J Control Autom Electr Syst (2016) 27:119–131
DOI 10.1007/s40313-016-0231-9

Robust Model Predictive Control of a Benchmark Electromechanical System

José Roberto Colombo Junior[1] · **Rubens Junqueira Magalhães Afonso**[1] · **Roberto Kawakami Harrop Galvão**[1] · **Edvaldo Assunção**[2]

Received: 31 August 2015 / Revised: 16 November 2015 / Accepted: 19 January 2016 / Published online: 1 February 2016

Abstract This paper presents an experimental investigation concerning the use of robust model predictive control (RMPC) for a two-mass–spring system. This benchmark system has been employed as a numerical simulation example in several works involving RMPC formulations, but an actual experimental implementation has never been reported. Particular care was taken to solve the optimization problem with linear matrix inequalities within a small sampling period (15 ms). A discussion concerning the discretization of the uncertain model is presented to justify the use of the exact zero-order hold method. More specifically, the resulting loss of polytopic structure was found to be negligible with the adopted sampling period. Three experimental scenarios were considered, with different ranges for the uncertain spring stiffness coefficient. In all cases, the control task was successfully accomplished, with proper satisfaction of constraints on the input voltage and spring deformation.

✉ Roberto Kawakami Harrop Galvão
kawakami@ita.br

José Roberto Colombo Junior
colombojrj@gmail.com

Rubens Junqueira Magalhães Afonso
rubensjm@ita.br

Edvaldo Assunção
edvaldo@dee.feis.unesp.br

[1] Divisão de Engenharia Eletrônica, Instituto Tecnológico de Aeronáutica, Praça Marechal Eduardo Gomes, 50, São José dos Campos, SP 12228-900, Brazil

[2] Departamento de Engenharia Elétrica, Univ Estadual Paulista, Av. José Carlos Rossi, 1370, Ilha Solteira, SP 15835-000, Brazil

Keywords Predictive control · Robust control · Constrained control · Linear matrix inequalities · Two-mass–spring system

1 Introduction

Model predictive control (MPC) techniques were initially developed for oil refining applications in the 1970s. In more recent years, the use of MPC has become disseminated in several other fields, such as the chemistry, aerospace and food industries (Qin and Badgwell 2003). Novel applications have included, for example, the control of oxygen excess ratio in fuel cells (Gruber et al. 2012), management of battery/supercapacitor storage systems in hybrid electric vehicles (Santucci et al. 2014), exhaust emission regulation in turbocharged diesel engines (Zhao et al. 2014) and load voltage control of four-leg inverters (Yaramasu et al. 2014).

One of the key reasons for the wide acceptance of MPC in industrial applications is the possibility of handling constraints on manipulated and controlled variables (Capron and Odloak 2013), which are usually active at the most profitable operating conditions (Maciejowski 2002). Nominal stability and constraint satisfaction guarantees can be obtained with adequate formulation of the optimization problem to be solved at each sampling time (Mayne et al. 2000). However, such properties may be lost in the presence of a mismatch between the internal model of the controller and the actual dynamics of the plant, resulting from modeling approximations, parametric uncertainties or faults.

In this context, much research has been conducted to develop robust model predictive control (RMPC) formulations. Early propositions involved uncertainties expressed in the form of bounds on the impulse response of FIR (finite impulse response) models (Campo and Morari 1987; Allwright and Papavasiliou 1992; Zheng and Morari 1993). A more elaborate approach introduced by Kothare et al. (1996) allowed for the use of more general uncertainty structures, either in polytopic or structured feedback forms. The resulting optimization problem could be cast into a semidefinite programming (SDP) format, with constraints in the form of linear matrix inequalities (LMIs). Efficient numerical solvers (Boyd et al. 1994; Gahinet and Nemirovski 1997) could then be used to obtain the optimal control to be applied at each sampling time.

For illustration, Kothare et al. (1996) presented numerical simulation results using a two-mass–spring system, which is often employed as a benchmark in robust control studies (Wie and Bernstein 1992). Several subsequent works involving LMI-based RMPC strategies derived from the formulation of Kothare et al. (1996) have also employed this example. Cuzzola et al. (2002) proposed the use of LMIs with less conservatism, involving a different Lyapunov function for each vertex of the uncertainty polytope. Less conservative results were also obtained by Wada et al. (2006) and Feng et al. (2006) through the use of parameter-dependent Lyapunov functions. Cao and Lin (2005) developed an improved approach to handle input constraints, based on a set invariance condition. Tahir and Jaimoukha (2013) addressed the problem of external disturbances, in addition to model uncertainty. Zhang (2013) proposed a method to reduce the computational workload required for real-time implementation of the RMPC control law.

It is worth noting that all these papers presented the two-mass–spring example within the context of numerical simulations. It may thus be argued that experimental studies involving this benchmark system would be of much value to link theory and practice in the LMI-based RMPC literature. Within this scope, the present work presents an experimental demonstration of the RMPC formulation originally proposed by Kothare et al. (1996), using an actual two-mass–spring system with electromechanical actuation. The control task involves both input and output constraints, as well as model uncertainty. Particular care was taken to enable the update of the control actions with a sampling period of 15 ms, which is relatively small in comparison with typical industrial applications of predictive control (Zhang et al. 2014), (Capron and Odloak 2013).

A preliminary version of this work was presented in a recent conference (Colombo Junior et al. 2014). The present paper is a much improved version, with more detailed descriptions of the theoretical background and experimental work. In particular, a discussion on the adopted model discretization procedure (exact zero-order hold method) is presented, with respect to the possible loss of the polytopic structure of the uncertainty. Moreover, a more appropriate initialization procedure for the SDP solver is employed. Finally, more elaborate scenarios are considered for the RMPC control task, with different ranges for the model uncertainty.

The remainder of this paper is organized as follows. Section 2 reviews the adopted RMPC formulation. Section 3 introduces the case study, with a description of the two-mass–spring system employed in the experiments, the adopted state-space model and the RMPC implementation details. The results are presented in Sect. 4, and final remarks are given in Sect. 5.

1.1 Notation

$I_{\bullet\times\bullet}$ and $0_{\bullet\times\bullet}$ denote an identity matrix and a matrix of zeros, with dimensions indicated by subscripts. The pth diagonal element of a square matrix X is denoted as X_{pp}. The star symbol $\star$ is used to indicate the blocks below the main diagonal of a symmetric matrix.

2 Robust Model Predictive Control Formulation

It is assumed that the plant dynamics can be described by an uncertain discrete-time state-space model of the form:

$$x(k+1) = Ax(k) + Bu(k),\ (A, B) \in \Omega, \tag{1}$$

where $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$ denote the state and control vectors, respectively, and Ω is an uncertainty polytope with known vertices (A_i, B_i), with $A_i \in \mathbb{R}^{n_x \times n_x}$ and $B_i \in \mathbb{R}^{n_x \times n_u}$, $i = 1, 2, \ldots, L$. It is also assumed that component-wise amplitude constraints are to be imposed on the control vector $u(k)$, as well as on a vector of output variables $y(k) \in \mathbb{R}^{n_y}$ given by

$$y(k) = Cx(k) \tag{2}$$

where $C \in \mathbb{R}^{n_y \times n_x}$ is a known matrix.

The predictive control formulation adopted herein involves an infinite-horizon cost function $J_\infty(k)$ given by

$$J_\infty(k) = \sum_{j=0}^{\infty} \Big[x(k+j|k)^T Sx(k+j|k) + u(k+j|k)^T Ru(k+j|k) \Big], \tag{3}$$

where $S \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ are symmetric, positive-definite weight matrices and $x(k+j|k)$, $u(k+j|k)$ denote future state and control values, which are related by a prediction equation of the form

$$x(k+j|k) = Ax(k+j-1|k) + Bu(k+j-1|k),\ \ j \geq 1, \tag{4}$$

with $x(k|k) = x(k)$.

In view of the uncertainty on the A, B matrices, the optimization problem can be formulated in a min–max framework as

$$\min_{u(k+j|k),\ j\geq 0}\ \max_{(A,B)\in\Omega} J_\infty(k), \tag{5}$$

subject to

$$|u_p(k+j|k)| \leq u_{p,\max}, \quad p=1,2,\ldots,n_u,\ j\geq 0 \tag{6}$$

$$|y_q(k+j|k)| \leq y_{q,\max}, \quad q=1,2,\ldots,n_y,\ j\geq 1 \tag{7}$$

where $u_{p,\max}$ and $y_{q,\max}$ denote the bounds on the excursion of the pth control variable and qth output variable, respectively.

As shown by Kothare et al. (1996), this min–max problem can be replaced with the following semidefinite programming (SDP) problem, which involves the minimization of an upper bound γ for the cost $J_\infty(k)$:

$$\min_{\gamma, Q>0, Y, X} \gamma \tag{8}$$

subject to

$$\begin{bmatrix} Q & x(k) \\ \star & 1 \end{bmatrix} \geq 0, \tag{9}$$

$$\begin{bmatrix} Q & QA_i^T + Y^T B_i^T & QS^{1/2} & Y^T R^{1/2} \\ \star & Q & 0_{n_x\times n_x} & 0_{n_x\times n_u} \\ \star & \star & \gamma I_{n_x\times n_x} & 0_{n_x\times n_u} \\ \star & \star & \star & \gamma I_{n_u\times n_u} \end{bmatrix} \geq 0,\ i=1,2,\ldots,L, \tag{10}$$

$$\begin{bmatrix} X & Y \\ \star & Q \end{bmatrix} \geq 0, \tag{11}$$

$$X_{pp} \leq u_{p,\max}^2,\ p=1,2,\ldots,n_u \tag{12}$$

$$\begin{bmatrix} Q & (A_iQ+B_iY)^T C_q^T \\ \star & y_{q,\max}^2 \end{bmatrix} \geq 0,\ i=1,2,\ldots,L,\ q=1,2,\ldots,n_y \tag{13}$$

where $Q = Q^T \in \mathbb{R}^{n_x\times n_x}$, $X = X^T \in \mathbb{R}^{n_u\times n_u}$ and $Y \in \mathbb{R}^{n_u\times n_x}$ are matrix variables of the optimization problem. The constraints on the control and output variables are imposed through the LMIs (11), (12) and (13), respectively.

It is worth noting that the solution of the SDP problem (8)–(13) depends on the current state $x(k)$, in view of the LMI (9). Therefore, this problem will be henceforth denoted by $\mathbb{P}(x(k))$. To emphasize that $\mathbb{P}(x(k))$ must be solved at each time k, the solution of the problem will be denoted by $(\gamma_k, Q_k, Y_k, X_k)$. If $\mathbb{P}(x(k))$ is feasible, the control action $u(k)$ is obtained as

$$u(k) = F_k x(k) \tag{14}$$

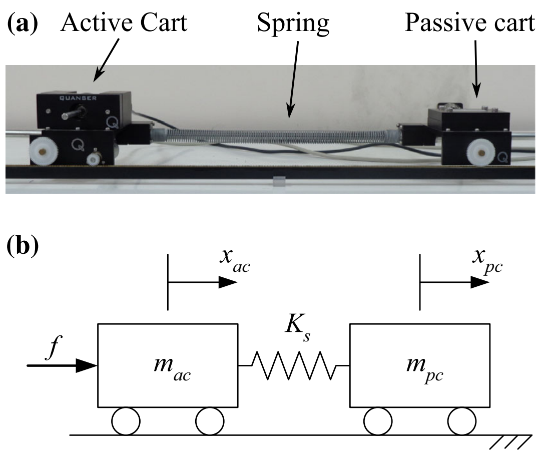

Fig. 1 Two-mass–spring system: **a** photograph and **b** schematic representation

where F_k is a gain matrix given by

$$F_k = Y_k Q_k^{-1} \tag{15}$$

As demonstrated by Kothare et al. (1996), if $\mathbb{P}(x(0))$ is feasible for the initial condition $x(0)$, then $\mathbb{P}(x(k))$ will remain feasible for $k>0$ and the state $x(k)$ will converge to the origin asymptotically, with satisfaction of the input and output constraints.

3 Case Study

3.1 System Description

Figure 1a presents a photograph of the two-mass–spring system (*Quanser Consulting*, model LFJC-E) employed in this work, with a schematic representation in Fig. 1b. The control variable is the voltage ϑ applied to the active cart motor, which results in the generation of a force f through a rack and pinion transmission system. The masses and positions of the active and passive carts are denoted by m_{ac}, m_{pc} and x_{ac}, x_{pc}, respectively. The stiffness coefficient of the spring is denoted by K_{s}. It is worth noting that x_{ac} and x_{pc} are the relative displacements of each cart with respect to an initial resting condition. The spring will be compressed if $x_{\mathrm{ac}} > x_{\mathrm{pc}}$ and elongated if $x_{\mathrm{ac}} < x_{\mathrm{pc}}$.

A continuous-time model for this system can be written in the form

$$\dot{x}(t) = A_c x(t) + B_c \vartheta(t) \tag{16}$$

where t denotes the continuous-time variable, ϑ is the motor voltage and $x = [x_{\mathrm{ac}}\ \ x_{\mathrm{pc}}\ \ v_{\mathrm{ac}}\ \ v_{\mathrm{pc}}]^T$ is the state vector, comprising the cart positions x_{ac}, x_{pc} and the corresponding velocities $v_{\mathrm{ac}} = \dot{x}_{\mathrm{ac}}$, $v_{\mathrm{pc}} = \dot{x}_{\mathrm{pc}}$. The electrical dynamics of

Table 1 Model parameters

Parameter	Physical meaning	Value
K_s	Spring stiffness coefficient	142 N/m
m_ac	Active cart mass	1.15 kg
m_pc	Passive cart mass	0.54 kg
b_ac	Viscous friction coefficient (active cart)	5.4 Ns/m
b_pc	Viscous friction coefficient (passive cart)	2.2 Ns/m
K_g	Gearbox ratio	3.71
K_m	Counter-electromotive force constant	7.67×10^{-3} Vs/rad
K_t	Motor torque constant	7.67×10^{-3} Nm/A
R_m	Motor armature resistance	2.6 Ω
r_mp	Motor pinion radius	6.35 mm

the motor are not considered in the model because they are much faster than the mechanical dynamics of the two-mass–spring system. The A_c, B_c matrices are of the form (Apkarian et al. 2013):

$$A_c = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{K_\mathrm{s}}{m_\mathrm{ac}} & \frac{K_\mathrm{s}}{m_\mathrm{ac}} & -\frac{b_\mathrm{ac}+\frac{K_\mathrm{g}^2 K_\mathrm{m} K_\mathrm{t}}{R_\mathrm{m} r_\mathrm{mp}^2}}{m_\mathrm{ac}} & 0 \\ \frac{K_\mathrm{s}}{m_\mathrm{pc}} & -\frac{K_\mathrm{s}}{m_\mathrm{pc}} & 0 & -\frac{b_\mathrm{pc}}{m_\mathrm{pc}} \end{bmatrix}, \tag{17}$$

$$B_c = \begin{bmatrix} 0 \; 0 \; \frac{K_\mathrm{g} K_\mathrm{t}}{R_\mathrm{m} r_\mathrm{mp}} \; 0 \end{bmatrix}^T, \tag{18}$$

with parameter values given in Table 1, which were taken from the user manuals (Quanser 2012), (Quanser Document Number 501) and companion MATLAB code. This model comprises viscous friction terms, which are a linear approximation to more complex friction effects (both viscous and dry) that are present in the actual system. It is worth noting that the active cart parameters m_ac and b_ac are linear-motion equivalent coefficients, which include the rotary parameters of the motor.

By using the parameter values in Table 1, the resulting eigenvalues for matrix A_c are $-3.1 \pm 19.1j$, -9.3 and 0. As can be seen, the state-space model has a stable second-order mode (natural frequency $\omega_n = 19.3$ rad/s and damping ratio $\zeta = 0.16$), a stable first-order mode and an integrator.

For illustration, Fig. 2 presents the results of an open-loop experiment with an input voltage $\vartheta = 2$ V from $t = 0$ s to $t = 0.5$ s and $\vartheta = 0$ V after $t = 0.5$ s. The amplitude of this input excitation is similar to the voltage level observed in the transient period of the closed-loop experiments, as shown in the Sect. 4. As can be seen in Fig. 2, the model response is in good agreement with the experimental results in terms of the active and passive cart positions, as well as the spring deformation. More details concerning the experimental validation of the model can be found in (Colombo Junior 2014).

The positions x_ac and x_pc of the active and passive carts are measured by optical encoders with a resolution equivalent to 0.023 mm of linear displacement. The velocities v_ac and v_pc are estimated by using the following approximations:

$$\hat{v}_\mathrm{ac}(kT) = \frac{x_\mathrm{ac}(kT) - x_\mathrm{ac}((k-1)T)}{T}, \tag{19}$$

$$\hat{v}_\mathrm{pc}(kT) = \frac{x_\mathrm{pc}(kT) - x_\mathrm{pc}((k-1)T)}{T}, \tag{20}$$

where T is the sampling period. Henceforth, with a small abuse of notation, the time indication (kT) will be stated simply as (k), in agreement with the discrete-time notation adopted in Sect. 2.

Remark 1 A state observer could be used for the estimation of the velocities, as an alternative to the finite difference method adopted herein. However, the design of a suitable observer for use with the predictive controller may be challenging in the presence of model uncertainty. It is also worth noting that the encoder resolution (2.3×10^{-3} cm) is much smaller than the cart displacements per sampling period during the transient response. Indeed, as will be seen in the Sect. 4, the carts travel approximately 18 cm in one second, which corresponds to 0.27 cm within a sampling period of 15 ms. Therefore, the quantization noise in the position measurements was not a significant issue regarding the use of the approximations (19), (20).

3.2 Model Uncertainty

In order to evaluate the RMPC control law in the presence of model uncertainty, the spring stiffness coefficient K_s will be considered an uncertain parameter, as in the original simulation example presented by Kothare et al. (1996). More specifically, three cases will be considered, with different uncertainty ranges, as illustrated in Fig. 3. In Case I, the uncertainty range is centered around the nominal stiffness

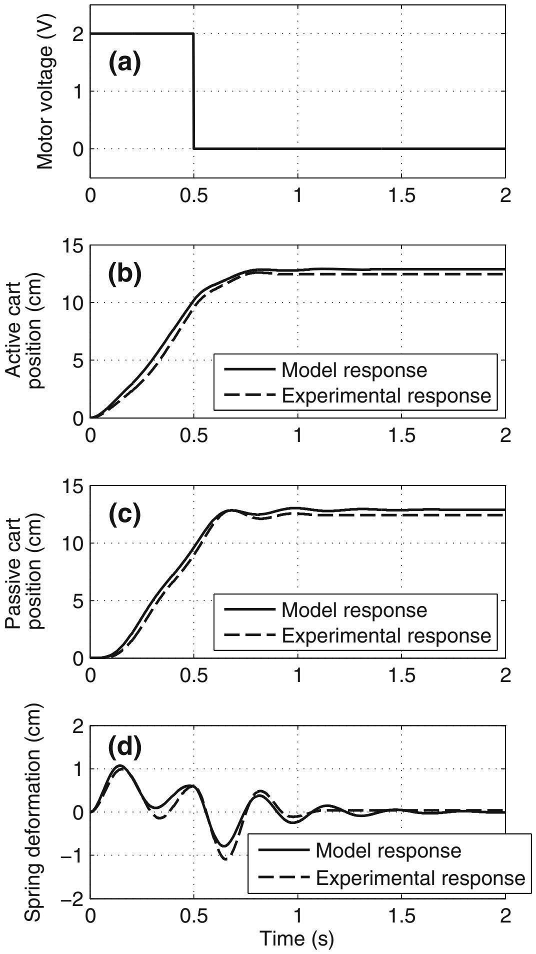

Fig. 2 Open-loop results: **a** input voltage, **b** active cart position, **c** passive cart position, **d** spring deformation

coefficient K_s (in the sense that the nominal value is the geometrical mean of the uncertainty bounds). In Cases II and III, the ranges are shifted toward larger and smaller values with respect to the nominal stiffness coefficient K_s, respectively. In what follows, subscripts 1 and 2 will be used to indicate the smallest and largest values of K_s in each case, as well as the corresponding model vertices.

In each case, an uncertainty polytope for matrix A_c is defined with two vertices A_{c1}, A_{c2} associated with the extreme values of K_s through (17). The input matrix B_c is not affected by the uncertainty, because the corresponding expression (18) does not depend on K_s. Therefore, the uncertainty polytope for the continuous-time model is formed with vertices (A_{c1}, B_c) and (A_{c2}, B_c).

With the extreme values $K_{s1} = 71$ N/m and $K_{s2} = 284$ N/m, the eigenvalues of A_{c1} and A_{c2} are $(-3.0 \pm 13.2j, -9.6, 0)$ and $(-3.2 \pm 27.4j, -9.2, 0)$, respectively. As can be seen, a change in K_s is mainly reflected in the natural frequency of the second-order mode ($\omega_{n1} = 13.5$ rad/s and $\omega_{n2} = 27.6$ rad/s).

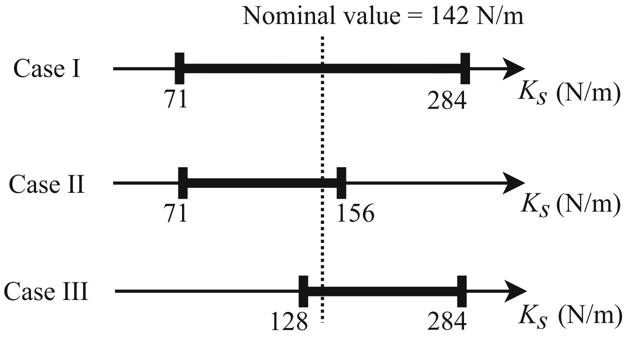

Fig. 3 Uncertainty ranges for the spring stiffness coefficient K_s in three cases considered in the experiments

3.3 Model Discretization

The state-space Eq. (16) needs to be discretized in the form (1) for use in the RMPC control law. Since the control is applied through a digital-to-analog converter, which keeps the motor voltage constant between the sampling times, an exact discretization can be carried out by using the zero-order hold (ZOH) method (Franklin et al. 1998) as

$$A = e^{A_c T} = \sum_{n=0}^{\infty} \frac{(A_c T)^n}{n!} \tag{21}$$

$$B = \left(\int_0^T e^{A_c \tau} \mathrm{d}\tau\right) B_c = \left(\sum_{n=0}^{\infty} \frac{(A_c T)^n}{(n+1)!}\right) B_c T \tag{22}$$

The sampling period was set to $T = 15$ ms, which is an order of magnitude smaller than the natural period of oscillation T_n of the second-order mode in either vertex of the uncertain model ($T_{n1} = 2\pi/\omega_{n1} = 465$ ms and $T_{n2} = 2\pi/\omega_{n2} = 228$ ms). As will be seen in the Sect. 4, the use of smaller values for T would not be appropriate, in view of the time required for completion of the RMPC calculations within each sampling period.

With the extreme values $K_{s1} = 71$ N/m and $K_{s2} = 284$ N/m, the eigenvalues of the A_1, A_2 matrices resulting from the discretization of A_{c1}, A_{c2} are $(0.94 \pm 0.19j, 0.87, 1)$ and $(0.87 \pm 0.38j, 0.87, 1)$, respectively. As can be seen, the stability properties of the continuous-time model are preserved, with a marginally stable mode (an integrator associated with the unity eigenvalue), a stable first-order mode and a stable second- order mode.

By using the ZOH method, the continuous-time model vertices (A_{c1}, B_c) and (A_{c2}, B_c) are mapped to the discrete-time counterparts (A_1, B_1) and (A_2, B_2). The vertices B_1, B_2 for the input matrix, are obtained through (22) by using A_{c1} and A_{c2}, respectively, in addition to the single matrix B_c.

It is worth noting that this exact discretization may not preserve the polytopic structure of the uncertainty, because of the nonlinear mapping between the continuous and discrete models involved in (21), (22). As an alternative, a first-order approximation (forward Euler method) could be used as (Franklin et al. 1998):

$$A_e = I_{n_x \times n_x} + A_c T \tag{23}$$

$$B_e = B_c T \tag{24}$$

where A_e and B_e are the resulting matrices of the discrete-time model. However, this approximation presents the following inconvenience. By using $T = 15$ ms, the eigenvalues of the A_{e1}, A_{e2} matrices for the extreme values $K_{s1} = 71$ N/m and $K_{s2} = 284$ N/m are $(0.96 \pm 0.20j, 0.86, 1)$ and $(0.95 \pm 0.41j, 0.86, 1)$, respectively. As can be seen, the second-order mode of the discretized model for K_{s2} is unstable, since the complex conjugate poles $(0.95 \pm 0.41j)$ are outside the unit circle. This is an inconvenience because the prediction model employed in the RMPC formulation would differ in a drastic manner from the dynamics of the actual physical system, in terms of stability properties. In order to obtain a stable second-order mode for both vertices of the uncertain model, the sampling period T would need to be smaller than 8 ms, which would not be enough to complete the RMPC calculations (as will be shown in the Sect. 4).

In view of these issues related to the stability of the discretized model, the ZOH method was adopted to obtain the (A_1, B_1) and (A_2, B_2) matrices employed in the RMPC control law. It is worth noting that the deviation from the polytopic structure of the uncertainty is negligible with the adopted sampling period, as shown in "Appendix 1."

3.4 Handling the Computational Delay

Since the RMPC control law involves the real-time solution of an optimization problem, the time delay between the acquisition of sensor readings and the control voltage update may correspond to a significant fraction of the sampling period. If this delay was known and constant, it would be possible to include it in the plant model, as described by Maciejowski (2002). However, in the present work there is no guarantee that the delay will remain constant during the entire control task. Therefore, the following procedure was adopted to account for the computational delay.

After finding the solution of the optimization problem, the resulting control value is stored to be used at the next sampling time. By doing so, the delay will always be equal to one sampling period, as depicted in Fig. 4.

Such a delay can be included in the model by defining an augmented state vector ξ as

$$\xi(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}, \tag{25}$$

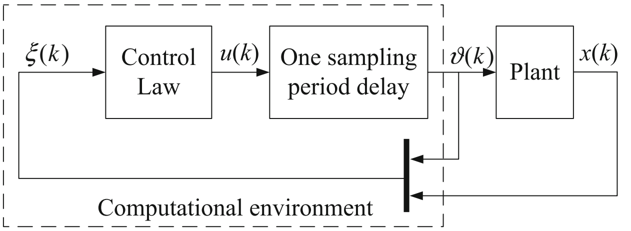

Fig. 4 Control loop with inclusion of an artificial delay

where $x(k) \in \mathbb{R}^4$ is the original state vector for the two-mass–spring system and $u(k-1) = \vartheta(k)$ is the control voltage actually applied to the plant at time k. As a result, the model employed in the RMPC control law becomes

$$\xi(k+1) = \bar{A}\xi(k) + \bar{B}u(k), \tag{26}$$

with $\bar{A}$ and $\bar{B}$ defined as

$$\bar{A} = \begin{bmatrix} A & B \\ 0_{1\times 4} & 0 \end{bmatrix}, \tag{27}$$

$$\bar{B} = \begin{bmatrix} 0_{4\times 1} \\ 1 \end{bmatrix}. \tag{28}$$

This augmented model can be regarded as a limit case of the procedure presented by Maciejowski (2002), with the delay equal to one full sampling period.

In short, the solution of the optimization problem $\mathbb{P}(\xi(k))$ is used to obtain the control action $u(k) = F_k\xi(k)$, which is applied to the plant at the next sampling time.

3.5 Parameters of the RMPC Controller

The control task considered herein consists of moving the active and passive carts to a target position $r = 20.0$ cm, starting from a rest condition. Since the RMPC control law is formulated in regulation form, with the purpose of steering the state to the origin, this task can be carried out by replacing the state vector $\xi(k)$ with a translated vector $\tilde{\xi}(k) = \xi(k) - [r \;\; r \;\; 0 \;\; 0]^T$. Therefore, by steering the state $\tilde{\xi}(k)$ to the origin, the controller will move the cart positions x_{ac} and x_{pc} to the target value r.

The weight matrices were set to $S = diag(10^3, 10^4, 10^{-1}, 10^{-1}, 10^{-3})$ and $R = 10^{-1}$. In order to place greater emphasis on the minimization of the cart position errors, a larger value was assigned to the first and second diagonal elements of matrix S.

The bounds on the control variable were set to ±6.0 V (i.e., $u_{\max} = 6.0$ V, which corresponds to the nominal voltage of the active cart motor). Experimentally, it was observed that larger control values could cause a derailment of the motor pinion over the rack.

In addition, bounds on the spring deformation were imposed to evaluate the ability of the RMPC controller to handle output constraints. For this purpose, an output variable $y(k)$ was defined as

$$y(k) = x_{\mathrm{ac}}(k) - x_{\mathrm{pc}}(k) = \left[\, 1 \; -1 \; 0 \; 0 \; 0 \,\right] \, \xi(k). \tag{29}$$

with positive and negative values corresponding to the compression and elongation of the spring, respectively. The bounds on this output variable were set to ±1.0 cm (i.e., $y_{\max} = 1.0$ cm).

3.6 Implementation Details

The RMPC control law was implemented in a computer with Intel i5-3470S processor (2.90 GHz), 6 GB of RAM memory and Windows 7 operational system. The interface with the plant hardware was accomplished by using a Q2-USB data acquisition module and the QuaRC software (both from Quanser Consulting). The SDP problem was solved by using the *Robust Control Toolbox* of MATLAB 2012a.

Remark 2 It is worth noting that industrial controllers are typically implemented by using dedicated hardware and software. However, general-purpose personal computers are often employed in the predictive control literature for experimental proof-of-concept studies (Gruber et al. 2009; Herceg et al. 2009; Rahideh and Shaheed 2012). This is the approach adopted herein.

It is worth noting that (9) is the only LMI that needs to be altered during the control task (with $\tilde{\xi}(k)$ in place of $x(k)$), which simplifies the recoding of the optimization problem in real time. Indeed, after the set of LMIs (9)–(13) is encoded in the memory of the computer upon the initialization of the controller, it is sufficient to change the value of $\tilde{\xi}(k)$ in LMI (9) at each new sampling time before invoking the SDP solver.

Before the beginning of the control task, a solution $(\gamma_0, Q_0, Y_0, X_0)$ was obtained *offline* for the initial problem $\mathbb{P}(\tilde{\xi}(0))$, in order to calculate the control $u(0) = Y_0 Q_0^{-1} \tilde{\xi}(0)$. Obtaining this solution required approximately 55 ms. Since the calculation of the subsequent control actions needed to be completed within the sampling period $T = 15$ ms, the solution $(\gamma_{k-1}, Q_{k-1}, Y_{k-1}, X_{k-1})$ was used to initialize the solver at each time k. Moreover, the maximum number of iterations in the SDP solution algorithm was reduced from 100 (default value of the solver) to 10. As shown in "Appendix 2," this stopping criterion resulted in a cost γ close to the solution obtained with the default number of iterations.

Finally, it was observed that infeasibility of the previous solution occasionally occurred when the carts were already close to the target positions. Such a problem can be ascribed to nonlinear friction effects, which were neglected in the model and become more significant at low speeds (Olsson et al. 1998). This issue was circumvented by fixing the gain matrix (15) upon the first infeasibility event, i.e., the optimization problem was no longer solved and the last calculated gain F_{k-1} was used in the remaining part of the control task.

The time required by the SDP solver and the overall computation time spent in the implementation of the control law (including I/O operations and auxiliary calculations) were stored for presentation alongside the results of the control task.

Remark 3 Even by using a predictive control law with robustness properties, recursive feasibility problems may occur in practice because the adopted uncertainty framework may not capture all the mismatches between the design model and the actual system. A possible approach to circumvent this problem consists of using slack variables to soften the constraints (Erdem et al. 2004; Minh and Hashim 2011). However, such an approach implies an increase in computational workload, since more variables are involved in the optimization problem. The fixed-gain strategy adopted herein has the advantage of not requiring the use of extra optimization variables. Moreover, it preserves the robust stability guarantees of the closed-loop system. Indeed, let $\gamma_{k-1}, Q_{k-1}, Y_{k-1}$ be the values of γ, Q, Y obtained as the solution of $\mathbb{P}(x(k-1))$, with $x(k-1) \neq 0$. Since γ_{k-1} is an upper bound for cost (3) and $x(k-1) \neq 0$, it follows that $\gamma_{k-1} > 0$. Moreover, Q_{k-1} is positive-definite as imposed in (8), and $\gamma_{k-1}, Q_{k-1}, Y_{k-1}$ satisfy the LMI constraint (10). Therefore, by applying Schur's complement to the LMI (10) and letting $F = Y_{k-1} Q_{k-1}^{-1}$, it follows that

$$(A_i + B_i F)^T P (A_i + B_i F) - P \leq -S - F^T R F, \quad i = 1, 2, \ldots, L \tag{30}$$

with $P = \gamma_{k-1} Q_{k-1}^{-1} > 0$. Since $S > 0$ and $R > 0$, it follows that

$$(A_i + B_i F)^T P (A_i + B_i F) - P < 0, \quad i = 1, 2, \ldots, L \tag{31}$$

which guarantees that the fixed gain F will stabilize the closed-loop system for any (A, B) in the uncertainty polytope Ω with vertices (A_i, B_i), $i = 1, 2, \ldots, L$ (Boyd et al. 1994).

4 Results and Discussion

Figure 5 presents the results obtained in Case I, which involves an uncertainty range for the spring stiffness coefficient $K_s \in [71 - 284]$ N/m. As can be seen in Fig. 5a–c,

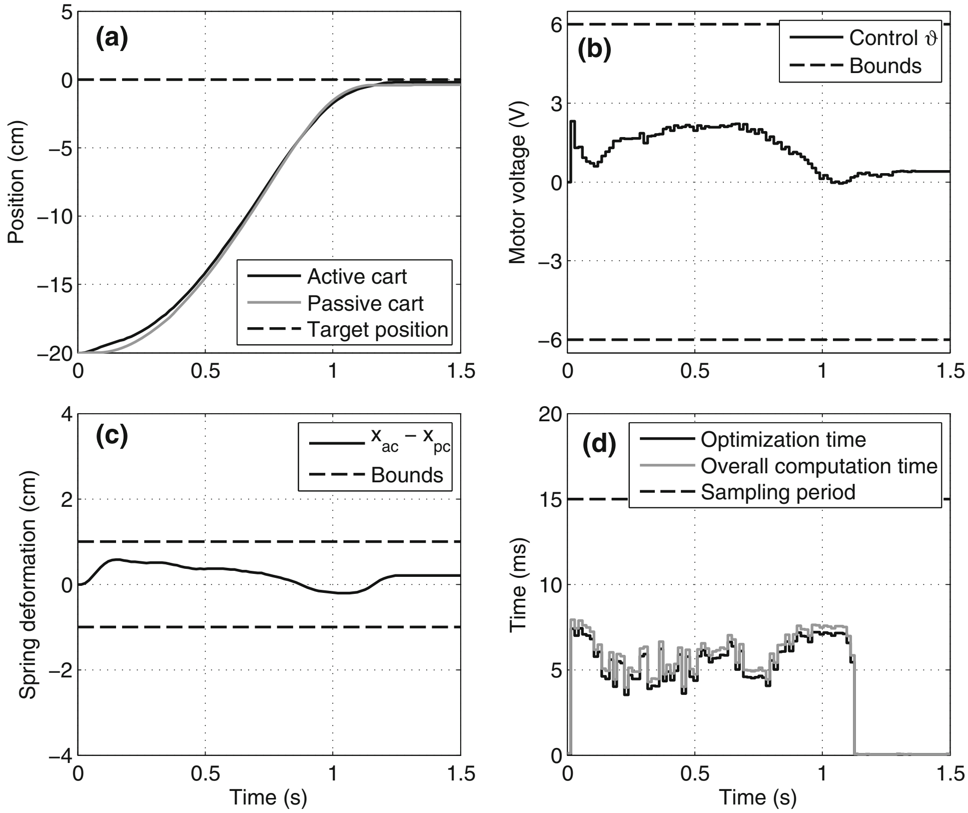

Fig. 5 Results obtained in Case I. **a** Encoder position readings and associated target value. **b** Control voltage ϑ and associated constraint bounds. **c** Spring deformation and associated constraint bounds. **d** Time spent by the optimization solver and overall computation time required to implement the control law at each sampling period

both the active and passive carts were driven to the setpoint, with proper enforcement of the constraints on the voltage and spring deformation. The steady-state positioning error (approximately 1.5 %) can be ascribed to dry friction effects and was considered sufficiently small so that changes in the controller design were not necessary. The gain matrix was fixed after $t = 1.11$ s, because the previous SDP solution was found to be infeasible, as discussed in Sect. 3.6. For this reason, the computation time decreased to very small values after this event, as can be seen in Fig. 5d. It is worth noting that the carts were already close to the target positions at this time, and thus, further optimization of the gain matrix would not result in substantial performance improvement.

Figure 5d reveals that the optimization process accounted for a substantial part of the computational workload. However, the overall computation time was always smaller than the sampling period, as required for the implementation of the control law. Moreover, by measuring the time intervals between the control action updates (with an Agilent MSO-X 2012A oscilloscope), the relative deviations with respect to the nominal sampling period were found to be smaller than 1 %.

It is worth noting that both the control voltage (Fig. 5b) and the spring deformation (Fig. 5c) values exhibit a noticeable gap with respect to the constraint bounds. However, as shown in “Appendix 3,” the removal of the spring deformation constraint leads to deformations larger than 3 cm, with control values very close to the upper bound of 6 V. Therefore, it may be argued that the spring deformation constraint was the limiting factor for the control actions.

The results for Cases II and III are presented in Figs. 6 and 7, respectively. In these cases, the controller was designed with uncertainty ranges $K_s \in [71-156]$ N/m and $K_s \in [128-284]$ N/m. Again, the control task was performed with very small steady-state error (approximately 1.5 and 0.5 %, respectively), satisfaction of the voltage and spring deformation constraints, and computation time always smaller than the sampling period.

It is interesting to notice that the transient response in Case III (Fig. 7a) was faster compared to Cases I (Fig. 5a) and II (Fig. 6a). Indeed, as shown in Table 2, the rise time was considerably smaller in Case III. Such a finding may be explained because the controllers in Cases I and II were faced with a more difficult task, in view of the possibility of having a less stiff spring connecting the carts (smaller lower bound for the uncertain coefficient K_s). It is worth pointing out that the spring was physically the same in all cases, but the control actions were more cautious in Cases I and II. Indeed, the gap between the maximum deformation value and the constraint bound (1 cm) is wider in Cases I and II compared to Case III, as shown in Table 2.

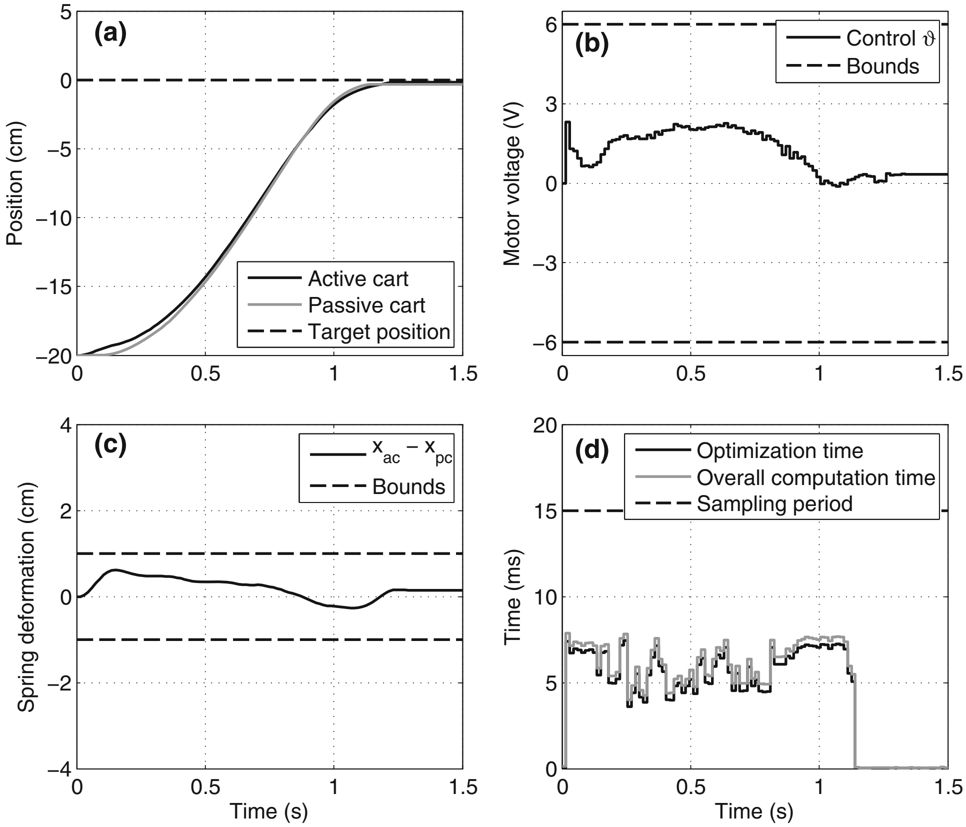

Fig. 6 Results obtained in Case II. **a** Encoder position readings and associated target value. **b** Control voltage ϑ and associated constraint bounds. **c** Spring deformation and associated constraint bounds. **d** Time spent by the optimization solver and overall computation time required to implement the control law at each sampling period

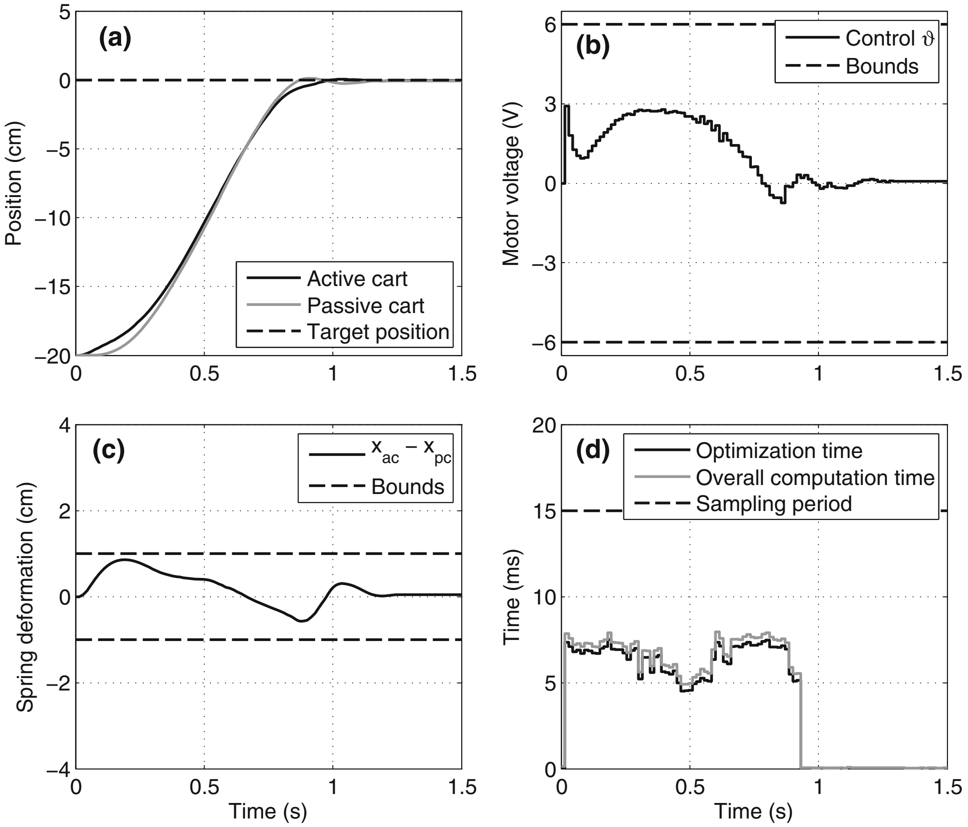

Fig. 7 Results obtained in Case III. **a** Encoder position readings and associated target value. **b** Control voltage ϑ and associated constraint bounds. **c** Spring deformation and associated constraint bounds. **d** Time spent by the optimization solver and overall computation time required to implement the control law at each sampling period

Table 2 Maximum spring deformation and rise time[a] values

Case	Uncertainty range for K_s (N/m)	Max. spring deformation (magnitude) (cm)	Rise time[a] (active cart) (s)	Rise time[a] (passive cart) (s)
I	71–284	0.58	0.98	0.97
II	71–156	0.62	0.99	0.98
III	128–284	0.86	0.77	0.76

[a] (0–90 %) criterion

5 Conclusions

This paper presented an experimental demonstration of the effectiveness of the RMPC control law originally proposed by Kothare et al. (1996), using an actual two-mass–spring system with electromechanical actuation. This plant is often used as a benchmark in robust control studies and has been widely employed in the LMI-based RMPC literature. However, to the best of the authors' knowledge, this is the first work concerning an actual experimental RMPC implementation for such a system.

Particular care was taken to obtain the solution of the SDP optimization problem within a sampling period of 15 ms. A warm start of the optimizer was employed by always using the solution obtained at the previous sampling time. Moreover, the default settings of the solver were changed to reduce the maximum number of optimization iterations. Even so, the required computational time varied during the control task and could be as large as 8 ms, which is a significant fraction of the sampling period. In order to account for this time delay, the resulting control value was stored to be used only at the next sampling time. By doing so, the delay was always equal to one sampling period, which facilitated its inclusion in the RMPC model. It is worth noting that the loss of polytopic structure resulting from the zero-order hold discretization of the uncertain model was found to be negligible with the adopted sampling period, as shown in "Appendix 1."

Three experimental scenarios were considered, with different ranges for the uncertain value of spring stiffness coefficient. The control task was successfully accomplished in all cases, with satisfaction of the constraints imposed on the motor voltage and spring deformation. Due to the presence of dry friction, steady-state position errors ranging from 0.5 to 1.5 % were obtained. Future studies could be concerned with methods for compensating the dry friction effects in order to further reduce the magnitude of the steady-state errors. It is worth noting that the introduction of integrators in the control law may result in a *hunting* phenomenon involving sustained oscillations around the target position (Yao et al. 2013), (Hensen et al. 2003). Alternatively, disturbance estimators could be employed as in traditional predictive control formulations (Maeder et al. 2009). However, such an approach may not be appropriate in this case, because the friction force changes with the direction of motion and may exhibit complex features at low speeds (Olsson et al. 1998). Therefore, more detailed investigations are required to address this issue. Future works could also be concerned with an implementation of the RMPC control law using dedicated industrial hardware. For this purpose, the use of non-commercial SDP solvers such as CSDP (C library for semidefinite programming) (Borchers 1999) or SDPA (semidefinite programming algorithms) (Fujisawa et al. 2002) could be investigated. Finally, the experimental framework presented in this paper could be employed in comparative studies involving different RMPC approaches derived from the original formulation of Kothare et al. (1996).

Acknowledgments The authors acknowledge the support of Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP—Grant 2011/17610-0 and PhD scholarship 2011/18632-8), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq—Research Fellowships), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES—PhD scholarship) and Financiadora de Estudos e Projetos (FINEP—Captaer and Captaer II Grants). Part of this work was carried out during the first author's MSc program (Colombo Junior 2014), with a CAPES/Pró-Engenharias scholarship. The authors are also grateful to Prof. Elder Moreira Hemerly (Instituto Tecnológico de Aeronáutica, Brazil) for his valuable comments concerning the stability of the discretized models.

Appendix 1

A pair of matrices $(A, B) \in (\mathbb{R}^{n_x \times n_x} \times \mathbb{R}^{n_x \times n_u})$ belongs to a polytope $\Omega = \text{Co}\{(A_1, B_1), (A_2, B_2)\}$ if and only if there exists a scalar $\lambda \in [0, 1]$ such that

$$(A, B) = \lambda(A_1, B_1) + (1 - \lambda)(A_2, B_2). \tag{32}$$

In order to test whether a given pair (A, B) belongs to Ω, let $w_1, w_2, w \in \mathbb{R}^{(n_x^2 + n_x n_u)}$ be defined as

$$w_1 = vec(A_1, B_1) \tag{33}$$

$$w_2 = vec(A_2, B_2) \tag{34}$$

$$w = vec(A, B) \tag{35}$$

where vec is an operator that stacks the elements of the pair of matrices in a column vector form.

The problem then consists of testing whether w can be written as $w = \lambda w_1 + (1 - \lambda) w_2$ for some $\lambda \in [0, 1]$. For this purpose, a least-squares solution for λ can be obtained by solving the following optimization problem:

$$\min_{\lambda} J(\lambda) = (w - \hat{w})^T (w - \hat{w}), \tag{36}$$

with

$$\hat{w} = \lambda w_1 + (1 - \lambda) w_2. \tag{37}$$

In the absence of the constraint $\lambda \in [0, 1]$, the optimal solution $\hat{\lambda}$ for this problem is obtained by replacing (37) for $\hat{w}$ in (36) and imposing $dJ/d\lambda = 0$. The result is uniquely given by

$$\hat{\lambda} = \frac{(w - w_2)^T (w_1 - w_2)}{(w_1 - w_2)^T (w_1 - w_2)} \tag{38}$$

If $\hat{\lambda} \in [0, 1]$ and $J(\hat{\lambda}) = 0$, one can conclude that (A, B) belongs to Ω.

Within the scope of the present paper, the problem consists of testing whether the uncertain (A, B) matrices obtained through the ZOH discretization of (A_c, B_c) belong to the polytope $\Omega = \text{Co}\{(A_1, B_1), (A_2, B_2)\}$, with (A_1, B_1), (A_2, B_2) obtained through the ZOH discretization of (A_{c1}, B_c) and (A_{c2}, B_c), respectively (with B_c not subject to uncertainty).

For this purpose, four matrices $A_c^{(1)}, A_c^{(2)}, A_c^{(3)}, A_c^{(4)}$ were created through convex combinations of A_{c1} and A_{c2} as

$$A_c^{(n)} = \lambda^{(n)} A_{c1} + \left(1 - \lambda^{(n)}\right) A_{c2}, \quad n = 1, 2, 3, 4, \tag{39}$$

with $\lambda^{(1)} = 0.1$, $\lambda^{(2)} = 0.4$, $\lambda^{(3)} = 0.6$ and $\lambda^{(4)} = 0.9$. The pairs of matrices $(A_c^{(1)}, B_c)$, $(A_c^{(2)}, B_c)$, $(A_c^{(3)}, B_c)$, $(A_c^{(4)}, B_c)$ were then discretized to obtain $(A^{(1)}, B^{(1)})$, $(A^{(2)}, B^{(2)})$, $(A^{(3)}, B^{(3)})$, $(A^{(4)}, B^{(4)})$. The elements of these resulting matrices were stacked in column vectors $w^{(n)}$, in order to solve the least-squares problem (36), (37). As can be seen in Table 3, the resulting solutions $\hat{\lambda}^{(n)}$ were all inside the [0, 1] interval, displaying very small differences with respect to the actual $\lambda^{(n)}$ values. This finding indicates that the deviations from the polytopic structure resulting from the ZOH discretization are very small in this case. Such a conclusion is corroborated by the graphical representation in Fig. 8, which shows a close agreement between the corresponding components of vectors $\hat{w}^{(n)}$ and $w^{(n)}$.

Table 3 Least-squares solutions

n	$\lambda^{(n)}$	$\hat{\lambda}^{(n)}$	Difference (%)
1	0.100	0.0987	1.3
2	0.400	0.397	0.8
3	0.600	0.597	0.5
4	0.900	0.899	0.1

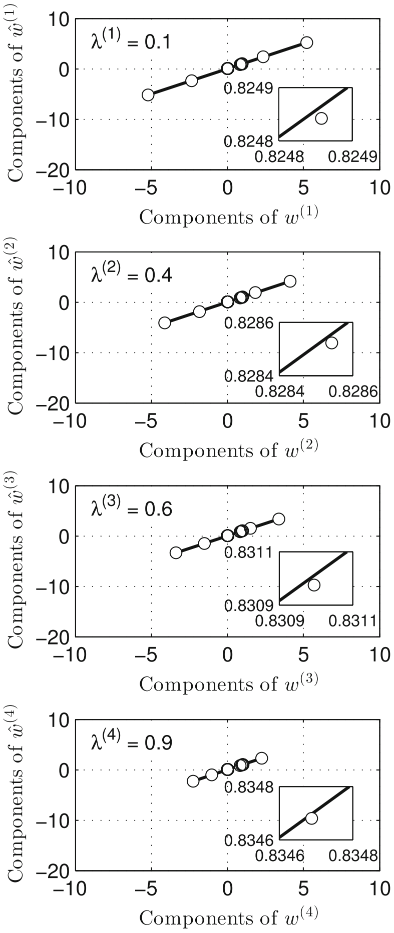

Fig. 8 Comparison of the corresponding components of vectors $\hat{w}^{(n)}$ and $w^{(n)}$ for $n = 1, 2, 3, 4$. The bisectrix of each plot is presented as a *straight line* between the extreme data points

Appendix 2

Figure 9 illustrates the evolution of the cost γ over 100 iterations of the solver. This analysis was carried out offline by using a state vector obtained in Case I during the transient response ($t = 0.69$ s). As in the online procedure, the solver was initialized with the solution obtained in the previous sampling period. As can be seen, the cost obtained after 10 iterations is very close to the value resulting from 100 iterations (default stopping criterion of the solver).

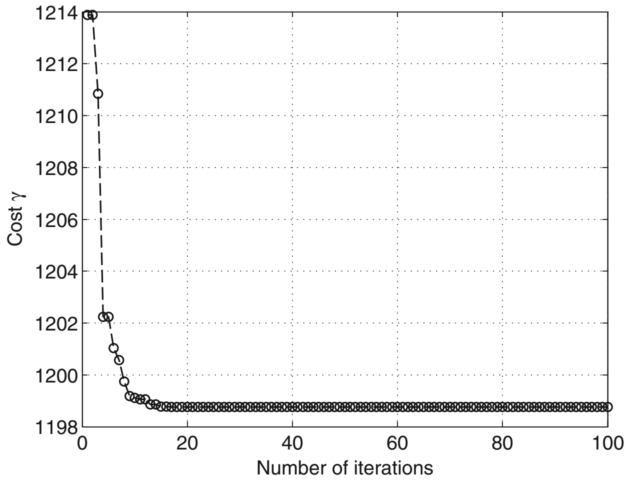

Fig. 9 Evolution of the cost γ over 100 iterations of the solver

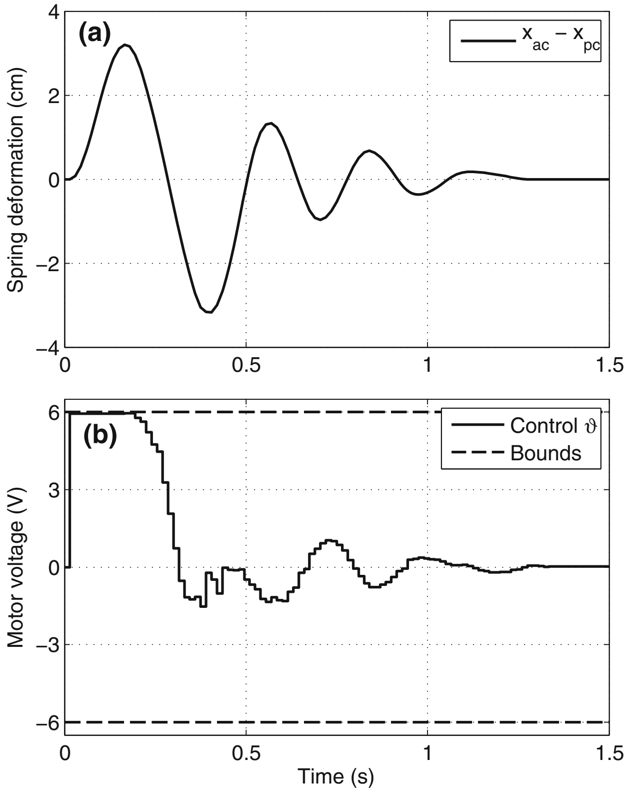

Fig. 10 Results obtained in Case I without the spring deformation constraint. **a** Spring deformation. **b** Control voltage ϑ and associated constraint bounds

Appendix 3

Figure 10 presents the spring deformation and control voltage signals obtained in Case I (uncertainty range $K_s \in [71-284]$ N/m), without the constraint on the spring deformation. As can be seen in Fig. 10a, the excursion of the spring deformation is now larger than the constraint bounds previously adopted (±1 cm), which indicates that this constraint was indeed active. It is worth noting that the control actions now get much closer to the upper bound of 6 V, as shown in Fig. 10b.

References

Allwright, J. C., & Papavasiliou, G. C. (1992). On linear programming and robust model-predictive control using impulse-responses. *Systems and Control Letters*, *18*(2), 159–164. doi:10.1016/0167-6911(92)90020-S.

Apkarian, J., Lacheray, H., & Martin, P. (2013). *Instructor workbook: Linear flexible joint experiment for Matlab/Simulink users*. Markham, ON: Quanser Consulting.

Borchers, B. (1999). CSDP, A C library for semidefinite programming. *Optimization Methods and Software*, *11*(1–4), 613–623.

Boyd, S. P., El Ghaoui, L., Feron, E., & Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*. Philadelphia, PA: SIAM.

Campo, P. J., & Morari, M. (1987). Robust model predictive control. In *Proceedings of the American control conference* (pp. 1021–1026). Minneapolis, MN.

Cao, Y. Y., & Lin, Z. (2005). Min-max MPC algorithm for LPV systems subject to input saturation. *IEE Proceedings-Control Theory and Applications*, *152*(3), 266–272. doi:10.1049/ip-cta:20041314.

Capron, B., & Odloak, D. (2013). LMI-based multi-model predictive control of an industrial C3/C4 splitter. *Journal of Control, Automation and Electrical Systems*, *24*(4), 420–429. doi:10.1007/s40313-013-0050-1.

Colombo Junior JR (2014) Implementação de um controlador preditivo robusto empregando desigualdades matriciais lineares para um sistema de dinâmica rápida. Master's dissertation, Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, Brazil.

Colombo Junior, J. R., Afonso, R. J. M., Galvão, R. K. H., & Assunção, E. (2014). Experimental evaluation of a robust predictive control using LMIs for a system with fast dynamics. In *Proceedings of the 20th Congresso Brasileiro de Automática* (pp 3244–3251). Belo Horizonte, Minas Gerais, Brazil (in Portuguese).

Cuzzola, F. A., Geromel, J. C., & Morari, M. (2002). An improved approach for constrained robust model predictive control. *Automatica*, *38*(7), 1183–1189. doi:10.1016/S0005-1098(02)00012-2.

Erdem, G., Abel, S., Morari, M., Mazzotti, M., & Morbidelli, M. (2004). Online optimization based feedback control of simulated moving bed chromatographic units. *Chemical and Biochemical Engineering Quarterly*, *18*(4), 319–328.

Feng, L., Wang, J. L., & Poh, E. K. (2006). Improved robust model predictive control with structural uncertainty. *Journal of Process Control*, *17*(8), 683–688. doi:10.1016/j.jprocont.2007.01.008.

Franklin, G. F., Workman, M. L., & Powell, J. D. (1998). *Digital control of dynamic systems* (3rd ed.). Half Moon Bay, CA: Ellis-Kagle Press.

Fujisawa, K., Kojima, M., Nakata, K., & Yamashita, M. (2002). *SDPA manual, version 6.2.0. Research Reports on Mathematical and Computer Sciences*.

Gahinet, P., & Nemirovski, A. (1997). The projective method for solving linear matrix inequalities. *Mathematical Programming*, *77*(1), 163–190. doi:10.1007/BF02614434.

Gruber, J., Ramirez, D., Alamo, T., Bordons, C., & Camacho, E. (2009). Control of a pilot plant using qp based min-max predictive control. *Control Engineering Practice*, *17*(11), 1358–1366.

Gruber, J., Bordons, C., & Oliva, A. (2012). Nonlinear MPC for the airflow in a PEM fuel cell using a Volterra series model. *Control*

Engineering Practice, *20*(2), 205–217. doi:10.1016/j.conengprac.2011.10.014.

Hensen, R. H. A., Van de Molengraft, M. J. G., & Steinbuch, M. (2003). Friction induced hunting limit cycles: A comparison between the LuGre and switch friction model. *Automatica*, *39*(12), 2131–2137. doi:10.1016/S0005-1098(03)00234-6.

Herceg, M., Kvasnica, M., & Fikar, M. (2009). Minimum-time predictive control of a servo engine with deadzone. *Control Engineering Practice*, *17*(11), 1349–1357.

Kothare, M. V., Balakrishnan, V., & Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, *32*(10), 1361–1379. doi:10.1016/0005-1098(96)00063-5.

Maciejowski, J. (2002). *Predictive Control with Constraints*. Upper Saddle River, NJ: Prentice Hall.

Maeder, U., Borrelli, F., & Morari, M. (2009). Linear offset-free model predictive control. *Automatica*, *45*(10), 2214–2222. doi:10.1016/j.automatica.2009.06.005.

Mayne, D. Q., Rawlings, J. B., Rao, C. V., & Scokaert, P. O. M. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, *36*(6), 789–814. doi:10.1016/S0005-1098(99)00214-9.

Minh, V. T., & Hashim, F. B. M. (2011). Tracking setpoint robust model predictive control for input saturated and softened state constraints. *International Journal of Control, Automation and Systems*, *9*(5), 958–965. doi:10.1007/s12555-011-0517-4.

Olsson, H., Åström, K. J., Canudas de Wit, C., Gäfvert, M., & Lischinsky, P. (1998). Friction models and friction compensation. *European Journal of Control*, *4*(3), 176–195. doi:10.1016/S0947-3580(98)70113-X.

Qin, S. J., & Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, *11*(7), 733–764. doi:10.1016/S0967-0661(02)00186-7.

Quanser (2012). *User manual—Linear flexible joint experiment: Set up and configuration*. Quanser Consulting, Markham, ON.

Quanser (Document Number 501) *User manual—Linear Motion Servo Plants: IP01 or IP02. Revision 03*. Quanser Consulting, Markham, ON.

Rahideh, A., & Shaheed, M. (2012). Constrained output feedback model predictive control for nonlinear systems. *Control Engineering Practice*, *20*(4), 431–443.

Santucci, A., Sorniotti, A., & Lekakou, C. (2014). Power split strategies for hybrid energy storage systems for vehicular applications. *Journal of Power Sources*, *258*, 395–407. doi:10.1016/j.jpowsour.2014.01.118.

Tahir, F., & Jaimoukha, I. M. (2013). Causal state-feedback parameterizations in robust model predictive control. *Automatica*, *49*(9), 2675–2682. doi:10.1016/j.automatica.2013.06.015.

Wada, N., Saito, K., & Saeki, M. (2006). Model predictive control for linear parameter varying systems using parameter dependent Lyapunov function. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *53*(12), 1446–1450. doi:10.1109/TCSII.2006.883832.

Wie, B., & Bernstein, D. S. (1992). Benchmark problems for robust control design. *Journal of Guidance, Control, and Dynamics*, *15*(5), 1057–1059. doi:10.2514/3.20949.

Yao, W. H., Tung, P. C., Fuh, C. C., & Chou, F. C. (2013). Suppression of hunting in an ILPMSM driver system using hunting compensator. *IEEE Transactions on Industrial Electronics*, *60*(7), 2586–2594. doi:10.1109/TIE.2012.2196014.

Yaramasu, V., Rivera, M., Narimani, M., Wu, B., & Rodriguez, J. (2014). Model predictive approach for a simple and effective load voltage control of four-leg inverter with an output LC filter. *IEEE Transactions on Industrial Electronics*, *61*(10), 5259–5270. doi:10.1109/TIE.2013.2297291.

Zhang, L. (2013). Automatic offline formulation of robust model predictive control based on linear matrix inequalities method. *Abstract and Applied Analysis*,. doi:10.1155/2013/380105.

Zhang, R., Wu, S., & Gao, F. (2014). Improved PI controller based on predictive functional control for liquid level regulation in a coke fractionation tower. *Journal of Process Control*, *24*(3), 125–132. doi:10.1016/j.jprocont.2014.01.004.

Zhao, D., Liu, C., Stobart, R., Deng, J., Winward, E., & Dong, G. (2014). An explicit model predictive control framework for turbocharged diesel engines. *IEEE Transactions on Industrial Electronics*, *61*(7), 3540–3552. doi:10.1109/TIE.2013.2279353.

Zheng, Z. Q., & Morari, M. (1993). Robust stability of constrained model predictive control. In *Proceedings of the American control conference* (pp. 379–383). San Francisco, CA.