An efficient numerical method for optimal control problems with low dimensional nonlinearities

Yifei Zheng[1], Peter M. Dower[2], William M. McEneaney[1]

***Abstract*—A class of finite time horizon optimal control problems with nonlinear dynamics and non-quadratic costs is considered. Stat-quad duality is used to transform the problem into a canonical form. A derivative-free numerical method that only uses fixed-point iterations is devised to solve it efficiently, the convergence of which is limited only by the existence of the staticizing control process** ($\arg\operatorname{stat}$)**. For problems with mild and low-dimensional nonlinearities, this leads to dimension reduction of the control space. A 4-D and a 25-D control problem are solved to demonstrate its accuracy and scalability.**

I. Introduction

Numerical solution of optimal control problems is often challenging, especially for nonlinear problems with high-dimensional state space and long time horizons. Such nonlinear optimizations arise not only from control design in robotics and other real-time applications [1], [2], but also naturally from models of conservative mechanical systems, such as gravitational systems [3], [4].

Staticization (i.e. locating stationary points) of a function is a notion related to extremization, but with distinct properties and some peculiarities. The definition of staticization will be given in §II-A. Here we present some cases that differentiate staticization from extremization. In [5], Gray and Taylor noted that conservative mechanical systems evolve along stationary trajectories of the action functional, which are not necessarily the minimizers. Furthermore, the equilibria of differential games can sometimes be viewed as the stationary points of some cost functional. This allows optimal control and game problems to be considered as stationary control problems in a unified way. The reader is referred to [6], [7], [8] for motivations and connections to extremization.

Traditionally, optimal control problems are solved using some form of dynamic programming, Pontryagin's principle, or other direct methods from nonlinear programming [9]. Certain differential games can also be solved using similar tools to those used in optimal control, such as Hamilton-Jacobi-Issac equation and Pontryagin's principle [10]. Continuous-time dynamic programming typically requires solving the Hamilton-Jacobi-Bellman (HJB) equation, which suffers from the curse-of-dimensionality, whereas Pontryagin's principle requires solving a two-point boundary value problem (TPBVP), which is computationally challenging for long time durations. As such, these methods are usually seen as distinct approaches to the same problem with very dissimilar characteristics. That said, some hybrid methods have also been proposed, which mitigate the shortcomings of each method [11], [12].

In this paper, we treat a class of finite time horizon control problems with non-quadratic cost function and nonlinear dynamics, where we *staticize* the cost functional.

The effort here is based on the results in [13], [14], [15], where stat-quad duality is used to obtain a representation of the value function that corresponds to a "simpler" staticization problem. The dimension of the space of control input in the new problem roughly corresponds to the "dimension of nonlinearity" of the original problem. In particular, if the nonlinearity in state dynamics and the non-quadratic part in the running cost are defined on a subspace of the state-costate space (which are common, for example, in robotics and other rigid-body systems [1]), the control dimension of the staticization problem may be reduced.

In the current development, we propose a new algorithm that subdivides long duration problems into smaller problems using the dynamic programming principle, and solves each subproblem using Pontryagin's principle. A time-varying coordinate transform is used on a per-subinterval basis to avoid propagating differential Riccati equations (DRE) directly for long durations, which ensures that the DRE solution does not blow up.

The proposed algorithm has a multiple shooting structure due to division of the interval. Although commonly applied to optimal control problems, multiple shooting is usually solved using derivative-based solvers such as Newton's method or gradient descent [16], [17], which incur additional computational cost to upkeep the Jacobian. However, the proposed algorithm involves fixed-point iterations only and is derivative free. As such, the algorithm performs relatively well for high dimensional problems.

This paper is structured as follows. The optimal control problem of interest and assumptions are specified in Section II, followed by the definition of stat-quad duality and the equivalent problem with linear time-invariant dynamics. The numerical method is proposed in Section III. Two numerical examples are shown in Section IV. Conclusions and future directions are indicated in Section V. In the interest of space, many results will be stated without proof.

Notationally, we adopt the following conventions: $'$ denotes transpose. I_n denotes the $n \times n$ identity matrix and $0_{m\times n}$ the $m \times n$ zero matrix. Standard notation ($\frac{\partial}{\partial}$) is used for partial derivatives with respect to time. ∇ denotes spatial derivative and ∇^2 second derivative. Where unambiguous, we

[1] Dept. Mech. and Aero. Eng., UC San Diego, La Jolla, CA 92093 USA. {yiz152,wmceneaney}@uscd.edu

[2] Dept. Elec. & Electronic Eng., Univ. Melbourne, Victoria 3010, Australia. pdower@unimelb.edu.au

This work was supported by AFOSR Grant FA9550-22-1-0015.

omit the variable(s) with respect to which the derivative is taken; otherwise we indicate them using subscripts on ∇. $i, j, k, l, m, n \in \mathbb{N}$ are used freely for indexing and counting. $\alpha, \beta, \mu, \nu, \xi, \zeta$ are used to denote processes in general. $\doteq$ is used to differentiate definitions from equalities, and the over-dot notation is used for total time derivative.

II. The optimal control problem and its stat-quad dual equivalent problem

A. Problem Definition

Let $0 \le t \le T < \infty$ denote the initial and terminal time and $s \in [t, T]$ denote an arbitrary intermediate time. We consider the control problem with state process ξ given by

$$\dot{\xi}(s) = A\xi(s) + Lf(M\xi(s)) + Bu(s), \quad \xi(t) = x \in \mathbb{R}^n, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times j}$, $L \in \mathbb{R}^{n\times l}$, $M \in \mathbb{R}^{k\times n}$, $f : \mathbb{R}^k \to \mathbb{R}^l$, and $u \in L^2((t,T);\mathbb{R}^j)$ denotes the control process. The cost function J is given by

$$J(u;t,x) \doteq \psi(\xi(T)) + \int_t^T \ell(M\xi(s)) + \frac{1}{2}\xi(s)'C\xi(s) + \frac{1}{2}u(s)'Du(s)\,ds, \tag{2}$$

where $C \in \mathbb{R}^{n\times n}$, $D \in \mathbb{R}^{j\times j}$ are symmetric invertible matrices, $\ell : \mathbb{R}^k \to \mathbb{R}$, and $\psi : \mathbb{R}^n \to \mathbb{R}$.

The following regularity conditions on f, ℓ, ψ are assumed.

Assumption 1: f and ℓ are twice continuously differentiable, and have uniformly bounded first and second derivative. $\psi \in C^\infty(\mathbb{R}^n;\mathbb{R})$ has bounded second derivative.

This problem deviates from the usual linear-quadratic-regulator problem due to the nonlinear term f in the dynamics, the nonquadratic term ℓ in the running cost, and that the terminal cost ψ need not be quadratic.

The proposed algorithm will exploit the fact that f, ℓ do not depend on the full state x but on Mx, and that f does not map directly into $\mathbb{R}^n$, but "broadcast" through a linear map L. As the nonlinearity f, ℓ are introduced through rank-deficient matrices M and L, we say that the control problem (1)–(2) has "low-dimensional nonlinearity".

We consider the staticization (stat) of the cost function. Loosely speaking, staticization of a differentiable function is equivalent to locating its critical points and critical values. We take the following characterization of $\arg\operatorname{stat}$ from [7, Lemma 1] as its definition here.

Definition 1 ($\arg\operatorname{stat}, \operatorname{stat}$): Let V be a Hilbert space and let $U \subset V$ be open. For a (Fréchet) differentiable function $G : U \to \mathbb{R}$, we define

$$\operatorname*{arg\,stat}_{u\in U} G(u) \doteq \{\bar{u} \in U : \tfrac{dG}{du}(\bar{u}) = 0\}.$$

If $\{G(\bar{u}) : \bar{u} \in \arg\operatorname{stat}_{u\in U} G(u)\}$ is a singleton; that is, if there exists some $a \in \mathbb{R}$ such that

$$\{G(\bar{u}) : \bar{u} \in \operatorname*{arg\,stat}_{u\in U} G(u)\} = \{a\}.$$

Then $\operatorname{stat}_{u\in U} G(u)$ is defined to be a. Otherwise, $\operatorname{stat}_{u\in U} G(u)$ is undefined.

If G is continuously differentiable and convex in u and has a minimum, then $\operatorname{stat}_{u\in U} G(u)$ exists and is its minimum.

Assuming $\operatorname{stat}_{u\in L^2((t,T);\mathbb{R}^j)} J(u;t,x)$ exists (i.e. is single-valued) for all $t, x \in (0,T)\times\mathbb{R}^n$, we consider the value function

$$\tilde{W}(t,x) \doteq \operatorname*{stat}_{u\in L^2((t,T);\mathbb{R}^j)} J(u;t,x).$$

The HJB equation corresponding to the staticization problem above is given by

$$0 = -\tfrac{\partial}{\partial s}W(s,x) - \tilde{H}(x, \nabla W(s,x)) \tag{3}$$

with terminal condition $W(T,\cdot) = \psi$, where the Hamiltonian $\tilde{H}$ is given by

$$\tilde{H}(x,p) \doteq \operatorname*{stat}_{v\in\mathbb{R}^j} \{\tfrac{1}{2}x'Cx + \tfrac{1}{2}v'Dv + \ell(Mx) + p'[Ax + Lf(Mx) + Bv]\}.$$

We see that the stat is attained at $v = -D^{-1}B'p$, and thus

$$\tilde{H}(x,p) = \tfrac{1}{2}x'Cx + p'Ax - \tfrac{1}{2}p'BD^{-1}B'p + \underbrace{\ell(Mx) + (L'p)'f(Mx)}_{\doteq \mathcal{N}(Mx, L'p)}. \tag{4}$$

It was shown in [18, §2.3, §3.3] that the value function $\tilde{W}$ is the viscosity solution to (3) if H and ψ are convex. A similar verification theorem for more general staticization problems can be found in [19].

B. Stat-quad duality

We group the non-quadratic terms in (4) as $\mathcal{N}$. The following assumption is made in order to take the stat-quad dual of $\mathcal{N}(y_0, y_1) = \ell(y_0) + (y_1)'f(y_0)$.

Let $\hat{C} \doteq -\begin{bmatrix} c_1 I_k & 0_{k\times l} \\ 0_{l\times k} & c_2 I_l \end{bmatrix}$. For any $a \in \mathbb{R}^k, b \in \mathbb{R}^l$, define

$$Q_{\hat{C}}(a,b) \doteq \frac{1}{2}\begin{bmatrix} a \\ b \end{bmatrix}' \hat{C} \begin{bmatrix} a \\ b \end{bmatrix}.$$

Assumption 2: $c_1, c_2 \in \mathbb{R}\setminus\{0\}$ are such that

$$\mathcal{N}(y_0,y_1) = \operatorname*{stat}_{(a,b)\in\mathbb{R}^{k+l}} \left\{\breve{\mathcal{N}}(a,b) + Q_{\hat{C}}(y_0,y_1,a,b)\right\}, \tag{5}$$

where

$$\breve{\mathcal{N}}(a,b) \doteq \operatorname*{stat}_{(y_0,y_1)\in\mathbb{R}^{k+l}} \left\{\mathcal{N}(y_0,y_1) - Q_{\hat{C}}(y_0,y_1,a,b)\right\}.$$

That is, $\mathcal{N}$ and $\breve{\mathcal{N}}$ are stat-quad dual of each other.

Remark 1: The convexity of $\mathcal{N}$ is less critical here compared to the Legendre-Fenchel dual, because stat is taken instead of inf or sup. In this regard, stat-quad dual is more akin to semiconvex dual, since a quadratic function is added to $\mathcal{N}$ to aid the duality.

If $\mathcal{N}$ has uniformly bounded second derivative, taking $c_1, c_2 > 2\sup_{y_0,y_1}|\nabla^2\mathcal{N}(y_0,y_1)|$ is sufficient for Assumption 2. However, since the y_1 argument represents $L'\nabla\tilde{W}$, this usually does not hold a priori. Instead, one may use a local bound of $L'\nabla\tilde{W}$ on the region of interest to restrict the range of y_1.

Remark 2: $\hat{C}$ does not need to be diagonal in general and can be chosen to facilitate computation. The reader may refer to [7], [20] for discussions of properties of stat-quad dual.

We now rewrite (3) using stat-quad duality. For the sake of space, we omit the arguments for W, and W shall implicitly refer to $W(s,x)$ in the rest of this section. Using (5) in (4) and expanding $Q_{\hat{C}}$, we may rewrite $\tilde{H}$ as:

$$\begin{aligned} H(x,p) &= \tfrac{1}{2}x'Cx + p'Ax - \tfrac{1}{2}p'BD^{-1}B'p \\ &\quad + \operatorname*{stat}_{(a,b)\in\mathbb{R}^{k+l}}[\breve{\mathcal{N}}(a,b) + Q_{\hat{C}}(Mx-a, L'p-b)] \\ &= \tfrac{1}{2}x'Cx + p'Ax - \tfrac{1}{2}p'\Gamma p + \\ &\operatorname*{stat}_{(a,b)\in\mathbb{R}^{k+l}}[\breve{\mathcal{N}}(a,b) - \frac{c_1}{2}|Mx-a|^2 - \frac{c_2}{2}|b|^2 + c_2 p'Lb], \end{aligned} \tag{6}$$

where $\Gamma \doteq BD^{-1}B' + c_2LL'$. We note that Γ is symmetric. By considering its singular value decomposition, there exist matrices $Q \in \mathbb{R}^{n\times m}$ and $\Lambda \in \mathbb{R}^{m\times m}$ such that $\Gamma = Q\Lambda Q'$ and Λ is invertible. Observe then

$$-\tfrac{1}{2}p'\Gamma p = \operatorname*{stat}_{v\in\mathbb{R}^m}[p'Qv + \tfrac{1}{2}v'\Lambda^{-1}v] \quad \forall\, p\in\mathbb{R}^n,$$

and therefore,

$$\begin{aligned} &H(x,p) \\ &= \tfrac{1}{2}x'Cx + p'Ax + \operatorname*{stat}_{v\in\mathbb{R}^m}[p'Qv + \tfrac{1}{2}v'\Lambda^{-1}v] \\ &\quad + \operatorname*{stat}_{(a,b)\in\mathbb{R}^{k+l}}[\breve{\mathcal{N}}(a,b) - \frac{c_1}{2}|Mx-a|^2 - \frac{c_2}{2}|b|^2 + c_2p'Lb], \\ &= \tfrac{1}{2}x'Cx + p'Ax + \operatorname*{stat}_{(v,a,b)\in\mathbb{R}^{m+k+l}}\{p'Qv + \tfrac{1}{2}v'\Lambda^{-1}v + \\ &\qquad \breve{\mathcal{N}}(a,b) - \frac{c_1}{2}|Mx-a|^2 - \frac{c_2}{2}|b|^2 + c_2p'Lb\}. \end{aligned} \tag{7}$$

The stat over v and a,b are independent of each other, and are combined to form a Hamiltonian of a new staticizing control problem. Substituting H in (7) into (3), we obtain

$$\begin{aligned} 0 = -\{&\tfrac{\partial W}{\partial s} + \tfrac{1}{2}x'Cx + (\nabla W)'Ax \\ &+ \operatorname*{stat}_{(v,a,b)\in\mathbb{R}^{m+k+l}}\{(\nabla W)'Qv + \tfrac{1}{2}v'\Lambda^{-1}v + c_2(\nabla W)'Lb \\ &\qquad + \breve{\mathcal{N}}(a,b) - \frac{c_1}{2}|Mx-a|^2 - \frac{c_2}{2}|b|^2\}\}. \end{aligned} \tag{8}$$

Observe that (8), along with the original boundary condition $W(T,\cdot)=\psi$, is the attendant HJB equation for an optimal control problem with *linear time-invariant* dynamics

$$\dot{\zeta}(s) = A\zeta(s) + Q\nu(s) + c_2L\beta(s), \quad \zeta(t) = x \in \mathbb{R}^n, \tag{9}$$

and cost function $\breve{J}$ given by

$$\begin{aligned} &\breve{J}(\nu,\alpha,\beta;t,x) \doteq \\ &\int_t^T \frac{1}{2}\zeta(s)'C\zeta(s) + \frac{1}{2}\nu(s)'\Lambda^{-1}\nu(s) + \breve{\mathcal{N}}(\alpha(s),\beta(s)) \\ &\qquad - \frac{c_1}{2}|M\zeta(s)-\alpha(s)|^2 - \frac{c_2}{2}|\beta(s)|^2\,ds + \psi(\zeta(T)). \end{aligned} \tag{10}$$

We again consider the value function

$$\breve{W}(t,x) \doteq \operatorname*{stat}_{(\nu,\alpha,\beta)\in\mathcal{V}\times\mathcal{A}\times\mathcal{B}} \breve{J}(\nu,\alpha,\beta;t,x), \tag{11}$$

where $\mathcal{V} \doteq L^2((t,T);\mathbb{R}^m)$, $\mathcal{A} \doteq L^2((t,T);\mathbb{R}^k)$, and $\mathcal{B} \doteq L^2((t,T);\mathbb{R}^l)$.

We note that (3) and (8) are equivalent. By uniqueness of viscosity solution to (3), one concludes that $\breve{W} = \tilde{W}$.

C. An equivalent ODE formulation for $\arg\operatorname{stat}$

Let $t\in[0,T], x\in\mathbb{R}^n$ be given. The staticization problem (9)–(11) can be viewed as a problem of finding stationary points of $\breve{J}(\cdot;t,x)$, subject to an ODE constraint (9).

Introducing Lagrange multipliers to the problem (9)–(11) converts it into an equivalent unconstrained staticization problem. Let $W^{1,2} = W^{1,2}((t,T);\mathbb{R}^n)$ denote the respective Sobolev space. Define $G: W^{1,2}\times\mathcal{V}\times\mathcal{A}\times\mathcal{B} \to (W^{1,2})^*$

$$\begin{aligned} G(\xi,\nu,\alpha,\beta) : h \mapsto &(x-\xi(t))'h(t) + \\ &\int_t^T (A\xi(s) + Q\nu(s) + c_2L\beta(s) - \dot{\xi}(s))'h(s)\,ds. \end{aligned}$$

We use ξ (instead of ζ) to denote the state process in the unconstrained problem, to emphasize that ξ is an arbitrary state process a priori decoupled from ν,α,β. G now carries information about the state dynamics, in the sense of the following lemma.

Lemma 1: $G(\xi,\nu,\alpha,\beta)=0$ iff (9) holds (in ξ).

We now introduce the multiplier (costate) λ and define $\mathcal{J}: W^{1,2}\times\mathcal{V}\times\mathcal{A}\times\mathcal{B}\times W^{1,2}\to\mathbb{R}$ as

$$\begin{aligned} \mathcal{J}(\xi,\nu,\alpha,\beta,\lambda) \doteq\ & G(\xi,\nu,\alpha,\beta)[\lambda] + \\ &\int_t^T \frac{1}{2}\xi(s)'C\xi(s) + \frac{1}{2}\nu(s)'\Lambda^{-1}\nu(s) + \breve{\mathcal{N}}(\alpha(s),\beta(s)) \\ &\quad - \frac{c_1}{2}|M\xi(s)-\alpha(s)|^2 - \frac{c_2}{2}|\beta(s)|^2\,ds + \psi(\xi(T)). \end{aligned}$$

Each critical point $(\xi,\nu,\alpha,\beta,\lambda)$ of the Lagrangian $\mathcal{J}$ corresponds a constrained critical point (ν,α,β) of $\breve{J}$ [21, Prop. 43.21].

For $y,z\in\mathbb{R}^n, a\in\mathbb{R}^k, b\in\mathbb{R}^l, v\in\mathbb{R}^m$, define

$$\begin{aligned} H(z,v,a,b,y) \doteq\ & \tfrac{1}{2}z'Cz + \tfrac{1}{2}v'\Lambda^{-1}v + \breve{\mathcal{N}}(a,b) \\ &- \frac{c_1}{2}|Mz-a|^2 - \frac{c_2}{2}|b|^2 + y'(Az+Qv+c_2Lb). \end{aligned}$$

Then

$$\begin{aligned} \mathcal{J}(\xi,\nu,\alpha,\beta,\lambda) &= \int_t^T H(\xi(s),\nu(s),\alpha(s),\beta(s),\lambda(s))\,ds \\ &\quad + \psi(\xi(T)) - \int_t^T \lambda(s)'\dot{\xi}(s)\,ds + \lambda(t)'(x-\zeta(t)) \\ \text{(by parts)} \qquad &= \int_t^T H(\xi(s),\nu(s),\alpha(s),\beta(s),\lambda(s))\,ds \\ &\quad + \int_t^T \dot{\lambda}(s)'\xi(s)\,ds + \psi(\xi(T)) + \lambda(t)'x - \lambda(T)'\xi(T). \end{aligned}$$

Considering the Fréchet derivative of $\mathcal{J}$ with respect to $\xi,\lambda\in W^{1,2}$ and $\nu,\alpha,\beta\in\mathcal{V}\times\mathcal{A}\times\mathcal{B}$, we conclude that ν^*,α^*,β^* must satisfy

$$\begin{aligned} &\frac{\partial H}{\partial z}(\xi(s),\nu(s),\alpha(s),\beta(s),\lambda(s)) + \dot{\lambda}_s = 0, \\ &\frac{\partial H}{\partial \lambda}(\xi(s),\nu(s),\alpha(s),\beta(s),\lambda(s)) - \dot{\xi}_s = 0, \\ &(\nu(s),\alpha(s),\beta(s)) = \operatorname*{arg\,stat}_{(v,a,b)\in\mathbb{R}^{m+k+l}} H(\xi(s),v,a,b,\lambda(s)), \end{aligned} \tag{12}$$

for almost every $s \in [t,T]$, with boundary conditions $\lambda(T) = \nabla\psi(\xi(T))$ and $\xi(t) = x$. These equations are comparable to those obtained by Pontryagin's principle.

We see given each (z,y), in finding

$$(v^*,a^*,b^*) \doteq \underset{(v,a,b)\in\mathbb{R}^{m+k+l}}{\arg\operatorname{stat}} H(z,v,a,b,y),$$

v can be staticized independently from a,b. In particular,

$$v^*(z,y) = -\Lambda Q'y \text{ and } \begin{bmatrix} a^*(z,y) \\ b^*(z,y)\end{bmatrix} = \eta^{-1}\left(\begin{bmatrix} Mz \\ L'y\end{bmatrix}\right) \quad (13)$$

where $\eta(a,b) \doteq (a,b) + \hat{C}^{-1}\nabla\breve{\mathcal{N}}(a,b)$.

For the present problem, (12) reduces to

$$\begin{aligned} -\dot{\lambda}(s)' &= \zeta(s)'C - c_1(M\zeta(s) - a^*(s))'M + \lambda(s)'a^*(s) \\ \dot{\zeta}(s) &= A\zeta(s) + Qv^*(s) + c_2Lb^*(s), \end{aligned}$$

or more compactly:

$$\begin{aligned} \begin{bmatrix}\dot{\zeta}(s) \\ \dot{\lambda}(s)\end{bmatrix} &= \begin{bmatrix} A & -\Gamma \\ c_1M'M - C & -A'\end{bmatrix}\begin{bmatrix}\zeta(s)\\ \lambda(s)\end{bmatrix} \\ &\quad + \begin{bmatrix} c_2Lb^*(\zeta(s),\lambda(s)) \\ -c_1M'a^*(\zeta(s),\lambda(s))\end{bmatrix} \\ &\doteq \bar{A}\begin{bmatrix}\zeta(s)\\ \lambda(s)\end{bmatrix} + \bar{B}\begin{bmatrix} a^*(\zeta(s),\lambda(s)) \\ b^*(\zeta(s),\lambda(s))\end{bmatrix}, \end{aligned} \quad (14)$$

with boundary condition $\lambda(T) = \nabla\psi(\zeta(T))$ and $\zeta(t) = x$.

By [21, Prop. 43.21], we have

Theorem 2: $\operatorname{stat}_{(\nu,\alpha,\beta)\in\mathcal{V}\times\mathcal{A}\times\mathcal{B}} \breve{J}(\nu,\alpha,\beta;t,x)$ is single-valued if and only if there exists a unique solution (ζ,λ) to the TPBVP (14). In that case, for a.e. $s \in (t,T)$,

$$\underset{(\nu,\alpha,\beta)\in\mathcal{V}\times\mathcal{A}\times\mathcal{B}}{\arg\operatorname{stat}} \breve{J}(\nu,\alpha,\beta;t,x) = \underset{(v,a,b)\in\mathbb{R}^{m+k+l}}{\arg\operatorname{stat}} H(\zeta(s),v,a,b,\lambda(s)).$$

III. Numerical Method

We proceed to develop the numerical method for the staticization problem (9)–(11). We note that the solution (ζ^*,λ^*) to the TPBVP (14) with boundary condition $\lambda(T) = \nabla\psi(\zeta(T))$, $\zeta(t) = x$ also solves (14) with boundary condition $\zeta(t) = x$ and $\lambda(\tau) = \nabla\psi(\zeta^*(\tau))$ for any $\tau \in [t,T]$. The ODE (14) can be solved on $[t,T]$ regardless of τ. This observation is similar to the dynamic programming principle, which allows us to partition problems with long time horizons into shorter sub-problems, so as to solve longer time horizon problems efficiently.

A. Short Duration Fixed-Point Iteration

We begin by treating the case where the time duration $T-t$ is "short". For notational simplicity, we suppress the time of evaluation in ODEs where unambiguous.

The ζ and λ processes in (14) can be partially decoupled to exploit the low dimensionality of this new problem and simplify computation. Let R,S be the solution to the matrix ODE

$$\begin{bmatrix}\dot{R}\\ \dot{S}\end{bmatrix} = \bar{A}\begin{bmatrix}R\\ S\end{bmatrix} \quad (15)$$

with boundary condition $R(t) = 0_{n\times n}$, $S(T) = I_n$.

We make the following assumption on the time horizon, which will later be bypassed by employing the dynamic programming principle.

Assumption 3: $T-t$ is sufficiently small so that $S(s)$ is invertible for all $s \in [t,T]$.

In the following discussion, we assume R,S have been found and aim to evaluate $\breve{W}$ at some given (t,x).

For brevity, we denote $\mu \doteq [\alpha',\beta']'$. Consider the "open-loop" system

$$\begin{bmatrix}\dot{\zeta}\\ \dot{\lambda}\end{bmatrix} = \bar{A}\begin{bmatrix}\zeta\\ \lambda\end{bmatrix} + \bar{B}\mu. \quad (16)$$

Instead of treating ζ,λ directly, we first apply a linear time-varying change of variable. For each $s \in [t,T]$, let

$$\begin{aligned} \begin{bmatrix}p(s)\\ q(s)\end{bmatrix} &\doteq \begin{bmatrix} I_n & -R(s)S(s)^{-1} \\ 0_{n\times n} & S(s)^{-1}\end{bmatrix}\begin{bmatrix}\zeta(s)\\ \lambda(s)\end{bmatrix} \\ &= \begin{bmatrix} I_n & R(s) \\ 0_{n\times n} & S(s)\end{bmatrix}^{-1}\begin{bmatrix}\zeta(s)\\ \lambda(s)\end{bmatrix}. \end{aligned}$$

Then

$$\begin{aligned} \begin{bmatrix}\dot{p}\\ \dot{q}\end{bmatrix} &= \begin{bmatrix} I_n & R\\ 0 & S\end{bmatrix}^{-1}\begin{bmatrix}\dot{\zeta}\\ \dot{\lambda}\end{bmatrix} + \left(\frac{d}{ds}\begin{bmatrix} I_n & R\\ 0 & S\end{bmatrix}^{-1}\right)\begin{bmatrix}\zeta\\ \lambda\end{bmatrix}, \\ &= \begin{bmatrix} I_n & R\\ 0 & S\end{bmatrix}^{-1}\left(\bar{A}\begin{bmatrix} I_n & R\\ 0 & S\end{bmatrix}\begin{bmatrix}p\\ q\end{bmatrix} + \bar{B}\mu\right) \\ &\quad - \begin{bmatrix} I_n & R\\ 0 & S\end{bmatrix}^{-1}\begin{bmatrix} 0_{n\times n} & \dot{R}\\ 0_{n\times n} & \dot{S}\end{bmatrix}\begin{bmatrix}p\\ q\end{bmatrix} \\ &= \begin{bmatrix} I_n & -RS^{-1}\\ 0 & S^{-1}\end{bmatrix}\left(\begin{bmatrix} A\\ c_1M'M - C\end{bmatrix}p + \bar{B}\mu\right) \\ &= \begin{bmatrix} A + RS^{-1}(C - c_1M'M)\\ -S^{-1}(C - c_1M'M)\end{bmatrix}p \\ &\quad + \begin{bmatrix} c_1RS^{-1}M' & c_2L\\ -c_1S^{-1}M' & 0_{n\times n}\end{bmatrix}\mu \end{aligned} \quad (17)$$

with boundary condition $p(t) = x$ and $q(T) = \nabla\psi(p(T) + R(T)q(T))$.

Note that the dynamics of (p,q) does not involve q. Given any $\mu \in L^2((t,T);\mathbb{R}^{k+l})$, the solution to (17) is given by

$$\begin{aligned} p^\mu(s) &\doteq \bar{\Phi}_{s,t}x + \\ &\quad \int_t^s \bar{\Phi}_{s,\tau}(c_1R(\tau)S(\tau)^{-1}M'\alpha(\tau) + c_2L\beta(\tau))\,d\tau \end{aligned} \quad (18)$$

$$\begin{aligned} q^\mu(s) &\doteq \nabla\psi(p(T) + R(T)q(T)) + \\ &\quad \int_s^T S(\tau)^{-1}(C - c_1M'M)p^\mu(\tau) + c_1S(\tau)^{-1}M'\alpha(\tau)\,d\tau, \end{aligned} \quad (19)$$

where $\bar{\Phi}_{s,\tau}$ is the state transition matrix associated with the linear ODE

$$\frac{\partial}{\partial s}\bar{\Phi}_{s,\tau} = (A + RS^{-1}(C - c_1M'M))\bar{\Phi}_{s,\tau}.$$

Define $\Upsilon : L^2((t,T);\mathbb{R}^{k+l}) \to C([t,T];\mathbb{R}^{2n})$,

$$\Upsilon : \mu \mapsto (p^\mu, q^\mu).$$

Every μ process generates a (p,q) process via Υ, which in turn generates a new μ process by closing the loop using (13), as given by

$$\mu^*(p,q) = \eta^{-1}\left(\begin{bmatrix} M & MR\\ 0_{l\times n} & L'S\end{bmatrix}\begin{bmatrix}p\\ q\end{bmatrix}\right). \quad (20)$$

The solution to the closed-loop system (14) will be found by fixed-point iteration.

The next few results show that each of the maps involved is Lipschitz and estimate their Lipschitz constant.

Lemma 3: $\eta^{-1}(y) = y - \hat{C}^{-1}\nabla\mathcal{N}(y)$. The inverse of η is C^1 with Lipschitz constant 2. In particular, $(\nabla\eta(\mu))^{-1}$ is bounded by 2 and Lipschitz in μ with Lipschitz constant 4. Similar to [13], the proof uses properties of stat-quad dual, which we omit here.

The following result gives a Lipschitz bound on the terminal condition.

Lemma 4: There exists some $\delta \in \mathbb{R}^+$ such that whenever $T - t < \delta$, there exists a unique continuously differentiable function $g : \mathbb{R}^n \to \mathbb{R}^n$ such that

$$g(p) = \nabla\psi(p + R(T)g(p)) \quad \forall p \in \mathbb{R}^n. \tag{21}$$

In particular, δ can be chosen such that g is Lipschitz with Lipschitz constant $K_g \doteq 2\sup_{x\in\mathbb{R}^n} |\nabla^2\psi(x)|$.

Proof: Recall that $R(t) = 0_{n\times n}$ and $\nabla^2\psi$ is bounded. By continuity of $R(\cdot), S(\cdot)$, there exists some $\delta > 0$ such that $2|R(\tau)| < (\sup_{x\in\mathbb{R}^n} |\nabla^2\psi(x)|)^{-1}$ for all $\tau \in (t, t+\delta)$. In particular, whenever $T - t < \delta$, we have $|\nabla^2\psi(x)R(T)| < \frac{1}{2}$ for all $x \in \mathbb{R}^n$.

We claim that (21) defines a contractive fixed-point iteration. Indeed,

$$\left|\frac{d}{dq}\nabla_x\psi(p + R(T)q)\right| = |[\nabla_x^2\psi(p + R(T)q)]R(T)| < \frac{1}{2}.$$

By Banach fixed-point theorem, there exists a unique function $g : \mathbb{R}^n \to \mathbb{R}^n$ such that (21) holds.

Denote $F(p, q) \doteq q - \nabla\psi(p + R(T)q)$. By construction of g, $F(p, g(p)) = 0$. Applying the implicit function theorem on F yields that $g(p)$ is differentiable and

$$(I_n - \nabla^2\psi(p+R(T)g(p))R(T))\nabla g(p) = \nabla^2\psi(p+R(T)g(p)).$$

Therefore,

$$\begin{aligned}&|\nabla g(p)|\\ &\le |(I_n - \nabla^2\psi(p + R(T)g(p))R(T))^{-1}||\nabla^2\psi(p + R_T g(p))|\\ &< 2\sup_{x\in\mathbb{R}^n} |\nabla^2\psi(x)|.\end{aligned}$$

■

Lemma 5: Υ is Lipschitz with Lipschitz constant $\mathcal{O}(\sqrt{T-t})$. The Lipschitz constant also depends on $|\hat{C}|$, $|M|$, $|L|$, $\sup_{s,\tau\in[t,T]} |\Phi_{s,t}|$, K_g, $\sup_{s\in[t,T]} |S(\tau)^{-1}|$, $\sup_{s\in[t,T]} |R(\tau)|$.

Finally, we have the existence and uniqueness result.

Theorem 6: There exists a unique control pair $(\alpha^*, \beta^*) \in \mathcal{A} \times \mathcal{B}$ such that (16) and (20) hold, provided $T - t$ is sufficiently small.

Proof: Suppose $T - t$ is sufficiently small so that Υ is Lipschitz with Lipschitz constant less than $\frac{1}{2}$. Then the composition $\mu^* \circ \Upsilon$ defines a contractive fixed-point iteration on $L^2((t, T); \mathbb{R}^{k+l})$. By Banach fixed-point theorem, it admits a unique fixed-point $\mu^* = [(\alpha^*)', (\beta^*)']'$. ■

This shows that if $\nabla^2\psi$ is uniformly bounded and $T - t$ is sufficiently small, the staticizing control problem (9)–(11) can be solved on $[t, T]$ using fixed-point iterations and the convergence is guaranteed.

We note that one may find the value $\tilde{W}(t, x)$ using (10), (2), or rewrite $\breve{\mathcal{N}}$ using the stat-quad duality in Assumption 2 to obtain an expression involving only $\mathcal{N}, p, q$.

B. Extension to Longer Durations

For longer durations, we partition the interval $[t, T]$ as follows. Let $t_k \in [t, T]$ be a strictly increasing sequence with $t_0 = t$ and $t_N = T$. On each sub-interval $[t_k, t_{k+1}]$, the optimal control problem can be solved using fixed-point iteration as shown in Lemma 5, whereas each sub-intervals are stitched together by the following algorithm, where the terminal gradient is propagated towards t_0. The convergence of the iteration below is based on the dynamic programming principle and requires that the second derivative of the value function $\breve{W}$ is bounded on the domain concerned and over $[t, T]$. This tacitly assumes that $\breve{W}$ is C^2 in space for all $(0, T]$.

function SOLVE_P($(t_k)_{k=0}^N$, x, μ)
  Returns $(p^\mu(t_k))_{k=0}^N$ as given by (18).
end function
function Q_INT($(t_k)_{k=0}^N$, μ)
  Returns $(q^\mu(t_k) - \nabla\psi(p(T) + R(T)q(T)))_{k=0}^N$.
end function
function VALUE(t, x, $(t_k)_{k=0}^N$, n, g, ϵ)
  $\mu \leftarrow$ Initial guess for μ
  $P \leftarrow$ SOLVE_P($(t_k)_{k=0}^N, x, \mu$)
  $Q \leftarrow$ empty array of length $N + 1$
  $k \leftarrow N$
  while $k > 0$ **do**
    $k \leftarrow k - 1$
    $\tau \leftarrow$ LINSPACE(t_k, t_{k+1}, n)
    Find or recall $(R(\tau_k))_{k=0}^n, (S(\tau_k))_{k=0}^n$
    repeat
      $p \leftarrow$ SOLVE_P(τ, P_k, μ)
      if $k + 1 = N$ **then**
        $Q_{k+1} \leftarrow g(p_n)$
      else if $|p_N + R(\tau_n)Q_{k+1} - P_{k+1}| > \epsilon$ **then**
        $P_{k+1} \leftarrow p_N + R(\tau_n)Q_{k+1}$
        $k \leftarrow k + 2$
        break
      end if
      $q \leftarrow$ Q_INT(τ, μ) $+ Q_k$.
      $\mu|_{[t_k,t_{k+1})} \leftarrow \eta^{-1}(M(p + Rq), L'Sq)$
      $Q_k \leftarrow S_{\tau_0} q_0$
    until $\mu|_{[t_k,t_{k+1})}$ converges
  end while
  return $\breve{W}(t_0, x)$
end function

Remark 3: We note that R, S are solutions of a time-independent ODE (15).

The algorithm discretized all sub-intervals using n points; that is, τ has length n for each k. This is not necessary, but this allows us to exploit the time-independence of (15), since R and S can be reused rather than re-recomputed for each sub-interval.

The inner loop that finds $\mu|_{[t_k,t_{k+1})}$ via fixed-point iteration is similar to the Forward-Backward-Sweep method, whose convergence is usually limited by the time horizon and the problem structure [22]. The outer loop allows us to extend its convergence to longer time horizons.

The relation between outer and inner loop can also be thought of as a mutual recursion, where the inner loop solves the optimal control problem on a short subinterval, but need to solve it with the correct terminal gradient $\lambda(t_{k+1}) = \nabla\psi(\xi(t_{k+1}))$. This is delegated to the outer loop, which retrieves that information from the next subinterval, that is, the outer loop then runs the inner loop on the next subinterval and returns the correct gradient. The process repeats until μ converges on the first subinterval.

IV. Numerical Examples

Two numerical experiments are run on an Intel Core i9-11900 processor. The algorithm is implemented in Python and C++.

A. Motor control

A variant of the motor control problem in [1, §2.2] is solved to test the accuracy of the algorithm.

Consider the state process (1) where

$$A \doteq \begin{bmatrix} 0 & 1 & 0 & 0 \\ -20 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}, \qquad B \doteq \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

and $L \doteq [0,-1,0,0]'$, $M \doteq [1,0,0,0]'$, $f(x) \doteq 20(\sin(x) - x)$. The cost function is given by

$$J(u;t,x) \doteq \frac{1}{4\pi}\int_t^{2\pi} u(s)^2 + (\xi_1(s) - \xi_3(s))^2 + (\xi_2(s) - \xi_4(s))^2 \, ds.$$

The dualizing coefficients are taken to be $c_1 = c_2 = -2$.

The time interval is divided into $N = 22$ parts, each taking $n = 50$ steps. The value function was evaluated at 51×51 points on a 2-D subspace of the state space given by

$$\{(x_1, x_2, x_1, x_2)' : x_1, x_2 \in \mathbb{R}\}$$

with target accuracy 10^{-4} (in terms of the boundary condition of (14)). The numerical results are plotted in Fig. 1.

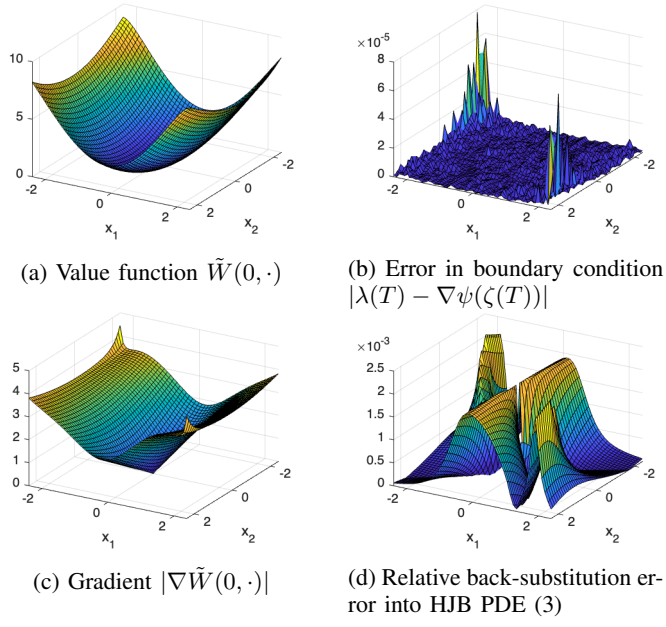
The table below compares the proposed algorithm with the generic `solve_bvp` function from SciPy. The two columns list the average time to evaluate the value function at a point, and the number of points at which the solver converged.

Solver	Time per eval.	# converged
Stat-quad dual	142ms	2601
`solve_bvp`	122ms	2149

B. Diffusion equation with nonlocal spatial term

High dimensional control problems similar to the one considered in [23] is solved to test the scalability of the algorithm. We consider a square domain discretized into a 5×5 uniform grid as shown in Fig. 2.

Applying finite difference method on the grid yields a 25-dimensional control problem as follows.



(a) Value function $\tilde{W}(0,\cdot)$

(b) Error in boundary condition $|\lambda(T) - \nabla\psi(\zeta(T))|$

(c) Gradient $|\nabla\tilde{W}(0,\cdot)|$

(d) Relative back-substitution error into HJB PDE (3)

Fig. 1: Results for Example A.

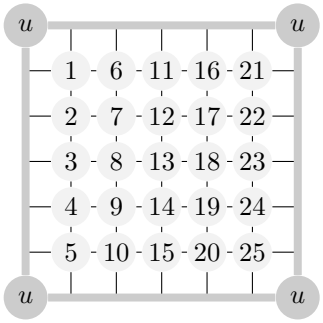


Fig. 2: Grid configuration: numbers are the index for each node.

Let A be the discrete Laplace operator and let $B \in \mathbb{R}^{25\times 1}$ be given by

$$B_i = \begin{cases} 2 & i = 1,5,21,25 \\ 1 & i = 2,3,4,6,10,11,15,16,20,22,23,24 \\ 0 & \text{otherwise} \end{cases};$$

then $A\xi + Bu$ is exactly the discrete Laplacian over the grid. Let $\mathbb{1}_{25\times 1}$ denote the 25×1 vector of ones. We consider the state process given by

$$\dot{\xi} = A\xi + Bu + \frac{\mathbb{1}_{25\times 1}}{5} f\left(\frac{1}{\sqrt{2\pi}}\sum_{i=1}^{25} e^{-\frac{1}{2}d(i,13)^2}\xi_s^i\right), \xi(0) = x,$$

where $f(x) \doteq \ln(1 + e^x)$ and $d(i,j)$ denotes the Euclidean distance between node i and j.

The cost function is given by

$$J(u;t,x) \doteq \int_t^1 \frac{1}{2}u(s)^2 - \frac{1}{5}\xi(s)'(G_1'G_1 + G_2'G_2)\xi(s) + \cos\left(\tfrac{1}{\sqrt{2\pi}}\textstyle\sum_{i=1}^{25} e^{-\frac{1}{2}d(i,13)^2}\xi_i(s)\right) ds,$$

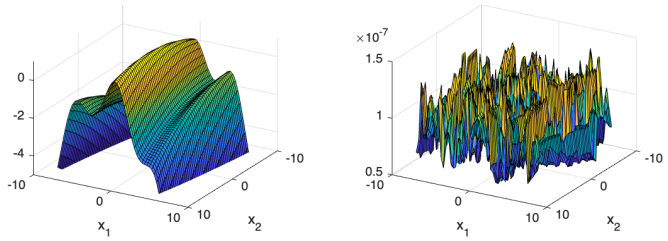
where G_1, G_2 computes the finite difference in the horizontal and vertical direction over the grid.

The time interval $[0, 1]$ was divided equally into $N = 5$ parts. The dualizing coefficients were taken to be $c_1 = 4$ and $c_2 = -3$. The value function was then evaluated at $51 \times 51 = 2601$ points on a 2-D subspace of the state space defined by

$$\{x_1 \mathbb{1}_{25\times 1} + x_2\omega : x_1, x_2 \in \mathbb{R}\},$$

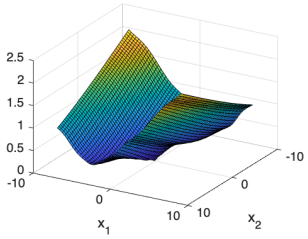
where $\omega \in \mathbb{R}^{25}$ is given by $\omega_i = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}d(i,13)^2}$; this means that initial distributions are Gaussian distributions scaled by x_2 and shifted by x_1.

The computations took about 140 minutes (3.2s per evaluation). The numerical results are plotted in Fig. 3.

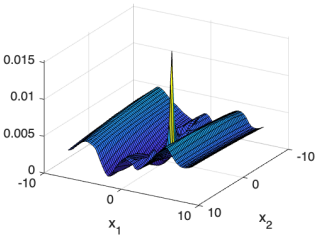


(a) Value function $\tilde{W}(0, \cdot)$

(b) Error in boundary condition $|\lambda(T) - \nabla\psi(\zeta(T))|$



(c) Gradient $|\nabla\tilde{W}(0, \cdot)|$



(d) Relative back-substitution error into HJB PDE (3)

Fig. 3: Results for Example B.

C. Discussions

The parameters used in the examples are somewhat arbitrary. In general, the length of subdivisions should be chosen to ensure a desirable convergence rate in fixed-point iterations. On the other hand, n affects the evaluation of R and S, which in turn affects the accuracy of p^μ, q^μ. The proposed algorithm is slightly slower than the standard TPBVP solver. But it does not employ any derivative-based root finding in shooting, and have demonstrated better convergence properties than the standard BVP solver.

V. Conclusions

A numerical method for solving optimal control problems and computing value function is proposed, which employs a multiple shooting structure to splice Pontryagin's principle and dynamic programming. The method exploits the low dimensionality of nonlinearity in the control problem and performs reasonably well for high dimensional problems.

In the current version of the algorithm, the size of the second derivative of the value function $V(t_{k+1}, \cdot)$ limits the step size $t_{k+1} - t_k$. This is not ideal numerically, but might be worked around by rearranging the fixed-point iteration equation. It would also be interesting to investigate whether this approach can be extended to more general control-affine problems.

References

[1] D. M. Dawson, J. Hu, and T. C. Burg, *Nonlinear Control of Electric Machinery*. CRC Press, 1 2019.

[2] H. J. Pesch, *Solving Optimal Control and Pursuit-Evasion Game Problems of High Complexity*. Birkhäuser Basel, 1994, vol. 115, pp. 43–61.

[3] H. Seywald and E. M. Cliff, "Goddard problem in presence of a dynamic pressure limit," *Journal of Guidance, Control, and Dynamics*, vol. 16, pp. 776–781, 7 1993.

[4] D. Malyuta, Y. Yu, P. Elango, and B. Açıkmeşe, "Advances in trajectory optimization for space vehicle control," *Annual Reviews in Control*, vol. 52, pp. 282–315, 2021.

[5] C. G. Gray and E. F. Taylor, "When action is not least," *American Journal of Physics*, vol. 75, pp. 434–458, 5 2007.

[6] I. Ekeland, "Legendre duality in nonconvex optimization and calculus of variations," *SIAM Journal on Control and Optimization*, vol. 15, pp. 905–934, 11 1977.

[7] W. M. McEneaney and P. M. Dower, "Static duality and a stationary-action application," *Journal of Differential Equations*, vol. 264, pp. 525–549, 1 2018.

[8] P. M. Dower and W. M. McEneaney, "Verification of stationary action trajectories via optimal control," in *2020 American Control Conference (ACC)*. IEEE, 7 2020, pp. 1779–1784.

[9] F. Biral, E. Bertolazzi, and P. Bosetti, "Notes on numerical methods for solving optimal control problems," *IEEJ Journal of Industry Applications*, vol. 5, pp. 154–166, 2016.

[10] A. Bressan, "Noncooperative differential games," *Milan Journal of Mathematics*, vol. 79, pp. 357–427, 12 2011.

[11] E. Cristiani and P. Martinon, "Initialization of the shooting method via the hamilton-jacobi-bellman approach," *Journal of Optimization Theory and Applications*, vol. 146, pp. 321–346, 8 2010.

[12] M. Sassano and A. Astolfi, "Combining pontryagin's principle and dynamic programming for linear and nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 65, pp. 5312–5327, 12 2020.

[13] P. M. Dower, W. M. McEneaney, and Y. Zheng, "Min-max and stat game representations for nonlinear optimal control problems," in *2023 American Control Conference (ACC)*. IEEE, 5 2023, pp. 2733–2738.

[14] W. M. McEneaney, P. M. Dower, and Y. Zheng, "Computational exploitation of low-dimensional nonlinearities in hamilton-jacobi pdes," in *2024 International Symposium on Mathematical Theory of Networks and Systems*. IFAC, 2024.

[15] ——, "Computational reduction for systems with low-dimensional nonlinearities via staticization-based duality," in *2024 Conference on Decision and Control*. IEEE, 2024.

[16] H. Bock and K. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," *IFAC Proceedings Volumes*, vol. 17, pp. 1603–1608, 7 1984.

[17] H. J. Pesch, *Optimal and Nearly Optimal Guidance by Multiple Shooting*. Éditions Cépaduès, 1990, pp. 761–771.

[18] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. Birkhäuser Boston, 1997.

[19] W. M. McEneaney, P. M. Dower, and Y. Zheng, "Stat-quad based representation for dimensional reduction of certain nonlinear optimal control problems," *SIAM Journal on Control and Optimization*, to appear.

[20] W. M. McEneaney, P. M. Dower, and T. Wang, "Second-order hamilton–jacobi pde problems and certain related first-order problems, part 1: Approximation," *SIAM Journal on Control and Optimization*, vol. 61, pp. 3280–3315, 12 2023.

[21] E. Zeidler, *Nonlinear Functional Analysis and its Applications*. Springer New York, 1985, vol. III.

[22] M. McAsey, L. Mou, and W. Han, "Convergence of the forward-backward sweep method in optimal control," *Computational Optimization and Applications*, vol. 53, pp. 207–226, 9 2012.

[23] E. Fernández-Cara, Q. Lü, and E. Zuazua, "Null controllability of linear heat and wave equations with nonlocal spatial terms," *SIAM Journal on Control and Optimization*, vol. 54, pp. 2009–2019, 1 2016.