Iterative optimal control syntheses illustrated on the Brockett integrator

Shen Zeng [*]

[*] *Department of Electrical & Systems Engineering, Washington University, St. Louis, MO, USA*
email: s.zeng@wustl.edu

Abstract: In this paper, we investigate computational methods for synthesizing optimal control inputs for nonholonomic control systems. We use the Brockett integrator as a benchmark example to illustrate different aspects of the computational trajectory optimization problem. The main result of this paper is the establishment of a rather attractive iterative scheme for practically constructing optimal control inputs. The functionality and efficiency of the proposed iterative scheme are illustrated on different optimal steering problems centered around the Brockett integrator.

Keywords: Nonholonomic systems; Optimal control; Computational methods

1. INTRODUCTION

Trajectory optimization for nonholonomic control systems has long been a central problem in control theory and robotics. Its applications range from attitude control of unmanned vehicles and satellites to explaining different aspects of locomotion in robots and animals. From a control theoretic standpoint, the study of nonholonomic systems has seen a significant surge around the 1990s, see, e.g., Murray and Sastry (1993). The studies in this time period were, however, more fundamental in nature, and mostly centered around the insights obtained from Lie algebraic considerations of the nonlinear control systems.

With the increase of computational power over the years, the study of trajectory optimization for nonholonomic control systems has steadily shifted from the more fundamental study of basic principles of motion planning of nonholonomic systems towards slightly more computational approaches aimed at solving the practical control problems in applications. This has been especially the case in the domain of robotics, where learning-based control strategies have seen a rapid development, see, e.g., Abbeel et al. (2010).

In this paper, we revisit one of the most studied examples of a nonholonomic control system: the Brockett integrator, see, e.g., Brockett (1983), Bloch and Drakunov (1996), Astolfi (1998), Khaneja and Brockett (1999), Morgansen and Brockett (1999). Using this example as a benchmark, we illustrate different aspects associated with the more computationally oriented perspective. We start out by considering the very central minimum energy point-to-point steering problem and attempt to solve it by computational means, both with and without the exploitation of the particular structure present in the Brockett integrator. In the latter more generic perspective, our consideration eventually leads us to introduce a novel iterative procedure for synthesizing optimal control inputs of nonholonomic, and, more generally, nonlinear control systems.

Iterative procedures for trajectory optimization present an attractive alternative to shooting and collocation methods as they allow for a simple and rather flexible implementation and operation. There are several related but in their core quite different implementations of this general idea. Notable examples include iterative learning control (see, e.g., Bristow et al. (2006) for a survey) and methods falling under the broader term of iterative LQR methods (Li and Todorov, 2004; Abbeel et al., 2010; Tassa et al., 2012; Bomela and Li, 2016), see also Jacobson and Mayne (1970) for the related idea of differential dynamic programming. In contrast to these existing schemes, the method proposed in this paper is rooted in very tangible basic systems theoretic considerations that, to the best knowledge of the author, have not been considered in this rather pure form.

The paper is organized as follows. In Section 2 we provide a rapid review of the Brockett integrator, which includes an elementary exposition of its well-known geometric interpretation as a “position-area system”. It will then serve as a benchmark example for the general problem of devising (optimal) steering laws for nonholonomic system. In Section 3 we address the problem of synthesizing optimal steering laws for the Brockett integrator in a computational fashion. This consideration extends the nowadays standard discussion for illustrating the emergence of a new control direction by virtue of taking the Lie bracket of the two control vector fields. It is shown that the minimum energy point-to-point steering problem can be cast as a quadratic program that can be solved quite efficiently. Since such favorable structure cannot be expected in a more general setting, it is aimed to devise a method that circumvents the formulation of the trajectory planning problem as a generic nonlinear program. In Section 4, we present such a method and illustrate its functionality and efficiency on different optimal steering problems for the Brockett integrator. Section 5 provides a summary and some further discussions regarding the significance of the proposed iterative scheme.

2. REVIEW OF THE BROCKETT INTEGRATOR

In this section, we repeat some basic geometric perspectives on the Brockett integrator which provide a better understanding of its mechanism for the remainder of our treatment. Given a planar curve $t \mapsto (x(t), y(t)) \in \mathbb{R}^2$, it is an interesting thought to attach to a point $(x(t), y(t))$ on the curve an angle via

$$\varphi(t) = \arctan\left(\frac{y(t)}{x(t)}\right),$$

where we ignore the singularity for $x(t) = 0$ for a moment, cf. the illustration in Figure 1.

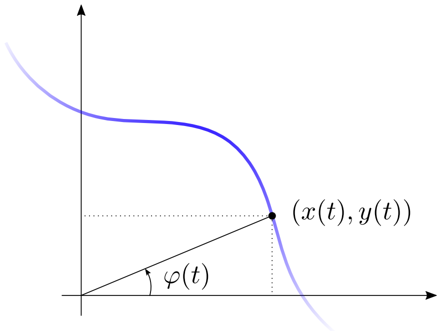

Fig. 1. Definition of the angle for a planar curve.

Formally differentiating φ with respect to time yields

$$\dot{\varphi} = \frac{1}{1+(y/x)^2}\frac{d}{dt}\left(\frac{y}{x}\right) = \frac{x^2}{x^2+y^2}\frac{x\dot{y}-\dot{x}y}{x^2} = \frac{x\dot{y}-\dot{x}y}{x^2+y^2},$$

which is well-defined for all $(x, y) \in \mathbb{R}^2 \setminus \{0\}$. At this point, one could already impose that the curve $t \mapsto (x(t), y(t))$ be produced by a planar single integrator

$$\dot{x} = u_1, \quad \dot{y} = u_2$$

to introduce controls into the picture. This would then provide us with a manageable nonlinear control system with inputs to analyze. But, instead, let us rearrange the equation for $\dot{\varphi}$ to

$$(x^2 + y^2)\dot{\varphi} = x\dot{y} - \dot{x}y \tag{1}$$

to eliminate the somewhat unpleasant singularity at the origin. By the same line of argument as before, this directly corresponds to the control system

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ x_1u_2 - u_1x_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -x_2 & x_1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \end{pmatrix},$$

typically referred to as the Brockett integrator, or, also as the Heisenberg system. While the influence of the control (u_1, u_2) to (x_1, x_2) is rather clear, our previous geometric considerations, specifically the relation (1), also shed light on the dynamics of the state x_3 as the area generated by the (x_1, x_2) curve, as illustrated in Figure 2.

By comparing the dynamics of the Brockett integrator with the introductory geometric consideration, we can explain many known facts in an elementary way. One such fact is that solutions to the optimal steering problem

$$\begin{aligned} &\underset{u}{\text{minimize}} && \|u\|^2 \\ &\text{subject to} && x(t_0) = x_0 \\ & && x(t_f) = x_f \end{aligned}$$

are given by circular arcs in $\mathbb{R}^2$ that are connecting the initial state x_0 and the desired terminal state x_f.

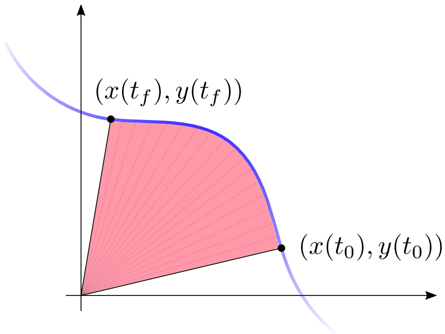

Fig. 2. The value of $\int_{t_0}^{t_f} x\dot{y} - \dot{x}y \, dt$ is precisely given by the (signed) area of the highlighted red sector formed by the curve segment $(x(t), y(t))_{t\in[t_0,t_f]}$ and the origin.

In this paper, we are, however, taking the stance that we are dealing with an arbitrary nonholomic system for which we would like to devise a broadly applicable procedure for practically computing optimal control inputs. Our full understanding of the mechanism behind the Brockett integrator shall be merely used as a means to compare and analyze the quality of the synthesized solutions.

3. ON COMPUTING OPTIMAL CONTROLS FOR THE BROCKETT INTEGRATOR

In this section, we consider the practical problem of synthesizing optimal control inputs for the Brockett integrator by computational means. For the sake of rendering the optimal control problem computationally tractable, in the remainder of the paper we consider the discretized version of the Brockett integrator given by

$$x(k+1) = x(k) + \Delta T \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -x_2(k) & x_1(k) \end{pmatrix}\begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix}, \tag{2}$$

which amounts to a simple first-order expansion of the control vector field. Our consideration essentially constitutes an extension of the nowadays classical consideration of showing the existence of "hidden control directions" given by the Lie bracket of the first two control vector fields and yields some new perspectives and results.

We first note that the first two states are governed by

$$\begin{pmatrix} x_1(N) \\ x_2(N) \end{pmatrix} = \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \Delta T \sum_{j=0}^{N-1}\begin{pmatrix} u_1(j) \\ u_2(j) \end{pmatrix} \tag{3}$$

and evidently quite easy to steer. Any difficulty in steering the system is due to the nonholonomic part for the third state. Nevertheless, given the particular structure of the Brockett integrator, it can be verified that the equation for the third state can be favorably written in closed form as a quadratic function in the control input

$$\mathscr{U} = (u_1(0)\ u_2(0)\ \cdots\ u_1(N-1)\ u_2(N-1))^\top.$$

This is summarized in the following explicit relation.

Proposition 1. For the discretized Brockett integrator (2), the final state $x_3(N)$ can be expressed in terms of $\mathscr{U}$ as

$$x_3(N) = x_3(0) + \mathscr{U}^\top \left(\Delta T \begin{pmatrix} 0 & 1 & \cdots & 1 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 \\ 0 & \cdots & \cdots & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\right)\mathscr{U}.$$

Proof. The third state can be seen to satisfy

$$x_3(N) = x_3(0) + \Delta T \sum_{j=0}^{N-1} (-x_2(j)\ x_1(j)) \begin{pmatrix} u_1(j) \\ u_2(j) \end{pmatrix}. \quad (4)$$

Let e_i denote the ith unit vector of $\mathbb{R}^N$. Then we can write

$$(-x_2(j)\ x_1(j)) = \mathscr{U}^\top \left(\left(\sum_{i=1}^{j} e_i \right) \otimes \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \right),$$

$$\begin{pmatrix} u_1(j) \\ u_2(j) \end{pmatrix} = \left(e_{j+1}^\top \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \mathscr{U}.$$

Given these equations and using the mixed-product property of the Kronecker product, we can write

$$(-x_2(j)\ x_1(j)) \begin{pmatrix} u_1(j) \\ u_2(j) \end{pmatrix} = \mathscr{U}^\top \left(\left(\sum_{i=1}^{j} e_i e_{j+1}^\top \right) \otimes \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \right) \mathscr{U}.$$

The claim follows from plugging this into the sum in (4) and recognizing that

$$\sum_{j=0}^{N-1} \left(\sum_{i=1}^{j} e_i e_{j+1}^\top \right) = \begin{pmatrix} 0 & 1 & \cdots & 1 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 \\ 0 & \cdots & \cdots & 0 \end{pmatrix}.$$

What these considerations show is that the problem of steering the Brockett integrator between two states $x(0)$ and x_f with minimal control energy can be cast as the minimization of $\mathscr{U}^\top \mathscr{U}$ subject to the linear constraint

$$\begin{pmatrix} x_1(N) \\ x_2(N) \end{pmatrix} - \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \Delta T \left(\begin{array}{cc|c|cc} 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \end{array} \right) \mathscr{U},$$

and the quadratic constraint specified in Proposition 1. This constrained quadratic program is evidently very easy to set up and can be solved quite efficiently and reliably, in contrast to, e.g., a generic constrained nonlinear program.

However, because not all nonholonomic systems admit such a favorable structure that one may exploit, there is in fact still a great interest in finding more generally applicable computational schemes for practically synthesizing optimal control inputs. The remainder of this paper is dedicated to this particular problem.

4. AN ITERATIVE SCHEME FOR COMPUTING OPTIMAL CONTROL INPUTS

In this section, we propose a general iterative scheme for computing optimal control inputs for nonlinear systems and illustrate its mechanism and efficiency on the Brockett integrator. Iterative schemes for trajectory optimization for rather nonlinear dynamic problems have long been recognized as a valuable addition to other established methods such as shooting and collocation methods. Notable examples include iterative learning control (see, e.g., Bristow et al. (2006) for a survey) and methods falling under the broader term of iterative LQR methods (Li and Todorov, 2004; Abbeel et al., 2010; Tassa et al., 2012; Bomela and Li, 2016), see also Jacobson and Mayne (1970) for the related idea of differential dynamic programming. It is noted that despite a strong similarity on a high level, the aforementioned methods are quite different in their implementations and their underlying assumptions.

To the best knowledge of the author, the underlying idea and principles of the methodology proposed in the following is furthermore new and significantly different from the aforementioned iterative methods. It is based on the following deceptively simple perspective. If we fix an input $\tilde{\mathscr{U}}$ and apply it to the system in (2), we obtain the resulting state trajectory $\tilde{x}(0), \tilde{x}(1), \ldots, \tilde{x}(N)$. Given this state evolution, we define the matrix

$$\tilde{\mathscr{H}} := \Delta T \left(\begin{array}{cc|c|cc} 1 & 0 & & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ -\tilde{x}_2(0) & \tilde{x}_1(0) & & -\tilde{x}_2(N-1) & \tilde{x}_1(N-1) \end{array} \right). \quad (5)$$

The relevance of this matrix is that $\mathscr{U} \mapsto \tilde{\mathscr{H}} \mathscr{U} + \tilde{x}(N)$ approximately describes the terminal state of the Brockett integrator (2) when applying the input $\tilde{\mathscr{U}} + \mathscr{U}$, assuming that $\|\mathscr{U}\|$ is sufficiently small. In this argument, it is as if we deliberately freeze the state trajectory to $\tilde{x}(0), \ldots, \tilde{x}(N)$ in the definition of the matrix $\tilde{\mathscr{H}}$, which although ultimately a falsification of the true system model, is what leads to the significant simplification of the nonlinear problem to a linear time-varying problem. Another way of viewing this simpliciation is as a breaking up of the nonlinear coupling of the input $\mathscr{U}$ to the system and the corresponding state evolution $x(0), x(1), \ldots, x(N)$ into two seperate steps, which are simpler. Moreover, it can be verified both theoretically and practically that the simplication is sufficiently accurate for the purpose of iteratively generating optimal controls (under the aforementioned condition that $\|\mathscr{U}\|$ is sufficiently small).

Remark 1. It is in fact straightforward to extend the general idea described above to more general discrete-time nonlinear systems, say, of the form

$$x_{k+1} = A(x_k)x_k + B(x_k)u_k.$$

By iterating the above equation starting with an initial state $x_0 \in \mathbb{R}^n$, we obtain the explicit solution

$$\begin{aligned} x_N = \left(\prod_{k=0}^{N-1} A(x_k) \right) x_0 &+ A(x_{N-1}) \cdots A(x_1) B(x_0) u_0 \\ &+ A(x_{N-1}) \cdots A(x_2) B(x_1) u_1 \\ &\ \vdots \\ &+ A(x_{N-1}) B(x_{N-2}) u_{N-2} \\ &+ B(x_{N-1}) u_{N-1}. \end{aligned}$$

Analogously to the basic idea described for the Brockett integrator, we start with a predetermined control input $\tilde{\mathscr{U}}$ and compute the state evolution $\tilde{x}(0), \tilde{x}(1), \ldots, \tilde{x}(N)$. Now we fix $\tilde{A}_k = A(\tilde{x}_k)$ and $\tilde{B}_k = B(\tilde{x}_k)$ and again pretend that the passing from the predetermined control input $\tilde{\mathscr{U}}$ to the slightly perturbed input $\tilde{\mathscr{U}} + \mathscr{U}$ is somewhat faithfully represented via the approximation

$$x_N \approx \tilde{x}_N + \left(\tilde{A}_{N-1} \cdots \tilde{A}_1 \tilde{B}_0 \quad \cdots \quad \tilde{B}_{N-1} \right) \mathscr{U}.$$

This simple yet very practical observation forms the basis for the iterative procedure for synthesizing minimum energy steering laws for the Brockett integrator.

The overall problem can be split into two main parts. First, only paying attention to fulfilling the terminal constraint, and secondly, a fine tuning so as to ensure a minimal control energy. These two steps are described in more detail in the following.

4.1 Achieving satisfaction of the terminal constraints

Given a predetermined control $\tilde{\mathscr{U}}$ along with the corresponding state evolution $\tilde{x}(0), \tilde{x}(1), \ldots, \tilde{x}(N)$, we consider the unconstrained quadratic optimization problem

$$\underset{\mathscr{U}}{\text{minimize}} \; \|\tilde{\mathscr{H}}\mathscr{U} + \tilde{x}(N) - x_f\|^2 + \lambda\|\mathscr{U}\|^2, \qquad (6)$$

where λ is a regularization parameter that enforces a penalty on the magnitude of $\mathscr{U}$ so as to ensure that the incremental change in each iteration is small. We stress that the regularization parameter is key here as without it, one would not be able to guarantee that the system description obtained from plugging in the frozen state evolution is a faithful representation of the real system behavior. Another way of thinking about the regularization parameter is that it provides us with a small budget for incrementally altering our predetermined input $\tilde{\mathscr{U}}$ (as our "belief" about the system mechanism may greatly change when walking different paths later on) so as to gradually come closer to our goal of achieving $\tilde{x}_N \approx x_f$. Thus, if the initial control input $\tilde{\mathscr{U}}$ is chosen very small, we can intuitively expect that the terminal constraints will eventually be achieved by a somewhat economic control input. For the quadratic program (6), one can readily write down the explicit solution as

$$\mathscr{U}^\star = -(\tilde{\mathscr{H}}^\top \tilde{\mathscr{H}} + \lambda I)^{-1} \tilde{\mathscr{H}}^\top (\tilde{x}(N) - x_f). \qquad (7)$$

In summary, we have the following iterative scheme for achieving satisfaction of the terminal constraints.

Algorithm 1 Part 1: Achieving the terminal constraint

Require: Desired terminal state x_f, initial input $\tilde{\mathscr{U}}$.

1: Apply the input $\tilde{\mathscr{U}}$ to the system and store the resulting state trajectory $\tilde{x}(0), \tilde{x}(1), \ldots, \tilde{x}(N)$.
2: Compute $\tilde{\mathscr{H}}$ as defined in (5).
3: Update the control input via $\tilde{\mathscr{U}} \leftarrow \tilde{\mathscr{U}} + \mathscr{U}^\star$, cf. (7).
4: Repeat until $\|\tilde{x}(N) - x_f\|$ falls below desired tolerance.

Figure 3 illustrates the iteration process for the problem of steering $x(0) = (0, 0, 0)$ to the terminal state $x_f = (0, 0, 2)$, with the parameters $\Delta T = 0.02$, $N = 200$, and $\lambda = 0.01$. Since the initial state is the origin and the fact that the Brockett integrator is a driftless system, our initial choice for the predetermined control input $\tilde{\mathscr{U}}$ is significant. While it was mentioned earlier, that it is reasonable to start out with a small control input, we cannot choose $\tilde{\mathscr{U}}$ to be zero here, as otherwise $\tilde{\mathscr{H}}^\top(\tilde{x}(N) - x_f)$ would stay zero, resulting in no update in $\tilde{\mathscr{U}}$ and ultimately a deadlock. There are, however, many other reasonable small but non-zero choices for the initial control input $\tilde{\mathscr{U}}$, such as the randomized choice $\tilde{\mathscr{U}}(k) \sim N(0, \sigma^2)$ with $\sigma \ll 1$. In this particular case, the general iteration scheme can be naively thought of as letting the algorithm explore the system behavior with a random initial control input to get a rough understanding of the dynamic mechanism based on which the algorithm computes an incremental correction to come closer to the goal of satisfying the terminal constraint. It is to be stressed that the purpose of this procedure is not to learn about the model like in other learning-based iteration schemes as the model is fully known to us, but rather to turn the nonlinear (quadratic) problem into a sequence of significantly simpler classical LQR problems.

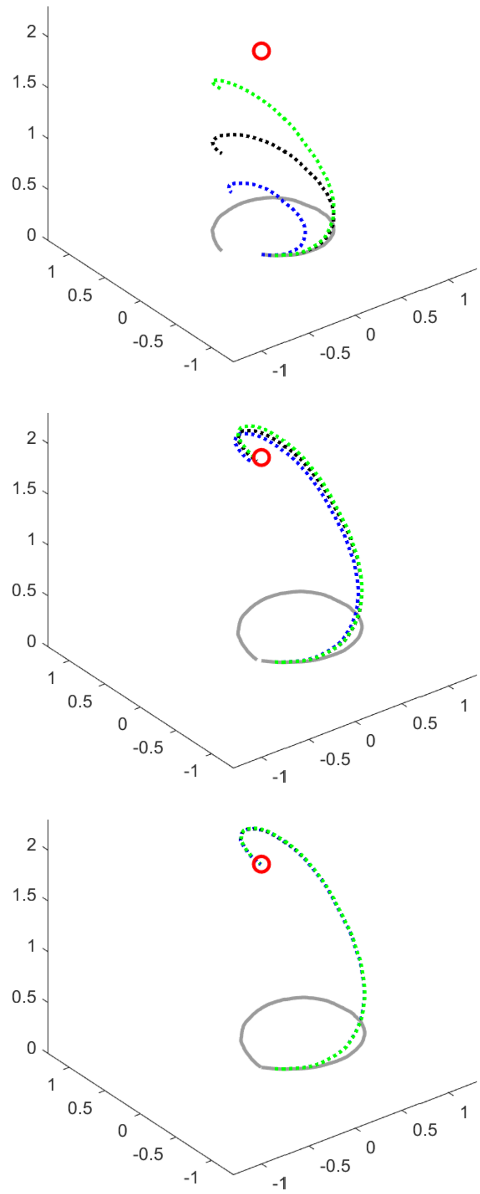

Fig. 3. This figure illustrates different intermediate results in the first set of iterations aimed at achieving satisfaction of the terminal constraint. The blue dotted line illustrates the state trajectory obtained from the predetermined control input $\tilde{\mathscr{U}}$. The black dotted line illustrates the state evolution according to the frozen system model when applying the adjusted control input $\tilde{\mathscr{U}} + \mathscr{U}^\star$. The green dotted line illustrates the real state evolution when applying the adjusted control input $\tilde{\mathscr{U}} + \mathscr{U}^\star$. In this example, the regularization parameter λ has been chosen rather small, so as to speed up the iteration process. Despite the significant mismatch between the black and green trajectories in the beginning, the overall trend of the system is in fact captured sufficiently well as the system "learns" that tracing out larger circles with an increasing size is key. After roughly a dozen iterations, the algorithm converges. We observe that the projection of the resulting state-trajectory in the (x_1, x_2)-plane is close to but slightly different from a circle. This is precisely what will be corrected in the second part.

4.2 Minimizing the control energy

In the foregoing part of the overall iterative scheme, the iterative computation of control inputs that leads to the satisfaction of the terminal constraint is parsimonious by design in the sense that at each step in the iteration, only a small budget for altering the control profile is provided to the algorithm. Thus, it is natural to expect a relatively moderate control energy for the final control obtained from the iterations. On the other hand, it may be argued that the procedure in the first step is a somewhat greedy one, where the choices to be made in future iterations already depend to a large degree on the foregoing choices.

The idea is to establish a second iteration to ensure that the input computed in the first step will have minimal control energy. Since the procedure in the first step has ensured the satisfaction of terminal constraints, we can consider a linearly constrained quadratic program for the second part. More specifically, we consider

$$\begin{aligned} &\underset{\mathscr{U}}{\text{minimize}} \quad \|\tilde{\mathscr{U}} + \mathscr{U}\|^2 + \mu\|\mathscr{U}\|^2 \\ &\text{subject to} \quad \check{\mathscr{H}}\mathscr{U} = x_f - \tilde{x}(N), \end{aligned} \tag{8}$$

where μ is a regularization parameter with the same purpose as the regularization parameter λ in the first part.

Algorithm 2 Part 2: Achieving minimal control energy

Require: Desired final state x_f, input $\tilde{\mathscr{U}}$ steering to x_f.
1: Apply the input $\tilde{\mathscr{U}}$ to the system and store the resulting state trajectory $\tilde{x}(0), \tilde{x}(1), \ldots, \tilde{x}(N)$.
2: Compute $\check{\mathscr{H}}$ as defined in (5).
3: Solve for $\mathscr{U}^\star$ of the constrained quadratic program (8).
4: Update the control input via $\tilde{\mathscr{U}} \leftarrow \tilde{\mathscr{U}} + \mathscr{U}^\star$.
5: Repeat until improvement $\|\mathscr{U}^\star\|$ is too incremental.

Figure 4 illustrates the effect of the second part of iterative scheme to our earlier example with the choice $\mu = 2$. In view of our understanding of optimality in terms of the closeness of the (x_1, x_2)-projection of the state trajectory to arcs of a circle, we find a noticeable improvement of the solution after this second iteration.

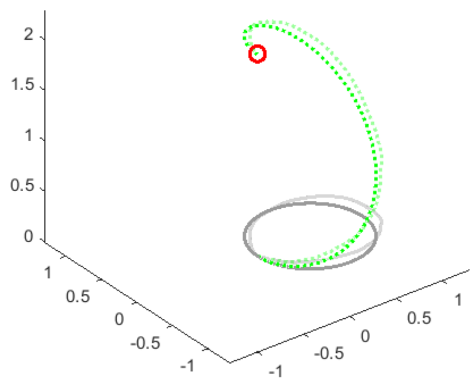

Fig. 4. The transparent green dotted line depicts the result of the first part (aimed at satisfying the terminal constraint). The dark green dotted line depicts the results of applying the second part of the iteration to the control input from the first part.

In Figure 5 we further illustrate the result of two minimum energy stabilization problems with different initial states.

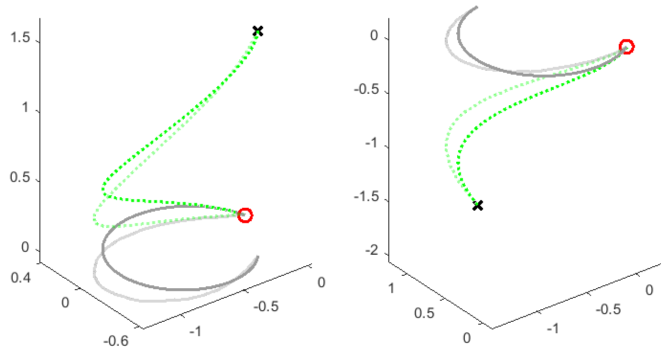

Fig. 5. The results of the first and second part of the iterative procedure for steering to the origin (red circle) with minimal control energy.

More general quadratic cost functionals The presented procedure can be rather directly extended to more general dynamics, such as

$$x_{k+1} = A(x_k)x_k + B(x_k)u_k$$

mentioned earlier before, as well as more general quadratic cost functionals of the form

$$J = \sum_{k=1}^{N} x(k)^\top Q_k x(k) + \sum_{k=0}^{N-1} u(k)^\top R_k u(k)$$

both with or without fixed terminal constraints. The minor modification required here is that instead of the previously defined matrix $\check{\mathscr{H}}$ associated with the terminal state, one would consider the full version, given by

$$\check{\mathscr{H}} = \begin{pmatrix} \tilde{B}_0 & 0 & \cdots & 0 \\ \tilde{A}_1\tilde{B}_0 & \tilde{B}_1 & & \vdots \\ \tilde{A}_2\tilde{A}_1\tilde{B}_0 & \tilde{A}_2\tilde{B}_1 & & \vdots \\ \vdots & \vdots & & 0 \\ \tilde{A}_{N-1}\cdots\tilde{A}_1\tilde{B}_0 & \tilde{A}_{N-1}\cdots\tilde{A}_2\tilde{B}_1 & \cdots & \tilde{B}_{N-1} \end{pmatrix}.$$

Then, defining the extended weight matrices

$$\mathscr{Q} = \operatorname{diag}(Q_1, \ldots, Q_N), \quad \mathscr{R} = \operatorname{diag}(R_0, \ldots, R_{N-1}),$$

the rewritten cost functional of interest reads as

$$\begin{aligned} \tilde{J} = \frac{1}{2}\mathscr{U}^\top(\check{\mathscr{H}}^\top\mathscr{Q}\check{\mathscr{H}} + \mathscr{R} + \mu I)\mathscr{U} \\ + \mathscr{U}^\top\left(\check{\mathscr{H}}^\top\mathscr{Q}\begin{pmatrix}\tilde{x}(1)\\ \vdots \\ \tilde{x}(N)\end{pmatrix} + \mathscr{R}\tilde{\mathscr{U}}\right), \end{aligned}$$

where μ again is the regularization parameter for limiting the magnitude of the incremental adjustment $\mathscr{U}$.

In Figure 6 we illustrate the solution of an optimal stabilization problem ($x(N) \overset{!}{=} 0$) with the cost functional

$$J = \sum_{k=1}^{N} \|x(k)\|^2 + \sum_{k=0}^{N-1} \|u(k)\|^2.$$

The solution to this optimal stabilization problem was obtained by first applying Algorithm 1 to achieve satisfaction of the terminal constraint $x(N) = 0$ and then applying a modified version of Algorithm 2 as detailed above.

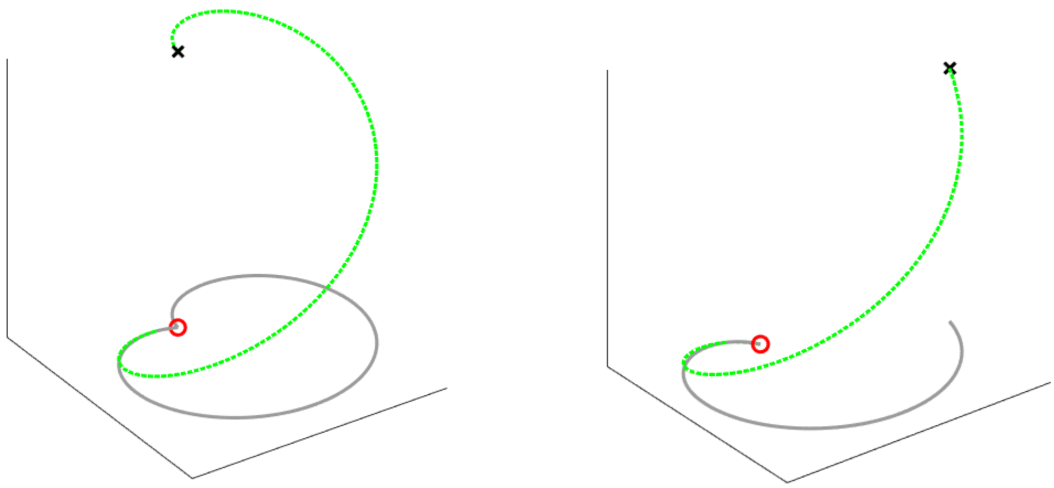

Fig. 6. Left: The state trajectory corresponding to the solution of the optimal stabilization problem with $x(0) = (0, 0, 1)$. Rather interestingly, the (x_1, x_2)-projection appears to depict a *cardioid*. To the best knowledge of the author, this observation is new. Right: The solution to the same optimal stabilization problem with some other initial state.

5. SUMMARY AND DISCUSSION

We have revisted the classical example of the Brockett integrator and discussed interesting aspects of it in a new light. The main focus of this paper has been on the practical computation of optimal control inputs in different ways. While we first showed that making full use of the present structure in the Brockett integrator leads to a simple quadratic program, our desire to obtain a more generic methodology that does not require the presence of such structures led us to introduce a new iterative scheme for synthesizing (open-loop) optimal controls for nonholonomic systems where the cost functional is quadratic. This provided a very convenient and quite elementary way for solving the trajectory optimization problem. In view of the existing iterative methods, we note that the fully open-loop formulation proposed in this paper can be described as more transparent in its operation and also allows for a simpler implementation.

In contrast to other established methods such as collocation methods, the effort in formulating the optimization problem is minimal. However, due to the fact that the proposed method operates on the central premise that at each iteration only minor adjustments are to be implemented, a longer computation time may be expected. Nevertheless, the proposed method establishes a very convenient computational methodology for obtaining the solution of a nonlinear quadratic optimal control problem in a quick straightforward fashion.

In ongoing work, we are further investigating the proposed iterative scheme with respect to rigorous convergence guarantees or the ability to devise an adaptive choice for the regularization parameters, as well as on extending its scope to the general class of continuous-time nonlinear systems (Zeng, 2019).

REFERENCES

Abbeel, P., Coates, A., and Ng, A.Y. (2010). Autonomous helicopter aerobatics through apprenticeship learning. *Int. J. Rob. Res.*, 29(13), 1608–1639.

Astolfi, A. (1998). Discontinuous control of the Brockett integrator. *European Journal of Control*, 4(1), 49–63.

Bloch, A. and Drakunov, S. (1996). Stabilization and tracking in the nonholonomic integrator via sliding modes. *Systems & Control Letters*, 29(2), 91–99.

Bomela, W. and Li, J.S. (2016). An iterative method for computing optimal controls for bilinear quadratic tracking problems. In *Proc. 2016 American Control Conference*, 2912–2917.

Bristow, D.A., Tharayil, M., and Alleyne, A.G. (2006). A survey of iterative learning control. *IEEE Control Systems*, 26(3), 96–114.

Brockett, R.W. (1983). Asymptotic stability and feedback stabilization. *Differential geometric control theory*, 27(1), 181–191.

Jacobson, D.H. and Mayne, D.Q. (1970). *Differential dynamic programming*. Elsevier.

Khaneja, N. and Brockett, R. (1999). Dynamic feedback stabilization of nonholonomic systems. In *Proc. 38th IEEE Conference on Decision and Control*, 1640–1645.

Li, W. and Todorov, E. (2004). Iterative linear quadratic regulator design for nonlinear biological movement systems. In *Proc. 1st International Conference on Informatics in Control, Automation and Robotics*, 222–229.

Morgansen, K.A. and Brockett, R.W. (1999). Nonholonomic control based on approximate inversion. In *Proc. 1999 American Control Conference*, 3515–3519.

Murray, R.M. and Sastry, S.S. (1993). Nonholonomic motion planning: Steering using sinusoids. *IEEE Transactions on Automatic Control*, 38(5), 700–716.

Tassa, Y., Erez, T., and Todorov, E. (2012). Synthesis and stabilization of complex behaviors through online trajectory optimization. In *Proc. International Conference on Intelligent Robots and Systems (IROS)*, 4906–4913.

Zeng, S. (2019). On linearization, discretization, and numerical integration of nonlinear control systems with applications to optimal control design. In preparation.