Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 48-23 (2015) 422–427

Real-time MHE-based nonlinear MPC of a Pendubot system ⋆

M. Gulan, M. Salaj, M. Abdollahpouri, B. Rohaľ-Ilkiv ⋆

⋆ *Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology, Bratislava, Slovakia (e-mail: <martin.gulan@stuba.sk).*

Abstract: This paper addresses the real-time optimal control of a Pendubot using nonlinear model predictive control (NMPC) combined with nonlinear moving horizon estimation (MHE). This fast, under-actuated nonlinear mechatronic system apparently poses a challenging benchmark problem that may benefit from a nonlinear optimization scheme. To overcome the related computational difficulties we make use of the ACADO Code Generation tool allowing to export a highly efficient Gauss-Newton real-time iteration algorithm tailored to the nonlinear optimal control and estimation problem while respecting the imposed constraints. We show experimental results illustrating the overall closed-loop control performance, as well as the advantages of the nonlinear MHE-based NMPC scheme.

Keywords: Pendubot control, under-actuated system, fast NMPC, nonlinear optimization, real-time control and estimation.

1. INTRODUCTION

The double inverted Pendubot, introduced in Spong and Block (1995), is a well-known academic benchmark from the class of under-actuated mechatronic systems. These tend to be often used to demonstrate various concepts in linear and nonlinear control such as swing-up, stabilization or trajectory tracking of unstable systems; see e.g. Xin and Liu (2014); Choukchou-Braham et al. (2014). With regards to process nonlinearities, constraints, and stability guarantees, optimization-based techniques such as MPC can be thus sought as a systematic methodology to address these challenging problems. The Pendubot, as a nonlinear under-actuated mechatronic system with fast dynamics, though lacks a complex control strategy capable of addressing the aforementioned control problems and phenomena.

With the recent development of efficient convex programming solvers, linear MPC quickly became well-established in both academia and industry, in particular when aiming at fast systems and processes, often by means of embedded control hardware. Similarly, in case of fast nonlinear mechatronic systems, as is the Pendubot, an increasing interest is in achieving reliable and accurate performance over the widest operating range possible while accounting for model nonlinearities. In such control scenarios, instead of re-linearizing the system dynamics as often as possible, the related optimal control problem (OCP) can be instead cast as a nonlinear MPC problem. Its non-convex nature and inherent convergence issues, however, typically come at price of fairly higher computational requirements for effective solution, as compared to linear formulations. The applicability of NMPC was therefore primarily associated with systems exhibiting rather slower dynamics. This was the motivation for several approaches to tackle the control of fast, nonlinear, and constrained systems, which have been proposed over the last decade; among them the continuation/GMRES method, the advanced step controller, or the real-time iteration scheme. For a comprehensive overview of algorithms aiming at nonlinear optimal control we refer to Diehl et al. (2009); Johansen (2011).

Within this study we propose to employ an MHE-NMPC algorithm tailored for the Pendubot's unstable equilibria tracking problem. It builds upon the real-time iteration (RTI) scheme of Diehl et al. (2002) exploiting the method of direct multiple shooting (Bock and Plitt. (1984)). Since targeting a system with fast dynamics at high sampling rates, to address the restrictive computational times typically arising in conventional NMPC/MHE we make use of an efficient customized source code generated by means of the `ACADO` toolkit of Houska et al. (2011a), namely its `Code Generation` tool (Houska et al. (2011b)). The efficacy of the RTI scheme stems mainly from performing only one Newton-type sequential quadratic programming (SQP) iteration per sampling instant (Diehl et al. (2005)). In contrast to numerous works, wherein the Pendubot-like systems are usually controlled by a switching swing-up/balancing strategy for a single equilibrium, the proposed NMPC scheme exploits the full nonlinear dynamic model and thus allows for more complex control tasks. As outlined above, the control framework is augmented by the complementary nonlinear MHE scheme to estimate the unmeasured states and unknown parameters of the system. The overall performance is verified experimentally.

⋆ The authors gratefully acknowledge the contribution of the Slovak Research and Development Agency under the projects APVV-0090-10 and APVV-0015-12, the Scientific Grant Agency of the Slovak Republic under the grant 1/0144/15, and the internal grant of the Slovak University of Technology for support of young researchers. This work has also benefited from collaboration within the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement no 607957 (TEMPO).

10.1016/j.ifacol.2015.11.315

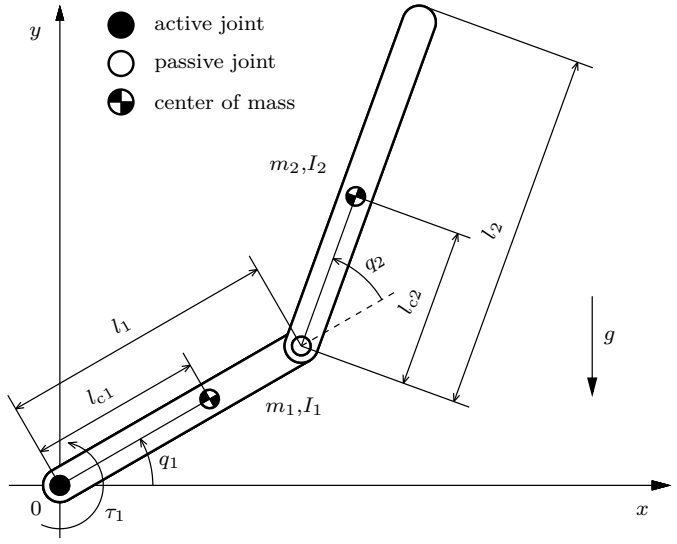

Fig. 1. Schematic of the Pendubot in the relative coordinate system.

2. SYSTEM MODEL

The Pendubot is essentially a two-link planar robot that is referred to as under-actuated since the number of its control inputs is less than the number of its degrees of freedom, which renders it challenging to control. It has a single actuator at the base, "shoulder", of the first link, whilst the "elbow" joint between the two links is unactuated and thus allowed to swing freely. Its mathematical model can be derived by means of the Lagrange formalism

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial}{\partial \dot{q}_k}\mathcal{L}(q,\dot{q}) - \frac{\partial}{\partial q_k}\mathcal{L}(q,\dot{q}) + \frac{\partial}{\partial \dot{q}_k}\mathcal{R}(\dot{q}) = \tau \quad k = 1,2 \quad (1)$$

with the Lagrange function $\mathcal{L}(q,\dot{q}) = \mathcal{T}(q,\dot{q}) - \mathcal{V}(q)$ defined as the difference between the kinetic energy and the potential energy, respectively, and the Rayleigh dissipation function $\mathcal{R}$ accounting for friction. The generalized coordinates summarized in the vector $q = [q_1, q_2]^T$ here stand for the angular positions of the two links, and $\tau = [\tau_1, 0]^T$ denotes the external control force vector.

By applying (1), the resulting equation of motion can be cast in the standard vector/matrix form:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + g(q) = \tau \quad (2)$$

where $D(q)$ is the symmetric positive definite inertia matrix, $C(q,\dot{q})$ contains the Coriolis and centrifugal terms, $F(\dot{q})$ is the vector of viscous frictional terms, and $g(q)$ denotes the vector of gravitational terms. For the Pendubot system, schematically illustrated in Fig. 1, the following quantities can be obtained:

$$D(q) = \begin{bmatrix} \theta_1 + \theta_2 + 2\theta_3\cos q_2 & \theta_2 + \theta_3\cos q_2 \\ \theta_2 + \theta_3\cos q_2 & \theta_2 \end{bmatrix},$$

$$C(q,\dot{q}) = \begin{bmatrix} -\theta_3\sin(q_2)\dot{q}_2 & -\theta_3\sin(q_2)\dot{q}_2 - \theta_3\sin(q_2)\dot{q}_1 \\ \theta_3\sin(q_2)\dot{q}_1 & 0 \end{bmatrix},$$

$$g(q) = \begin{bmatrix} \theta_4 g\cos q_1 + \theta_5 g\cos(q_1+q_2) \\ \theta_5 g\cos(q_1+q_2) \end{bmatrix},$$

where

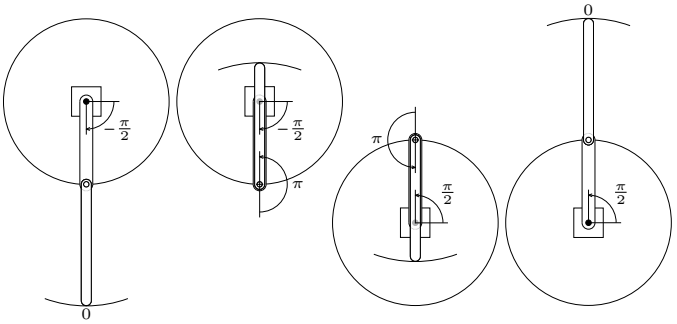

Fig. 2. Illustration of the Pendubot's un/stable equilibria w.r.t. the arrangement of its links; from left to right: down-down ⇊, down-up ⇵, up-down ⇅, up-up ↑↑.

$$\theta_1 = m_1 l_{c1}^2 + m_2 l_1^2 + I_1 \qquad \theta_4 = m_1 l_{c1} + m_2 l_1$$
$$\theta_2 = m_2 l_{c2}^2 + I_2 \qquad \theta_5 = m_2 l_{c2}$$
$$\theta_3 = m_2 l_1 l_{c2}$$

are the parameter equations necessary for the subsequent control design. Taking into account viscous friction in both joints, the dynamic system (2) can be rewritten as follows:

$$\ddot{q}_1 = \frac{1}{\theta_1\theta_2 - \theta_3^2\cos^2 q_2}\Big[\theta_2\theta_3\sin q_2(\dot{q}_1+\dot{q}_2)^2 + \theta_3^2\cos q_2\sin(q_2)\dot{q}_1^2 - \theta_2\theta_4 g\cos q_1 + \theta_3\theta_5 g\cos q_2\cos(q_1+q_2) + \theta_2\tau_1 - \theta_2 b_1\dot{q}_1 + (\theta_2+\theta_3\cos q_2)b_2\dot{q}_2\Big] \quad (3)$$

$$\ddot{q}_2 = \frac{1}{\theta_1\theta_2 - \theta_3^2\cos^2 q_2}\Big[-\theta_3(\theta_2+\theta_3\cos q_2)\sin q_2(\dot{q}_1+\dot{q}_2)^2 - (\theta_1+\theta_3\cos q_2)\theta_3\sin(q_2)\dot{q}_1^2 + (\theta_2+\theta_3\cos q_2)(\theta_4 g\cos q_1 - \tau_1) - (\theta_1+\theta_3\cos q_2)\theta_5 g\cos(q_1+q_2) + (\theta_2+\theta_3\cos q_2)b_1\dot{q}_1 - (\theta_1+\theta_2+2\theta_3\cos q_2)b_2\dot{q}_2\Big] \quad (4)$$

yielding Pendubot's non-linear equations of motion, which are used for design of a nonlinear predictive controller.

By introducing the state vector $x(t) = [x_1, \dot{x}_1, x_2, \dot{x}_2]^T = [q_1, \dot{q}_1, q_2, \dot{q}_2]^T$ and denoting $u = \tau_1$, the equations of motion of the Pendubot may be formulated as follows:

$$\begin{cases} \dot{x} = f(x,u,b) \\ y = h(x,b) \end{cases} \quad (5)$$

Note that for $u = 0$ the system (5) exhibits four equilibrium positions, see Fig. 2, the first being a stable and the other three unstable ones. From these, challenging from the control standpoint are mainly the down-up (further ⇵) and up-up (↑↑) Pendubot's unstable positions. In the scope of this work is a point-to-point transition task between different equilibria, by exploiting the system's nonlinear dynamics (5) subject to constraints.

The parameters such as masses m_i, the pendulum links' lengths l_i, and the distance l_{ci} of the center of mass to the corresponding joint are directly measurable, and read as $m_1 = 0.265\,\mathrm{kg}$, $m_2 = 0.226\,\mathrm{kg}$, $l_1 = 0.206\,\mathrm{m}$, $l_2 = 0.298\,\mathrm{m}$, $l_{c1} = 0.107\,\mathrm{m}$ and $l_{c2} = 0.133\,\mathrm{m}$. Even the moments of inertia I_1 and I_2 can be analytically determined accurately enough. On the other hand, the viscous friction coefficients $b = [b_1\, b_2]^T$ are due to their uncertainty and sensitivity intended to be estimated together with the unmeasured system states via the MHE scheme discussed further.

3. CONTROLLER SYNTHESIS

This section states the formulation and implementation of the present control scheme based on combined nonlinear MHE and MPC for the fast Pendubot system. For clarity, we introduce the formulation first by deriving the estimation scheme.

3.1 MHE formulation

As it tends to be the case in many practical applications, also the number of available measurements provided by Pendubot's sensors is smaller than the number of states within its dynamic model. Specifically, only the angular positions of its links are measured by means of dedicated encoders, leaving the respective velocities up to the estimator. The state vector is moreover augmented by the unknown model parameters $b = [b_1\, b_2]^T$, i.e. viscous friction coefficients. We remark, that compared to other nonlinear estimation methods, the optimization-based MHE allows for joint state and parameter estimation within one problem, and importantly, for the incorporation of constraints.

The nonlinear MHE state and parameter estimation problem is proposed to take form of a constrained least-squares (LSQ) dynamic optimization problem

$$\begin{aligned}\min_{x(\cdot),b,u(\cdot)}\ & \frac{1}{2}\Big(\Big\|\begin{matrix}x-\bar{x}\\ b-\bar{b}\end{matrix}\Big\|^2_{P_\mathrm{E}}\Big)_{t=t_0-T_\mathrm{E}} \\ & + \frac{1}{2}\int_{t_0-T_\mathrm{E}}^{t_0}\left(\|h(x,b)-\bar{y}\|^2_{Q_\mathrm{E}} + \|u-\bar{u}\|^2_{R_\mathrm{E}}\right)\mathrm{d}t, \qquad (6a)\\ \text{s.t. }\ & \frac{\mathrm{d}}{\mathrm{d}t}x = f(x,u,b), \qquad (6b)\\ & h_\mathrm{ineq}(x,u,b) \le 0 \qquad (6c)\end{aligned}$$

solved repeatedly at every time instant $t_k = kT_\mathrm{s}$ ($k = 0,1,\dots$), where T_s is the sampling time. The moving horizon objective penalizes the deviation between the measurement model $h(\cdot)$ and the set of measurement data $\bar{y}$. Note that the controls are made decision variables and included in the objective to account for the noise collected during signal transfer and actuator inaccuracies. Additionally, the first summand in (6a) is considered to approximate the so-called arrival costs and thus summarize the information prior to $t_0 - T_\mathrm{E}$, where T_E is the estimation horizon. Within this study we use the approach of Kühl et al. (2011) to obtain the a priori estimates $\bar{x}$ and $\bar{b}$. The LSQ terms are penalized by appropriately chosen weighting matrices $Q_\mathrm{E} \succeq 0$, and $P_\mathrm{E}, R_\mathrm{E} \succ 0$. The equality (6b) describes the propagation of the system state. The upper and lower bounds for states and parameters can be imposed via (6c). This way estimated states and unknown model parameters are subsequently fed as the initial state estimate $(\hat{x}, \hat{b})$ to the NMPC controller.

3.2 NMPC formulation

The considered nonlinear MPC scheme is based on repeatedly solving the following optimal control problem (OCP) in effort of finding a function

$$\begin{aligned}u^\star(\cdot) = \underset{u(\cdot),x(\cdot)}{\arg\min}\ & \frac{1}{2}\left(\|x-x^\mathrm{ref}\|^2_P\right)_{t=t_0+T_\mathrm{P}} \\ & + \frac{1}{2}\int_{t_0}^{t_0+T_\mathrm{P}}\left(\|x-x^\mathrm{ref}\|^2_Q + \|u-u^\mathrm{ref}\|^2_R\right)\mathrm{d}t, \qquad (7a)\\ \text{s.t. }\ & x(t_0) = \hat{x}(t_0), \qquad (7b)\\ & \frac{\mathrm{d}}{\mathrm{d}t}x = f(x,u,\hat{b}), \qquad (7c)\\ & h_\mathrm{ineq}(x,u) \le 0 \qquad (7d)\end{aligned}$$

where T_P denotes the NMPC control horizon. This LSQ cost function penalizes the deviation of the process control inputs u and states x from their reference trajectories, u^ref and x^ref, respectively. Its first summand evaluates the final costs raised by controlled variables at the given end time t_{k+T_P}. This so-called Mayer term, in this context usually referred to as cost-to-go or terminal penalty function is often included for stability reasons. As usual in tracking MPC applications, the norms in the objective are weighted with matrices $Q, P \succeq 0$ and $R \succ 0$.

The nonlinear right-hand side function $f(x,u,\hat{b})$ in (7c) describes the system dynamics with ordinary differential equations (ODEs). With regards to the dependence of the optimal control $u^\star(t_k, x_k, b_k)$ on state x_k and model parameters b, the full initial state (x_k, b_k) must be known. For this purpose we employ the aforementioned estimation scheme implied by the initial state constraint (7b). Vectors $\hat{x}(t_k)$, $\hat{b}(t_k)$ are the system state and chosen model parameters estimated at time instant t_k by solving the nonlinear MHE problem (6). This way the NMPC problem (7) can be regarded as a parametric nonlinear OCP. Finally, the inequality (7d) lumps together state and input constraints imposed on the system.

The nonlinear MHE and NMPC are treated here together since they are almost identical in the approach and implementation, even though they solve two different, yet complementary problems. Therefore, we apply the same solution methods to tackle both the MHE and the NMPC problem, as shown in e.g. Zanon et al. (2013); Debrouwere et al. (2014). In the following, we recall an efficient procedure to handle the infinite-dimensional least-squares objectives subject to nonlinear dynamics and constraints, as in (6) and (7).

3.3 Real-time implementation approach

With respect to the unstable model dynamics, the problems (6) and (7) are conveniently treated using the simultaneous approaches Diehl et al. (2009); Johansen (2011), such as the direct multiple shooting, which transforms the infinite-dimensional optimal control or estimation problem into a finite-dimensional optimization problem (Bock and Plitt. (1984)). The resulting discretized OCP in fact renders a least-square nonlinear programming (NLP) problem, which can be efficiently solved e.g. by the sequential quadratic programming or interior-point approach.

It is well known that the computational effort necessary to solve nonlinear optimal control and estimation problems exactly tends to easily become prohibitive, in particular for systems with fast-evolving dynamics. To overcome this issue, the real-time iteration scheme of Diehl et al. (2002) performs only a single SQP iteration with Gauss-Newton Hessian approximation per sampling time. Although solv-

ing the problems only approximately, its established efficacy is supported by employing the aforementioned simultaneous NLP parametrization, direct multiple-shooting method with/without condensing, and the so-called initial value embedding which constraints the initial value in the NLP to coincide with the estimated state of the system. This way most of the computations are performed before the current estimated state becomes available. The computations within each iteration are therefore divided into a more demanding preparation phase (linearization, condensing, etc.) and a shorter feedback phase solving just one dense/sparse QP subproblem.

Needless to say, to solve a whole sequence of "neighbouring" NLPs, in both the MHE and NMPC formulation we efficiently utilize a so-called shift initialization Diehl et al. (2009) to initialize the subsequent problems based on previous information. For a detailed description, contractivity and stability proof of the RTI scheme we refer to Diehl et al. (2002, 2005). As outlined in the introductory section, in order to numerically solve the optimization problems we make use of the `ACADO Code Generation` tool (Houska et al. (2011b)), exploiting direct multiple-shooting, RTI and sequential quadratic programming.

4. EXPERIMENTAL RESULTS

In this section we put forward and analyze the experimental results obtained for the nonlinear Pendubot system using the on-line estimation-based predictive control scheme introduced in Section 3.

4.1 Hardware & software approach

The real-time experiments were performed on a real-life laboratory setup designed at the authors' workplace. The actuator of the system is a Mitsubishi HC-KFS43 servo motor with a WITTENSTEIN alpha CP060 gearbox. It utilizes a Mitsubishi MR-J2S-40A control unit working in torque control mode. It is also equipped with an absolute encoder allowing the real-time measurement of the angular position of the first link. In order to measure the angle that the second – free link makes with the actuated arm, an incremental rotary encoder OMRON E6B2-C is used. For more details, model parameters and previous work, we refer to Gulan et al. (2014).

As outlined, we make use of the `ACADO Code Generation` tool for exporting an instance of the RTI scheme, tailored for both the NMPC and the MHE optimization problem. The exported solvers, provided as a self-contained plain C code, are utilized together within one custom executable code, and subsequently called repeatedly at each sampling instant. To efficiently solve the underlying QP subproblems of the SQP-based RTI scheme, we make use of efficient quadratic programming solvers. In particular, for the MHE problem we utilize a condensing-based `qpOASES` parametric solver implementing the online active set strategy proposed in Ferreau et al. (2008). Based on experimental observations that lead to choosing a rather longer NMPC prediction horizon of 100 steps (compared to the sufficient 50 steps for MHE), we address the QPs arising in the NMPC problem by an open-source implementation of the `qpDUNES` solver Frasch et al. (2014), based upon the dual Newton strategy which combines the warm-starting capabilities of active-set methods and the structure-exploiting features of interior-point methods. We remark that the choice to use a dense solver within MHE problem is relevant due to a still tractable problem size, yet supported by the fact that the latter solver strategy is not interfaced for arrival cost update's computation within the `ACADO` framework at the time of herein presented study.

The presented estimator-controller scheme is implemented on an ADLINK MXC-6321 embedded computer (Intel i7, 2.3 GHz, 12 GB RAM) featuring Ubuntu 14.04 with an RT preemptive real-time kernel as operating system.

4.2 Experimental setup

System dynamics given by the continuous ODE model (3)-(4) is parametrized by the multiple-shooting technique using uniform intervals of $T_\mathrm{s} = 10\,\mathrm{ms}$. For discretization over the shooting intervals we use an implicit Gauss-Legendre Runge-Kutta integrator of order two. The estimation and prediction horizon are chosen as $T_\mathrm{E} = 0.5\,\mathrm{s}$ and $T_\mathrm{C} = 1\,\mathrm{s}$, respectively. Moreover, at each sampling time the related nonlinear optimization problems are solved with the following bounds on controls and state

$$-2\,\mathrm{N\,m} \le u \le 2\,\mathrm{N\,m}$$
$$-2\pi\,\mathrm{rad} \le x_1 \le 0\pi\,\mathrm{rad}.$$

while requiring nonnegative values for the estimated parameters, i.e. $b_1, b_2 \ge 0$. Within the above box constraints, the system input u is constrained by the internal limitations of the actuator. Similarly, the constraint imposed on the Pendubot arm's angular position x_1 is inherited from its physical limits to prevent twisting of the encoder cable.

The state and control references within the NMPC scheme are kept zero except of the references for angular positions of both links which are changed online to track the desired equilibria, i.e. $[\cos x_1^{\mathrm{ref}}, \sin x_1^{\mathrm{ref}}, 0, \cos x_2^{\mathrm{ref}}, \sin x_2^{\mathrm{ref}}, 0]^T$ [1] and $u^{\mathrm{ref}} = 0$. The respective NMPC weighting matrices were set as $P = Q = \mathrm{diag}([1e2, 1e2, 1, 1e2, 1e2, 1])$, with input weights $R_i = \{1.5e2, 1e1, 4e1, 3e1\}$ heuristically tuned for each set-point in order of switching. Next, the MHE weights were selected as $Q_E = \mathrm{diag}([2, 2])$, $R_E = 0.1$ and P_E defined following Kühl et al. (2011). Note that the units of the weights are consistent with the variables in order to yield a dimensionless cost, and are omitted here for brevity.

4.3 Real-time estimator-controller performance

The performance of the exported MHE-NMPC scheme is demonstrated in a point-to-point motions' scenario where multiple set-point changes are imposed to the controller, corresponding to the unstable equilibrium positions of the system. Specifically, starting off from its slightly disturbed ⇊ stable equilibrium, the Pendubot is shortly afterwards stepwise prompted to achieve the unstable equilibria in order of ⇵-⇅-↑↑. This unified swing-up+balancing task is finally concluded by performing the so-called swing-down maneuver, i.e. returning both links into the initial resting

[1] This form of state and its associated reference vector was utilized within the NMPC objective to ease the reference angle tracking since each of the two links may, in theory, exhibit infinitely many possible angular values (with 2π period) for a given equilibrium.

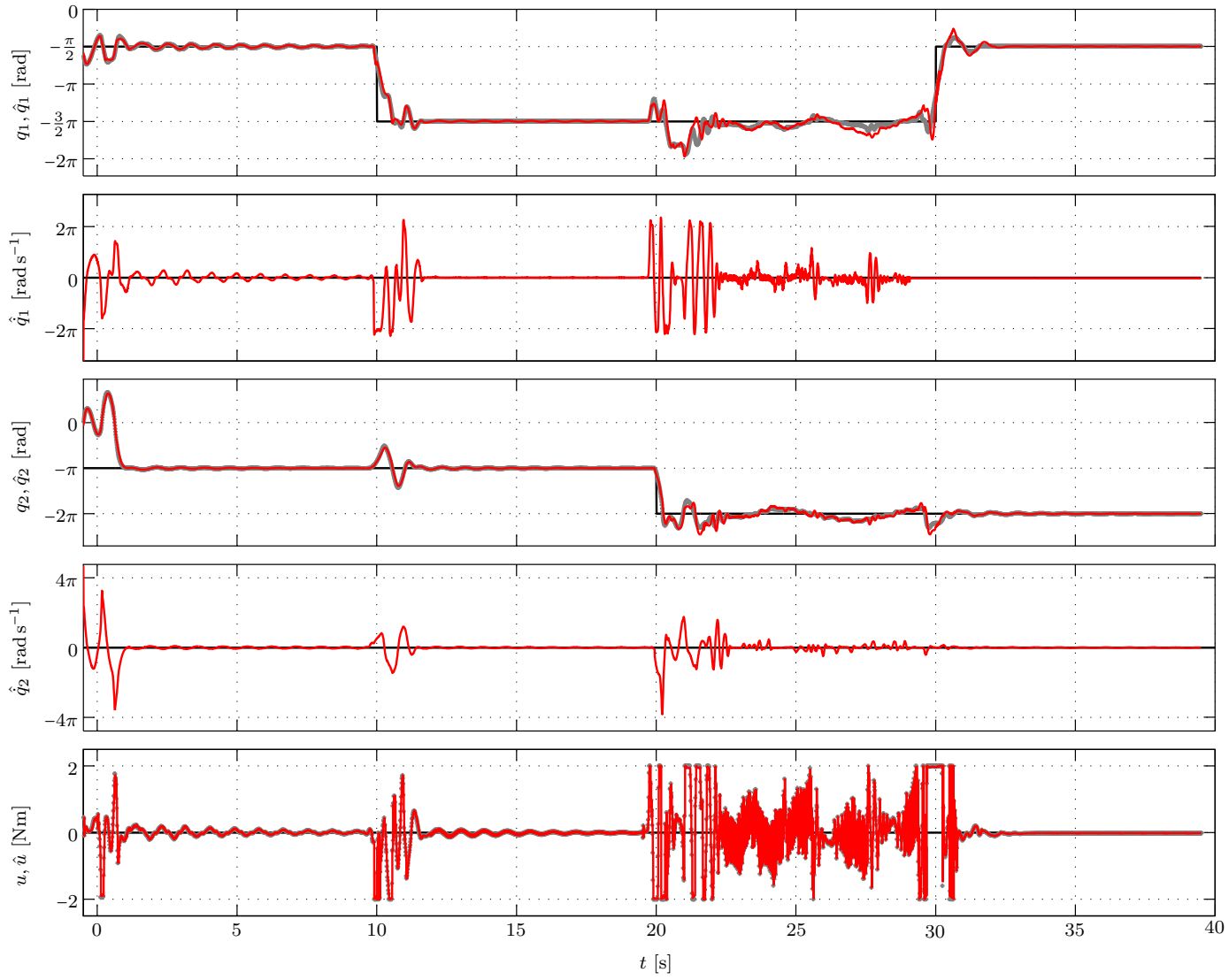

Fig. 3. Experimental results: red line - system state and control input estimates, gray markers - measurements, black line - references; the equilibrium set-point tracking task is performed in 10 s intervals, in the order of ⇵-↑↓-↑↑-⇊.

position in the shortest time possible. Notice that within the 50 samples prior to the main part of the experiment, only the MHE routine is engaged in order to fill its buffer for providing accurate estimates for the NMPC algorithm which is triggered thereafter (Fig. 3). During this short initialization phase the system is excited by a sinusoidal signal in the vicinity of the ⇊ stable equilibrium position.

The acquired angular position measurements for the given control scenario are denoted in Fig. 3. by grey markers (+), together with their estimates depicted with solid red lines. The remaining states, i.e. estimated angular velocities, and the corresponding control input acting on the first link are shown as well. We recall that the relatively smooth velocity estimates are fed to the controller instead of the rather noisy ones calculated by means of finite differences (not shown). The reliable MHE estimates fairly contribute to the robustness of the NMPC scheme, which has been observed mainly in the otherwise critical transfers between set-points. As can be seen in Fig. 3, expectedly the most challenging set-point to track is the ↑↑ position when both links simultaneously aim to achieve their unstable equilibria. In this configuration, the system is clearly most sensitive to any external disturbances, unmodelled dynamics or mismatch between the actual system behaviour and its mathematical representation. This fact has also reflected into the monitored Karush-Kuhn-Tucker tolerance which indicates the measure of suboptimality of the solution, closely related to the problem nonlinearity. Similarly, the estimates of the friction coefficients (not shown for brevity) were fairly influenced during this control phase, possibly suggesting to devise a more advanced friction model that would capture this phenomenon in a more complex fashion, and thus extend and further improve the overall performance of the estimator-controller scheme.

Nevertheless, based on the obtained experimental results, the investigated scheme has been shown to perform adequately well for the equilibrium point-to-point transition objective of a fast mechatronic system, while respecting the upper and the lower bounds on both input and state variable(s).

Table 1. Average and maximum computation times of one MHE-NMPC iteration in [ms].

		qpOASES		qpDUNES	
	phase	t_{avg}	$t_{\max}$	t_{avg}	$t_{\max}$
MHE ($N_{\text{E}} = 50$)	*estimation*	0.243	1.770		
	preparation	0.766	1.410		
	full RTI	1.092	2.683		
NMPC ($N_{\text{C}} = 100$)	*feedback*			0.256	0.600
	preparation			1.073	1.400
	full RTI			1.374	1.800
total execution time†		$t_{\text{avg}} = 2.426$, $t_{\max} = 4.083$			

† Based on particular MHE-NMPC iterations.

To assess the computational complexity of the scheme, in Table 1 reports the execution times of its main algorithmic routines. For this purpose, we treat the iterations of the RTI scheme for both the MHE and NMPC problem separately, and split the computation effort of each into a preparation phase and a feedback (estimation, in case of MHE) phase. As outlined earlier, for solution of the nonlinear MHE problem we utilize the condensing qpOASES-based approach while the longer-horizon NMPC problem is tackled by the sparsity exploiting qpDUNES-based approach. Both approaches clearly evidence that the RTI execution time is dominated by the preparation phase, where the effort is put mainly into linearization of the NLP. This is in particular significant for the former approach which additionally carries out a condensing routine resulting in the reduced-size QP. Overall, the remaining feedback/estimation phase is subsequently devoted to the solution of a single sparse/dense QP subproblem. The results indicate that the lower per-iteration complexity of the sparse QP strategy (see Frasch et al. (2014)) allows for computation times to scale better with the problem size, as compared to the condensing-based approach. The worst-case computational effort over all real-time steps is reported as well. The total execution time of the estimator-controller including all the auxiliary tasks (data communication, logging, etc.) on average yields 2.43 ms, with peak computation time at 4.08 ms, which still offers a reasonable safety margin within the sampling time of 10 ms.

5. CONCLUSION

In this paper, we presented a real-time implementation of the combined NMHE-NMPC framework for the Pendubot control task. This efficient RTI-based scheme exploiting the full nonlinear model allowed for a good unstable equilibria-tracking performance of the NMPC controller, to certain extent due to the necessary MHE estimates of unmeasured system states. The solution approach has been experimentally verified to be fast enough for real-time deployment on a laboratory Pendubot system, leading to feasible computation times and a good estimation/control performance. The further work may clearly benefit from a more advanced friction model, and computationally from parallelization of the two nonlinear optimization routines, which is the subject of ongoing research.

REFERENCES

Bock, H.G. and Plitt., K.J. (1984). A multiple shooting algorithm for direct solution of optimal control problems. In *9th IFAC World Congress*, 243–247.

Choukchou-Braham, A., Cherki, B., Djemaï, M., and Busawon, K. (2014). *Analysis and Control of Underactuated Mechanical Systems*. Springer.

Debrouwere, F., Vukov, M., Quirynen, R., Diehl, M., and Swevers, J. (2014). Experimental validation of combined nonlinear optimal control and estimation of an overhead crane. In *19th IFAC World Congress*, 9617–9622.

Diehl, M., Bock, H.G., Schlöder, J., Findeisen, R., Nagy, Z., and Allgöwer, F. (2002). Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, 12(4), 577–585.

Diehl, M., Bock, H.G., and Schlöder, J. (2005). A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Control and Optimization*, 43(5), 1714–1736.

Diehl, M., Ferreau, H.J., and Haverbeke, N. (2009). Efficient numerical methods for nonlinear MPC and moving horizon estimation. In L. Magni, D.M. Raimondo, and F. Allgöwer (eds.), *Nonlinear Model Predictive Control - Towards New Challenging Applications*, number 384 in Lecture Notes in Control and Information Sciences, 391–417. Springer-Verlag.

Ferreau, H.J., Bock, H.G., and Diehl, M. (2008). An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18(8), 816–830.

Frasch, J.V., Vukov, M., Ferreau, H.J., and Diehl, M. (2014). A new quadratic programming strategy for efficient sparsity exploitation in SQP-based nonlinear MPC and MHE. In *19th IFAC World Congress*, 2945–2950.

Gulan, M., Salaj, M., and Rohaľ-Ilkiv, B. (2014). Achieving an equilibrium position of pendubot via swing-up and stabilizing model predictive control. *Journal of Electrical Engineering*, 65(6).

Houska, B., Ferreau, H.J., and Diehl, M. (2011a). ACADO toolkit - an open-source framework for automatic control and dynamic optimization. *Optimal Control Applications and Methods*, 32(3), 298–312.

Houska, B., Ferreau, H.J., and Diehl, M. (2011b). An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range. *Automatica*, 47(10), 2279–2285.

Johansen, T.A. (2011). Introduction to nonlinear model predictive control and moving horizon estimation. In M. Huba, S. Skogestad, M. Fikar, M. Hovd, T.A. Johansen, and B. Rohaľ-Ilkiv (eds.), *Selected Topics on Constrained and Nonlinear Control*, 187–239. STU Bratislava - NTNU Trondheim.

Kühl, P., Diehl, M., Kraus, T., Schlöder, J.P., and Bock, H.G. (2011). A real-time algorithm for moving horizon state and parameter estimation. *Computers & Chemical Engineering*, 35(1), 71–83.

Spong, M.W. and Block, D.J. (1995). The pendubot: a mechatronic system for control research and education. In *34th Conference on Decision and Control*, 555–556.

Xin, X. and Liu, Y. (2014). *Control Design and Analysis for Underactuated Robotic Systems*. Springer-Verlag.

Zanon, M., Gros, S., and Diehl, M. (2013). Rotational start-up of tethered airplanes based on nonlinear MPC and MHE. In *European Control Conference 2013*, 1023–1028.