Nonlinear Optimal Control Synthesis Using Basis Functions: Algorithms and Examples

Minh Vu[1], Hao Fang[1], and Shen Zeng[1]

Abstract— **The ability to quickly synthesize an optimal control signal for a nonlinear system is critical for practical implementations. In previous work, we have introduced a computational procedure to iteratively synthesize an optimal control signal for a very broad class of nonlinear control systems. This paper presents an extension of the approach to allow for different parameterizations of control inputs, resulting in a substantial reduction in the number of decision variables and thus computation time. The highlighted efficiency and effectiveness of the proposed approach are illustrated and compared against other methods using various control examples.**

I. INTRODUCTION

For challenging control applications, the ability to obtain a *dynamically feasible* open-loop control solution which provides a certain control performance as well as safety is particularly important. The ability to do so is a critical element in building an overall sound and reliable feedback control system, because it would minimize the unnecessary effort of forcing a system to follow (or track) what was assumed (by a user) to be a reasonable trajectory which, in fact, might not be even dynamically feasible, thus potentially leading to detrimental and obstructing effects.

As a result, a good amount of recent progress has been made in generating an open-loop control solution for nonlinear complex constrained systems. These methods include both sample-based approaches such as the Rapidly Exploring Randomized Trees (RRTs) family [1], and optimization-based approaches such as the trajectory optimization [2]. However, using the above approaches, one faces significant challenges in obtaining an optimal control solution that is strictly dynamically feasible (e.g., see the exponential computational complexity of nonlinear kinodynamic motion planning [3], or the effects of transcription errors on the accuracy of trajectory optimization [4], [5]).

As a new direction, a computationally attractive procedure to iteratively synthesize an optimal control signal that is always dynamically feasible for a general nonlinear control system has been recently established [6], [7], [8]. In this paper, we present an extension of this framework to allow for different parameterizations of the control input. In particular, by parameterizing the control input using suitable basis functions, we significantly reduce the number of decision variables and computation time compared to the original approach with only a minor penalty on the objective cost.

[1]Department of Electrical and System Engineering, Washington University in St. Louis, St. Louis, MO, USA, emails: {minhvu, f.hao, s.zeng}@wustl.edu.

This work was supported by the NSF under the award CMMI-1933976.

Piecewise-constant control inputs are among the most commonly applied control parameterizations because of their simplicity and versatility across different control problems; however, it has been shown that different control parameterizations (e.g., basis functions) in practice tend to substantially reduce the number of decision variables, and thus minimize computation time [9], [10]. The time complexity can then be traded-off with the number of basis functions instead of the number of time steps as in the standard piecewise-constant control inputs. This advantage provides one with the ability to quickly compute a reasonable control solution even for a considerably long control horizon [11], [12]. Therefore, in this paper, we propose a general mechanism to readily equip the iterative framework with different control parameterization schemes, making it an attractive and viable option for practical implementations.

To this end, the content of this paper is organized as follows. In the next section, we first state the optimal control problem. In Section III, we give an overview of the original iterative optimal control synthesis framework with the standard piecewise-constant control input. In Section IV, we present the proposed approach to parameterize the control signal using a linear combination of basis functions. In Section V, we provide a comparative study of different control parameterizations for a classical control problem and investigate the tradeoff between computational complexity and optimality. In this section, we also present different implementations to illustrate the computational reduction and robustness of the proposed approach compared to other methods such as trajectory optimization.

II. PROBLEM STATEMENT

Consider the minimum energy nonlinear optimal control problem in the following form

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & \int_0^T \|u(t)\|^2 dt \\ \text{subject to} \quad & \dot{x}(t) = f(x(t), u(t)) \\ & x(0) = x_0,\ x(T) = x_{\text{target}} \\ & u \in L^2([0,T], \mathbb{R}^m) \end{aligned} \tag{1}$$

where $\|.\|$ denotes the 2-norm (which will be used through out in this paper), $L^2([0,T], \mathbb{R}^m)$ represents the space of square integrable functions mapping from $[0,T]$ to $\mathbb{R}^m$, and x_0, $x_{\text{target}} \in \mathbb{R}^n$ are the initial condition and the desired terminal state, respectively. In the following, we will give an overview of the computationally iterative procedure to solve this rather general nonlinear optimal control problem.

III. AN OVERVIEW OF AN ITERATIVE OPTIMAL CONTROL SYNTHESIS

To discretize the continuous-time dynamics of $\dot{x}(t) = f(x(t), u(t))$, we first employ the commonly adopted zero-order hold assumption. More specifically, by letting $u(t) \equiv \bar{u}_k$ for some $\bar{u}_k \in \mathbb{R}^m$, $t \in [k\Delta T, (k+1)\Delta T]$ where ΔT is the step size and $k \in \mathbb{N}$ is the time index, we compute $x((k+1)\Delta T)$ from the values of $x(k\Delta T)$ and $\bar{u}_k$ by considering the autonomous system

$$\frac{d}{dt}\begin{bmatrix}\xi_1(t)\\ \xi_2(t)\end{bmatrix} = \begin{bmatrix}f(\xi_1(t), \xi_2(t))\\ 0\end{bmatrix}, \begin{bmatrix}\xi_1(0)\\ \xi_2(0)\end{bmatrix} = \begin{bmatrix}x(k\Delta T)\\ \bar{u}_k\end{bmatrix}$$

for $t \in [0, \Delta T]$. Denoting the flow of the above autonomous system as $(\Phi_t)_{t\in\mathbb{R}}$, we simply have

$$\begin{bmatrix}x((k+1)\Delta T)\\ \bar{u}_k\end{bmatrix} = \Phi_{\Delta T}\left(\begin{bmatrix}x(k\Delta T)\\ \bar{u}_k\end{bmatrix}\right).$$

This representation gives us a constructive formulation (though not a closed analytic one) for the discrete-time evolution of the continuous-time system, i.e.,

$$x_{k+1} = F(x_k, u_k) \tag{2}$$

where F is a part of $\Phi_{\Delta T}$ corresponding to the mapping to $x((k+1)\Delta T)$. Moreover, if the Jacobian of F could be calculated (e.g., see [13]), we will also have a constructive way to compute the linearization of the system at each time step.

In our practical computations, given an initial state x_0 and a nominal control signal $U = \left[u_0^\top, u_1^\top, \dots, u_{N-1}^\top\right]^\top$, rather than using traditional first-order discretizations and linearizations, we employ a recently established numerical approach utilizing high-order Taylor series expansion [13], to accurately obtain the aforementioned Jacobian of F, and therefore, the following linearization of the continuous-time dynamics

$$\delta x_{k+1} = A_k \delta x_k + B_k \delta u_k, \qquad \delta x_0 = 0 \tag{3}$$

where $A_k := \frac{\partial F}{\partial x}(x_k, u_k), B_k := \frac{\partial F}{\partial u}(x_k, u_k)$. Iterating (3), we can unfold the above dynamics into

$$\begin{aligned}
\delta x_1 &= B_0 \delta u_0\\
\delta x_2 &= A_1 B_0 \delta u_0 + B_1 \delta u_1\\
&\ \vdots\\
\delta x_N &= A_{N-1} \dots A_1 B_0 \delta u_0 + \dots + B_{N-1}\delta u_{N-1}.
\end{aligned}$$

Given that δu_k is sufficiently small to ensure the appropriateness of the above linearizations, our idea under this approach is to approximate the terminal state of the system trajectory when applying a slightly perturbed control input by the following expression

$$\tilde{x}_N \approx x_N + \underbrace{\begin{bmatrix}A_{N-1}\dots A_1 B_0 & \cdots & B_{N-1}\end{bmatrix}}_{=:\,H} \underbrace{\begin{bmatrix}\delta u_0\\ \vdots\\ \delta u_{N-1}\end{bmatrix}}_{=:\,\Delta U}$$

where $\tilde{x}_N$ is the resulting terminal state of the system trajectory due to the perturbed control signal $U + \Delta U$. The main advantage of this approximation is that it breaks the nonlinear "coupled" dynamics and significantly simplifies the nonlinear control problem. Using this simple yet powerful idea, we will lay out in the following an iterative procedure to systematically synthesize a minimal energy steering control for a board class of nonlinear control systems. The overall procedure comprises of the two following parts: iteratively steer the system to the target state, then fine-tune the control input to ensure the achievement of a minimal control energy.

A. Iterative Steering

Given an (arbitrary) initial control inputs U along with the corresponding state evolution $x_0, x_1, \dots, x_N$, we consider the following optimization problem

$$\underset{\Delta U}{\text{minimize}} \quad \|x_N + H\Delta U - x_{\text{target}}\|^2 + \lambda\|\Delta U\|^2 \tag{4}$$

where $\lambda \geq 0$ is a regularization parameter that enforces a penalty on the magnitude of ΔU. This penalty helps us to ensure a sufficiently incremental update of U at each iteration and thus guarantees the appropriateness of the aforementioned linearization (approximation) of the system. For the quadratic program (4), one can immediately obtain the following explicit solution

$$\Delta U^* = -(H^\top H + \lambda I)^{-1} H^\top (x_N - x_{\text{target}}).$$

From [6], the iterative scheme for steering the system to the target state is as follows.

Algorithm 1 Steering to the target

Require: Desired terminal state x_{target}, initial input U.
1: Apply the input U to the system and store all A_k, B_k.
2: Calculate H.
3: Solve for ΔU^* of the optimization problem (4).
4: Update the control input via $U = U + \Delta U^*$.
5: Repeat step 1-4 until $\|x_N - x_{\text{target}}\| \leq \epsilon_{1,\text{tol}}$.

B. Optimal Steering

The first part of the overall procedure will result in a steering of the system to a small neighborhood of the target. The second part is established to ensure that the input computed in the first step will have a minimal control energy. More specifically, we consider

$$\begin{aligned}
&\underset{\Delta U}{\text{minimize}} && \|U + \Delta U\|^2 + \gamma\|\Delta U\|^2\\
&\text{subject to} && x_N + H\Delta U = x_{\text{target}}
\end{aligned} \tag{5}$$

where $\gamma \geq 0$ is the regularization parameter, which has the same purpose as λ in the first part. The second iterative scheme to achieve a minimal energy control signal is presented in Algorithm 2.

In summary, under this framework, we strategically transform the general nonlinear optimal control problem into two iterative sequences of quadratic programs (4) and (5). Given

Algorithm 2 Minimizing the control energy

Require: Nominal input U steering the system to x_{target}.
1: Apply the input U to the system and store all A_k, B_k.
2: Calculate H.
3: Solve for ΔU^* of the optimization problem (5).
4: Update the control input via $U = U + \Delta U^*$.
5: Repeat step 1-4 until $\|\Delta U^*\| \leq \epsilon_{2,\text{tol}}$.

an arbitrary initial control input, we first apply Algorithm 1 to iteratively steer the system to a desired target. Then, using the input thereof as a nominal control, we employ Algorithm 2 to iteratively minimize the control energy and thus eventually obtain a minimal energy point-to-point steering control solution.

Due to the favorable (convex) structure of the approach, this computationally attractive procedure has been utilized by the authors in various nonlinear control applications ranging from motion planning for non-holonomic control systems to optimal charging control of Lithium-ion batteries in which complicated nonlinear constraints can be seamlessly included [7], [14]. To even further reduce the computational load of this framework for practical applications, in the next section, we propose an approach to parameterize the control signal using a linear combination of basis functions.

IV. AN ITERATIVE OPTIMAL CONTROL SYNTHESIS USING BASIS FUNCTIONS

The most commonly applied numerical approach to optimal control synthesis is to partition the control horizon into smaller sub-intervals and parameterize the control signal with piecewise-constant control values. However, for applications with long control horizons and/or fast sampling rates, this approach often results in a rather sizable problem with a large number of decision variables, which significantly increases the computational complexity [15], [16].

To reduce the number of decision variables and therefore avoid the above problem, we propose an approach to parameterize the control signal by a linear combination of arbitrary basis functions, i.e.,

$$u(t) = I_{m\times m} \otimes \mathcal{B}(t)\,\alpha \tag{6}$$

where $I_{m\times m}$ is a $m \times m$ identity matrix, $\otimes$ denotes the Kronecker product, $\mathcal{B}(t) := [b_1(t), b_2(t), \ldots, b_L(t)] \in \mathbb{R}^{1\times L}$ represents the basis in which, without loss of generality, we assume $\langle b_i, b_j \rangle = \delta_{ij}$, i.e., the basis functions are orthonormal, and $\alpha \in \mathbb{R}^{mL}$ are the coefficients.

With this expression, our goal is to find an appropriate α such that its linear combination with the basis via the expression of (6) will best approximate the optimal control signal. The best approximation here is the orthogonal projection of the optimal control signal (with respect to the inner product defined in $L_2([0,T],\mathbb{R}^m)$) onto the finite dimensional vector space spanned by the orthonormal basis elements of $b_1, \ldots, b_L$. Now, under the zero-order hold assumption of the time-discretization, the control signal is again assumed to be constant (but not as decision variables) over each time step. To achieve that, we consider the following expression of the control signal

$$u(t) = I_{m\times m} \otimes \mathcal{B}(k\Delta T)\,\alpha, \text{ for } t \in [k\Delta T, (k+1)\Delta T].$$

To shorten the notation, we denote the above expression as

$$u_k = I_{m\times m} \otimes \mathcal{B}_k\,\alpha$$

where k denotes the time index. Then, a control input over the entire control horizon can be described as follows.

$$U = \begin{bmatrix} u_0 \\ \vdots \\ u_{N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} I_{m\times m} \otimes \mathcal{B}_0 \\ \vdots \\ I_{m\times m} \otimes \mathcal{B}_{N-1} \end{bmatrix}}_{=: K} \alpha. \tag{7}$$

Under the control parameterization using basis functions, a slight perturbation of the control signal, i.e., ΔU, is achieved by a corresponding perturbation of the basis coefficients, i.e.,

$$\Delta U = K\Delta\alpha \text{ where } \Delta\alpha = \left[\delta\alpha_1, \delta\alpha_2, \ldots, \delta\alpha_{mL}\right]^\top.$$

This simple expression will later allow us to conveniently substitute ΔU in the previous construction (as in Section III) with $K\Delta\alpha$. To this end, we are now ready to establish an iterative procedure to synthesize a minimal coefficient α such that its linear combination with the basis elements $b_1, \ldots, b_L$ will best approximate the minimum energy control signal.

A. Iterative Steering

Let $\mathscr{H} = HK$, under the control parameterization using basis functions, we rewrite (4) as

$$\underset{\Delta\alpha}{\text{minimize}} \quad \|x_N + \mathscr{H}\Delta\alpha - x_{\text{target}}\|^2 + \lambda\|\Delta\alpha\|^2 \tag{8}$$

where λ is an regularization parameter that enforces a penalty on the magnitude of $\Delta\alpha$ so as to ensure sufficiently incremental changes of the control signal at each iteration. Similarly, we have the following explicit solution of (8)

$$\Delta\alpha^* = -(\mathscr{H}^\top\mathscr{H} + \lambda I)^{-1}\mathscr{H}^\top(x_N - x_{\text{target}}).$$

From Algorithm 1, the modified iterative scheme for steering the system to the target state is as follows.

Algorithm 3 Steering to the target

Require: A desired x_{target}, an (arbitrary) initial coefficient α, and a finite set of orthonormal basis elements $b_1, \ldots, b_L$ in $L_2([0,T],\mathbb{R})$.
1: Calculate the control input U using equation (7).
2: Apply the input U to the system and store all A_k, B_k.
3: Calculate $\mathscr{H}$.
4: Solve for $\Delta\alpha^*$ of the optimization problem (8).
6: Update the control parameterization via $\alpha = \alpha + \Delta\alpha^*$.
7: Repeat step 1-6 until $\|x_N - x_{\text{target}}\| \leq \epsilon_{1,\text{tol}}$.

B. Optimal Steering

The previous subsection lays out the first part of the overall procedure, which iteratively steers the system to a small neighborhood of the target state. We now establish a second iteration to ensure that the control input under the parameterization scheme will have a minimal control energy. To this end, we consider the following optimization

$$\begin{aligned} \underset{\Delta\alpha}{\text{minimize}} \quad & \|\alpha + \Delta\alpha\|^2 + \gamma\|\Delta\alpha\|^2 \\ \text{subject to} \quad & x_N + \mathscr{H}\Delta\alpha = x_{\text{target}} \end{aligned} \tag{9}$$

where $\gamma \geq 0$ is the regularization parameter, which again has the same purpose as the regularization parameter λ in the first part. The modified second iterative scheme to achieve the minimum control energy is as follows.

Algorithm 4 Minimizing the control energy

Require: The finite basis elements $b_1, \ldots, b_L$ and a nominal coefficient α acquired from the result of Algorithm 3.
1: Calculate the control input U using equation (7).
2: Apply the input U to the system and store all A_k, B_k.
3: Calculate $\mathscr{H}$.
4: Solve for $\Delta\alpha^*$ of the optimization problem (9).
6: Update the control parameterization via $\alpha = \alpha + \Delta\alpha^*$.
7: Repeat step 1-6 until $\|\Delta\alpha\| \leq \epsilon_{2,\text{tol}}$.

Let $J_L := \|\alpha\|^2 = \sum_{i=1}^{mL} \alpha_i{}^2$, we observe that the energy of the approximated control signal monotonically decreases as the number of basis elements increases, i.e., $J_L \leq J_{L+1}$. In fact, given that $[\alpha_1, \ldots, \alpha_{mL}]^\top$ are the optimal coefficients of $b_1, \ldots, b_L$, we can always choose $[\alpha_1, \ldots, \alpha_m, 0, \ldots, \alpha_{m(L-1)+1}, \ldots, \alpha_{mL}, 0]^\top$ as a feasible candidate and search for the optimal coefficients of $b_1, \ldots, b_{L+1}$. This property effectively allows us to better approximate the optimal control signal by increasing the number of basis functions. In other words, the trade-off between computational complexity and optimality can be analyzed and balanced via the number of basis functions. This advantage allows us to not only reduce the number of decision variables but also maintain the accuracy and the favorable (convex) structure of the iterative framework.

V. A COMPARATIVE STUDY WITH ILLUSTRATIVE EXAMPLES

The previous section presents a general mechanism to parameterize (or approximate) an optimal control signal using a finite set of basis functions. In this section, we demonstrate the effectiveness of the proposed approach under different control examples and discuss the trade-off between computational complexity and optimality via a simple comparative study.

A. Minimum energy swing up control

We consider the problem of swinging up an inverted pendulum on a cart using a minimum control energy. The motion of the system is described by the following dynamics

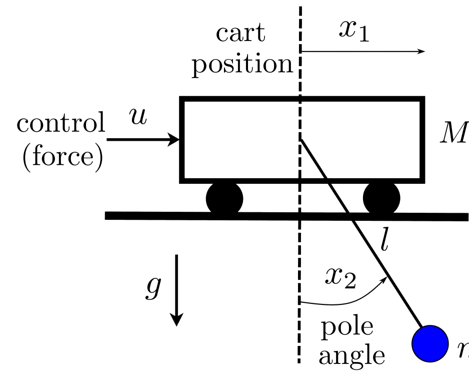

Fig. 1: The model of an inverted pendulum on a cart with the following parameters: $M = 1$, $m = 0.3$, $l = 0.5$, $g = 9.81$.

$$\begin{aligned} \dot{x}_1 &= x_3 \\ \dot{x}_2 &= x_4 \\ \dot{x}_3 &= \frac{ml\sin(x_2)x_4^2 + mg\cos(x_2)\sin(x_2) + u}{M + m(1-\cos^2(x_2))} \\ \dot{x}_4 &= \frac{g(M+m)\sin(x_2) + ml\cos(x_2)\sin(x_2)x_4^2 + u\cos(x_2)}{-Ml - ml(1-\cos^2(x_2))} \end{aligned}$$

where x_1, x_2, x_3, and x_4 are the position of the cart, the angle of the pendulum, the speed of the cart, and the angular velocity of the pendulum, respectively, as illustrated in Figure 1.

In our experiment, we consider four different approaches to the minimum-energy swing-up control problem. First, we apply the original iterative method as introduced in Section III. For the second and third cases, we implement the proposed method, using the first 20 elements of the Legendre polynomial basis and those of the Fourier basis, respectively. Finally, we employ a trajectory optimization technique using a trapezoidal collocation on a standard mesh-grid. To keep the same parameters for all cases, we set $N = 200$, $\Delta T = 0.01$, $\lambda = 0.01$, $\mu = 2$, $\epsilon_{1,\text{tol}} = 10^{-2}$, $\epsilon_{2,\text{tol}} = 10^{-4}$, $x_0 = [0, 0, 0, 0]^\top$, and $x_{\text{target}} = [1, \pi, 0, 0]^\top$.

The computational complexities of the four approaches are presented in Table I. All the computations are done on a personal laptop with an IntelCore i7-4770HQ (2.50 Ghz) CPU and 16GB RAM using Matlab. One can notice that the first three approaches using the iterative optimal control synthesis framework are significantly faster than the trajectory optimization approach. The reason is that in order for the solution of the trajectory optimization to be (dynamically) accurate, a significantly dense grid (of 400

	Part1		Part2	
	Iterations	Time	Iterations	Time
Original iterative	106	4.2s	335	15.1s
Legendre basis	16	0.7s	173	8.0s
Fourier basis	12	0.6s	212	10s
Trajectory optimization	-	-	-	932s

TABLE I: Time complexity of four control approaches.

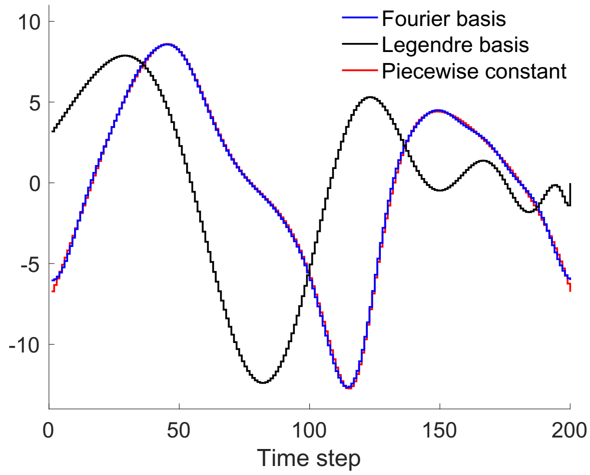

Fig. 2: Optimal control signals obtained from the iterative framework under different control parameterization schemes.

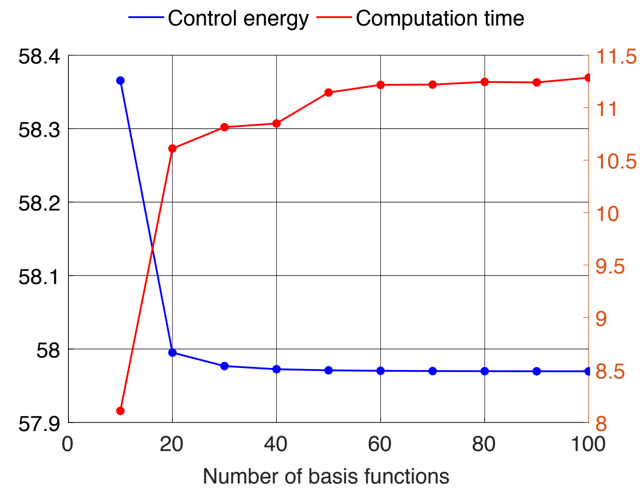

Fig. 3: The trade-off between the number of basis elements, the computational complexity (in seconds) and the control energy (cost).

points) is required, which results in a considerably time-consuming computation. This requirement is necessary for the accuracy because although the transcription errors (often calculated as the difference between a candidate trajectory and the actual system dynamics between each collocation points) are small, these errors can propagate over time and eventually lead to a considerably large error at the end [17]. As a result, we emphasize that due to the inherent transcription errors, a solution generated by any trajectory optimization process must be cautiously verified with an independent test so as to check whether the generated control signal can actually steer the system to a desired target in open loop (e.g., see [4], [5], [17], [18] for similar conclusions).

As presented in Table I, the two methods using the proposed approach converge to their optimal solutions significantly faster than the original iterative method. This rapid convergence agrees with the previous argument that due to a small number of decision variables, control parameterizations using basis functions naturally result in more lightweight computations. However, also due to this property, these control parameterizations might become less flexible than the original iterative method. Although the choice of a particular basis hardly affects the success of the steering task (i.e., the first part of the framework), it tends to influence the minimal cost of the final control solution.

From our investigation, we notice that a basis which shares some similar properties to the system often results in a better (or lower cost) control solution. In deed, one can observe from Figure 2 that due to the same intrinsic periodic property of Fourier basis and the cart-pendulum system, the solution using Fourier basis is almost identical to that of the original iterative method, while the control solution using Legendre polynomial basis is quite different. As a result, the energy (cost) of the control input using Fourier basis is 57.99 which is very close to (though slightly higher than) the energy of the original iterative method of 57.97, while the control energy using Legendre polynomial basis is 63.78.

B. Analyzing the trade-off between the number of basis elements, time complexity and optimality

Recall from Section IV that our goal under a basis parameterization is to find a linear combination of finite basis elements to best approximate the optimal control signal in which the approximation is the orthogonal projection of the optimal control signal onto the parametric space (i.e., the finite dimensional vector space spanned by these orthonormal basis elements). A natural question that often arises in this context is that how many bases should we choose so that we can obtain a good approximation of the optimal control signal. To study the effects of the number of basis elements, we repeatedly apply the proposed approach using the Fourier basis with different numbers of basis elements (varying from 10 to 100).

As the numbers of basis elements increases, the parametric space (i.e., the space spanned by these orthonormal basis elements) becomes larger, which naturally leads to a better approximation of the optimal control signal. As a result, the control energy (cost) decreases and quickly plateaus at a certain level as the number of basis elements increases, as illustrated in Figure 3. However, the increase in number of basis elements also raises the number of decision variables, leading to a larger problem thus a longer computation. It is observed that the decrease in control energy only comes with a minor cost of increasing computation. In this example, the proposed approach reaches almost the exact same cost as the optimal control signal of the original iterative method (at 40 basis elements) and still maintains a faster convergence (under 11 seconds), as illustrated in Figure 3. This favorable nature is also observed in the upcoming example.

C. Minimum energy attitude control

We consider the attitude control problem, also known as the control of rigid-body rotations. The set of attitudes of a rigid body is represented by the set of 3×3 orthogonal matrices (rotation matrices) whose determinant is 1, which is often referred to as the special orthogonal group $SO(3)$ [19]. The rotation of a rigid body is described by the following dynamics

$$\dot{R} = R\omega^{\times}$$
$$J\dot{\omega} = J\omega \times \omega + u$$

where $R(t) \in SO(3)$ is the rotation matrix representing an attitude of a rigid body, $J \in \mathbb{R}^{3\times 3}$ is the moment of inertia of the body, $\omega := [w_1, w_2, w_3]^\top \in \mathbb{R}^3$ are the angular velocities, and

$$\omega^{\times} := \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix}.$$

The control objective is to design an input u which steers the rigid-body with $J = diag\{3, 4, 5\}$ from an initial position of $R_0 = I_{3\times 3}$ with $\omega_0 = [0, 0, 0]^\top$ to a desired target $R_{\text{target}} = [-0.3995, 0.8201, 0.4097, 0.1130, -0.3995, 0.9097, 0.9097, 0.4097, 0.0670]^\top$ with $\omega_{\text{target}} = [0, 0, 0]^\top$.

To this end, we first rewrite the system in the form of $\dot{x} = f(x, u)$, where $x := [R_{11}, R_{12}, R_{13}, R_{21}, R_{22}, R_{23}, R_{31}, R_{32}, R_{33}, \omega_1, \omega_2, \omega_3]^\top \in \mathbb{R}^{12}$, and $u := [u_1, u_2, u_3]^\top \in \mathbb{R}^3$. Then, using $N = 500$, $\Delta T = 0.01$, $\lambda = 0.01$, $\mu = 2$, $\epsilon_{1,\text{tol}} = 10^{-2}$, $\epsilon_{2,\text{tol}} = 10^{-4}$, and 10 elements of the Legendre polynomial basis, we apply the proposed approach to synthesize a minimal energy steering control, as illustrated in Figure 4. This maneuver is equivalent to a 150° counter-clockwise rotation around $v = [\frac{1}{2}, \frac{1}{2}, \frac{1}{\sqrt{2}}]$. In this example, our computation time of the control parameterization is on average around 6 seconds with the control energy (cost) of 14.62. On the other hand, it takes approximately 28 seconds to synthesize an optimal control input with the energy (cost) of 11.24 using the original iterative method.

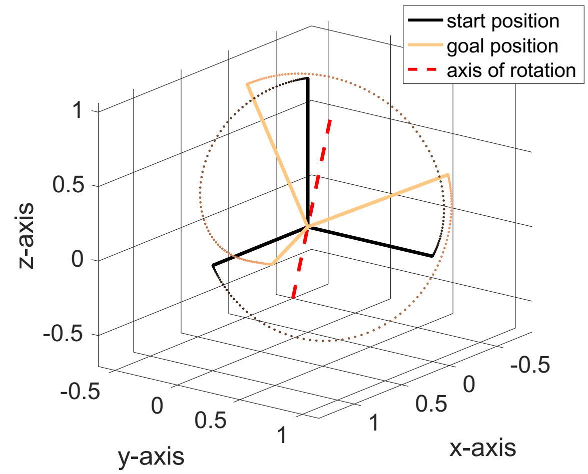

Fig. 4: The simulated rotation of a body-fixed frame represented by the three axes. The rotating frame is uniformly sampled in time but only its tip is plotted, going from (dark) black to (light) brown as the trajectory progresses.

VI. Summary and Conclusions

It has been shown with the help of different examples that parameterization of the control input signal using basis functions has a great potential to significantly reduce the computation time of nonlinear optimal control syntheses with only a minor penalty on the objective cost. The overarching framework presented in this paper thus shows great promise for the fast generation of optimal steering inputs for a broad class of nonlinear control systems. In ongoing work, we are investigating the application of the presented framework towards the efficient synthesis of (optimal) feedback control laws for nonlinear control systems.

References

[1] S. M. LaValle, "Motion planning: Wild frontiers," *IEEE Robotics Automation Magazine*, vol. 18, no. 2, pp. 108–118, 2011.

[2] M. Kelly, "An introduction to trajectory optimization: How to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.

[3] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *Journal of guidance, control, and dynamics*, vol. 25, no. 1, pp. 116–129, 2002.

[4] I. Ross, "Enhancements to the dido optimal control toolbox," *arXiv preprint arXiv:2004.13112*, 2020.

[5] F. Biral, E. Bertolazzi, and P. Bosetti, "Notes on numerical methods for solving optimal control problems," *IEEJ Journal of Industry Applications*, vol. 5, no. 2, pp. 154–166, 2016.

[6] S. Zeng, "Iterative optimal control syntheses illustrated on the Brockett integrator," in *Proc. 11th IFAC Symposium on Nonlinear Control Systems*, 2019, pp. 138–143.

[7] M. Vu and S. Zeng, "Iterative optimal control syntheses for nonlinear systems in constrained environments," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1731–1736.

[8] ——, "Iterative optimal control synthesis for nonlinear switching systems," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 995–1000.

[9] Q. Lin, R. Loxton, and K. L. Teo, "The control parameterization method for nonlinear optimal control: a survey," *Journal of Industrial and management optimization*, vol. 10, no. 1, pp. 275–309, 2014.

[10] G. Naus, J. Ploeg, M. Van de Molengraft, W. Heemels, and M. Steinbuch, "Design and implementation of parameterized adaptive cruise control: An explicit model predictive control approach," *Control Engineering Practice*, vol. 18, no. 8, pp. 882–892, 2010.

[11] M. Muehlebach and R. D'Andrea, "Parametrized infinite-horizon model predictive control for linear time-invariant systems with input and state constraints," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 2669–2674.

[12] M. Muehlebach, C. Sferrazza, and R. D'Andrea, "Implementation of a parametrized infinite-horizon model predictive control scheme with stability guarantees," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2723–2730.

[13] S. Zeng, "On linearization, discretization, and numerical integration of nonlinear control systems with applications to optimal control design." Preprint, 2019.

[14] M. Vu, S. Zeng, and H. Fang, "Health-aware battery charging via iterative nonlinear optimal control syntheses," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 12 485–12 490, 2020.

[15] S. Shin, T. Faulwasser, M. Zanon, and V. M. Zavala, "A parallel decomposition scheme for solving long-horizon optimal control problems," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 5264–5271.

[16] R. Scattolini and S. Bittanti, "On the choice of the horizon in long-range predictive controlsome simple criteria," *Automatica*, vol. 26, no. 5, pp. 915–917, 1990.

[17] J. T. Betts, *Practical methods for optimal control and estimation using nonlinear programming*. SIAM, 2010.

[18] C. L. Darby, W. W. Hager, and A. V. Rao, "An hp-adaptive pseudospectral method for solving optimal control problems," *Optimal Control Applications and Methods*, vol. 32, no. 4, pp. 476–502, 2011.

[19] M. D. Shuster *et al.*, "A survey of attitude representations," *Navigation*, vol. 8, no. 9, pp. 439–517, 1993.