Memorias del Congreso Nacional de Control Automático
ISSN: 2594-2492

A novel LMI computed-torque technique for stabilization of underactuated systems [⋆]

Jesús Alonso Díaz [*] **Miguel Bernal** [*]

[*] *Department of Electrical and Electronics Engineering, Sonora Institute of Technology, Ciudad Obregón, Sonora, Mexico (e-mail: miguel.bernal@itson.edu.mx)*

Abstract: A novel computed-torque technique for stabilization of a class of underactuated robot manipulators is proposed in this paper. Instead of obtaining the equivalent of a linear error system by model inversion as usually done when all the actuators are available, it is shown that these plants are amenable to a nonlinear form of the error system by only employing the available torques. The resulting nonlinear system can thus be exactly rewritten as a polytope, based on which a control law can be designed in the form of parallel distributed compensation via linear matrix inequalities. The whole scheme has been successfully applied both in simulation and real-time implementation to an inverted pendulum on a cart.

Keywords: Computed torque, Underactuated system, Lyapunov stability, Linear matrix inequality.

1. INTRODUCTION

Usually, the control of nonlinear plants is achieved by linearization around an operating point, as linear methods such as pole placement (Ogata, 2001; Richardson et al., 2005) or solving the Ricatti equation (Chen, 1984) are easy to apply and mostly effective. Naturally, this comes at the price of poor performance when the plant is driven too far from the nominal values, a situation that has been tackled by the wide range of model-based nonlinear control techniques such as geometric control (Isidori, 1995), sliding modes (Shtessel et al., 2013), passivity-based techniques (Ortega et al., 1998), etcetera. Nevertheless, these approaches are usually too involved for being of practical use.

A simpler, yet effective approach for control of robot manipulators is the computed-torque technique, which is based on model inversion to cancel out the system nonlinearities as to obtain a linear error system that can be driven to zero by ordinary linear tools (Lewis et al., 2003). This accomplishment is linked to the structure of Lagrange-Euler dynamics of robotic manipulators, since model inversion is possible thanks to the positive-definiteness of the inertia matrix (Marion, 1965). Thus, if the arm parameters are known with enough precision, computed-torque control may provide a good performance for stabilization as well as trajectory tracking in real-time applications (Khosla and Kanade, 1989), provided an adequate path is generated (Lee and Chen, 1983; Shin and McKay, 1985).

Problem statement: Traditional computed-torque control requires the presence of all generalized torques in the Lagrange-Euler equations, i.e., it is assumed to be fully actuated; thus, its application to underactuated manipulators has been hindered, prompting few solutions by the control community: switching variations (Udawatta et al., 2002), genetic algorithms (Udawatta et al., 2003b), and fuzzy logic (Udawatta et al., 2003a), among others. Importantly, most of these refinements of the computed-torque technique yield the systematicness of the traditional approach to heuristic solutions from the field of soft computing.

Contribution: In this paper, a novel solution for computed-torque control of underactuated robotic manipulators is proposed; it is based on obtaining a nonlinear error system instead of a linear one by means of the available actuators. Once this partial model inversion is performed, the nonlinear error system is exactly rewritten as a polytope (convex interpolation) of linear systems via the sector nonlinearity approach (Taniguchi et al., 2001); based on the polytope vertices, a controller is designed via parallel distributed compensation (PDC) to complete the feedforward signal (Tanaka and Wang, 2001). The PDC gains are obtained by solving a set of linear matrix inequalities (Boyd et al., 1994) via commercially available software (Gahinet et al., 1995). Input/output saturation limits, decay rate, and other performance measures can be easily incorporated as LMIs in the controller design, which may help providing a controller that takes into account real-time specifications (Duan and Yu, 2013). In order to test the proposal, an inverted pendulum on a cart is considered, which is a very well-known underactuated mechanism with 2 degrees of freedom (DOF), 1 pole, and 2 joints, one translational and another rotational (Angeli, 2001). The actuator only displaces the cart along the translational axis to stabilize the pole in its upright position. Both simulation and real-time results are presented to illustrate the usefulness of the proposal on a Feedback plant (Instruments, 2015).

Organization: This work is organized as follows: section 2 introduces the ordinary computed-torque technique in order to point out the difficulty of applying it to unde-

[⋆] This work has been supported by the ITSON PROFAPI Project 2019-0002 for the PRODEP CA-18.

ractuated manipulators; section 3 presents our solution to the previous problem, first for the feedforward part, then for stabilizing the nonlinear error system trough convex optimization techniques; section 4 presents the simulation and real-time implementation results of the designed controller along with a discussion on related issues; concluding remarks are given in section 5.

2. PROBLEM STATEMENT

Consider a robotic manipulator consisting of rigid beams along with translational and rotational joints, grouped in a generalized coordinate vector $\bar{q} \in \mathbb{R}^n$, amenable to the following Lagrange-Euler form:

$$M(\bar{q})\ddot{\bar{q}}(t) + N(\bar{q}, \dot{\bar{q}}) = \bar{\tau}(t), \qquad (1)$$

where $M(\bar{q})$ is the inertia matrix, $N(\bar{q}, \dot{\bar{q}})$ gathers the Coriolis, friction, and gravity vectors, and $\bar{\tau}$ is the generalized torque vector. If fully actuated, every entry in $\bar{\tau}$ will be available; otherwise, some of these entries might be zero.

Given a desired trajectory $\bar{q}_d(t)$ whose first and second time derivatives are available, we define $e(t) = \bar{q}_d(t) - \bar{q}(t)$ as the error vector, from which it is clear that

$$-M(\bar{q})\left(\ddot{e}(t) - \ddot{\bar{q}}_d(t)\right) + N(\bar{q}, \dot{\bar{q}}) = \bar{\tau}. \qquad (2)$$

Thus, defining $u(t) \equiv \ddot{e}(t)$ and making the proper simplifications, we have the following linear state representation of the error system:

$$\underbrace{\begin{bmatrix} \dot{e}(t) \\ \ddot{e}(t) \end{bmatrix}}_{\dot{\bar{e}}(t)} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}}_{\bar{e}(t)} + \begin{bmatrix} 0 \\ I \end{bmatrix} u(t), \qquad (3)$$

with $\bar{e} \in \mathbb{R}^{2n}$, for which a control law u can be designed to guarantee $\bar{e}(t) \to 0$ as $t \to \infty$ by any linear control technique such as pole placement (Chen, 1984) or LMI formulations (Duan and Yu, 2013). This control law is of course should be transferred to τ which, in addition, has the inverse dynamics part incorporated, i.e.,

$$\bar{\tau}(t) = M(\bar{q})(\ddot{\bar{q}}_d(t) - u) + N(\bar{q}, \dot{\bar{q}}), \qquad (4)$$

where the dependency on the accuracy of the model is evident.

Problem statement: If some of the signals in $\bar{\tau}$ are not available, i.e., if the robotic manipulator is underactuated, the right-hand side of (4) may most likely produce an expression incompatible with the fact that some entries of $\bar{\tau}$ are fixed to 0. The goal of the next section is to provide a solution to this issue for stabilization of the nonlinear system (1) via the computed-torque technique above, i.e., $\bar{q}_d(t) \equiv 0$, $\forall t \geq 0$.

3. MAIN RESULTS

A strong limitation of the classical computed-torque technique is that the designer is obliged to adopt a linear error system such as (3), thus precluding the possibility of using nonlinear terms for achieving control objectives. This is particularly critical for underactuated systems, where the control law is supposed to deal with a variety of objectives. In this section, we will show a methodology to judiciously include nonlinear terms in the error system as to deal with the underactuated characteristics of the plant, for stabilization purposes.

To begin with, in order to distinguish the available inputs from those which are not available, consider the following rewriting of the plant model (1):

$$M(\bar{q})\ddot{\bar{q}}(t) + N(\bar{q}, \dot{\bar{q}}) = \bar{E}\bar{\tau}(t), \qquad (5)$$

where $\bar{E} \in \mathbb{R}^{n \times n}$ is defined in terms of its i-j entries as follows

$$\bar{E}_{ij} = \begin{cases} 0, \text{ if } i \neq j \text{ or } i = j \text{ with } \bar{\tau}_i \text{ not available,} \\ 1, \text{ if } i = j \text{ and } \bar{\tau}_i \text{ is available.} \end{cases}$$

If the number of equations in (5) with available torque $\bar{\tau}_i$ is m, we can write m equations as

$$M_{i,*}(\bar{q})\ddot{\bar{q}}(t) + N_i(\bar{q}, \dot{\bar{q}}) = \bar{\tau}_i(t), \qquad (6)$$

and $n - m$ equations as

$$M_{i,*}(\bar{q})\ddot{\bar{q}}(t) + N_i(\bar{q}, \dot{\bar{q}}) = 0, \qquad (7)$$

where the subscript $\{i, *\}$ stands for the i-th row of the corresponding matrix and the subscript $\{i\}$ for the i-th entry of the corresponding vector.

From the aforementioned discussion, it is clear that a suitable rewriting of the system (5) will split actuated $q_1 \in \mathbb{R}^m$ from non-actuated entries $q_2 \in \mathbb{R}^{n-m}$ as follows:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \qquad (8)$$

where arguments of matrices M_{ij} and N_i have been omitted for brevity, their dimensions can be straightforwardly deduced from the splitting; naturally $\tau \in \mathbb{R}^m$ groups the actuated entries of the original $\bar{\tau}$.

Correspondingly, the transformed desired trajectory for stabilization will be $q_d(t) = 0$ as $\bar{q}_d(t) = 0$, which means the tracking error $e = \begin{bmatrix} e_1^T & e_2^T \end{bmatrix}^T$ is such that $e_1 = q_{d1} - q_1 = -q_1$ and $e_2 = q_{d2} - q_2 = -q_2$. Thus, taking into account equation (8) we have that the error system is:

$$\begin{aligned} M_{11}\ddot{e}_1 + M_{12}\ddot{e}_2 &= N_1 - \tau \\ M_{21}\ddot{e}_1 + M_{22}\ddot{e}_2 &= N_2, \end{aligned}$$

from which $\ddot{e}_2$ can be solved from the second equation as $\ddot{e}_2 = M_{22}^{-1}\left(N_2 - M_{21}\ddot{e}_1\right)$ due to the positive-definiteness of the inertia matrix, and substituted in the first one yielding

$$\begin{aligned} & M_{11}\ddot{e}_1 + M_{12}M_{22}^{-1}\left(N_2 - M_{21}\ddot{e}_1\right) = N_1 - \tau \\ \iff & \left(M_{12}M_{22}^{-1}M_{21} - M_{11}\right)\ddot{e}_1 + N_1 - M_{12}M_{22}^{-1}N_2 = \tau, \end{aligned}$$

where the last equation has the same structure as (2), which suggests that choosing $u = \ddot{e}_1$ may help writing the error system in a simpler manner as $\ddot{e}_1 = u$, $\ddot{e}_2 = M_{22}^{-1}\left(N_2 - M_{21}u\right)$, from which τ keeps the following relationship with u:

$$\tau = \left(M_{12}M_{22}^{-1}M_{21} - M_{11}\right)u + N_1 - M_{12}M_{22}^{-1}N_2. \qquad (9)$$

Recalling that for stabilization $M_{22}^{-1}N_2$ in $\ddot{e}_2$ depends on q_1, q_2, $\dot{q}_1$, and $\dot{q}_2$, which in turn are equivalent (up to a sign change) to e_1, e_2, $\dot{e}_1$, and $\dot{e}_2$, respectively, it is always possible to write $M_{22}^{-1}N_2$ as a sum of nonlinear expressions multiplied by these four signals; in other words, the nonlinear expression $M_{22}^{-1}N_2$ admits a right-hand factorization of $\bar{e} = \begin{bmatrix} e_1^T & e_2^T & \dot{e}_1^T & \dot{e}_2^T \end{bmatrix}^T$ Taniguchi et al. (2001). Thus, a nonlinear version of the error system (3) can now be written for the stabilization problem as:

$$\begin{bmatrix}\dot{e}_1\\ \dot{e}_2\\ \ddot{e}_1\\ \ddot{e}_2\end{bmatrix} = \underbrace{\begin{bmatrix}0&0&I&0\\0&0&0&I\\0&0&0&0\\A_{41}&A_{42}&A_{43}&A_{44}\end{bmatrix}}_{A(\bar{e})}\begin{bmatrix}e_1\\e_2\\\dot{e}_1\\\dot{e}_2\end{bmatrix}+\underbrace{\begin{bmatrix}0\\0\\I\\-M_{22}^{-1}M_{21}\end{bmatrix}}_{B(\bar{e})}u, \tag{10}$$

where, again, arguments for matrices A_{4j}, M_{21}, and M_{22} have been omitted; yet, it should be kept in mind that they are nonlinear as indicated in the shorthand notation $A(\bar{e})$ and $B(\bar{e})$.

The nonlinear error system (10) will be stabilized by a PDC control law of the form (Wang et al., 1996):

$$u(t) = \sum_{i=1}^{r} h_i(z(\bar{e}))F_i\bar{e}(t), \tag{11}$$

where $z(\bar{e}) \in \mathbb{R}^p$ is a vector collecting the different nonlinearities in A_{41}, A_{42}, A_{43}, A_{44}, M_{22}^{-1} and M_{21}, which are assumed to be bounded in a compact set of the state space, i.e., $z_j(\bar{e}) \in [z_j^0, z_j^1]$, $j \in \{1, 2, \ldots, p\}$; $h_i(z)$, $i \in \{1, 2, \ldots, r\}$, $r = 2^p$, are defined from

$$w_0^j(z) = \frac{z_j^1 - z_j(\bar{e})}{z_i^1 - z_i^0}, \quad w_1^i(z) = 1 - w_0^i(z),$$

$$h_i(z) = \prod_{j=1}^{p} w_{i_j}^j, \quad (i_1, i_2, \ldots, i_p) \in \{0,1\}^p;$$

and, finally, $F_i \in \mathbb{R}^{m\times 2n}$ are constant gains to be determined. Notice that (11) is a nonlinear control law because of the presence of the system nonlinearities trough the different $h_i(z)$.

It is a standard result of quasi-LPV stabilization techniques as those in Tanaka and Wang (2001) that the origin $\bar{e} = 0$ of the nonlinear error system (10) under the PDC control law (11) is asymptotically stable if there exist matrices $X \in \mathbb{R}^{2n\times 2n}$, $M_i \in \mathbb{R}^{m\times 2n}$ such that the LMIs $X = X^T > 0$ and

$$\frac{2}{r-1}\Gamma_{ii}+\Gamma_{ij}+\Gamma_{ji}<0,\ i, j \in \{1, 2 \ldots, r\}, \tag{12}$$

hold for $\Gamma_{ij} = A(\bar{e})|_{h_i=1}X + B(\bar{e})|_{h_i=1}M_j + (*)$, with $(*)$ denoting the transpose of the expression at its left-hand side. The corresponding gains are thus $F_i = M_iX^{-1}$, $i \in \{1, 2, \ldots, r\}$.

Remark 1. Note that $A(\bar{e})|_{h_i=1}$ and $B(\bar{e})|_{h_i=1}$ are constant matrices corresponding to the vertex systems of a quasi-LPV exact convex representation of (10); they are formed with the 2^p combinations of minima-maxima of the nonlinearities gathered in $z(\bar{e})$. There are many ways of obtaining such representations, which are not unique (Lendek et al., 2010).

Remark 2. The convex sum relaxation in Tuan et al. (2001) has been used to obtain the LMIs in (12), but any othere relaxation scheme can be employed instead, e.g., those in Tanaka et al. (1998) and Sala and Ariño (2007).

Remark 3. The LMI nature of conditions (12) allows a straightforward incorporation of decay rate specifications and input/output constraints, also in the form of LMIs. In this work all of them were employed as they play an important role in the real-time applicability of the control scheme. Decay rate of $\alpha > 0$ can be imposed by adding the term $2\alpha X$ to Γ_{ij} above. An input constraint of $\mu > 0$ such that $\|u(t)\| < \mu$ as well as an output constraint of $\lambda > 0$ such that $\|y(t)\| < \lambda$ for a possibly nonlinear output $y(t) = \sum_{i=1}^{r} h_i(z)C_i\bar{e}$ under initial conditions $\bar{e}(0)$, can be achieved if the following LMIs are added to those above (Tanaka and Wang, 2001):

$$\begin{bmatrix}X & M_j^T\\ M_j & \mu^2 I\end{bmatrix} \geq 0,\ \begin{bmatrix}1 & x^T(0)\\ x(0) & X\end{bmatrix} \geq 0,\ \begin{bmatrix}X & XC_i^T\\ C_iX & \lambda^2 I\end{bmatrix} \tag{13}$$

Summing up, the proposed computed-torque technique will apply (9) with u defined as in (11) achieving stabilization of the underactuated system (5). The next section illustrates the effectiveness of the proposal, both in simulation and real-time, for an inverted pendulum on a cart.

4. SIMULATION AND REAL-TIME RESULTS

The inverted pendulum on a cart, shown in Fig.1, is a very well-known underactuated system which consists on a pole that freely swings around the cart center; the cart, in turn, is mounted on a linear rail and moves in any direction thanks to a DC motor, whose voltage is the only actuator of the system (Angeli, 2001). From the point of view of a Lagrange-Euler modelling, this system consists on 2 joints, one translational and the other rotational, which are marked as x_p and θ in Fig.1; they constitute the vector of generalized coordinates $q = [q_1\ q_2]^T$, where $q_1 = \theta$ (pole angle w.r.t. the upright position) and $q_2 = x_p$ (the cart distance to the rail center). Thus, the Lagrange-Euler model is:

$$\underbrace{\begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}}_{M(q)}\ddot{q} + \underbrace{\begin{bmatrix}N_1\\ N_2\end{bmatrix}}_{N(q,\dot{q})} = \underbrace{\begin{bmatrix}\tau\\ 0\end{bmatrix}}_{\bar{\tau}} \tag{14}$$

with $M_{11} = M_{22} = m_pl\cos q_1$, $M_{12} = m_p+m_c$, $M_{21} = I+m_pl^2$, $N_1 = -m_pl\dot{q}_1^2\sin q_1+b\dot{q}_2$ y $N_2 = -m_plg\sin q_1+d\dot{q}_1$, where the parameters g, l, m_c, m_p, I, b, and d are all specified in Table 1; these values have been taken from the provider in Instruments (2015). Our goal is to stabilize the pole in its upright position.

Table 1. Parameters of the inverted pendulum

Parameter	Symbol	Value
Gravity	g	9.81 m/s^2
Pole length	l	0.36 m
Cart mass	m_c	2.3 kg
Pole mass	m_p	0.2 kg
Moment of inertia of the pole	I	0.0099 $kg{\cdot}m^2$
Cart friction coefficient	b	0.00005 $N{\cdot}m/s$
Pendulum damping coefficient	d	0.005 $N\cdot m{\cdot}s/rad$

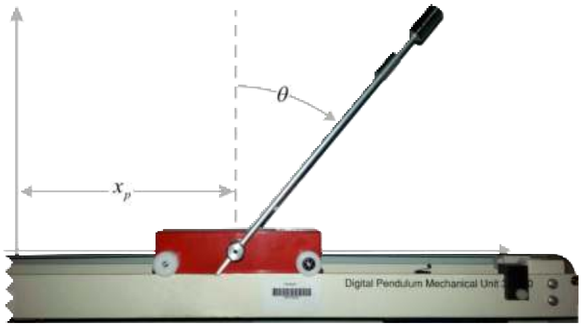

Fig. 1. Inverted pendulum

By taking into account that $e_1 = -q_1$, $e_2 = -q_2$, $\dot{e}_1 = -\dot{q}_1$, $\dot{e}_2 = -\dot{q}_2$, it is clear that $M_{22}^{-1}N_2$ and $M_{22}^{-1}M_{21}$ can be written as

$$\begin{aligned}
M_{22}^{-1}N_2 &= \frac{1}{m_p l\cos q_1}\left(d\dot{q}_1 - m_p gl\sin q_1\right)\\
&= \frac{1}{m_p l\cos(-e_1)}\left(-d\dot{e}_1 - m_p gl\sin(-e_1)\right)\\
&= \frac{1}{m_p l\cos e_1}\left(-d\dot{e}_1 + m_p gl\sin e_1\right)\\
&= -\frac{d}{m_p l}z_1\dot{e}_1 + \frac{m_p gl}{m_p l}z_1 z_2 e_1\\
M_{22}^{-1}M_{21} &= \frac{M_{21}}{m_p l}z_1,
\end{aligned}$$

where z_1 and z_2 are defined in Table 2 along with their bounds for $|e_1| \leq 0.2$.

It is now possible to write the error system for the inverted pendulum in the form (10) as follows

$$\dot{\bar{e}} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ gz_1z_2 & 0 & -\frac{d}{m_p l}z_1 & 0\end{bmatrix}}_{A(z_1,z_2)}\bar{e} + \underbrace{\begin{bmatrix}0\\0\\1\\-\frac{M_{21}}{m_p l}z_1\end{bmatrix}}_{B(z_1)}u, \qquad (15)$$

where $\bar{e} = [e_1\ e_2\ \dot{e}_1\ \dot{e}_2]^T$.

The design of a PDC control law (11) for u above relies on the construction of functions h_i as well as the corresponding matrices A_i, B_i, $i \in \{1,2,3,4\}$ described in the previous section. Such task begins by constructing the following functions, which take into account the definitions and bounds of z_i, $i \in \{1,2\}$ in Table 2:

$$\begin{aligned}
w_0^1(z_1) &= \frac{1.0203 - z_1}{1.0203 - 1}, \quad w_1^1(z_1) = 1 - w_0^1(z_1),\\
w_0^2(z_2) &= \frac{1 - z_2}{1 - 0.9933}, \quad w_1^2(z_2) = 1 - w_0^2(z_2).
\end{aligned}$$

Thus, $h_1(z) = w_0^1 w_0^2$, $h_2(z) = w_0^1 w_1^2$, $h_3(z) = w_1^1 w_0^2$, and $h_4(z) = w_1^1 w_1^2$, with matrices $A_i = A(z_1,z_2)|_{h_i=1}$, $B_i = B(z_1)|_{h_i=1}$ given by:

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ 9.7447 & 0 & -0.0694 & 0\end{bmatrix} \quad A_2 = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ 9.81 & 0 & -0.0694 & 0\end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ 9.9429 & 0 & -0.0709 & 0\end{bmatrix} \quad A_4 = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ 10.0095 & 0 & -0.0709 & 0\end{bmatrix}$$

$$B_1 = B_2 = \begin{bmatrix}0\\0\\1\\-0.4975\end{bmatrix} \qquad B_3 = B_4 = \begin{bmatrix}0\\0\\1\\-0.5076\end{bmatrix}.$$

We can now proceed to find $X = X^T > 0$ such that (12) hold, which will give a stabilizing PDC u for the error system; once this u is substituted in (9) the full control law τ will be obtained. Yet, for real-time implementation further considerations must be done, namely, the bounds on u guaranteeing bounds on τ, bounds on the rail length q_2 (i.e., e_2), as well as a prescribed decay rate to accelerate the settling time. How this information is produced and included is now discussed.

Table 2. Definitions of nonlinear terms

Term	Definition	Lower bounds z_i^0	Upper bounds z_i^1
z_1	$1/\cos e_1$	1	$1/\cos(0.2)$
z_2	$\sin e_1/e_1$	$\sin(0.2)/0.2$	1

According to the provider, the torque τ in (14) is limited to the range $\tau \in [-20, 20]$ newtons, which can be translated into bounds for u solved from (9) as:

$$u = \left(M_{12}M_{22}^{-1}M_{21} - M_{11}\right)^{-1}\left(\tau - N_1 + M_{12}M_{22}^{-1}N_2\right).$$

By taking into account that $\dot{e}_1$, $\dot{e}_2 \in [-10, 10]$, $e_1 \in [-0.2, 0.2]$, we have the induced bounds

$$\begin{gathered}
1.1718 \leq M_{12}M_{22}^{-1}M_{21} - M_{11} \leq 1.1985,\\
-8.1739 \leq N_1 - M_{12}M_{22}^{-}1N_2 \leq 5.3120,
\end{gathered}$$

which in turn mean that

$$\begin{gathered}
\min(u) = (1.1718)^{-1}(\min\tau - 5.3120) = -20.6009\\
\max(u) = (1.1718)^{-1}\left(\max\tau - (-8.1739)\right) = 24.0432
\end{gathered}$$

Therefore, if $|u(t)| \leq 21.6009$, it is guaranteed that $|\tau_1| \leq 20$ as desired, which means $\mu = 21.6009$ in (13).

The bounds on the rail length can be imposed by considering it as a constrained output $-0.4 \leq e_2 = -q_2 \leq 0.4$, which requires defining a constant $C_i = [0\ 1\ 0\ 0]$, $i \in \{1,2,3,4\}$, and $\lambda = 0.4$ in (13). We can now proceed to find $X = X^T > 0$ such that (12) hold along with (13) for initial condition $\bar{e}(0) = [-0.1\ 0.1\ 0\ 0]^T$, imposing an additional decay rate of $\alpha = 1$ to accelerate the settling time of the stabilization transient. These LMIs produce a feasible solution X and M_j, $j \in \{1,2,3,4\}$, from which

$$P = X^{-1} = \begin{bmatrix} 124.3233 & 79.6337 & 28.5935 & 42.0703\\ 79.6337 & 61.7024 & 18.1400 & 27.9064\\ 28.5935 & 18.1400 & 6.6088 & 9.6500\\ 42.0703 & 27.9064 & 9.6500 & 14.4803\end{bmatrix},$$

which defines a quadratic Lyapunov function $V(\bar{e}) = \bar{e}^T P\bar{e}$ associated to the nonlinear error system (15), and $F_j = M_j X^{-1}$

$$\begin{aligned}
F_1 &= [-126.1094\ -52.7746\ -31.8751\ -36.7767],\\
F_2 &= [-125.5813\ -52.2159\ -31.7360\ -36.5388],\\
F_3 &= [-128.9144\ -54.0761\ -32.4622\ -37.5994],\\
F_4 &= [-128.7243\ -53.6262\ -32.3999\ -37.4642],
\end{aligned}$$

are the gains in the PDC controller.

Remark 4. The purely LMI-based PDC controller in Quintana et al. (2017) consists of 16 gains; the PDC part in our work contains only 4 at the price of incorporating nonlinear terms in the computed-torque part.

The PDC controller u in (11) with $r = 4$ can now be incorporated to τ as in (9); since τ is a torque and the DC motor requires a voltage $v(t)$ as an input, the additional relationship $v(t) = -\tau(t)/9.6$ should be taken into account for simulation and real-time implementation via the interface of (Instruments, 2015), which is the case in this work. This linear relationship comes from the internal dynamics of the DC motor, which are

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix} -\frac{d_m}{J}x_1 + \frac{K_t}{J}x_2\\ -\frac{K_b}{L_m}x_1 - \frac{R}{L_m}x_2 - \frac{9.6}{L_m}v(t)\end{bmatrix}, \ \tau = [0\ \ 600K_t]\begin{bmatrix}x_1\\ x_2\end{bmatrix},$$

where $J = 1.4 \times 10^{-5}$, $K_t = 0.05$, $K_b = 0.05$, $d_m = 1 \times 10^{-6}$, $R = 2.5$, $L_m = 2.5 \times 10^{-3}$.

In order to test the ability of the controller to recover from perturbations, the signal $w(t) = 5/6\mathscr{U}(t-8) - 5/6\mathscr{U}(t-8.3)$ will be added to $v(t)$, where $\mathscr{U}(t)$ stands for the step function. In Figures 2, 3, 4, and 5, the pole angle w.r.t. the upright position, the position of the cart w.r.t. the center of the rail, the angular velocity of the pole, and the cart speed along the rail, respectively, are shown. In each figure, the the signal at the top is for simulation and that at the bottom the real-time one; bounds imposed via the LMI design are shown in dashed lines, if any. The effect of the perturbation can be noticed at $t = 8$; the controller is able to overcome its effect. In Fig. 6 the voltage signal is displayed; again, the simulation result is shown at the top and the real-time one at the bottom. The bounds imposed on the control signal are also shown in dashed lines.

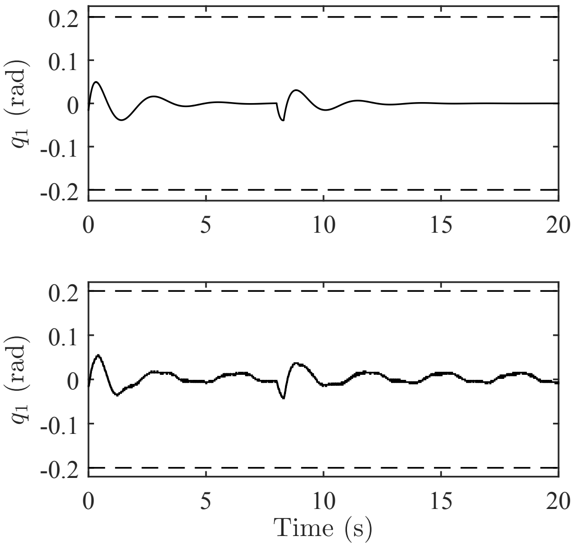

Fig. 2. Beam angle $q_1(t)$: simulation (top), real-time (bottom).

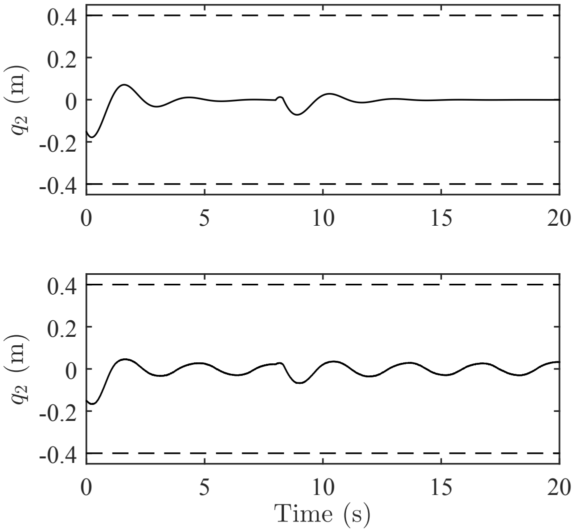

Fig. 3. Cart position $q_2(t)$: simulation (top), real-time (bottom).

5. CONCLUSIONS

A generalization of the computed-torque technique for stabilization of a class of underactuated robot manipulators has been proposed in this paper; it is based on performing partial model inversion through the actuated parts as to obtain a nonlinear error system, which in turn is stabilized through parallel distributed compensation. The latter has been designed using linear matrix inequalities that allow incorporating further plant restrictions and specifications within the same framework. Simulation and real-time implementation of the proposed control scheme on an inverted pendulum on a cart has been successfully done. Future work should address a generalization for trajectory tracking.

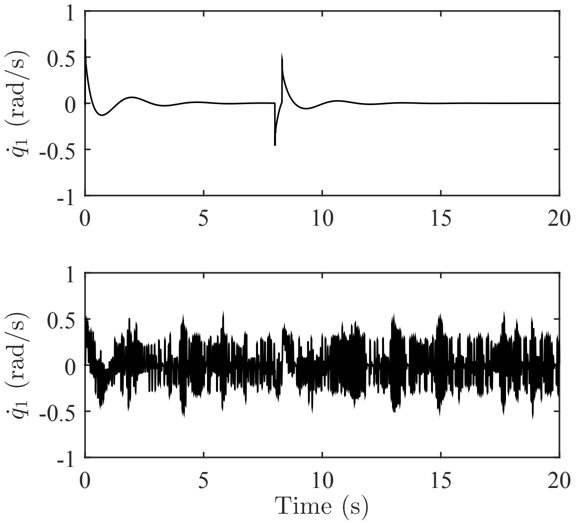

Fig. 4. Angular velocity $\dot{q}_1(t)$: simulation (top), real-time (bottom).

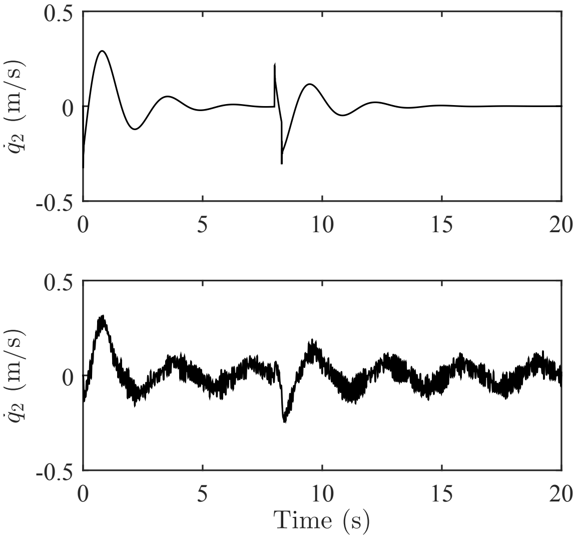

Fig. 5. Cart speed $\dot{q}_2(t)$: simulation (top), real-time (bottom).

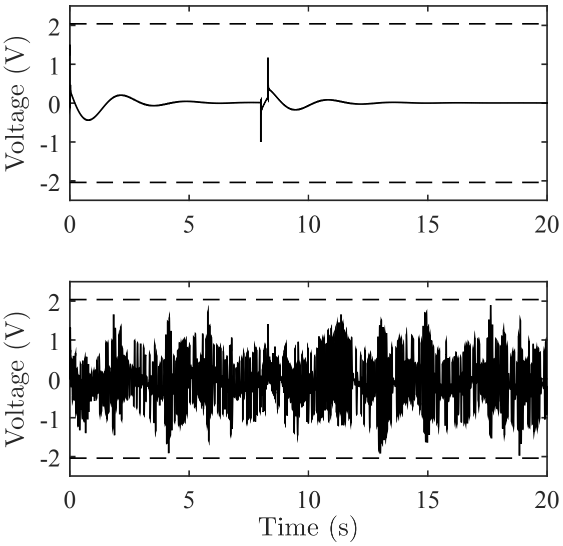

Fig. 6. Voltage input $v(t)$: simulation (top), real-time (bottom).

REFERENCES

Angeli, D. (2001). Almost global stabilization of the inverted pendulum via continuous state feedback. *Automatica*, 37(7), 1103–1108.

Boyd, S., Ghaoui, L.E., Feron, E., and Belakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*, volume 15. SIAM: Studies In Applied Mathematics, Philadelphia, USA.

Chen, C.T. (1984). *Linear System Theory and Design.* Saunders College Publishing, New York, USA.

Duan, G. and Yu, H. (2013). *LMIs in Control Systems: Analysis, Design and Applications.* CRC Press, Boca Raton, Florida.

Gahinet, P., Nemirovski, A., Laub, A.J., and Chilali, M. (1995). *LMI Control Toolbox.* Math Works, Natick, USA.

Instruments, F. (2015). *Digital Pendulum Control Experiments 33-936S.* Feedback instruments Ltd, East Sussex, U.K.

Isidori, A. (1995). *Nonlinear Control Systems.* Springer, London, 3 edition.

Khosla, P. and Kanade, T. (1989). Real-time implementation and evaluation of the computed-torque scheme. *IEEE Transactions on Robotics and Automation*, 5(2), 245–253.

Lee, C. and Chen, M. (1983). A suboptimal control design for mechanical manipulators. In *1983 American Control Conference*, 1056–1061. IEEE.

Lendek, Z., Guerra, T., Babuška, R., and De-Schutter, B. (2010). *Stability Analysis and Nonlinear Observer Design Using Takagi-Sugeno Fuzzy Models.* Springer-Verlag, Netherlands.

Lewis, F., Dawson, D., and Abdallah, C. (2003). *Robot manipulator control: theory and practice.* CRC Press.

Marion, J.B. (1965). *Classical dynamics.* Academic Press.

Ogata, K. (2001). *Modern control engineering.* Prentice Hall PTR, NJ, USA.

Ortega, R., Loría, J., Nicklasson, P., and Sira-Ramirez, H. (1998). *Passivity-based Control of Euler-Lagrange Systems: Mech., Elect. and Electromechanical Applications.* Springer, London.

Quintana, D., Estrada-Manzo, V., and Bernal, M. (2017). Real-time parallel distributed compensation of an inverted pendulum via exact takagi-sugeno models. In *2017 14th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, 1–5. IEEE.

Richardson, R., Brown, M., Bhakta, B., and Levesley, M. (2005). Impedance control for a pneumatic robot-based around pole-placement, joint space controllers. *Control Engineering Practice*, 13(3), 291–303.

Sala, A. and Ariño, C. (2007). Asymptotically necessary and sufficient conditions for stability and performance in fuzzy control: Applications of Polya's theorem. *Fuzzy Sets and Systems*, 158(24), 2671–2686.

Shin, K. and McKay, N. (1985). Minimum-time control of robotic manipulators with geometric path constraints. *IEEE Transactions on Automatic Control*, 30(6), 531–541.

Shtessel, Y., Edwards, C., Fridman, L., and Levant, A. (2013). *Sliding mode control and observation.* Birkhäuser, New York, USA.

Tanaka, K., Ikeda, T., and Wang, H. (1998). Fuzzy regulators and fuzzy observers: Relaxed stability conditions and LMI-based designs. *IEEE Transactions on Fuzzy Systems*, 6(2), 250–264.

Tanaka, K. and Wang, H. (2001). *Fuzzy Control Systems Design and Analysis: A linear matrix inequality approach.* John Wiley & Sons, New York.

Taniguchi, T., Tanaka, K., and Wang, H. (2001). Model construction, rule reduction and robust compensation for generalized form of Takagi-Sugeno fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 9(2), 525–537.

Tuan, H., Apkarian, P., Narikiyo, T., and Yamamoto, Y. (2001). Parameterized linear matrix inequality techniques in fuzzy control system design. *IEEE Transactions on Fuzzy Systems*, 9(2), 324–332.

Udawatta, L., Watanabe, K., Izumi, K., and Kiguchi, K. (2002). Obstacle avoidance of three-dof underactuated manipulator by using switching computed torque method. *Trans. on Control, Automation and Systems Engineering*, 4(4), 347–355.

Udawatta, L., Watanabe, K., Izumi, K., and Kiguchi, K. (2003a). Control of underactuated manipulators using fuzzy logic based switching controller. *Journal of Intelligent and Robotic Systems*, 38(2), 155–173.

Udawatta, L., Watanabe, K., Izumi, K., and Kiguchi, K. (2003b). Control of underactuated robot manipulators using switching computed torque method: Ga based approach. *Soft Computing*, 8(1), 51–60.

Wang, H., Tanaka, K., and Griffin, M. (1996). An approach to fuzzy control of nonlinear systems: Stability and design issues. *IEEE Transactions on Fuzzy Systems*, 4(1), 14–23.