Radial Basis Function Interpolation and Galerkin Projection for Direct Trajectory Optimization and Costate Estimation

Hossein Mirinejad, *Member, IEEE*, Tamer Inanc, *Senior Member, IEEE*, and Jacek M. Zurada, *Life Fellow, IEEE*

Abstract—This work presents a novel approach combining radial basis function (RBF) interpolation with Galerkin projection to efficiently solve general optimal control problems. The goal is to develop a highly flexible solution to optimal control problems, especially nonsmooth problems involving discontinuities, while accounting for trajectory accuracy and computational efficiency simultaneously. The proposed solution, called the RBF-Galerkin method, offers a highly flexible framework for direct transcription by using any interpolant functions from the broad class of global RBFs and any arbitrary discretization points that do not necessarily need to be on a mesh of points. The RBF-Galerkin costate mapping theorem is developed that describes an exact equivalency between the Karush–Kuhn–Tucker (KKT) conditions of the nonlinear programming problem resulted from the RBF-Galerkin method and the discretized form of the first-order necessary conditions of the optimal control problem, if a set of discrete conditions holds. The efficacy of the proposed method along with the accuracy of the RBF-Galerkin costate mapping theorem is confirmed against an analytical solution for a bang-bang optimal control problem. In addition, the proposed approach is compared against both local and global polynomial methods for a robot motion planning problem to verify its accuracy and computational efficiency.

Index Terms—Costate estimation, direct trajectory optimization, Galerkin projection, numerical optimal control, radial basis function interpolation.

I. INTRODUCTION

D IRECT methods are extensively used for solving optimal control problems, mainly due to their ability to handle path constraints, robustness to initial guess of parameters, and greater radii of convergence compared to indirect methods [1]–[3]. Direct transcription is based on approximating states

and/or controls with a specific function with unknown coefficients and discretizing the optimal control problem with a set of proper points (nodes) to transcribe it into a nonlinear programming (NLP) problem. The resulting NLP can then be efficiently solved by NLP solvers available. Many direct methods are collocation-based approaches using either local or global polynomials depending on the type of function used in the approximation. Runge-Kutta methods [4], [5] and B-spline approaches [6], [7] are examples of local collocation methods that leverage low-degree local polynomials for the approximation of states and controls. The main drawback of these methods is their algebraic convergence rate, so their solution is not usually as accurate as the solution of global polynomial methods [3].

Pseudospectral (PS) methods [8]–[14], on the other hand, use a high-degree global polynomial for the approximation of states and controls and a set of orthogonal nodes associated with the family of the polynomial for the discretization of the optimal control problem. Due to their spectral (exponential) accuracy and ease of implementation, PS methods have been widely used for direct trajectory optimization in recent years [1], [3]. However, their spectral accuracy only holds for sufficiently smooth functions. If the problem formulation or the optimal solution contains discontinuities (nonsmoothness), PS methods will converge poorly even with a high-degree polynomial [2]. Also, the use of a PS method is limited to a specific mesh of points; For instance, the Gauss PS [11] method can only use Legendre-Gauss nodes, or the Legendre PS method [12] is tied with the Legendre-Gauss-Lobatto nodes for the problem discretization. This limitation becomes problematic when the optimal solution has discontinuities requiring denser nodes around them to accurately capture the switching times of the solution, as will be later demonstrated in Section V. Variations of PS methods were proposed in the literature [15]–[18] to overcome this issue, but such modified PS schemes impose new limitations to the mathematical formulation of the problem, are usually sensitive to the initial guess of parameters, and cannot typically find an accurate solution to non-sequential optimal control problems [15], [19]. Thus, a significant gap exists in the literature with respect to a computationally efficient numerical approach that can find high accuracy solution to *nonsmooth* optimal control problems without applying further limitations to the problem and this work intends to address it.

This paper presents a novel optimal control approach that employs global radial basis functions (RBFs) for the approximation of states and controls and arbitrarily selected

points for the problem discretization. The proposed solution, called the RBF-Galerkin, combines the RBF interpolation with Galerkin projection for direct trajectory optimization and costate estimation. Since the global RBFs comprise a broad class of interpolating functions, including Gaussian (GA) RBFs, multiquadrics (MQ), and inverse multiquadrics (IMQ), the proposed method offers a great flexibility in terms of basis functions (interpolants) for parameterizing an optimal control problem. In addition, unlike a PS method tied with specific type of points, the proposed method leverages a completely arbitrary discretization scheme—which do not even need to be on a mesh of points— providing a highly efficient framework for solving nonsmooth optimal control problems such as a bang-bang problem [16], as will be later demonstrated in Section V. It should be noted that there have been recent attempts at leveraging global RBFs as the interpolants in direct transcription methods [20]–[25], but the use of RBFs was limited to specific type of problems (e.g., quadratic problems [20]) and specific discretization points (e.g., Legendre-Gauss-Lobatto points [21]–[25]). In contrast, the proposed method leverages a completely arbitrary discretization scheme and can be used in solving any general optimal control problems. In addition, unlike many direct methods, including previous RBF-based approaches [20]–[25], the RBF-Galerkin method possesses proof of optimality for solving optimal control problems. It will be shown through the *RBF-Galerkin costate mapping theorem* in Section IV that there will be an exact equivalency between the Karush-Kuhn-Tucker (KKT) multipliers of the NLP resulted from the RBF-Galerkin method and the discretized form of the costates of the original optimal control problem, if a set of discrete conditions (closure conditions) holds.

The major contribution of this work is to present a highly flexible numerical solution to general optimal control problems, especially the nonsmooth problems whose formulation or optimal solution involves discontinuities that cannot be accurately estimated by classical optimal control methods. Another contribution is the proof of optimality via *RBF-Galerkin costate mapping theorem* guaranteeing that the solution of the proposed method is equivalent to the solution of the original optimal control problem. To the best of the authors' knowledge, this is the first time that a *highly flexible, computationally efficient, accurate* solution with the *proof of optimality* is presented for general optimal control problems.

The rest of the paper is organized as follows: A general optimal control problem is formulated in Section II. The RBF-Galerkin solution is described in Section III. The costate estimation along with the proof of optimality is presented in Section IV. Numerical examples are provided in Section V, and finally, the conclusions are drawn in Section VI.

II. OPTIMAL CONTROL PROBLEM FORMULATION

The general optimal control problem is defined in Bolza form [3] as to determine the state $x(\tau) \in \mathbb{R}^n$, control $u(\tau) \in \mathbb{R}^m$, initial time t_0, and final time t_f that minimize the cost functional

$$J = \Gamma\big(x(-1), t_0, x(1), t_f\big) + \frac{t_f - t_0}{2} \int_1^{-1} L(x(\tau), u(\tau)) d\tau \quad (1)$$

subject to state dynamics,

$$\dot{x}(\tau) = \frac{t_f - t_0}{2} f(x(\tau), u(\tau)) \quad (2)$$

boundary conditions,

$$\gamma(x(-1), t_0, x(1), t_f) = 0 \in \mathbb{R}^\gamma \quad (3)$$

and mixed state-control path constraints,

$$q(x(\tau), u(\tau)) \le 0 \in \mathbb{R}^q. \quad (4)$$

It is assumed that the optimal solution to this problem exists. Please note that (1)–(4) can be transformed from the time interval $\tau \in [-1, 1]$ to the time interval $t \in [t_0, t_f]$ using an affine transformation

$$t = \frac{(t_f - t_0)}{2} \tau + \frac{t_f + t_0}{2} \quad (5)$$

where t_0 and t_f are the initial and the final optimization time, respectively.

III. RBF-GALERKIN METHOD FOR DIRECT TRAJECTORY OPTIMIZATION

A direct method combining global RBF parameterization, Galerkin projection, and arbitrary discretization is proposed to discretize the optimal control problem of (1)–(4). The discretized problem can be solved with the NLP solvers available.

RBF Definition: RBF is a real-valued function whose value depends on the distance from a fixed point, called center [26]

$$\rho(y, c) = \rho(\|y - c\|) \quad (6)$$

where ρ, y, c, and $\| \|$ denote the RBF, function variable, RBF center, and the Euclidean norm, respectively. Any function satisfying (6) is called an RBF function, which can be either infinitely smooth such as GA, MQ, and IMQ RBFs, or piecewise smooth such as Polyharmonic Splines. Infinitely smooth RBFs are also called global RBFs.

In the proposed method, global RBFs are leveraged as the basis functions for parameterizing the optimal control problem. For brevity and without loss of generality, it is assumed that the same type of RBFs, ρ, and the same number of RBFs, N, are used for the approximation of states $x(\tau)$ and controls $u(\tau)$ as

$$x(\tau) \approx x^R(\tau) = \sum_{i=1}^N \alpha_i \, \rho(\|\tau - \tau_i\|) = \sum_{i=1}^N \alpha_i \, \rho_i(\tau) \quad (7)$$

$$u(\tau) \approx u^R(\tau) = \sum_{i=1}^N \beta_i \, \rho(\|\tau - \tau_i\|) = \sum_{i=1}^N \beta_i \, \rho_i(\tau) \quad (8)$$

where $x^R(\tau)$ and $u^R(\tau)$ denote the RBF approximation of $x(\tau)$ and $u(\tau)$, respectively. Also, $\rho_i(\tau)$ is the RBF and α_i and β_i are RBF weights for $x^R(\tau)$ and $u^R(\tau)$, respectively.

A set of global RBFs $\{\rho_1(\tau), \rho_2(\tau), \ldots, \rho_N(\tau)\}$ forms a space of continuous, linearly independent basis functions. Taking derivative of (7) with respect to τ yields

$$\dot{x}(\tau) \approx \dot{x}^R(\tau) = \sum_{i=1}^N \alpha_i \, \dot{\rho}(\|\tau - \tau_i\|) = \sum_{i=1}^N \alpha_i \, \dot{\rho}_i(\tau). \quad (9)$$

By substituting (9) in (2), the *defect constraints* (residuals) $\psi(\tau)$ are defined as

$$\psi(\tau) = \frac{t_f - t_0}{2} f(\alpha_i, \beta_i, \rho_i(\tau)) - \sum_{i=1}^{N} \alpha_i \dot{\rho}_i(\tau) \quad (10)$$

for $i = 1, \ldots, N$. The Galerkin method [27] is applied to (10) in which the projection of defect constraints $\psi(\tau)$ on the space of global RBFs $\{\rho_1(\tau), \rho_2(\tau), \ldots, \rho_N(\tau)\}$ is set to zero. This will be obtained by setting the projection of defect constraints on each element of the RBF basis set equal to zero, i.e.,

$$\int_{-1}^{1} \psi(\tau) \rho_j(\tau) \, d\tau = 0 \quad \text{for } j = 1, 2, \ldots, N. \quad (11)$$

According to [27], it implies that the defect constraints converge to zero in the mean (in the limit $N \to \infty$). If x^R satisfies the boundary conditions of (3) and ψ converges to zero in the mean, the approximated solution x^R converges to the exact solution x in the mean, i.e.,

$$\lim_{N \to \infty} \|x^R - x\|_2 = 0. \quad (12)$$

In other words, by applying the Galerkin projection, the defect constraints are being minimized in L^2-norm sense. Now, substituting (10) in (11) and approximating the integral of (11) with a proper quadrature yield

$$\sum_{k=1}^{N} w_k \left(\frac{t_f - t_0}{2} f(\alpha_i, \beta_i, \rho_i(\tau_k)) - \sum_{i=1}^{N} \alpha_i \dot{\rho}_i(\tau_k) \right) \rho_j(\tau_k) = 0 \quad (13)$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, N$, where w_k, $k = 1, 2, \ldots, N$, are quadrature weights corresponding to the type of quadrature points used for approximating the integral.

A slack variable function $p(\tau)$ is defined to convert the inequality path constraints of (4) to equality constraints. $p(\tau)$ can be approximated using N global RBFs as

$$p(\tau) \approx p^R(\tau) = \sum_{r=1}^{N} \kappa_r \rho(\|\tau - \tau_r\|) = \sum_{r=1}^{N} \kappa_r \rho_r(\tau) \quad (14)$$

where $p^R(\tau)$ is the RBF approximation of $p(\tau)$ and κ_r denote the RBF weights for the $p^R(\tau)$. The residual of path constraints, R_q, is calculated as

$$R_q = q(\alpha_i, \beta_i, \rho_i(\tau)) + p^R(\tau) \circ p^R(\tau)$$

$$= q(\alpha_i, \beta_i, \rho_i(\tau)) + \sum_{r=1}^{N} \kappa_r \rho_r(\tau) \circ \sum_{l=1}^{N} \kappa_l \rho_l(\tau) \quad (15)$$

for $i = 1, \ldots N$, where $\circ$ is the entry-wise product of two vectors. Similar to (11), a Galerkin projection is applied to the residual R_q to set it orthogonal to every member of the RBF basis set and can be shown in the discretized form as

$$\sum_{k=1}^{N} w_k \left(q(\alpha_i, \beta_i, \rho_i(\tau_k)) + \left(\sum_{r=1}^{N} \kappa_r \rho_r(\tau) \circ \sum_{l=1}^{N} \kappa_l \rho_l(\tau) \right) \right) \rho_j(\tau_k) = 0 \quad (16)$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, N$, where w_k are the same quadrature weights used in (13). By applying the same numerical quadrature to approximate the running cost L in (1), the optimal control problem of (1)–(4) is transcribed into the following NLP problem:

Determine $A = (\alpha_1 \alpha_2 \cdots \alpha_N)^T_{N \times n}$, $B = (\beta_1 \beta_2 \cdots \beta_N)^T_{N \times m}$, $K = (\kappa_1, \kappa_2, \ldots, \kappa_N)^T_{N \times q}$, t_0, and t_f that minimize the cost

$$\overline{J} = \Gamma\left(\alpha_i, \rho_i(-1), \rho_i(1), t_0, t_f\right)$$

$$+ \frac{t_f - t_0}{2} \sum_{k=1}^{N} w_k L(\alpha_i, \beta_i, \rho_i(\tau_k)) \quad (17)$$

subject to:

$$\sum_{k=1}^{N} w_k \left(\frac{t_f - t_0}{2} f(\alpha_i, \beta_i, \rho_i(\tau_k)) - \sum_{i=1}^{N} \alpha_i \dot{\rho}_i(\tau_k) \right) \rho_j(\tau_k) = 0$$

$$\gamma\left(\alpha_i, \rho_i(-1), \rho_i(1), t_0, t_f\right) = 0$$

$$\sum_{k=1}^{N} w_k \left(q(\alpha_i, \beta_i, \rho_i(\tau_k)) + \left(\sum_{r=1}^{N} \kappa_r \rho_r(\tau) \circ \sum_{l=1}^{N} \kappa_l \rho_l(\tau) \right) \right) \rho_j(\tau_k) = 0 \quad (18)$$

for $i = 1, \ldots, N$ and $j = 1, 2, \ldots, N$. The discretization method is called the RBF-Galerkin approach for solving optimal control problems.

The proposed method is flexible in terms of both interpolant functions and discretization points, as it can use any type of global RBFs as the interpolants and any arbitrary-selected points as the discretization points. The arbitrary discretization scheme is based on the fact that the RBF interpolation is always unique for global RBFs, regardless of the type and number of points used in the interpolation [26].

IV. COSTATE ESTIMATION

In this Section, it will be shown that the KKT optimality conditions of the NLP problem of (17) and (18), are exactly equivalent to the discretized form of the first-order necessary conditions of the optimal control problem of (1)–(4), if a set of conditions will be added to the KKT conditions.

A. KKT Optimality Conditions

Lagrangian or augmented cost of the NLP problem is written as

$$\overline{J}_a = \Gamma\left(\alpha_i, \rho_i(-1), \rho_i(1), t_0, t_f\right)$$

$$+ \frac{t_f - t_0}{2} \sum_{k=1}^{N} w_k L(\alpha_i, \beta_i, \rho_i(\tau_k))$$

$$+ \sum_{j=1}^{N} \tilde{\xi}_j^T \sum_{k=1}^{N} w_k \left(\frac{t_f - t_0}{2} f(\alpha_i, \beta_i, \rho_i(\tau_k)) - \sum_{i=1}^{N} \alpha_i \dot{\rho}_i(\tau_k) \right) \rho_j(\tau_k)$$

$$+ \tilde{v}^T \gamma\left(\alpha_i, \rho_i(-1), \rho_i(1), t_0, t_f\right)$$

$$+ \sum_{j=1}^{N} \tilde{\eta}_j^T \sum_{k=1}^{N} w_k \left(q(\alpha_i, \beta_i, \rho_i(\tau_k)) + \left(\sum_{r=1}^{N} \kappa_r \rho_r(\tau) \circ \sum_{l=1}^{N} \kappa_l \rho_l(\tau) \right) \right) \rho_j(\tau_k) \quad (19)$$

for $i = 1,\ldots,N$, where $\tilde{\boldsymbol{\xi}}_j, \tilde{\boldsymbol{\nu}}, \tilde{\boldsymbol{\eta}}_j$ are KKT multipliers associated with the NLP constraints of (18). Differentiating $\bar{J}_a$ with respect to $\boldsymbol{\alpha}_m, \boldsymbol{\beta}_m, \tilde{\boldsymbol{\xi}}_m, \tilde{\boldsymbol{\eta}}_m, \tilde{\boldsymbol{\nu}}, \boldsymbol{\kappa}_m, t_0, t_f$ and setting them equal to zero provide the KKT optimality conditions: To save space and make it easier to follow, shortened notation $\boldsymbol{x}_1^R = \boldsymbol{x}^R(-1)$, $\boldsymbol{x}_N^R = \boldsymbol{x}^R(1)$, $\boldsymbol{u}_1^R = \boldsymbol{u}^R(-1)$, $\boldsymbol{u}_N^R = \boldsymbol{u}^R(1)$, $\boldsymbol{f}_k = \boldsymbol{f}(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \rho_i(\tau_k))$, $\boldsymbol{q}_k = \boldsymbol{q}(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \rho_i(\tau_k))$, and $L_k = L(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \rho_i(\tau_k))$ are used throughout the paper.

$$
\frac{\partial \bar{J}_a}{\partial \boldsymbol{\alpha}_m} = \frac{\partial \Gamma}{\partial \boldsymbol{x}_1^R} \rho_m(-1) + \frac{\partial \Gamma}{\partial \boldsymbol{x}_N^R} \rho_m(1)
$$

$$
+ \frac{t_f - t_0}{2} \sum_{k=1}^{N} w_k \frac{\partial L_k}{\partial \boldsymbol{x}^R} \rho_m(\tau_k)
$$

$$
+ \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \sum_{k=1}^{N} w_k \left(\frac{t_f - t_0}{2} \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{x}^R} \rho_m(\tau_k) - \dot{\rho}_m(\tau_k) \right) \rho_j(\tau_k)
$$

$$
+ \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial \boldsymbol{x}_1^R} \rho_m(-1) + \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial \boldsymbol{x}_N^R} \rho_m(1)
$$

$$
+ \sum_{j=1}^{N} \tilde{\boldsymbol{\eta}}_j^T \sum_{k=1}^{N} w_k \frac{\partial \boldsymbol{q}_k}{\partial \boldsymbol{x}^R} \rho_m(\tau_k) \rho_j(\tau_k). \tag{20}
$$

Lemma 1:

$$
\sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \sum_{k=1}^{N} w_k \dot{\rho}_m(\tau_k) \rho_j(\tau_k) = \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_m(1) \rho_j(1)
$$

$$
- \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_m(-1) \rho_j(-1) - \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \sum_{k=1}^{N} w_k \rho_m(\tau_k) \dot{\rho}_j(\tau_k). \tag{21}
$$

Proof: Using integration by parts, it can be written that

$$
\int_{-1}^{1} \dot{\rho}_m(\tau) \rho_j(\tau) d\tau = \rho_m(1) \rho_j(1) - \rho_m(-1) \rho_j(-1)
$$

$$
- \int_{-1}^{1} \rho_m(\tau) \dot{\rho}_j(\tau) d\tau. \tag{22}
$$

Approximating the integrals of (22) with a numerical quadrature where w_k, $k = 1, 2, \ldots, N$, are the quadrature weights and multiplying both sides of (22) with $\sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T$ complete the proof. ∎

Now, applying *Lemma 1* to (20) and rearranging the equations yield

$$
\frac{\partial \bar{J}_a}{\partial \boldsymbol{\alpha}_m} = \frac{t_f - t_0}{2} \sum_{k=1}^{N} w_k \left\{ \frac{\partial L_k}{\partial \boldsymbol{x}^R} + \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(\tau_k) \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{x}^R} \right.
$$

$$
\left. + \frac{2}{t_f - t_0} \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \dot{\rho}_j(\tau_k) + \frac{2}{t_f - t_0} \sum_{j=1}^{N} \tilde{\boldsymbol{\eta}}_j^T \rho_j(\tau_k) \frac{\partial \boldsymbol{q}_k}{\partial \boldsymbol{x}^R} \right\} \rho_m(\tau_k)
$$

$$
+ \left(\frac{\partial \Gamma}{\partial \boldsymbol{x}_1^R} + \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial \boldsymbol{x}_1^R} + \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(-1) \right) \rho_m(-1)
$$

$$
+ \left(\frac{\partial \Gamma}{\partial \boldsymbol{x}_N^R} + \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial \boldsymbol{x}_N^R} - \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(1) \right) \rho_m(1) = \boldsymbol{0}. \tag{23}
$$

Similarly,

$$
\frac{\partial \bar{J}_a}{\partial \boldsymbol{\beta}_m} = \frac{t_f - t_0}{2} \sum_{k=1}^{N} w_k \left\{ \frac{\partial L_k}{\partial \boldsymbol{u}^R} + \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(\tau_k) \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{u}^R} \right.
$$

$$
\left. + \frac{2}{t_f - t_0} \sum_{j=1}^{N} \tilde{\boldsymbol{\eta}}_j^T \rho_j(\tau_k) \frac{\partial \boldsymbol{q}_k}{\partial \boldsymbol{u}^R} \right\} \rho_m(\tau_k) = \boldsymbol{0} \tag{24}
$$

$$
\frac{\partial \bar{J}_a}{\partial \tilde{\boldsymbol{\xi}}_m} = \sum_{k=1}^{N} w_k \left(\frac{t_f - t_0}{2} \boldsymbol{f}_k^T - \sum_{i=1}^{N} \boldsymbol{\alpha}_i^T \dot{\rho}_i(\tau_k) \right) \rho_m(\tau_k) = \boldsymbol{0}
$$

$$
\frac{\partial \bar{J}_a}{\partial \tilde{\boldsymbol{\eta}}_m} = \sum_{k=1}^{N} w_k \left(\boldsymbol{q}_k^T + \left(\sum_{r=1}^{N} \boldsymbol{\kappa}_r^T \rho_r(\tau_k) \circ \sum_{l=1}^{N} \boldsymbol{\kappa}_l^T \rho_l(\tau_k) \right) \right) \rho_m(\tau_k) = \boldsymbol{0}
$$

$$
\frac{\partial \bar{J}_a}{\partial \tilde{\boldsymbol{\nu}}} = \boldsymbol{\gamma}^T \left(\boldsymbol{\alpha}_i, \rho_i(-1), \rho_i(1), t_0, t_f \right) = \boldsymbol{0}
$$

$$
\frac{\partial \bar{J}_a}{\partial \boldsymbol{\kappa}_m} = 2 \sum_{k=1}^{N} w_k \left(\sum_{r=1}^{N} \boldsymbol{\kappa}_r \rho_r(\tau_k) \circ \sum_{j=1}^{N} \tilde{\boldsymbol{\eta}}_j^T \rho_j(\tau_k) \right) \rho_m(\tau_k) = \boldsymbol{0}
$$

$$
\frac{\partial \bar{J}_a}{\partial t_0} = \frac{\partial \Gamma}{\partial t_0} + \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial t_0} - \frac{1}{2} \sum_{k=1}^{N} w_k \left(L_k + \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(\tau_k) \boldsymbol{f}_k \right) = 0
$$

$$
\frac{\partial \bar{J}_a}{\partial t_f} = -\frac{\partial \Gamma}{\partial t_f} - \tilde{\boldsymbol{\nu}}^T \frac{\partial \boldsymbol{\gamma}}{\partial t_f} - \frac{1}{2} \sum_{k=1}^{N} w_k \left(L_k + \sum_{j=1}^{N} \tilde{\boldsymbol{\xi}}_j^T \rho_j(\tau_k) \boldsymbol{f}_k \right) = 0 \tag{25}
$$

for $m = 1, 2, \ldots, N$. Equations (23)–(25) are KKT optimality conditions for the NLP problem of (17) and (18).

B. First-Order Necessary Conditions of the Optimal Control Problem

Assuming $\boldsymbol{\lambda}(\tau) \in \mathbb{R}^n$ is the costate, and $\boldsymbol{\mu}(\tau) \in \mathbb{R}^q$ is the Lagrange multiplier associated with the path constraints, Lagrangian of the Hamiltonian (augmented Hamiltonian) of the optimal control problem of (1)–(4) can be written as

$$
\bar{H}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = L(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\mu}^T (\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{p} \circ \boldsymbol{p}) \tag{26}
$$

where $\bar{H}$ is the augmented Hamiltonian and $\boldsymbol{p}$ is the slack variable function. Please note that the notation τ has been removed from (26) for simplicity. The first-order necessary conditions of the optimal control problem are derived as

$$
\dot{\boldsymbol{x}}^T = \frac{t_f - t_0}{2} \boldsymbol{f}^T(\boldsymbol{x}, \boldsymbol{u}) = \frac{t_f - t_0}{2} \frac{\partial \bar{H}}{\partial \boldsymbol{\lambda}}
$$

$$
\boldsymbol{\lambda}^T = \frac{d\boldsymbol{\lambda}}{d\tau} = -\frac{t_f - t_0}{2} \left(\frac{\partial L}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} + \boldsymbol{\mu}^T \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{x}} \right)
$$

$$
= -\frac{t_f - t_0}{2} \frac{\partial \bar{H}}{\partial \boldsymbol{x}}
$$

$$
\frac{\partial L}{\partial \boldsymbol{u}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}} + \boldsymbol{\mu}^T \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{u}} = \frac{\partial \bar{H}}{\partial \boldsymbol{u}} = \boldsymbol{0}
$$

$$
\boldsymbol{\gamma}^T (\boldsymbol{x}(-1), t_0, \boldsymbol{x}(1), t_f) = \boldsymbol{0}
$$

$$
\boldsymbol{\lambda}^T(-1) = -\frac{\partial \Gamma}{\partial \boldsymbol{x}}\bigg|_{\tau=-1} - \boldsymbol{\nu}^T \frac{\partial \boldsymbol{\gamma}}{\partial \boldsymbol{x}}\bigg|_{\tau=-1}
$$

$$\lambda^T(1) = \left.\frac{\partial \Gamma}{\partial x}\right|_{\tau=1} + v^T \left.\frac{\partial \gamma}{\partial x}\right|_{\tau=1}$$

$$q^T(x,u) + p^T \circ p^T = 0, \quad \frac{\partial \bar{H}}{\partial p} = 2\mu^T \circ p = 0$$

$$\bar{H}(t_0) = \frac{\partial \Gamma}{\partial t_0} + v^T \frac{\partial \gamma}{\partial t_0}, \quad \bar{H}(t_f) = -\frac{\partial \Gamma}{\partial t_f} - v^T \frac{\partial \gamma}{\partial t_f} \qquad (27)$$

where $v \in \mathbb{R}^\gamma$ is the Lagrange multiplier associated with the boundary conditions γ.

C. RBF-Galerkin Discretized Form of First-Order Necessary Conditions

The first-order necessary conditions of (27) are discretized using the RBF-Galerkin method. To this end, the costates $\lambda(\tau) \in \mathbb{R}^n$ and Lagrange multipliers $\mu(\tau) \in \mathbb{R}^q$ are approximated using N global RBFs as

$$\lambda(\tau) \approx \lambda^R(\tau) = \sum_{j=1}^N \xi_j \rho(\|\tau - \tau_j\|) = \sum_{j=1}^N \xi_j \rho_j(\tau) \qquad (28)$$

$$\mu(\tau) \approx \mu^R(\tau) = \sum_{j=1}^N \eta_j \rho(\|\tau - \tau_j\|) = \sum_{j=1}^N \eta_j \rho_j(\tau) \qquad (29)$$

where $\lambda^R(\tau)$ and $\mu^R(\tau)$ are the RBF approximations of $\lambda(\tau)$ and $\mu(\tau)$. Also, ξ_j and η_j are the RBF weights for $\lambda^R(\tau)$ and $\mu^R(\tau)$, respectively. By using (7), (8), and (14) along with (28) and (29), the first-order necessary conditions of (27) are parameterized with the global RBFs. Then, applying the Galerkin projection to the residuals and approximating the Galerkin integral with a numerical quadrature *discretize* the first-order necessary conditions as

$$\sum_{k=1}^N w_k \left(\frac{t_f - t_0}{2} f_k^T - \sum_{i=1}^N \alpha_i^T \dot{\rho}_i(\tau_k) \right) \rho_m(\tau_k) = 0 \qquad (30)$$

$$\frac{t_f - t_0}{2} \sum_{k=1}^N w_k \left\{ \frac{\partial L_k}{\partial x^R} + \sum_{j=1}^N \xi_j^T \rho_j(\tau_k) \frac{\partial f_k}{\partial x^R} \right.$$
$$\left. + \sum_{j=1}^N \eta_j^T \rho_j(\tau_k) \frac{\partial q_k}{\partial x^R} + \frac{2}{t_f - t_0} \sum_{j=1}^N \xi_j^T \dot{\rho}_j(\tau_k) \right\} \rho_m(\tau_k) = 0$$
$$\qquad (31)$$

$$\sum_{k=1}^N w_k \left\{ \frac{\partial L_k}{\partial u^R} + \sum_{j=1}^N \xi_j^T \rho_j(\tau_k) \frac{\partial f_k}{\partial u^R} \right.$$
$$\left. + \sum_{j=1}^N \mu_j^T \rho_j(\tau_k) \frac{\partial q_k}{\partial u^R} \right\} \rho_m(\tau_k) = 0 \qquad (32)$$

$$\gamma^T \left(\alpha_i, \rho_i(-1), \rho_i(1), t_0, t_f \right) = 0$$

$$\frac{\partial \Gamma}{\partial x_1^R} + v^T \frac{\partial \gamma}{\partial x_1^R} = -\sum_{j=1}^N \xi_j^T \rho_j(-1)$$

$$\frac{\partial \Gamma}{\partial x_N^R} + v^T \frac{\partial \gamma}{\partial x_N^R} = \sum_{j=1}^N \xi_j^T \rho_j(1)$$

$$\sum_{k=1}^N w_k \left(q_k^T + \left(\sum_{r=1}^N \kappa_r^T \rho_r(\tau_k) \circ \sum_{l=1}^N \kappa_l^T \rho_l(\tau_k) \right) \right) \rho_m(\tau_k) = 0$$

$$2\sum_{k=1}^N w_k \left(\sum_{j=1}^N \eta_j^T \rho_j(\tau_k) \circ \sum_{r=1}^N \kappa_r \rho_r(\tau_k) \right) \rho_m(\tau_k) = 0$$

$$\bar{H}(t_0) = \frac{\partial \Gamma}{\partial t_0} + v^T \frac{\partial \gamma}{\partial t_0}, \quad \bar{H}(t_f) = -\frac{\partial \Gamma}{\partial t_f} - v^T \frac{\partial \gamma}{\partial t_f}. \qquad (33)$$

D. Costate Mapping Theorem

So far, two sets of equations were derived corresponding to two different problems: The KKT optimality conditions of the RBF-Galerkin method shown by (23)–(25) in Section A and the discretized form of the first-order necessary conditions of the optimal control problem described by (30)–(33) in Section B. It was shown in the literature [3], [9] that dualization and discretization are not commutative operations, in general. In fact, when a continuous-time optimal control problem is discretized, a fundamental loss of information occurs in either primal or dual variables. Similar to the costate mapping theorem previously shown in the literature for PS methods [11], [28], [29], the RBF-Galerkin costate mapping theorem is developed here to restore this loss of information by adding a set of discrete equations to the problem defined in Section A.

To provide an exact equivalency between the KKT optimality conditions of the NLP derived from the RBF-Galerkin method and the discretized form of the first-order necessary conditions of the optimal control problem, a set of conditions must be added to (23)–(25). These conditions are

$$\frac{\partial \Gamma}{\partial x_1^R} + \tilde{v}^T \frac{\partial \gamma}{\partial x_1^R} = -\sum_{j=1}^N \tilde{\xi}_j^T \rho_j(-1)$$

$$\frac{\partial \Gamma}{\partial x_N^R} + \tilde{v}^T \frac{\partial \gamma}{\partial x_N^R} = \sum_{j=1}^N \tilde{\xi}_j^T \rho_j(1). \qquad (34)$$

Also, comparing (23)–(25) with (30)–(33) implies that

$$\bar{H}(t_0) = \bar{H}(t_f) = \frac{1}{2} \sum_{k=1}^N w_k \left(L_k + \sum_{j=1}^N \tilde{\xi}_j^T \rho_j(\tau_k) f_k \right). \qquad (35)$$

Discrete conditions of (34) and (35) are known as *closure conditions* in the literature [28], [29] and applied to the costates and Hamiltonian boundaries to guarantee that first-order necessary conditions of the NLP (i.e., KKT conditions) are equivalent to the discretized form of the first-order necessary conditions of the optimal control problem. In other words, by adding (34), (35) to the KKT conditions, the dualization and discretization are made commutative and hence the solution of the direct method is the same as the solution of indirect method.

RBF-Galerkin Costate Mapping Theorem: There is an exact equivalency between the KKT multipliers of NLP derived from the RBF-Galerkin method and Lagrange multipliers (costates) of the optimal control problem discretized by the RBF-Galerkin method.

Lemma 2: The Lagrange multipliers of the optimal control problem can be estimated from the KKT multipliers of NLP at

discretization points by the equations

$$\boldsymbol{\xi}_j = \tilde{\boldsymbol{\xi}}_j, \quad \boldsymbol{\eta}_j = \frac{2}{t_f - t_0} \tilde{\boldsymbol{\eta}}_j, \quad \boldsymbol{\upsilon} = \tilde{\boldsymbol{\upsilon}}, \quad j = 1, 2, \ldots, N. \quad (36)$$

Proof: Substitution of (36) in (31), (32), and (33) proves that (23)–(25) and (30)–(33) are the same, and hence the equivalency condition holds. ∎

V. NUMERICAL EXAMPLES

Two numerical examples are provided to demonstrate the efficiency of the proposed method. Example 1 is a bang-bang optimal control problem for which an analytical solution is available. The optimal trajectories calculated by the RBF-Galerkin method are evaluated against the exact solutions. Also, the costates computed by the RBF-Galerkin costate mapping theorem are compared against the exact costates from the analytical approach. Bang-bang is a typical nonsmooth optimal control problem in which the optimal solution has a switching time needed to be accurately estimated. Therefore, the efficacy of the RBF-Galerkin approach in solving a nonsmooth optimal control problem as well as the accuracy of the RBF-Galerkin costate mapping theorem will be thoroughly investigated in this Example.

Example 2 is a robot motion planning problem with obstacle avoidance in which the optimal trajectories from the RBF-Galerkin method are evaluated against those calculated from the two existing optimal control methods: DIDO [30], a commercial optimal control software tool using Legendre PS method (global polynomial), and OPTRAGEN [31], an academic optimal control software package using B-Spline approach (local polynomial) for direct trajectory optimization. Comparison studies between the proposed approach and the existing methods are presented to demonstrate the superior performance of the RBF-Galerkin solution for a typical motion planning problem.

A. Example 1

Consider a bang-bang optimal control problem with quadratic cost as to minimize

$$J = \frac{1}{2} \int_0^5 \left(x_1^2(t) + x_2^2(t) \right) dt \quad (37)$$

subject to:

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -x_1(t) + x_2(t) + u(t)$$
$$x_1(0) = 0.231, \quad x_2(0) = 1.126, \quad |u(t)| \le 0.8. \quad (38)$$

According to [16], the solution can be calculated from an analytical approach as

$$u^*(t) = \begin{cases} -0.8 & 0 \le t \le 1.275 \\ 0.8 & 1.275 \le t \le 5. \end{cases} \quad (39)$$

This example was thoroughly investigated in [16], in which the authors concluded that a PS method in the classic form cannot accurately solve it due to the discontinuity of the optimal control. For instance, it has been shown that the switching time of the solution cannot be accurately estimated from the Chebyshev PS method of [10] and the numerical

solution from the PS method includes undesired fluctuations at the boundaries (see [16] for more details).

On the other hand, modified PS schemes [15]–[18] may provide better performance than their classic counterparts for solving nonsmooth optimal control problems. However, they suffer from serious constraints limiting their applicability for solving such problems. For instance, a modified PS technique is more prone to the initial guess of parameters (reduced robustness), imposes higher computational loads, and can only handle limited form of state dynamics (i.e., dynamic constraints must be converted to explicit or implicit integral form), compared to a classic PS method [15].

In light of the current limitations with the existing methods, we investigated the efficiency of the RBF-Galerkin method for solving the nonsmooth optimal control problem of (37) and (38). To leverage the capability of arbitrary discretization of the proposed method, a set of pseudorandom points along with the trapezoidal quadrature was chosen for the discretization. 40 randomly distributed points were selected in the interval [0 5] from which at least five points were located between [1.2 1.3]. Increasing the density of discretization points around the discontinuity, i.e., $t = 1.275$, which is not typically possible in other direct methods, enhances the performance of the RBF-Galerkin method in accurately capturing the switching time of the solution. By parameterizing the states and control with the IMQ RBFs and applying the aforementioned discretization points, the problem of (37) and (38) was transcribed into an NLP, which was solved by SNOPT [32], a sparse NLP solver, with default feasibility/optimality tolerances ($\approx 10^{-6}$).

Fig. 1 shows the states and control trajectories obtained from the RBF-Galerkin method for 40 pseudorandomly distributed points along with their exact solutions. Also, the costates estimated from the proposed method are illustrated along with the exact costates in Fig. 2. The accuracy of the proposed method is clearly demonstrated in graphs even for
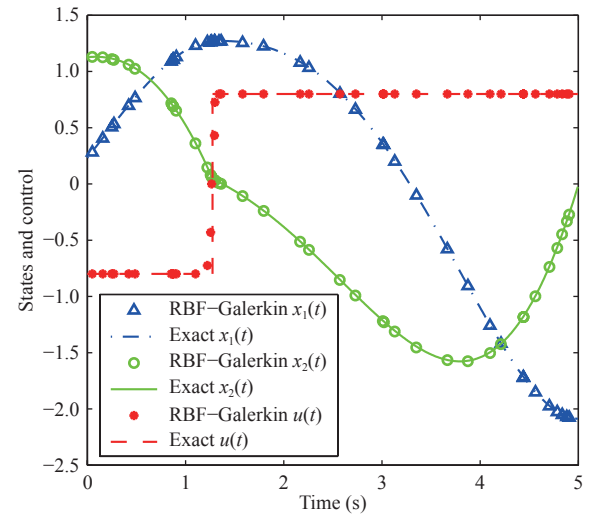

Fig. 1. States and control trajectories calculated by the RBF-Galerkin method for 40 pseudorandomly distributed points along with the exact solutions.
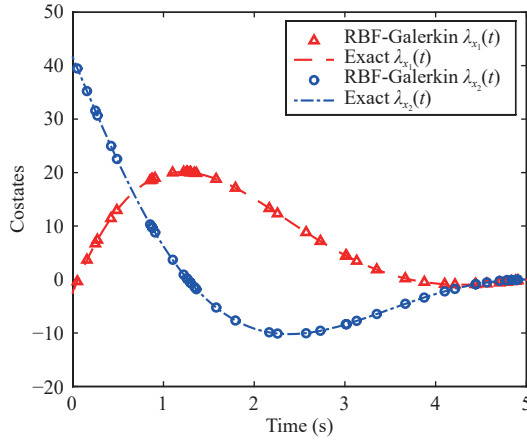
Fig. 2.   Costates computed by the RBF-Galerkin costate mapping theorem along with the exact costates for 40 pseudorandomly distributed points.

those points located near the control discontinuity ($t = 1.275$). The cost value calculated from the RBF-Galerkin method was 5.663 (error ≈ 0.003), and the switching time of the optimal control was estimtaed as 1.279 (error ≈ 0.004). The maximum absolute errors of the states $x_1(t)$ and $x_2(t)$ (over all 40 random points) were 2.5×10^{-6} and 7.9×10^{-6}, respectively. Also, the maximum absolute error of the optimal control $u(t)$ was 0.63. This occurs during the switching time (maximum error of the optimal control was 2.3×10^{-6} without considering the points located between [1.2 1.3]).

The maximum absolute errors for costates $\lambda_{x_1}(t)$ and $\lambda_{x_2}(t)$ (over 40 points) were 6.6×10^{-5} and 3.6×10^{-5}, respectively. This numerically verifies the accuracy of RBF-Galerkin costate mapping theorem as in (36). Even higher accuracy can be achieved by increasing the number of discretization points. For instance, the maximum absolute errors of $\lambda_{x_1}(t)$ and $\lambda_{x_2}(t)$ will be decreased to 3.4×10^{-6} and 3.0×10^{-6} (close to the level of feasibility and optimality tolerances set in the NLP solver) by increasing the discretization points to 80.

B. Example 2

A robot motion planning problem with obstacle avoidance in 2-dimensional space is considered. It is desired to find an optimal trajectory for a mobile robot that spends minimum kinetic energy to navigate through three circular obstacles in a fixed time span [0 20]. The obstacles are located at (40, 20), (55, 40), and (45, 65) with the radius $r = 10$. The horizontal and vertical speeds of the mobile robot cannot exceed 10. The optimal control problem is formulated as to minimize the cost function

$$J = \int_0^{20} (\ddot{x}^2(t) + \ddot{y}^2(t))\, dt \qquad (40)$$

subject to the constraints

$$|\dot{x}(\tau)| \le 10 \; , \; |\dot{y}(\tau)| \le 10$$
$$0 \le x(\tau) \le 80 \; , \; 0 \le y(\tau) \le 80$$
$$x(0) = 40 \; , \; y(0) = 5$$
$$x(20) = 55 \; , \; y(20) = 70 \qquad (41)$$

and nonlinear path constraints (obstacles)

$$10^2 \le (x(\tau) - 40)^2 + (y(\tau) - 20)^2 \le 80^2$$
$$10^2 \le (x(\tau) - 55)^2 + (y(\tau) - 40)^2 \le 80^2$$
$$10^2 \le (x(\tau) - 45)^2 + (y(\tau) - 65)^2 \le 80^2. \qquad (42)$$

The optimal trajectory for the mobile robot was computed from three different methods: The RBF-Galerkin approach, Legendre PS method (DIDO), and B-Spline approach (OPTRAGEN). All three methods use the same environment (MATLAB) along with the same NLP solver (SNOPT). To conduct a fair comparison, Legendre-Gauss-Lobatto points –type of points used in the Legendre PS method– were incorporated in the other two methods, as well. The cost and computation time of each method are demonstrated in Table I for different number of discretization points, i.e., $N = [10, 20, 30]$.

TABLE I
COST AND COMPUTATION TIME OF RBF-GALERKIN, LEGENDRE PS, AND B-SPLINE METHODS FOR ROBOT MOTION PLANNING EXAMPLE FOR $N = [10, 20, 30]$

Method	N	Cost	Time (s)
RBF-Galerkin	10	255.62	0.89
Legendre PS	10	278.43	2.50
B-Spline	10	260.67	0.98
RBF-Galerkin	20	255.40	1.02
Legendre PS	20	260.70	18.32
B-Spline	20	256.44	1.50
RBF-Galerkin	30	254.32	1.27
Legendre PS	30	254.37	44.63
B-Spline	30	254.35	1.87

By increasing the number of discretization points, the accuracy of trajectories improves at the expense of higher computation time. Among the three methods, the RBF-Galerkin had the least cost value and the shortest computation time for each value of N. For instance, the cost function from the RBF-Galerkin approach for $N = 10$ had about 2% and 8% less value (more accurate) than the B-Spline and Legendre PS method, respectively. Also, the computation time of the RBF-Galerkin approach was 9% faster than B-Spline and about 64% faster than Legendre PS method for the same number of discretization points. The computational efficiency of the RBF-Galerkin method is more profound for $N = 20$ and $N = 30$. This comparison studies clearly demonstrate superior accuracy and computational efficiency of the proposed approach against the state of the art in a motion planning example. The optimal trajectory calculated by the RBF-Galerkin approach for $N = 30$ is shown in Fig. 3.

VI. CONCLUSION

The RBF-Galerkin method combining RBF interpolation with Galerkin projection was presented for solving optimal control problems numerically. The proposed method incorporates arbitrary global RBFs along with the arbitrary discretization scheme offering a highly flexible framework for direct transcription. The RBF-Galerkin costate mapping
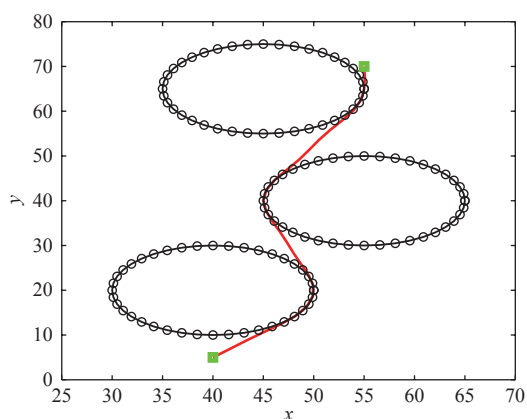
Fig. 3.   Optimal trajectory estimated by RBF-Galerkin method for robot motion planning with obstacle avoidance, $N = 30$.

theorem was developed through which the costates of the optimal control problem can be exactly estimated from the KKT multipliers of NLP at the discretization points. The efficacy of the proposed method for computing the states, costates, and optimal control trajectories as well as accurately capturing the switching time of the control function was verified through a bang-bang example for which an exact solution was available. Also, the superior accuracy and computational efficiency of the RBF-Galerkin approach were confirmed against a local and a global polynomial method for a motion planning example with obstacle avoidance. As the future extension, it is suggested to find an automated strategy to fine-tune the design parameters of global RBFs, including free shape parameter, to minimize the RBF interpolation error and promote the overall performance of the RBF-Galerkin approach.

REFERENCES

[1] X. Wang J. Liu, and H. Peng, "Computational techniques for nonlinear optimal control," in *Symplectic Pseudospectral Methods for Optimal Control*. Intelligent Systems, Control and Automation: Science and Engineering, vol. 97, Springer, Singapore, 2021.

[2] B. A. Conway, "A survey of methods available for the numerical optimization of continuous dynamic systems," *J. Optim. Theory Appl.*, vol. 152, no. 2, pp. 271–306, Feb. 2012.

[3] A. V. Rao, "Trajectory Optimization: A Survey," in: Optimization and Optimal Control in Automotive Systems, *Lecture Notes in Control and Information Sciences* vol. 455, Springer, Cham, pp. 3–21, 2014.

[4] W. W. Hager, "Runge-Kutta methods in optimal control and the transformed adjoint system," *Numer. Math.*, vol. 87, pp. 247–282, 2000.

[5] C. Jiang, K. Xie, C. Yu, *et al.*, "A sequential computational approach to optimal control problems for differential-algebraic systems based on efficient implicit Runge–Kutta integration," *Appl. Math. Modelling*, vol. 58 pp. 313–330, Jun. 2018.

[6] J. Mei, F. Zhang, J. Zang, Y. Zhao, and H. Yan, "Trajectory optimization of the 6-degrees-of-freedom high-speed parallel robot based on B-spline curve," *Scientific Progress*, vol. 103, no. 1, pp. 1–26, 2020.

[7] Y. E. Tabriz and A. Heydari, "Generalized B-spline functions method for solving optimal control problems," *Comput. Methods Differential Equations*, vol. 2, no. 4, pp. 243–255, 2014.

[8] X. Tang and J. Chen, "Direct trajectory optimization and costate

estimation of infinite-horizon optimal control problems using collocation at the flipped legendre-gauss-radau points," *IEEE/CAA J. Autom. Sinica*, vol. 3, pp. 174–183, 2016.

[9] I. M. Ross and M. Karpenko, "A review of pseudospectral optimal control: From theory to flight," *Annual Reviews in Control*, vol. 36, no. 2, pp. 182–197, 2012.

[10] H. Ma, T. Qin, and W. Zhang, "An efficient Chebyshev algorithm for the solution of optimal control problems," *IEEE Trans. Autom. Control*, vol. 56, no. 3, pp. 675–680, 2011.

[11] D. A. Benson, G. T. Huntington, T. P. Thorvaldsen, and A. V. Rao, "Direct trajectory optimization and costate estimation via an orthogonal collocation method," *J. Guidance Control Dynamics*, vol. 29, no. 6, pp. 1435–1440, 2006.

[12] G. Elnagar, M.A. Kazemi, and M. Razzaghi, "The pseudospectral Legendre method for discretizing optimal control problems," *IEEE Trans. Autom. Control*, vol. 40, no. 10, pp. 1793–1796, 1995.

[13] Q. Gong, I. M. Ross, and F. Fahroo, "Spectral and pseudospectral optimal control over arbitrary grids," *J. Optim. Theory Appl.*, vol. 169, no. 3, pp. 759–783, 2016.

[14] M. A. Mehrpouya and H. Peng, "A robust pseudospectral method for numerical solution of nonlinear optimal control problems," *Intern. J. Comput. Math.*, 2020. DOI: 10.1080/00207160.2020.1807521

[15] M. Shamsi, "A modified pseudospectral scheme for accurate solution of bang-bang optimal control problems," *Optim. Control Appl. Meth.*, vol. 32, pp. 668–680, 2011.

[16] E. Tohidi and S. L. Noghabi, "An efficient Legendre pseudospectral method for solving nonlinear quasi bang-bang optimal control problems," *J. Applied Math. Statistics Inform.*, vol. 8, no. 2, pp. 73–85, 2012.

[17] J. Wei, X Tang, and J. Yan, "Costate estimation for a multiple-interval pseudospectral method using collocation at the flipped Legendre-Gauss-Radau Points," *IEEE/CAA J. Autom. Sinica*, pp. 1–15, Nov. 2016. DOI: 10.1109/JAS.2016.7510028

[18] H. Ghassemi, M. Maleki, and M. Allame, "On the modification and convergence of unconstrained optimal control using pseudospectral methods," *Optimal Control Applications & Methods*, vol. 42, no. 3, pp. 717–743, 2021.

[19] H. Mirinejad, "A radial basis function method for solving optimal control problems," Ph.D. dissertation, University of Louisville, May 2016.

[20] J. A. Rad, S. Kazem, and K. Parand, "Optimal control of a parabolic distributed parameter system via radial basis functions," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 8, pp. 2559–2567, 2014.

[21] H. Mirinejad and T. Inanc, "A radial basis function method for direct trajectory optimization," in *Proc. American Control Conf.*, Chicago, IL. 2015, pp. 4923–4928.

[22] H. Mirinejad and T. Inanc, "An RBF collocation method for solving optimal control problems," *Robotics Autonomous Syst.*, vol. 87, pp. 219–225, 2017.

[23] H. Mirinejad and T. Inanc, "RBF method for optimal control of drug administration in the anemia of hemodialysis patients," in *Proc. 41st Annu. Northeast Biomed. Eng. Conf.*, Troy, NY, 2015.

[24] H. Mirinejad and T. Inanc, "Individualized anemia management using a radial basis function method," in *Proc. IEEE Great Lakes Biomed. Conf.*, Milwaukee, WI, 2015.

[25] H. Mirinejad, T. Inanc, M. Brier, and A. Gaweda, "RBF-based receding horizon control approach to personalized anemia treatment," in *Proc. 41st Annu. Northeast Biomed. Eng. Conf.*, Troy, NY, 2015.

[26] S. Hubbert, Q. Thong, L. Gia, and T. M. Morton, *Spherical Radial Basis Functions, Theory and Applications*. 1st ed., Springer International Publishing, 2015.

[27] K. E. Atkinson and W. Han, *Theoretical Numerical Analysis: A Functional Analysis Framework*. 3rd ed., Springer-Verlag, New York,

2009.

[28] I. M. Ross, " A historical introduction to the covector mapping principle," in *Proc. AAS/AIAA Astrodynamics Specialist Conf.*, Lake Tahoe, CA, 2005.

[29] Q. Gong, I. M. Ross, W. Kang, and F. Fahroo, "Connections between the covector mapping theorem and convergence of pseudospectral methods for optimal control," *Computat. Optim. and Appl.*, vol. 41, pp. 307–335, 2008.

[30] M. Ross and F. Fahroo, User's manual For DIDO 2002: A MATLAB application package for dynamic optimization, Dept. Aeronautics and Astronautics, Naval Postgraduate School, AA-02-002, Monterey, CA, 2002.

[31] R. Bhattacharya, " OPTRAGEN, a MATLAB toolbox for optimal trajectory generation," in *Proc. 45th IEEE Conf. Decision and Control*, San Diego, CA, Dec. 2006.

[32] P. E. Gill, W. Murray, and M. A. Saunders, " SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM Review*, vol. 47, no. 1, pp. 99–131, 2005.

Hossein Mirinejad (M' 12) received the Ph.D. degree in electrical engineering from the University of Louisville, USA, in 2016. He is an Assistant Professor in the College of Aeronautics and Engineering at Kent State University, USA. Prior to joining Kent State, he was a Research Fellow at the Food and Drug Administration (2017–2019) and a Postdoctoral Fellow at the University of Michigan-Ann Arbor (2016–2017). His research interests include modeling and control of autonomous systems, optimal control theory and applications, healthcare automation, and biomedical control systems.

Tamer Inanc (SM' 16) received the B.S. degree in electrical engineering from Dokuz Eylul University, Turkey in 1991. He received the M.S. and Ph.D. degrees in electrical engineering from Penn State University in 1994 and 2002, respectively. He was a Postdoctoral Scholar at Control and Dynamical Systems Department at Caltech, USA, between 2002 and 2004. He is currently an Associate Professor at Department of Electrical and Computer Engineering at the University of Louisville, USA. His research interests include autonomous robotics, nonlinear trajectory generation for UAVs, robust active vision systems, robust control and identification, biometrics, and biomedical problems.

Jacek M. Zurada (LF'15) received the Ph.D. degree in electrical engineering from Gdansk University of Technology, Poland, in 1975. He is a Professor of electrical and computer engineering at the University of Louisville, USA. He authored or coauthored several books and over 380 papers in computational intelligence, neural networks, machine learning, logic rule extraction, and bioinformatics, and delivered more than 100 presentations throughout the world. Prof. Zurada has been awarded numerous distinctions, including the 2013 Joe Desch Innovation Award, the 2015 Distinguished Service Award, and five honorary professorships. In 2014, he served as the IEEE V-President and Technical Activities (TAB Chair). He also chaired the IEEE TAB Periodicals Committee, and the TAB Periodicals Review and Advisory Committee. He was the Editor-in-Chief of the *IEEE Transactions on Neural Networks* from 1997 to 2003 and an Associate Editor of the *IEEE Transactions on Circuits and Systems*, *Neural Networks*, the *Proceedings of the IEEE*, *Neurocomputing*, and several other journals. From 2004 to 2005, he was the President of the IEEE Computational Intelligence Society. He has been a Board Member of IEEE, IEEE CIS, and IJCNN. He has been elected as a Member of Polish Academy of Sciences since 2005.