamiltonian. Based on the results of the structure detection, the original domain is partitioned into multiple-domains in a manner consistent with the aforementioned multiple-domain formulation. For any domain that is categorized as regular, the control is set equal to either its lower or upper limit (based on the sign of the switching function). For any domain that is categorized as singular (that is, the sign of the switching function in such a domain is indeterminate), a regularization method is employed to determine an accurate approximation of the singular control. The various parts of the method, that is, the multiple-domain formulation, the structure detection, and the categorization of the various intervals, are combined into a unified method that solves bang-bang and singular optimal control problems. It is also noted that, for problems where the solution contains no bang-bang or singular intervals, the method reverts to a standard single-domain LGR collocation method. The method is demonstrated on four examples. Three of the examples have either a bang-bang and/or singular solution while the fourth example contains no bang-bang or singular

intervals. It is found that the method efficiently produces accurate solutions to bang-bang and singular optimal control problems.

Acknowledgements The authors gratefully acknowledge support for this research from the U.S. National Science Foundation under grants DMS-1819002 and CMMI-2031213, the U.S. Office of Naval Research under grant N00014-19-1-2543, and from Lockheed-Martin Corporation under contract 4104177872.

Data Availability Statement The data that support the findings of this study are available from the corresponding author upon request.

References

1. Betts, J.T.: Practical Methods for Optimal Control and Estimation Using Nonlinear Programming. SIAM, New York (2010)
2. Gill, P.E., Murray, W., Saunders, M.A.: SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Rev.* **47**(1), 99–131 (2002). <https://doi.org/10.1137/S0036144504446096>
3. Biegler, L.T., Zavala, V.M.: Large-scale nonlinear programming using IPOPT: an integrating framework for enterprise-wide optimization. *Comput. Chem. Eng.* **33**(3), 575–582 (2008). <https://doi.org/10.1016/j.compchemeng.2008.08.006>
4. Benson, D.A., Huntington, G.T., Thorvaldsen, T.P., Rao, A.V.: Direct trajectory optimization and costate estimation via an orthogonal collocation method. *J. Guid. Control. Dyn.* **29**(6), 1435–1440 (2006). <https://doi.org/10.2514/1.20478>
5. Garg, D., Patterson, M.A., Hager, W.W., Rao, A.V., Benson, D.A., Huntington, G.T.: A unified framework for the numerical solution of optimal control problems using pseudospectral methods. *Automatica* **46**(11), 1843–1851 (2010). <https://doi.org/10.1016/j.automatica.2010.06.048>
6. Garg, D., Hager, W.W., Rao, A.V.: Pseudospectral methods for solving infinite-horizon optimal control problems. *Automatica* **47**(4), 829–837 (2011). <https://doi.org/10.1016/j.automatica.2011.01.085>
7. Garg, D., Patterson, M.A., Darby, C.L., Francolin, C., Huntington, G.T., Hager, W.W., Rao, A.V.: Direct trajectory optimization and costate estimation of finite-horizon and infinite-horizon optimal control problems via a radau pseudospectral method. *Comput. Optim. Appl.* **49**(2), 335–358 (2011). <https://doi.org/10.1007/s10589-009-9291-0>
8. Rao, A.V., Benson, D.A., Darby, C.L., Francolin, C., Patterson, M.A., Sanders, I., Huntington, G.T.: Algorithm 902: GPOPS. A MATLAB software for solving multiple-phase optimal control problems using the gauss pseudospectral method. *ACM Trans. Math. Software* **37**(2) (2010). <https://doi.org/10.1145/1731022.1731032>
9. Kameswaran, S., Biegler, L.T.: Convergence rates for direct transcription of optimal control problems using collocation at Radau points. *Comput. Optim. Appl.* **41**(1), 81–126 (2008). <https://doi.org/10.1007/s10589-007-9098-9>
10. Patterson, M.A., Hager, W.W., Rao, A.V.: A ph mesh refinement method for optimal control. *Opt. Control Appl. Methods* **36**(4), 398–421 (2015). <https://doi.org/10.1002/oca.2114>
11. Elnagar, G., Kazemi, M.A., Razzaghi, M.: The pseudospectral legendre method for discretizing optimal control problems. *IEEE Trans. Autom. Control* **40**(10), 1793–1796 (1995). <https://doi.org/10.1109/9.467672>
12. Hager, W.W., Hou, H., Rao, A.V.: Convergence rate for a gauss collocation method applied to unconstrained optimal control. *J. Optim. Theory Appl.* **169**(3), 801–824 (2016). <https://doi.org/10.1007/s10957-016-0929-7>
13. Hager, W.W., Hou, H., Rao, A.V.: Lebesgue constants arising in a class of collocation methods. *IMA J. Numer. Anal.* **37**(4), 1884–1901 (2017). <https://doi.org/10.1093/imanum/drw060>
14. Hager, W.W., Liu, J., Mohapatra, S., Rao, A.V., Wang, X.-S.: Convergence rate for a Gauss collocation method applied to constrained optimal control. *SIAM J. Control. Optim.* **56**, 1386–1411 (2018). <https://doi.org/10.1137/16M1096761>
15. Hager, W.W., Hou, H., Mohapatra, S., Rao, A.V., Wang, X.-S.: Convergence rate for a Radau hp-collocation method applied to constrained optimal control. *Comput. Optim. Appl.* **74**, 274–314 (2019). <https://doi.org/10.1007/s10589-019-00100-1>

16. Liu, F., Hager, W.W., Rao, A.V.: Adaptive mesh refinement method for optimal control using nonsmoothness detection and mesh size reduction. *J. Franklin Inst.* **352**(10), 4081–4106 (2015). <https://doi.org/10.1016/j.jfranklin.2015.05.028>
17. Gong, Q., Fahroo, F., Ross, I.M.: Spectral algorithm for pseudospectral methods in optimal control. *J. Guid. Control. Dyn.* **31**(3), 460–471 (2008). <https://doi.org/10.2514/1.32908>
18. Miller, A.T., Hager, W.W., Rao, A.V.: Mesh refinement method for solving optimal control problems with nonsmooth solutions using jump function approximations. *Opt. Control Appl. Methods* (2021). <https://doi.org/10.1002/oca.2719>
19. Schlegel, M., Marquardt, W.: Direct sequential dynamic optimization with automatic switching structure detection. *IFAC Proc. Vol.* **37**(9), 419–424 (2004). [https://doi.org/10.1016/s1474-6670\(17\)31845-1](https://doi.org/10.1016/s1474-6670(17)31845-1)
20. Schlegel, M., Marquardt, W.: Detection and exploitation of the control switching structure in the solution of dynamic optimization problems. *J. Process Control* **16**(3), 275–290 (2006). <https://doi.org/10.1016/j.jprocont.2005.06.008>
21. Wang, P., Yang, C., Yuan, Z.: The combination of adaptive pseudospectral method and structure detection procedure for solving dynamic optimization problems with discontinuous control profiles. *Ind. Eng. Chem. Res.* **53**(17), 7066–7078 (2014). <https://doi.org/10.1021/ie404148j>
22. Chen, W., Biegler, L.T.: Nested direct transcription optimization for singular optimal control problems. *AIChE J.* **62**(10), 3611–3627 (2016). <https://doi.org/10.1002/aic.15272>
23. Chen, W., Ren, Y., Zhang, G., Biegler, L.T.: A simultaneous approach for singular optimal control based on partial moving grid. *AIChE J.* **65**(6), e16584 (2019). <https://doi.org/10.1002/aic.16584>
24. Darby, C.L., Hager, W.W., Rao, A.V.: An hp-adaptive pseudospectral method for solving optimal control problems. *Opt. Control Appl. Methods* **32**(4), 476–502 (2010). <https://doi.org/10.1002/oca.957>
25. Liu, F., Hager, W.W., Rao, A.V.: Adaptive mesh refinement method for optimal control using decay rates of legendre polynomial coefficients. *IEEE Trans. Control Syst. Technol.* **26**(4), 1475–1483 (2018). <https://doi.org/10.1109/tcst.2017.2702122>
26. Agamawi, Y.M., Hager, W.W., Rao, A.V.: Mesh refinement method for solving bang-bang optimal control problems using direct collocation. *AIAA Scitech 2020 Forum* 0378 (2020). <https://doi.org/10.2514/6.2017-1506>
27. Aghaee, M., Hager, W.W.: The switch point algorithm. *SIAM J. Control. Optim.* **59**(4), 2570–2593 (2021)
28. Kaya, C., Noakes, J.: Computational method for time-optimal switching control. *J. Optim. Theory Appl.* **117**(1), 69–92 (2003). <https://doi.org/10.1023/a:1023600422807>
29. Mehropouya, M.A., Khaksar-e Oshagh, M.: An efficient numerical solution for time switching optimal control problems. *Comput. Methods Differ. Equ.* **9**(1) (2021). <https://doi.org/10.22034/cmde.2020.33529.1542>
30. Aronna, M.S., Bonnans, J.F., Martinon, P.: A shooting algorithm for optimal control problems with singular Arcs. *J. Optim. Theory Appl.* **158**(2), 419–459 (2013). <https://doi.org/10.1007/s10957-012-0254-8>
31. Mehra, R., Davis, R.: A generalized gradient method for optimal control problems with inequality constraints and singular arcs. *IEEE Trans. Autom. Control* **17**(1), 69–79 (1972). <https://doi.org/10.1109/tac.1972.1099881>
32. Jacobson, D., Gershwin, S., Lele, M.: Computation of optimal singular controls. *IEEE Trans. Autom. Control* **15**(1), 67–73 (1970). <https://doi.org/10.1109/tac.1970.1099360>
33. Maurer, H.: Numerical solution of singular control problems using multiple shooting techniques. *J. Optim. Theory Appl.* **18**(2), 235–257 (1976). <https://doi.org/10.1007/bf00935706>
34. Andrés-Martínez, O., Flores-Tlacuahuac, A., Kameswaran, S., Biegler, L.T.: An efficient direct/indirect transcription approach for singular optimal control. *AIChE J.* **65**(3), 937–946 (2018). <https://doi.org/10.1002/aic.16487>
35. Caponigro, M., Ghezzi, R., Piccoli, B., Trélat, E.: Regularization of chattering phenomena via bounded variation controls. *IEEE Trans. Autom. Control* **63**(7), 2046–2060 (2018). <https://doi.org/10.1109/TAC.2018.2810540>
36. Mall, K., Grant, M.J., Taheri, E.: Uniform trigonometrization method for optimal control problems with control and state constraints. *J. Spacecr. Rocket.* **57**(5), 995–1007 (2020). <https://doi.org/10.2514/1.a34624>

37. Fabien, B.C.: Indirect solution of inequality constrained and singular optimal control problems via a simple continuation method. *J. Dyn. Syst. Meas. Contr.* **136**(2), 021003 (2013). <https://doi.org/10.1115/1.4025596>
38. Andrés-Martínez, O., Biegler, L.T., Flores-Tlacuahuac, A.: An indirect approach for singular optimal control problems. *Comput. Chem. Eng.* **139**, (2020). <https://doi.org/10.1016/j.compchemeng.2020.106923>
39. Maga, L., Reverberi, A., et al.: A pattern recognition approach to the solution of optimal singular control problems. *Chem. Eng. J.* **68**(1), 35–40 (1997). [https://doi.org/10.1016/S1385-8947\(97\)00068-5](https://doi.org/10.1016/S1385-8947(97)00068-5)
40. Athans, M., Falb, P.L.: *Optimal Control: An Introduction to the Theory and Its Applications*. Courier Corporation, North Chelmsford (2013)
41. Kirk, D.E.: *Optimal Control Theory: An Introduction*. Courier Corporation, North Chelmsford (2004)
42. Bryson, A.E., Ho, Y.: *Applied Optimal Control: Optimization, Estimation, and Control*. Hemisphere Publishing Corporation, London (1975)
43. Schättler, H., Ledzewicz, U.: *Geometric Optimal Control: Theory, Methods and Examples*, vol. 38. Springer, Berlin (2012)
44. Kelley, H., Kopp, R.E., Moyer, H.G.: *Topics in Optimization*, edited by Leitman (1967)
45. Kopp, R.E., Moyer, H.G.: Necessary conditions for singular extremals. *AIAA J.* **3**(8), 1439–1444 (1965). <https://doi.org/10.2514/3.3165>
46. Archibald, R., Gelb, A., Yoon, J.: Polynomial fitting for edge detection in irregularly sampled signals and images. *SIAM J. Numer. Anal.* **43**(1), 259–279 (2005). <https://doi.org/10.1137/s0036142903435259>
47. Fritsch, F.N., Carlson, R.E.: Monotone piecewise cubic interpolation. *SIAM J. Numer. Anal.* **17**(2), 238–246 (1980). <https://doi.org/10.1137/0717021>
48. Patterson, M.A., Rao, A.V.: GPOPS-II. *ACM Trans. Math. Software* **41**(1), 1–37 (2014). <https://doi.org/10.1145/2558904>
49. Weinstein, M.J., Rao, A.V.: Algorithm 984: ADiGator, a toolbox for the algorithmic differentiation of mathematical functions in MATLAB using source transformation via operator overloading. *ACM Trans. Math. Software* **44**(2), 1–25 (2017). <https://doi.org/10.1145/3104990>
50. Dolan, E.D., More, J.J., Munson, T.S.: *Benchmarking optimization software with COPS 3.0*. Tech. rep., Argonne National Laboratory, Argonne, Illinois (2004). <https://doi.org/10.2172/834714>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.