Linear-Quadratic Optimal Control in Maximal Coordinates

Jan Brüdigam, Zachary Manchester

***Abstract*—The Linear-quadratic regulator (LQR) is an efficient control method for linear and linearized systems. Typically, LQR is implemented in minimal coordinates (also called generalized or "joint" coordinates). However, recent research suggests that there may be numerical and control-theoretic advantages when using higher-dimensional non-minimal state parameterizations for dynamical systems. One such parameterization is maximal coordinates, in which each link in a multi-body system is parameterized by its full SE(3) pose and joints between links are modeled with algebraic constraints. Such constraints can also account for closed kinematic loops or contact with the environment. This paper presents an extension to the standard LQR control method to handle dynamical systems with explicit physical constraints. Experiments comparing the basins of attraction and tracking performance of minimal- and maximal-coordinate LQR controllers suggest that maximal-coordinate LQR achieves greater robustness and improved tracking compared to minimal-coordinate LQR when applied to nonlinear systems.**

***Index Terms*—Maximal coordinates, linear-quadratic regulator (LQR), constrained optimal control**

I. Introduction

Minimal coordinates (also called generalized or "joint" coordinates) have historically dominated robotic simulation and control, possibly due to the perception that they lead to greater computational efficiency. However, rigid body dynamics in maximal coordinates with Lagrange multipliers can be computed with similar efficiency as unconstrained dynamics in minimal coordinates [1], and a substantial body of recent work, for example [2]–[4], has suggested that higher-dimensional linear models may have more descriptive power than minimal models when approximating nonlinear systems.

Maximal-coordinate models of robotic systems are mathematically expressed as differential-algebraic equations (DAEs) in continuous time or algebraic difference equations (ADEs) in discrete time. Unfortunately, the classical derivation of the Linear-Quadratic Regulator (LQR) is not well suited to such systems: If applied naively without explicitly accounting for "joint" or "manifold" constraints in the dynamics, the maximal-coordinate system will typically be mathematically uncontrollable, even if an equivalent minimal-coordinate realization is controllable.

It is important to emphasize that the constraints in maximal-coordinate systems are not artificial state or input constraints that can be controlled by an actuator. Rather, they are hard mechanical constraints that are enforced by the physical structure. Such constraints always arise when using maximal coordinates, but they can also arise in minimal-coordinate settings, for example, for structures with closed-kinematic loops, during contact with the environment, or in nonholonomic settings.

J. Brüdigam is with the Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany. (e-mail: jan.bruedigam@tum.de)

Z. Manchester is with The Robotics Institute, Carnegie Mellon University, Pittsburgh, USA (e-mail: zacm@cmu.edu)

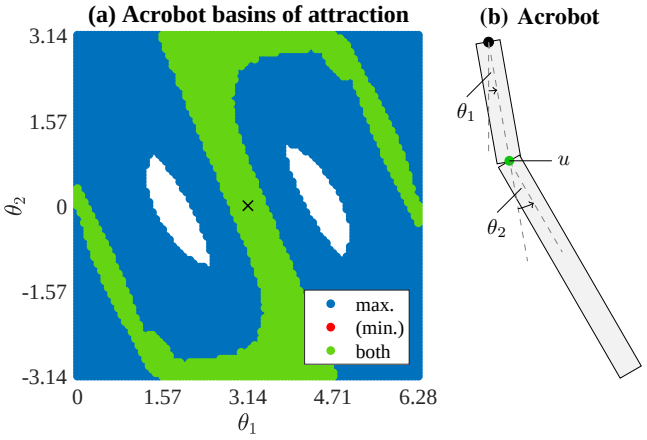

Fig. 1: Basins of attraction for maximal- and minimal-coordinate LQR when stabilizing the upright equilibrium (cross) of an acrobot from different initial configurations. The maximal-coordinate basin is shown in blue, minimal-coordinate basin in red, and basin for both in green. For this example, the maximal-coordinate ba-sin includes the entire minimal-coordinate basin and is significantly larger.

Some extensions of LQR to DAEs have been developed [5], [6], but the derivation of an LQR feedback law for discrete-time maximal-coordinate systems with explicit linear equality constraints presented in this paper has—to the best of our knowledge—not been studied previously. The novel contributions of this paper are:

- The derivation of an LQR feedback control law for maximal-coordinate systems with explicit linear constraints in the dynamics.
- An experimental comparison of the basins of attraction of time-invariant LQR feedback laws for systems described in minimal and maximal coordinates.
- An experimental comparison of trajectory tracking performance with time-varying LQR in minimal and maximal coordinates.

II. Related Work

Since this paper touches on a number of different subjects such as control and linearization of mechanically constrained systems, maximal-coordinate representations, and

linear-quadratic regulation, we want to provide an overview of selected work in these areas and their relation to the methods proposed in this paper.

A. Control of Mechanically Constrained Systems

There are several approaches to control mechanically constrained systems. One common type of constraints encountered with robotic systems are contact constraints. Manipulator contact has been treated by constraining the end effector to always be in contact with the environment. The system is then decomposed into subsystems relating to motion on and off the constraint manifold [7], [8]. Contact arising with walking robots has been handled by analyzing the constraint manifold and subsequently deploying specialized control algorithms for this manifold [9]–[11]. Systems with more general holonomic and nonholonomic constraints have been treated in a similar manner by either splitting the system into subsystems, see for example [12], or by analyzing the behavior on the constraint manifold, see for example [13].

Generally, the methods presented above are based on minimal (or reduced) coordinate representations and aim at reducing, splitting up, or eliminating the effect of constraints.

A more mathematical perspective on mechanically constrained systems can be obtained by treating them as differential-algebraic equations (DAEs). General control of DAEs can be achieved with a variety of nonlinear feedback laws [14]–[17]. Optimal control of nonlinear DAEs has also been proposed [18]–[20], and methods for systems described by linear DAEs have also been developed [5], [6], [21].

B. Linearization of Mechanically Constrained Systems

The linearization of mechanically constrained systems and DAEs has been investigated by several authors. For mathematical theory on the linearization of general DAEs see, for example, [22]. One common method for the linearization of mechanical systems described in minimal coordinates is to split the coordinates into independent and dependent variables [23]–[25]. For an overview of more general linearization schemes for arbitrary coordinates in continuous time see, for example, [26].

Linearization of constrained mechanical systems in discrete time has been proposed by [27] and [28]. However, in these methods the linear models incorporate the constraints directly and they are not explicitly retained.

C. Maximal Coordinates

Maximal coordinates represent the six degrees of freedom of each body in an articulated structure as well as explicit joint constraints between links. Several methods have been developed to efficiently compute dynamics of systems described in maximal coordinates, for example [1], [29], and [30]. An overview of various dynamics algorithms—not only for maximal coordinates—is presented in [31].

A related approach for trajectory optimization can be found in [32], in which partially extended coordinates with absolute orientations (but not absolute positions) are used to describe rigid body dynamics for improved computational performance.

Control of systems described in maximal coordinates has been mainly developed for individual vehicles, such as underwater, land, or aerial vehicles [33]–[35]. In contrast, control of articulated systems parameterized in maximal coordinates has found very little attention. Some ideas related to this can however be found in cooperative robotics, where a number of individual vehicles form a virtual articulated structure when acting together to perform a common task [36], [37].

Another related but fundamentally different control approach is cartesian control. With cartesian control, the goal is to derive a feedback law from the end effector state in some form of maximal or world coordinates [38]–[42]. However, except for the end effector, these systems are still described in minimal coordinates.

D. Linear-Quadratic Regulation

The linear-quadratic regulator is a well-established control scheme for (locally) linear systems. Control strategies based on LQR have been proposed in a variety of forms and for several systems, including the acrobot [43]–[46] and the cartpole [47]–[50], which are also investigated in this paper. All of the LQR variants cited above are built on minimal-coordinate representations.

LQR variants for systems with closed kinematic loops have been established as well. Common examples are parallel robots [51], [52], platform stabilization systems [53]–[55], and delta robots [56], [57]. In this paper, maximal-coordinate LQR will also be applied to balance a delta robot.

All of these systems consist of closed kinematic loops, but for the methods cited, they are nonetheless described in minimal (or reduced) coordinates. In addition, the control algorithms were mainly developed for these specific types of mechanical systems and not generalized to arbitrary structures and constraints.

Since maximal coordinates have been mostly used for individual vehicles, control of constrained systems based on LQR in maximal coordinates has thus been mainly developed for such systems [58]–[62]. Typical constraints occurring for such systems are, for example, nonholonomic constraints due to non-slipping wheels.

LQR for systems with environmental contact, for example walking robots, has been performed by deriving linearizations on the constraint manifold resulting in a linear model without constraints [63], [64].

Discrete-time LQR for systems with mechanical constraints has been proposed by [27] and [28]. However, as mentioned before, the constraints are directly incorporated in the linear system and not explicitly retained.

A different kind of constrained LQR is concerned with state and input constraints. There is a wide variety of approaches to derive LQR control laws for such scenarios [65]–[71], but these constraints are "virtual." As such, they can be physically violated, and part of the control task is to ensure constraint satisfaction by an appropriate choice of inputs. As described in Sec. I, the nature of mechanical constraints treated in this paper is fundamentally different.

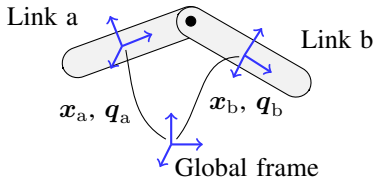

Fig. 2: Two links connected by a joint. Adopted from [29].

III. Background

We will briefly depict the treatment of rigid body dynamics in maximal coordinates (see [1] or [29] for details). Subsequently, we review the derivation of the standard unconstrained linear-quadratic regulator (LQR) from the dynamic programming principle with Riccati recursion, upon which we will build.

A. Maximal Coordinates

A single rigid body in space can be described by a position $\boldsymbol{x} \in \mathbb{R}^3$, a velocity $\boldsymbol{v} \in \mathbb{R}^3$, an orientation (unit quaternion) $\boldsymbol{q} \in \mathbb{R}^4$, and an angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$. We can group these quantities into a vector

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{v} \\ \boldsymbol{q} \\ \boldsymbol{\omega} \end{bmatrix}. \tag{1}$$

One or more bodies can be subject to constraints $\boldsymbol{g}$ which can, for example, represent joints. One example of such a physical constraint is the revolute joint connecting links a and b in Fig. 2. Such constraints on two bodies can generally be written as implicit equations

$$\boldsymbol{g}(\boldsymbol{z}_\mathrm{a}, \boldsymbol{z}_\mathrm{b}) = \boldsymbol{0}. \tag{2}$$

For each body we can generally write the unconstrained dynamics as implicit equations in discrete time:

$$\boldsymbol{d}_0(\boldsymbol{z}_k, \boldsymbol{z}_{k+1}, \boldsymbol{u}_k) = \boldsymbol{0}, \tag{3}$$

where k is the current time step and $\boldsymbol{u}_k$ are control inputs to the system, assuming zero-order hold on the controls.

Constraints can be treated by introducing Lagrange multipliers (constraint forces) $\boldsymbol{\lambda}$ into the dynamics equations:

$$\boldsymbol{d}(\boldsymbol{z}_k, \boldsymbol{z}_{k+1}, \boldsymbol{u}_k, \boldsymbol{\lambda}_k) = \boldsymbol{d}_0 - G^\mathrm{T}\boldsymbol{\lambda}_k = \boldsymbol{0}, \tag{4}$$

where G is the Jacobian of constraints $\boldsymbol{g}$ with respect to $\boldsymbol{z}$.

The dynamics equations $\boldsymbol{d}$ and the constraints $\boldsymbol{g}$ of multiple bodies form a system of algebraic difference equations (ADEs).

B. Unconstrained Riccati Recursion

The linear-quadratic regulator (LQR) can be derived for four different scenarios: finite or infinite horizon, and discrete or continuous time. The Riccati recursion is one method to derive LQR for all four scenarios. The procedure is based on the dynamic programming principle (see [72] for details) and, while not always the most efficient method for each scenario, we will build on it for the remainder of the paper for clarity.

For discrete-time systems with a finite horizon, the Riccati recursion produces feedback gains at each time step. Continuing the recursion further leads to convergence of the feedback gains, which is simply the solution to the infinite-horizon problem. Furthermore, by taking the limit of the discrete time step as it goes to zero, we can recover the continuous time feedback gains for both finite-horizon and infinite-horizon settings.

We will provide the derivation of a very basic recursion here to illustrate the general procedure and familiarize the reader with our notation.

The discrete-time optimal control problem for LQR with states $\boldsymbol{z}_k$, controls $\boldsymbol{u}_k$, and weight matrices Q and R is:

$$\begin{aligned} \min \quad & J_0 := \frac{1}{2}\boldsymbol{z}_N^\mathrm{T} Q \boldsymbol{z}_N + \frac{1}{2}\sum_{k=0}^{N-1}\left(\boldsymbol{z}_k^\mathrm{T} Q \boldsymbol{z}_k + \boldsymbol{u}_k^\mathrm{T} R \boldsymbol{u}_k\right) \\ \text{s.t.} \quad & \boldsymbol{z}_{k+1} = A\boldsymbol{z}_k + B\boldsymbol{u}_k, \end{aligned} \tag{5}$$

where the equation $\boldsymbol{z}_{k+1} = A\boldsymbol{z}_k + B\boldsymbol{u}_k$ describes the unconstrained linear dynamics.

From the principle of optimality it follows that the cost-to-go function at time step k is given by

$$J_k = \frac{1}{2}\left(\boldsymbol{z}_k^\mathrm{T} Q \boldsymbol{z}_k + \boldsymbol{u}_k^\mathrm{T} R \boldsymbol{u}_k\right) + J_{k+1}^\star, \tag{6}$$

with the optimal cost-to-go $J_k^\star = \min J_k$.

Since we are looking for a linear state feedback law, we assume that the control $\boldsymbol{u}$ depends linearly on the state $\boldsymbol{z}$ which leads to the conclusion that generally $J_k^\star$ is of the form

$$J_k^\star = \frac{1}{2}\boldsymbol{z}_k^\mathrm{T} P_k \boldsymbol{z}_k. \tag{7}$$

Inserting (7) into (6) yields

$$\begin{aligned} J_k &= \frac{1}{2}\left(\boldsymbol{z}_k^\mathrm{T} Q \boldsymbol{z}_k + \boldsymbol{u}_k^\mathrm{T} R \boldsymbol{u}_k\right) + \frac{1}{2}\boldsymbol{z}_{k+1}^\mathrm{T} P_{k+1} \boldsymbol{z}_{k+1} \\ &= \frac{1}{2}\Big(\boldsymbol{z}_k^\mathrm{T} Q \boldsymbol{z}_k + \boldsymbol{u}_k^\mathrm{T} R \boldsymbol{u}_k \\ &\qquad + \left(A\boldsymbol{z}_k + B\boldsymbol{u}_k\right)^\mathrm{T} P_{k+1}\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k\right)\Big). \end{aligned} \tag{8}$$

The optimal control $\boldsymbol{u}_k^\star$ can be found by taking the derivative of J_k with respect to $\boldsymbol{u}_k$ and setting it equal to zero:

$$\nabla_{\boldsymbol{u}_k} J_k = R\boldsymbol{u}_k + B^\mathrm{T} P_{k+1}\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k\right) = 0. \tag{9}$$

Rearranging yields

$$\boldsymbol{u}_k^\star = -\left(R + B^\mathrm{T} P_{k+1} B\right)^{-1} B^\mathrm{T} P_{k+1} A \boldsymbol{z}_k \tag{10a}$$

$$= -K_k \boldsymbol{z}_k. \tag{10b}$$

Setting (7) equal to (8) after inserting (10) yields an update rule for P_k after some rearranging:

$$P_k = Q + K_k^\mathrm{T} R K_k + \bar{A}_k^\mathrm{T} P_{k+1} \bar{A}_k, \tag{11}$$

where

$$\bar{A}_k = A - BK_k, \tag{12}$$

and we know that $P_N = Q$.

IV. Physically Constrained Linear-Quadratic Optimal Control

We now derive the linear-quadratic regulator (LQR) for maximal-coordinate systems. Note that, while maximal-coordinate rigid-body systems provide our main motivation, the derivation applies to general ADE systems.

Given an implicit discrete-time dynamical system with implicit constraints

$$\boldsymbol{d}(\boldsymbol{z}_k, \boldsymbol{z}_{k+1}, \boldsymbol{u}_k, \boldsymbol{\lambda}_k) = \boldsymbol{0}, \tag{13a}$$

$$\boldsymbol{g}(\boldsymbol{z}_{k+1}) = \boldsymbol{0}, \tag{13b}$$

we can apply the implicit function theorem to obtain linearized dynamics and constraints

$$\boldsymbol{z}_{k+1} = A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k, \tag{14a}$$

$$G\boldsymbol{z}_{k+1} = \boldsymbol{0}. \tag{14b}$$

Assuming linearized discrete-time dynamics and constraints of the form (14), we can extend the procedure presented in Sec. III-B. The optimal control problem (cf. (5)) now includes additional constraints:

$$\begin{aligned} \min \quad & J_0 := \frac{1}{2}\boldsymbol{z}_N^{\mathrm{T}} Q \boldsymbol{z}_N + \frac{1}{2}\sum_{k=0}^{N-1} \left(\boldsymbol{z}_k^{\mathrm{T}} Q \boldsymbol{z}_k + \boldsymbol{u}_k^{\mathrm{T}} R \boldsymbol{u}_k\right) \\ \text{s.t.} \quad & \boldsymbol{z}_{k+1} = A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k, \\ & G\boldsymbol{z}_{k+1} = G\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right) = \boldsymbol{0}. \end{aligned} \tag{15}$$

The structures of the cost-to-go function (6) and the optimal cost-to-go (7) are unchanged. However, inserting (7) into (6) (cf. (8)) now includes the constrained update rule for $\boldsymbol{z}_{k+1}$:

$$\begin{aligned} J_k = \frac{1}{2}\Big(& \boldsymbol{z}_k^{\mathrm{T}} Q \boldsymbol{z}_k + \boldsymbol{u}_k^{\mathrm{T}} R \boldsymbol{u}_k \\ & + \left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right)^{\mathrm{T}} P_{k+1} \left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right)\Big), \end{aligned} \tag{16a}$$

$$G\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right) = \boldsymbol{0}. \tag{16b}$$

Instead of minimizing (16) like a classical quadratic program with additional multipliers, we propose a more direct method to reduce the computational burden. Assuming linearly independent constraints—a common requirement for constrained optimization—the constraints (16b) uniquely define the constraint forces $\boldsymbol{\lambda}_k(\boldsymbol{u}_k)$ as implicit functions of $\boldsymbol{u}_k$ (and $\boldsymbol{z}_k$). Therefore, we obtain the necessary optimality condition by taking the gradient of (16a) with respect to $\boldsymbol{u}_k$ (cf. (9)):

$$\nabla_{\boldsymbol{u}_k} J_k = R\boldsymbol{u}_k + D^{\mathrm{T}} P_{k+1}\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right) = \boldsymbol{0}, \tag{17}$$

where

$$D = \frac{\mathrm{d}\left(A\boldsymbol{z}_k + B\boldsymbol{u}_k + C\boldsymbol{\lambda}_k\right)}{\mathrm{d}\boldsymbol{u}_k} = B - C\left(GC\right)^{-1} GB, \tag{18}$$

and $C\frac{\partial \boldsymbol{\lambda}_k}{\partial \boldsymbol{u}_k} = -C\left(GC\right)^{-1} GB$ is obtained by applying the implicit function theorem to the implicit constraints (16b).

We now have a linear system of equations at each time step:

$$\begin{bmatrix} R + D^{\mathrm{T}} P_{k+1} B & D^{\mathrm{T}} P_{k+1} C \\ GB & GC \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_k \\ \boldsymbol{\lambda}_k \end{bmatrix} = -\begin{bmatrix} D^{\mathrm{T}} P_{k+1} \\ G \end{bmatrix} A\boldsymbol{z}_k, \tag{19}$$

from which we can obtain the optimal feedback laws for $\boldsymbol{u}_k$ and $\boldsymbol{\lambda}_k$ (cf. (10)) by inverting the matrix on the left hand side of (19):

$$\begin{bmatrix} \boldsymbol{u}_k^\star \\ \boldsymbol{\lambda}_k^\star \end{bmatrix} = -\begin{bmatrix} K_k \\ L_k \end{bmatrix} \boldsymbol{z}_k. \tag{20}$$

The matrix K contains the control gains whereas L models the mapping of constraint forces during the controller derivation. However, in the physical system, these forces are "automatically" applied. So, for the actual controller only K is retained.

As before, the recursive update rule for P_k (cf. (11)) is

$$P_k = Q + K_k^{\mathrm{T}} R K_k + \bar{A}_k^{\mathrm{T}} P_{k+1} \bar{A}_k, \tag{21}$$

where

$$\bar{A}_k = A - BK_k - CL_k. \tag{22}$$

Iterating this procedure until convergence results in the discrete-time infinite-horizon control law, while taking the limit as $\Delta t \to 0$ results in the continuous-time control laws.

V. Experiments

The choice of coordinates used to describe a mechanical system does not have an effect on the nonlinear dynamical behavior. However, different coordinate choices can produce considerably different results when linearizing a nonlinear system. In this section, we apply LQR feedback control in minimal and maximal coordinates, i.e. we either generate control inputs from minimal-coordinate states or from maximal-coordinate states. As the dynamics simulation tool we chose the *ConstrainedDynamics.jl* package based on [29] written in the programming language *Julia*. The code for our algorithm and all experiments is available at: https://github.com/janbruedigam/ConstrainedControl.jl.

The first part of the experimental analysis consists of determining the basins of attraction for minimal- and maximal-coordinate LQR for a selection of systems. The infinite-horizon time-invariant linear controllers are calculated for a stationary reference point to compare for which initial configurations of the systems the control laws are applicable.

The stationary analysis is followed by a time-varying LQR tracking example. Given a nominal state and control trajectory, we calculate a linear time-varying model and the corresponding time-varying LQR.

In order to make the analysis of minimal- and maximal-coordinate LQR comparable, we only apply weights to minimal-coordinate states, even when using maximal coordinates. Additionally, the cost functions are defined such that, in a linearized sense, they have the same value in a region near the desired state. So, assuming a mapping $\boldsymbol{f}$ from maximal coordinates $\boldsymbol{z}$ to minimal coordinates $\boldsymbol{c}$

$$\boldsymbol{c} = \boldsymbol{f}(\boldsymbol{z}), \tag{23}$$

the linearized relation between minimal- and maximal-coordinate cost at the reference point $\boldsymbol{z}^*$ is given as

$$\boldsymbol{c}^{\mathrm{T}} Q \boldsymbol{c} = \boldsymbol{z}^{\mathrm{T}} F^{\mathrm{T}} Q F \boldsymbol{z}, \tag{24}$$

where F is the Jacobian of $\boldsymbol{f}$ and we choose $\boldsymbol{f}$ such that $\boldsymbol{f}(\boldsymbol{z}^*) = \boldsymbol{0}$.

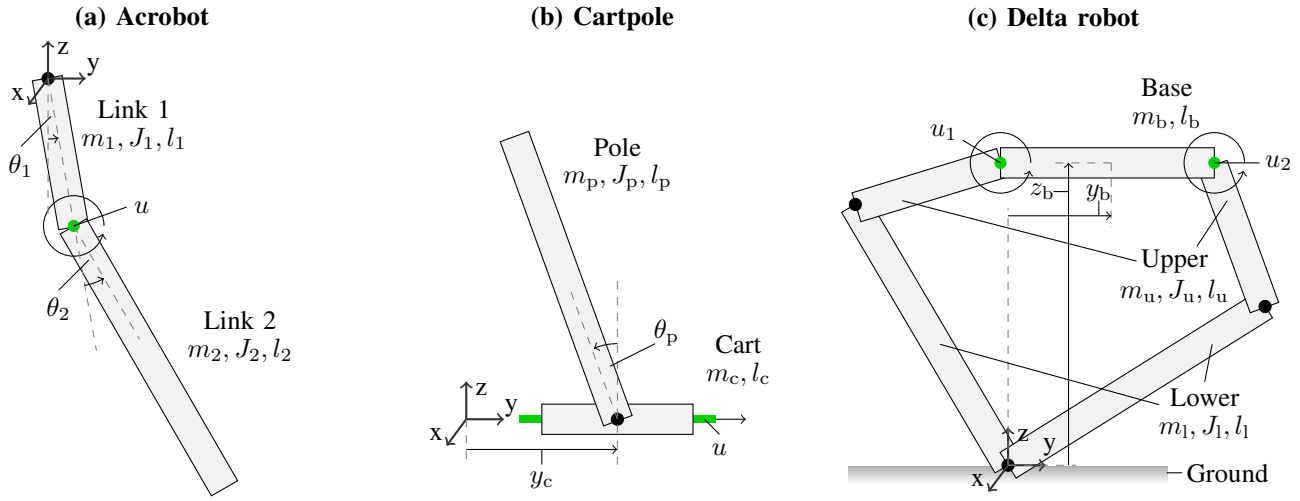

Fig. 3: Systems for the LQR basin of attraction analysis. Actuated joints highlighted in green. (a) Acrobot. (b) Cartpole. (c) Delta robot. The base angle of the delta robot is kept constant by parallel structures (not drawn).

	Link 1	Link 2	Cart	Pole	Base	Upper	Lower
m	1.0	1.0	0.5	1.0	0.71	0.5	1.0
J	0.084	0.334	-	0.084	-	0.011	0.084
l	1.0	2.0	0.5	1.0	0.71	0.5	1.0

TABLE I: Mechanical parameters of the systems' components.

A. Basin of Attraction Setup

For the analysis of the basins of attraction of each controlled system we chose initial conditions on a grid spanning the entire configuration space for initial velocities equal to zero. We simulated each system for 25 seconds with a time step of $\Delta t = 1$ms. For an initial condition to be counted as inside the basin of attraction, the system had to satisfy $\|\boldsymbol{z} - \boldsymbol{z}^*\| < 0.1$ within 25 seconds. We did not limit the magnitude of the control inputs applied to the systems. However, we stopped simulations if angular velocities exceeded 100π radians per second as the simulation no longer delivers reliable results for such excessive velocities. Velocities of this magnitude were indicative of instability.

For the experiments, we chose two systems commonly used for the analysis of control algorithms: the acrobot (Fig. 3a) and cartpole (Fig. 3b). Additionally, we investigated a 2-dimensional delta robot with closed kinematic loops shown in Fig. 3c. The control goal for all three systems was to stabilize an unstable equilibrium point at which the linearization was performed. As weight matrices Q and R we chose purely diagonal matrices. The mechanical properties for the components of all systems are stated in Table I.

1) Acrobot Setup: The acrobot is a double pendulum with a single actuator mimicking the dynamics of an acrobat holding on to a bar and generating force from his hips. The actuator is placed between the two links resulting in an underactuated but controllable system. Generally, inverted double pendulums are very nonlinear systems which makes them interesting from a control perspective. The control goal for the acrobot is to stabilize it in the upright position.

The desired state in minimal coordinates is $(\theta_1, \theta_2, \dot\theta_1, \dot\theta_2) = (\pi, 0, 0, 0)$ with nominal control input $u = 0$. The corresponding LQR weights are $Q = \mathrm{diag}(1, 1, 1, 1)$ and $R = 1$.

2) Cartpole Setup: The cartpole consists of pendulum (pole) mounted on a moving cart. The pendulum is connected by a passive joint. The actuated cart can only move along the y-axis. The control goal for the cartpole is to stabilize the inverted pendulum while keeping the cart near the origin.

The desired state in minimal coordinates is $(y_\mathrm{c}, \theta_\mathrm{p}, \dot y_\mathrm{c}, \dot\theta_\mathrm{p}) = (0, 0, 0, 0)$ with nominal control input $u = 0$. The corresponding LQR weights are $Q = \mathrm{diag}(1, 1, 1, 1)$ and $R = 1$.

3) Delta Robot Setup: Delta robots can be used to perform many tasks, including pick-and-place and milling. For the analysis we chose a 2-dimensional delta robot consisting of five links. The closed kinematic loops of the robot make a straight-forward application of minimal-coordinate LQR difficult. In contrast, the maximal-coordinate approach can handle this system without modifications to the algorithm. The system is driven by two actuators connected to the base resulting again in an underactuated but controllable system. For the control goal, we aimed at treating the delta robot as an inverted pendulum with the goal of stabilizing the system in an upright configuration close to the one depicted in Fig. 3c. The system can also be interpreted as an instance of a bipedal robot with feet pinned to the ground, underactuated articulated legs ("Lower" and "Upper"), and a torso ("Base") that is supposed to be balanced.

The desired state in minimal (reduced) coordinates is $(y_\mathrm{b}, z_\mathrm{b}, \dot y_\mathrm{b}, \dot z_\mathrm{b}) = (0, 1.061, 0, 0)$ with nominal control inputs $(u_1, u_2) = (6.788, -6.788)$ compensating gravity. The corresponding LQR weights are $Q = \mathrm{diag}(100, 100, 1, 1)$ and $R = 0.01$. Note that for all initial conditions we set the "legs" of the system to be pointing outwards, as depicted in Fig. 3c.

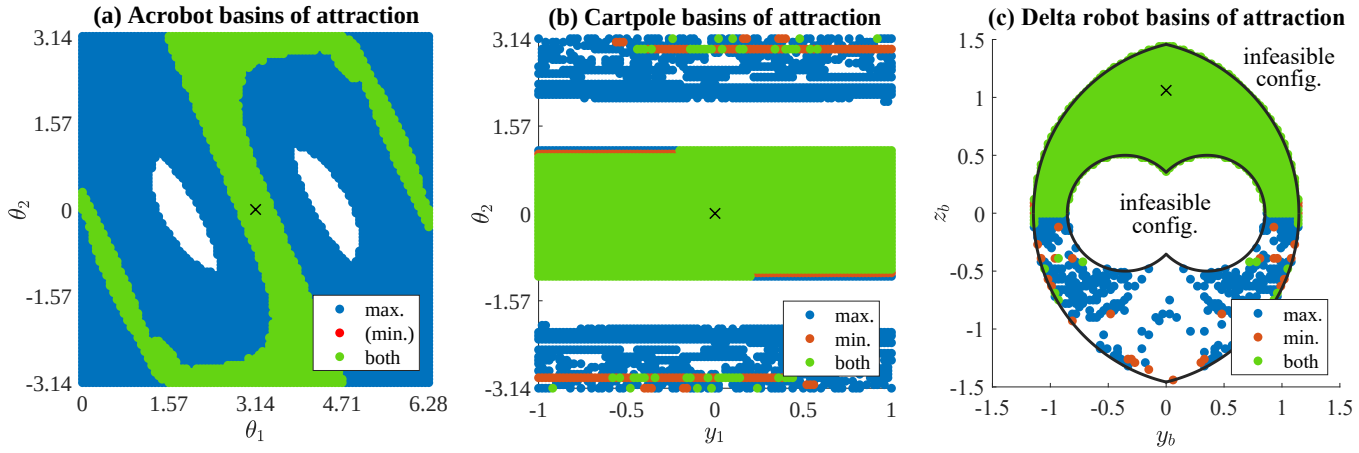

Fig. 4: Results for the LQR basin of attraction analysis. Maximal-coordinate basins (max.) shown in blue, minimal-coordinate basins (min.) in red, basins for both in green. The desired configurations are indicated by black crosses. (a) Results for acrobot. (b) Results for cartpole. (c) Results for the delta robot and indication of structurally infeasible initial configurations.

B. Basin of Attraction Results

The results for the simulations of the acrobot, cartpole, and delta robot are presented in Fig. 4. The regions for maximal coordinates are colored in blue, the regions for minimal coordinates in red. Regions for which both maximal- and minimal-coordinate LQR converged are colored in green.

1) Acrobot Results: The basin of attraction for the acrobot with a minimal-coordinate LQR shows "tilted" regions around the fully upright ($\theta_1 = \pi$, $\theta_2 = 0$) and fully hanging state ($\theta_1 = 0$, $\theta_2 = 0$) with connecting regions in between. A large area that is not stabilizable exists in between the tilted regions. In comparison, the stabilizable region for maximal-coordinate LQR has some resemblance in shape but is significantly larger.

2) Cartpole Results: The cartpole's basin of attraction for minimal-coordinate LQR consists of a stabilizable region for initial pendulum angle deviations within $\theta_2 \approx \pm\frac{1}{3}\pi$. The position of the cart does not have a significant effect on the stability. A few additional stabilizable initial conditions for hanging pendulum configurations ($\theta_2 \approx \pi$) were also detected. For angle deviations within $\theta_2 \approx \pm\frac{1}{3}\pi$ maximal-coordinate LQR yields almost the same region of attraction. Additionally, some more stabilizable points can be found for configurations around the hanging state within $\theta_2 \approx \pi \pm \frac{1}{3}\pi$.

3) Delta Robot Results: For z-positions above zero, both minimal- and maximal-coordinate LQR are capable of driving the delta robot to the desired state at any feasible initial configuration. The area below zero is less consistent, especially for minimal-coordinate LQR. In contrast, maximal-coordinate LQR is still able to drive the system into the desired position for several initial configurations.

C. Tracking LQR Setup

For the tracking of a nominal trajectory with time-varying LQR we chose a triple-pendulum cartpole system. The mechanical properties of cart and poles are the same as for the single-pendulum cartpole depicted in Fig. 3b and Table I. As reference trajectory we chose the swing-up of the triple pendulum into the upright position, while LQR is deployed to correct for noise and model errors during the swing-up.

The nominal state and control trajectory were calculated with the trajectory optimization tool *ALTRO* [73]. Subsequently, a time-varying model in minimal and maximal coordinates was calculated to obtain the respective LQR feedback laws. The weight matrices Q and R were kept constant for the entire trajectory and, as before, we applied cost only to minimal-coordinate states and matched the maximal-coordinate cost function to first order. The tracked state in minimal coordinates is $(y_\mathrm{c}, \theta_\mathrm{p1}, \theta_\mathrm{p2}, \theta_\mathrm{p3}, \dot{y}_\mathrm{c}, \dot{\theta}_\mathrm{p1}, \dot{\theta}_\mathrm{p2}, \dot{\theta}_\mathrm{p3})$ and the cart's control input is u. The corresponding LQR weights are $Q = \mathrm{diag}(10, 10, 10, 10, 1, 1, 1, 1)$ and $R = 0.1$.

Three sources of disturbances were applied to the nominal system: white Gaussian noise ($\mu = 0$, $\sigma = 2$) was added to the control input, the prismatic and revolute joints were impacted by viscous friction ($k = 0.1$), and the masses and inertias of cart and poles were distorted by multiplying uniformly distributed ($[0.9, 1.1]$) perturbation factors. Additionally, the initial configurations of cart and poles were uniformly distributed ($[-0.1, 0.1]$) around the nominal initial configurations. For each of the two controllers, 1000 runs were observed.

D. Tracking LQR Results

The results for the triple-pendulum cartpole tracking experiment are presented as cost per time step in Fig. 5a and accumulated cost in Fig. 5b.

Overall, the tracking of the perturbed system with minimal- and maximal-coordinate LQR shows some similarities. For the first 7 seconds, energy is supplied to the system by swinging back and forth. During this period, both controllers manage to keep the incurred cost low. After 7 seconds, the final swing-up into the upright position takes place.

An overall increase in cost per time step during the second phase becomes visible in Fig. 5a, and especially the minimal-

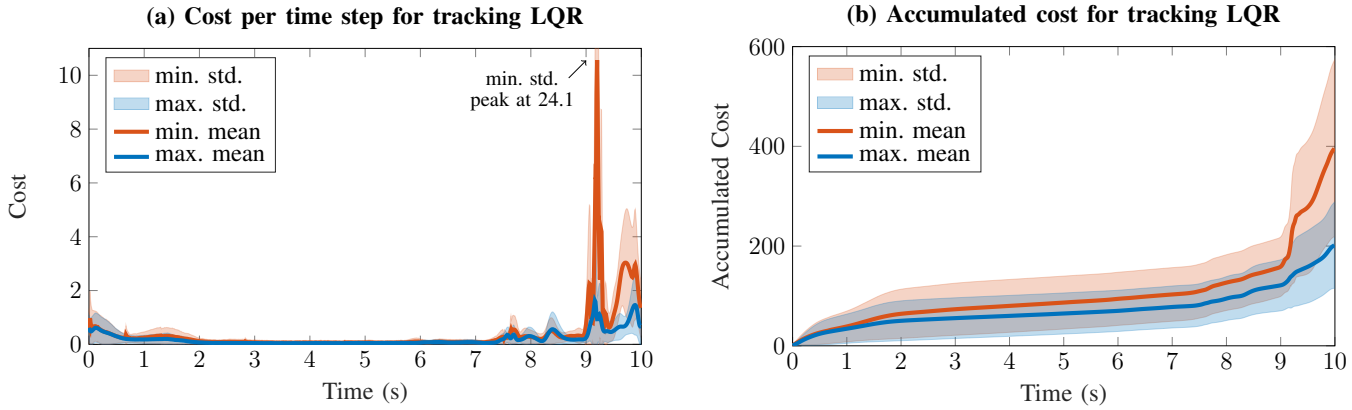

Fig. 5: Results for the tracking of the perturbed triple-pendulum cartpole swing-up with LQR. Maximal-coordinate mean cost (max. mean) and standard deviation (max. std.) in blue, minimal-coordinate mean cost (min. mean) and standard deviation (min. std.) in red. (a) Cost and standard deviation per time step. (b) Accumulated cost and corresponding standard deviation.

coordinate controller incurs increased cost. Additionally, a significantly higher cost variance for minimal-coordinate tracking can be observed in the final swing-up phase, when both systems diverge further from the nominal trajectory. Similarly, the accumulated costs for minimal- and maximal-coordinate LQR start to diverge visibly during the final swing-up phase in Fig. 5b. Additionally, for the entire run the cost variance of minimal-coordinate tracking is larger than the maximal-coordinate cost variance.

VI. Discussion

As expected, both minimal- and maximal-coordinate LQR are capable of stabilizing the systems under investigation from initial configurations close the reference point or trajectory. However, for larger deviations from the nominal trajectory, maximal-coordinate LQR appears to be more robust in our tests. This interpretation stems from the larger basins of attraction for acrobot, cartpole, and the delta robot, as well as from the lower cost and variance for tracking LQR.

One possible explanation for the higher robustness of maximal-coordinate LQR could lie in a larger region of validity of the linearization. This effect might relate to similar findings on advantages of higher-dimensional linear models [2]–[4]. The LQR tracking problem exhibits similar behavior: During the first phase, both systems only deviated slightly from the reference trajectory and, therefore, both linearizations were reasonably accurate. In the second phase, with overall larger deviations from the nominal trajectory, minimal-coordinate LQR performed increasingly worse.

It is important to note, however, that we only analyzed a small number of systems and settings and further analysis is necessary to determine the general validity of our results. It is possible that, for certain nonlinear systems or certain regions, minimal-coordinate LQR could outperform maximal-coordinate LQR.

VII. Conclusions

We presented an extension of the linear-quadratic regulator to systems with physical constraints represented in maximal coordinates. Our derivation for such systems retains linearized constraints explicitly and directly incorporates them into the feedback gain calculations. Since the LQR derivation is built on the standard Riccati recursion, extensions such as state or input constraints are applicable to our derivation as well.

The LQR controller can be directly deployed to stabilize systems around nominal trajectories. Selected experiments on nonlinear systems suggest preferable performance of maximal-coordinate LQR compared to minimal-coordinate LQR for comparable cost functions. This conclusion is based on the larger basins of attraction of maximal-coordinate LQR for the analyzed systems which generally points at a more robust controller.

Time-varying tracking LQR shows comparable behavior for minimal- and maximal-coordinate LQR when close to the nominal trajectory. However, maximal-coordinate control appears to be less sensitive to larger deviations.

Overall, the results make a strong case for using maximal coordinates for linear-quadratic control, even for unconstrained systems. The ease of incorporating additional physical constraints during the modeling process adds to this conclusion.

An interesting direction for future work could be to exploit sparsity when deriving the constrained LQR controller with additional Lagrange multipliers instead of, or in addition to, applying the more direct approach proposed in this paper.

The maximal-coordinate LQR approach could also be extended to trajectory optimization methods such as iterative LQR (iLQR) or model predictive control (MPC) schemes based on LQR. Future investigation is needed to determine if the larger basins of attraction and enhanced robustness hold for more complex systems, and to better understand the theory behind these observations.

REFERENCES

[1] D. Baraff, "Linear-time dynamics using Lagrange multipliers," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, pp. 137–146, ACM Press, 1996.

[2] D. Bruder, B. Gillespie, C. Remy, and R. Vasudevan, "Modeling and Control of Soft Robots Using the Koopman Operator and Model Predictive Control," in *Robotics: Science and Systems XV*, Robotics: Science and Systems Foundation, 2019.

[3] I. Abraham, G. de la Torre, and T. Murphey, "Model-Based Control Using Koopman Operators," in *Robotics: Science and Systems XIII*, Robotics: Science and Systems Foundation, 2017.

[4] H. Suh and R. Tedrake, "The Surprising Effectiveness of Linear Models for Visual Foresight in Object Pile Manipulation," in *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, Springer, 2020.

[5] D. Bender and A. Laub, "The linear-quadratic optimal regulator for descriptor systems," *IEEE Transactions on Automatic Control*, vol. 32, no. 8, pp. 672–688, 1987.

[6] P. Kunkel and V. Mehrmann, "The linear quadratic optimal control problem for linear descriptor systems with variable coefficients," *Mathematics of Control, Signals, and Systems*, vol. 10, no. 3, pp. 247–264, 1997.

[7] J. Mills and A. Goldenberg, "Force and position control of manipulators during constrained motion tasks," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 1, pp. 30–46, 1989.

[8] K. Khayati, P. Bigras, and L.-A. Dessaint, "A Multistage Position/Force Control for Constrained Robotic Systems With Friction: Joint-Space Decomposition, Linearization, and Multiobjective Observer/Controller Synthesis Using LMI Formalism," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 5, pp. 1698–1712, 2006.

[9] M. Townsend and T. Tsai, "On Optimal Control Laws for a Class of Constrained Dynamical Systems (With Application to Control of Bipedal Locomotion)," *Journal of Dynamic Systems, Measurement, and Control*, vol. 99, no. 2, pp. 98–102, 1977.

[10] M. Mistry, J. Buchli, and S. Schaal, "Inverse dynamics control of floating base systems using orthogonal decomposition," in *2010 IEEE International Conference on Robotics and Automation*, pp. 3406–3412, IEEE, 2010.

[11] M. Posa, S. Kuindersma, and R. Tedrake, "Optimization and stabilization of trajectories for constrained dynamical systems," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1366–1373, IEEE, 2016.

[12] L.-S. You and B.-S. Chen, "Tracking control designs for both holonomic and non-holonomic constrained mechanical systems: a unified viewpoint," *International Journal of Control*, vol. 58, no. 3, pp. 587–612, 1993.

[13] G. Liu and Z. Li, "A unified geometric approach to modeling and control of constrained mechanical systems," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 574–587, 2002.

[14] N. McClamroch, "Feedback stabilization of control systems described by a class of nonlinear differential-algebraic equations," *Systems & Control Letters*, vol. 15, no. 1, pp. 53–60, 1990.

[15] A. Kumar and P. Daoutidis, "Feedback control of nonlinear differential-algebraic-equation systems," *AIChE Journal*, vol. 41, no. 3, pp. 619–636, 1995.

[16] A. Kumar and P. Daoutidis, "Feedback regularization and control of nonlinear differential-algebraic-equation systems," *AIChE Journal*, vol. 42, no. 8, pp. 2175–2198, 1996.

[17] H. Krishnan and N. Mcclamroch, "Tracking in nonlinear differential-algebraic control systems with applications to constrained robot systems," *Automatica*, vol. 30, no. 12, pp. 1885–1897, 1994.

[18] P. Kunkel and V. Mehrmann, "Optimal control for unstructured nonlinear differential-algebraic equations of arbitrary index," *Mathematics of Control, Signals, and Systems*, vol. 20, no. 3, pp. 227–269, 2008.

[19] M. Bell and R. Sargent, "Optimal control of inequality constrained DAE systems," *Computers & Chemical Engineering*, vol. 24, no. 11, pp. 2385–2404, 2000.

[20] M. Gerdts, "Direct Shooting Method for the Numerical Solution of Higher-Index DAE Optimal Control Problems," *Journal of Optimization Theory and Applications*, vol. 117, no. 2, pp. 267–294, 2003.

[21] G. Kurina and R. März, "Feedback Solutions of Optimal Control Problems with DAE Constraints," *SIAM Journal on Control and Optimization*, vol. 46, no. 4, pp. 1277–1298, 2007.

[22] R. März, "On linear differential-algebraic equations and linearizations," *Applied Numerical Mathematics*, vol. 18, no. 1-3, pp. 267–292, 1995.

[23] V. Sohoni and J. Whitesell, "Automatic Linearization of Constrained Dynamical Models," *Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 108, no. 3, pp. 300–304, 1986.

[24] J. Kang, S. Bae, J. Lee, and T. Tak, "Force Equilibrium Approach for Linearization of Constrained Mechanical System Dynamics," *Journal of Mechanical Design*, vol. 125, no. 1, pp. 143–149, 2003.

[25] D. Negrut and J. Ortiz, "A Practical Approach for the Linearization of the Constrained Multibody Dynamics Equations," *Journal of Computational and Nonlinear Dynamics*, vol. 1, no. 3, pp. 230–239, 2006.

[26] F. González, P. Masarati, J. Cuadrado, and M. Naya, "Assessment of Linearization Approaches for Multibody Dynamics Formulations," *Journal of Computational and Nonlinear Dynamics*, vol. 12, no. 4, p. 041009, 2017.

[27] E. Johnson, J. Schultz, and T. Murphey, "Structured Linearization of Discrete Mechanical Systems for Analysis and Optimal Control," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 140–152, 2015.

[28] T. Fan and T. Murphey, "Structured linearization of discrete mechanical systems on Lie groups: A synthesis of analysis and control," in *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 1092–1099, IEEE, 2015.

[29] J. Brüdigam and Z. Manchester, "Linear-Time Variational Integrators in Maximal Coordinates," in *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, Springer, 2020.

[30] D.-S. Bae and E. Haug, "A Recursive Formulation for Constrained Mechanical System Dynamics: Part I. Open Loop Systems," *Mechanics of Structures and Machines*, vol. 15, no. 3, pp. 359–382, 1987.

[31] J. Bender, K. Erleben, and J. Trinkle, "Interactive Simulation of Rigid Body Dynamics in Computer Graphics: Interactive Rigid Body Simulation," *Computer Graphics Forum*, vol. 33, no. 1, pp. 246–270, 2014.

[32] A. Knemeyer, S. Shield, and A. Patel, "Minor Change, Major Gains: The Effect of Orientation Formulation on Solving Time for Multi-Body Trajectory Optimization," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5331–5338, 2020.

[33] Y. Zhao and S. BeMent, "Kinematics, dynamics and control of wheeled mobile robots," in *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pp. 91–96, IEEE Comput. Soc. Press, 1992.

[34] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*, pp. 2520–2525, IEEE, 2011.

[35] Z. Peng, J. Wang, and J. Wang, "Constrained Control of Autonomous Underwater Vehicles Based on Command Optimization and Disturbance Estimation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 5, pp. 3627–3635, 2019.

[36] M. Lewis and K.-H. Tan, "High Precision Formation Control of Mobile Robots Using Virtual Structures," *Autonomous Robots*, vol. 4, no. 4, pp. 387–403, 1997.

[37] P. Deshpande, P. Menon, C. Edwards, and I. Postlethwaite, "A distributed control law with guaranteed LQR cost for identical dynamically coupled linear systems," in *Proceedings of the 2011 American Control Conference*, pp. 5342–5347, IEEE, 2011.

[38] R. Paul, "Manipulator Cartesian Path Control," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 11, pp. 702–711, 1979.

[39] C.-Y. Kuo and S.-P. Wang, "Robust position control of robotic manipulator in Cartesian coordinates," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 5, pp. 653–659, 1991.

[40] J. Funda, R. Taylor, B. Eldridge, S. Gomory, and K. Gruben, "Constrained Cartesian motion control for teleoperated surgical robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 453–465, 1996.

[41] W. Chen, I.-M. Chen, W. Lim, and G. Yang, "Cartesian coordinate control for redundant modular robots," in *2000 IEEE International Conference on Systems, Man and Cybernetics (SMC).*, vol. 5, pp. 3253–3258, IEEE, 2000.

[42] V. Lippiello and F. Ruggiero, "Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3768–3773, IEEE, 2012.

[43] M. Spong, "The swing up control problem for the Acrobot," *IEEE Control Systems*, vol. 15, no. 1, pp. 49–55, 1995.

[44] S. Brown and K. Passino, "Intelligent Control for an Acrobot," *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 209–248, 1997.

[45] L. Wiklendt, S. Chalup, and R. Middleton, "A small spiking neural network with LQR control applied to the acrobot," *Neural Computing and Applications*, vol. 18, no. 4, pp. 369–375, 2009.

[46] S. Lee, M. Eom, and D. Chwa, "Robust swing up and balancing control of the acrobot based on a disturbance observer," in *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, pp. 48–53, IEEE, 2015.

[47] M. Shehu, M. Ahmad, A. Shehu, and A. Alhassan, "LQR, double-PID and pole placement stabilization and tracking control of single link inverted pendulum," in *2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, pp. 218–223, IEEE, 2015.

[48] W. Luhao and S. Zhanshi, "LQR-Fuzzy Control for Double Inverted Pendulum," in *2010 International Conference on Digital Manufacturing & Automation*, pp. 900–903, IEEE, 2010.

[49] E. Vinodh Kumar and J. Jerome, "Robust LQR Controller Design for Stabilizing and Trajectory Tracking of Inverted Pendulum," *Procedia Engineering*, vol. 64, pp. 169–178, 2013.

[50] Z. Manchester and S. Kuindersma, "DIRTREL: Robust Trajectory Optimization with Ellipsoidal Disturbances and LQR Feedback," in *Robotics: Science and Systems XIII*, Robotics: Science and Systems Foundation, 2017.

[51] R. Bordalba, J. Porta, and L. Ros, "A Singularity-Robust LQR Controller for Parallel Robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 270–276, IEEE, 2018.

[52] P. Karimi Eskandary, B. Belzile, and J. Angeles, "Trajectory-Planning and Normalized-Variable Control for Parallel Pick-and-Place Robots," *Journal of Mechanisms and Robotics*, vol. 11, no. 3, p. 031001, 2019.

[53] Y. Yun and Y. Li, "Design and analysis of a novel 6-DOF redundant actuated parallel robot with compliant hinges for high precision positioning," *Nonlinear Dynamics*, vol. 61, no. 4, pp. 829–845, 2010.

[54] Y. Yun and Y. Li, "Modeling and Control Analysis of a 3-PUPU Dual Compliant Parallel Manipulator for Micro Positioning and Active Vibration Isolation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 134, no. 2, p. 021001, 2012.

[55] E. Akgul, M. Mutlu, A. Saranli, and Y. Yazicioglu, "A comparative evaluation of adaptive and non-adaptive Sliding Mode, LQR & PID control for platform stabilization," in *2012 IEEE International Conference on Control Applications*, pp. 1547–1552, IEEE, 2012.

[56] W. Nan, X. Tang, and B. Song, "A new automatic motion planning algorithm for a 4-degree-of-freedom parallel kinematic manipulator based on the centre sphere method," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 229, no. 1_suppl, pp. 64–77, 2015.

[57] J. Fabian, C. Monterrey, and R. Canahuire, "Trajectory tracking control of a 3 DOF delta robot: a PD and LQR comparison," in *2016 IEEE XXIII International Congress on Electronics, Electrical Engineering and Computing (INTERCON)*, pp. 1–5, IEEE, 2016.

[58] N. Dayana Salim, D. Derawi, S. Abdullah, S. Mazlan, and H. Zamzuri, "PID plus LQR attitude control for hexarotor MAV in indoor environments," in *2014 IEEE International Conference on Industrial Technology (ICIT)*, pp. 85–90, IEEE, 2014.

[59] F. Lin, Z. Lin, and X. Qiu, "LQR controller for car-like robot," in *2016 35th Chinese Control Conference (CCC)*, pp. 2515–2518, IEEE, 2016.

[60] A. Gutierrez-Giles, F. Ruggiero, V. Lippiello, and B. Siciliano, "Non-prehensile Manipulation of an Underactuated Mechanical System With Second-Order Nonholonomic Constraints: The Robotic Hula-Hoop," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1136–1143, 2018.

[61] A. Tunik, "Simplified Path Tracking Control Laws for Quad-rotor Considered as Nonholonomic System," in *2018 IEEE 5th International Conference on Methods and Systems of Navigation and Motion Control (MSNMC)*, pp. 83–89, IEEE, 2018.

[62] J. Chen, W. Zhan, and M. Tomizuka, "Autonomous Driving Motion Planning With Constrained Iterative LQR," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 244–254, 2019.

[63] S. Mason, N. Rotella, S. Schaal, and L. Righetti, "Balancing and walking using full dynamics LQR control with contact constraints," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pp. 63–68, IEEE, 2016.

[64] S. Savin, S. Jatsun, and L. Vorochaeva, "Modification of constrained LQR for control of walking in-pipe robots," in *2017 Dynamics of Systems, Mechanisms and Machines (Dynamics)*, pp. 1–6, IEEE, 2017.

[65] P. Scokaert and J. Rawlings, "Constrained linear quadratic regulation," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1163–1169, 1998.

[66] J. Mare and J. De Doná, "Solution of the input-constrained LQR problem using dynamic programming," *Systems & Control Letters*, vol. 56, no. 5, pp. 342–348, 2007.

[67] G. Mancuso and E. Kerrigan, "Solving constrained LQR problems by eliminating the inputs from the QP," in *IEEE Conference on Decision and Control and European Control Conference*, pp. 507–512, IEEE, 2011.

[68] H.-N. Nguyen and P.-O. Gutman, "Fast Constrained LQR Based on MPC With Linear Decomposition," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, pp. 2585–2590, 2016.

[69] L. Ferranti, G. Stathopoulos, C. Jones, and T. Keviczky, "Constrained LQR using online decomposition techniques," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 2339–2344, IEEE, 2016.

[70] G. Stathopoulos, M. Korda, and C. Jones, "Solving the Infinite-Horizon Constrained LQR Problem Using Accelerated Dual Proximal Methods," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1752–1767, 2017.

[71] F. Laine and C. Tomlin, "Efficient Computation of Feedback Control for Equality-Constrained LQR," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6748–6754, IEEE, 2019.

[72] D. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, 1995.

[73] T. Howell, B. Jackson, and Z. Manchester, "ALTRO: A Fast Solver for Constrained Trajectory Optimization," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7674–7679, IEEE, 2019.