Comparison of Linear and Nonlinear Controllers on a Mechatronic System

Leo Patrlj*, Branimir Novoselnik*, Mato Baotić*
* University of Zagreb Faculty of Electrical Engineering and Computing,
Unska 3, 10000 Zagreb, Croatia
{leo.patrlj, branimir.novoselnik, mato.baotic}@fer.hr

***Abstract*—This technical paper presents the application of various control methods on a mechatronic laboratory setup of a rotary inverted pendulum. The full nonlinear dynamic model of the pendulum is obtained using Euler-Langrange method and it is further simplified for the purpose of designing different controllers for stabilization of the pendulum in the upright position. In particular, we compare two nonlinear control methods (sliding mode control and model predictive control) to two classical linear control methods (proportional-derivative control and linear-quadratic control). All considered control methods are briefly explained and corresponding controllers are derived and tested on the experimental setup.**

***Keywords*—sliding mode control, model predictive control, linear-quadratic control, proportional-derivative control, mechatronic system, rotary pendulum.**

I. INTRODUCTION

The rotary inverted pendulum, also called the Furuta pendulum [1], consists of a motor driven arm which rotates around a horizontal plane with a pendulum rod attached vertically to it at its end so it can swing freely. By controlling the input voltage of the DC motor the goal is to stabilize the vertical pendulum rod in the upright position as well as to control the angle of the pendulum arm in the horizontal plane. Because of its nonlinear dynamics and an unstable equilibrium point while in the upright position the pendulum is frequently used for testing various linear and nonlinear control methods.

In this brief technical paper two nonlinear control methods are compared to two linear control methods using the Furuta pendulum as an example. The goal is to observe the robustness, reference tracking and overall stabilization of the pendulum depending on the control method used and to analyze the resulting behavior of the system.

In Section II the mathematical model of the pendulum and the custom built experimental setup which is a physical realization of the pendulum is presented. In Section III various control methods for stabilization of the pendulum in the upright position are discussed. In particular, two linear methods (proportional-derivative and linear quadratic regulators) and two nonlinear methods (model predictive control and sliding-mode control) are briefly described. All four control methods are tested on the experimental setup in Section IV. Concluding remarks are given in Section V.

II. PENDULUM MODEL AND EXPERIMENTAL SETUP

The pendulum itself consists of two perpendicular arms, one horizontal and one vertical, attached to a DC motor that controls the horizontal arm. The angle at which the horizontal arm rotates is denoted as θ and the angle at which the vertical rod rotates is denoted as α.

Differential equations that describe the dynamic behavior of the pendulum are obtained using Euler-Lagrange equations as described in [2]:

$$\ddot{\theta}(\hat{J}_0 + \hat{J}_2 \sin^2(\alpha)) - m_2 L_1 l_2 \cos(\alpha) \cdot \ddot{\alpha} + \\ + m_2 L_1 l_2 \sin(\alpha) \cdot \dot{\alpha}^2 + \dot{\alpha}\dot{\theta}\hat{J}_2 \sin(2\alpha) = \tau - b_1 \dot{\theta}, \tag{1}$$

$$\ddot{\theta} m_2 L_1 l_2 \cos(\alpha) - \hat{J}_2 \cdot \ddot{\alpha} + \frac{1}{2}\hat{J}_2 \sin(2\alpha) \cdot \dot{\theta}^2 + \\ + g m_2 l_2 \sin(\alpha) = b_2 \dot{\alpha}. \tag{2}$$

In the equations above, $\hat{J}_0$ denotes the moment of inertia of the arm around the z axis, $\hat{J}_2$ denotes the moment of inertia of the pendulum around its pivot point, m_2 is the pendulum rod mass, l_2 is the length from the center of mass of the pendulum rod to its pivot point, L_1 is the length of the pendulum arm, b_1 the viscous friction coefficient of the DC motor, and b_2 is the viscous friction coefficient of the pendulum rod. The term τ is torque generated by the DC motor. By neglecting the dynamics of the electrical part of the DC motor, the generated torque can be expressed as the following static function of the voltage u applied to the motor and its current speed of rotation $\dot{\theta}$:

$$\tau = \frac{n c_m (u - n c_e \dot{\theta})}{R_a}, \tag{3}$$

where n represents the gearbox ratio, R_a armature resistance, and c_e and c_m are back EMF and torque motor constants. With this simplification, the nonlinear pendulum model (1)-(2) comprises four state variables (θ, α, $\dot{\theta}$, $\dot{\alpha}$) and one control input variable u.

A laboratory experimental setup, as described in [3], is shown in Fig. 1. Only the first two state variables θ and α are directly measurable by equipped encoders on the motor (for measuring angle θ) and on the pendulum module (for measuring angle α). The custom-made control board enables serial communication in real time with a computer (through USB port) where a controller can easily be implemented in Matlab/Simulink. The parameters of the experimental setup are summarized in Table I.

TABLE I: PARAMETERS OF A NONLINEAR PENDULUM MODEL.

Parameter	Value	Unit
$\hat{J}_0$	$5.5351 \cdot 10^{-4}$	$kg \cdot m^2$
$\hat{J}_2$	$3.8533 \cdot 10^{-4}$	$kg \cdot m^2$
m_2	0.018	kg
L_1	0.108	m
l_2	0.1369	m
b_1	$8.3336 \cdot 10^{-5}$	$kg \cdot m^2/s$
b_2	$2.5 \cdot 10^{-4}$	$kg \cdot m^2$
g	9.81	m/s^2
c_m	0.0256	Nm/A
c_e	0.0256	Vs/rad
R_a	2.19	Ω
n	3.9	-

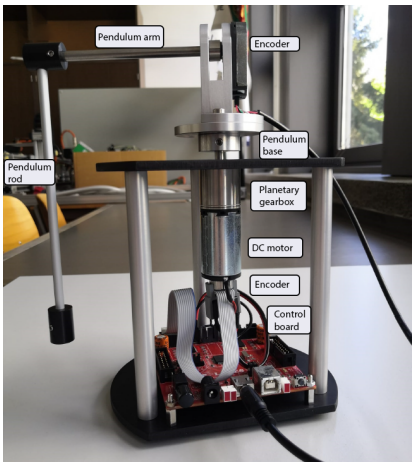


Fig. 1: The experimental setup with its main components labeled.

By linearizing nonlinear differential equations (1) and (2) around an upright equilibrium point ($(\theta_0, \alpha_0, \dot{\theta}_0, \dot{\alpha}_0) = (0, 0, 0, 0)$), the following linearized state equations around the upright position are obtained:

$$\dot{x} = Ax + Bu, \tag{4}$$

where $x = [\theta, \alpha, \dot{\theta}, \dot{\alpha}]^\top$ is the state vector and $A \in \mathbb{R}^{4\times4}$ and $B \in \mathbb{R}^{4\times1}$ are the following matrices:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{L_1 g l_2 m_2^2}{D} & -\frac{C\hat{J}_2 + \hat{J}_2 R_a b_1}{R_a D} & -\frac{L_1 b_2 l_2 m_2}{D} \\ 0 & \frac{\hat{J}_0 g l_2 m_2}{D} & -\frac{L_1 l_2 m_2 (C + R_a b_1)}{R_a D} & -\frac{\hat{J}_0 b_2}{D} \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{c_m \hat{J}_2 n}{R_a D} \\ \frac{c_m L_1 l_2 m_2 n}{R_a D} \end{bmatrix}, \tag{5}$$

where $C = c_e c_m n^2$ and $D = \hat{J}_0 \hat{J}_2 - L_1^2 l_2^2 m_2^2$.

III. CONTROLLER SYNTHESIS

A. Proportional-Derivative Regulator

The task of keeping the pendulum in the upright position (i.e. maintaining angle α around zero) and at the same time controlling the rotation of the pendulum arm (angle θ) can be achieved by using a simple linear regulator based on proportional-derivative (PD) actions. The parallel implementation of the PD regulator can generally be described by the following transfer function

$$G_{\text{PD}}(s) = K_p + K_d \frac{s}{T_f s + 1}, \tag{6}$$

where K_p is the proportional gain, K_d is the derivative gain and T_f is some small parasitic time constant that was introduced to ensure that the transfer function (6) is causal, i.e. feasible in reality (the transfer function of an ideal derivative component $K_d s$ is not causal).

Since two angles are to be controlled simultaneously (θ and α), two feedback loops and two corresponding PD regulators (connected in parallel) are needed, as depicted in Fig. 2. The control signal u can be described as follows

$$u = K_{p,\theta}(\theta_r - \theta) + K_{d,\theta}(\dot{\theta}_r - \dot{\theta}) - K_{p,\alpha}\alpha - K_{d,\alpha}\dot{\alpha}, \tag{7}$$

where θ_r denotes the reference for the angle θ and the reference for the angle α is simply zero (pendulum rod in the upright position).

The derivative components of PD regulators for angles θ and α provide an approximation of $\dot{\theta}$ and $\dot{\alpha}$, respectively. In other words, the control system in Fig. 2 can be interpreted as a state-space regulator, where $K_{p,\theta}$, $K_{p,\alpha}$, $K_{d,\theta}$ and $K_{d,\alpha}$ are state-space regulator gains for all four state variables in (4): θ, α, $\dot{\theta}$ and $\dot{\alpha}$, respectively. Therefore, values of regulator gains can be obtained from the linearized model (4) by any state-space linear control synthesis method, such as a pole placement method [4].

B. Linear Quadratic Regulator

The basic idea of optimal control methods is to find the best (optimal) control signals, which minimize (or maximize) a suitably chosen criterion function. For that purpose, a mathematical model is used to describe a dynamic behavior of the system (i.e. the dependence of regulated state variables and/or output variables) as a function of control signals. In order to limit the complexity of the underlying optimization problem, the analysis of system dynamics is usually carried out on a limited prediction horizon. The criterion, i.e. the cost function, usually penalizes the deviation of the state variables from the desired equilibrium state as well as the usage of control variables (i.e. the energy consumed for control purposes). The described control problem can be formulated as the following optimization problem:

$$\begin{aligned} \min_{u_0,\ldots,u_{N-1}} \quad & \frac{1}{2} x_N^\top P x_N + \frac{1}{2} \sum_{k=0}^{N-1} \left[x_k^\top Q x_k + u_k^\top R u_k \right] \\ \text{s.t} \quad & x_{k+1} = Ax_k + Bu_k,\ k = 0, \ldots, N-1, \\ & x_0 = x(0). \end{aligned} \tag{8}$$

where the cost function is quadratic (with suitably sized weighing matrices $P = P^\top \succeq 0$, $Q = Q^\top \succeq 0$ and $R = R^\top \succ 0$), N is the length od the prediction horizon, system dynamics are described in discrete-time by matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$, and $x(0)$ is a known initial state of the system. The solution of (8) can be obtained analytically in closed-form and the corresponding

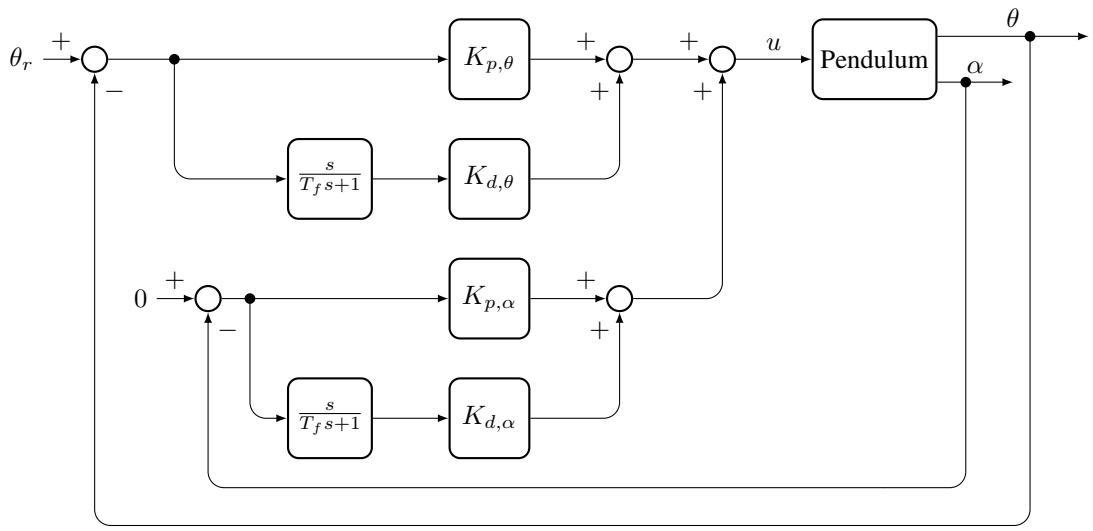


Fig. 2: Inverted pendulum control system based on two PD regulators.

optimal control signals can be expressed in the form of a control law [4]:

$$u_k = -G_k x_k,\ k = 0, \ldots, N-1, \tag{9}$$

where $G_k \in \mathbb{R}^{m \times n}$ is the time-varying gain matrix. In other words, the solution of the control problem (8) is a time-varying state-space regulator. The obtained optimal controller is called finite-horizon discrete-time linear quadratic regulator (FDLQR). The special case of FDLQR problem (8) is obtained when N approaches infinity. In that case, if the pair (A, B) is stabilizable and the pair (A, Q) is observable, all controller gains G_k converge to the stationary value G_∞ and the resulting time-invariant (static) discrete linear quadratic regulator (DLQR) in state-space is obtained [4]:

$$u_k = -G_\infty x_k,\ k = 0, \ldots \tag{10}$$

A simple implementation and inherent robustness with guaranteed gain and phase margins [5] make a DLQR controller a popular choice for control of multiple-input multiple-output (MIMO) systems. The regulator tuning, which is carried out by choosing different weighing matrices Q and R, is intuitive, although the relationship between controller parameters and the specified design goals in time-domain is not as clear as with other methods, like pole placement. However, the main disadvantage of LQR control is that control input and state constraints (e.g. saturation, actuator limits, physical constraints, etc.) cannot be systematically taken into account.

C. Model Predictive Controller

Model predictive control (MPC) is a form of optimal control and its principal idea is very similar to the linear quadratic control discussed in III-B, i.e. a mathematical model is used to predict the future behavior of the dynamical system as a function of input control signals and optimal control signals are obtained by minimizing a suitable cost function. The main difference is that MPC does not rely on any assumptions regarding linearity of the system (i.e. the mathematical model of the system dynamics can be nonlinear) and it can also take into account additional constraints on state variables and control input variables.

In case of a linear discrete-time system dynamic model and linear constraints on input and state variables, the model predictive control problem can be formulated as follows:

$$\begin{aligned} \min_{u_0,\ldots,u_{N-1}} \ & \frac{1}{2} x_N^\top P x_N + \frac{1}{2} \sum_{k=0}^{N-1} \left[x_k^\top Q x_k + u_k^\top R u_k \right] \\ \text{s.t} \quad & x_{k+1} = A x_k + B u_k,\ k = 0, \ldots, N-1, \\ & x_k \in \mathcal{X},\ u_k \in \mathcal{U}, \quad k = 0, \ldots, N-1. \\ & x_N \in \mathcal{X}_f,\ x_0 = x(0). \end{aligned} \tag{11}$$

where the cost is the same as in (8), N is the length of the prediction horizon, $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{X}_f$ are polyhedral sets that describe constraints on state vector x and input vector u, and $x(0)$ is a known initial state of the system.

Unlike LQ problem (8), the MPC problem (11) cannot be solved analytically in general. i.e. the optimization problem (11) needs to be solved numerically at each time sample. For that reason, the MPC problem generally cannot be analyzed for an infinite horizon, i.e. MPC is always implemented with a finite prediction horizon. The solution of (11) is an optimal control sequence $u_0^\star$, $u_1^\star$, ..., $u_{N-1}^\star$. However, in order to introduce feedback to the control algorithm, only the first element of the control sequence, $u_0^\star$, is applied to the system. At the next sampling instant, the new initial state of the system $x(0)$ is measured or estimated and the optimization problem (11) is solved again from that new initial state and with a shifted prediction horizon, in accordance with the so-called receding horizon control philosophy.

The closed-loop stability under the MPC control law $u = u_0^\star$ is not automatically guaranteed but it can be ensured by carefully choosing the terminal cost function and the terminal constraint set. In particular, if the stage

TABLE II: PARAMETERS OF A SIMPLIFIED NONLINEAR PENDULUM MODEL.

Parameter	Value	Unit
a_m	2.5585	$1/s$
K_p	82.33	$rad \cdot V/s^2$
C	$2.5 \cdot 10^{-3}$	$kg \cdot m^2/s$
$\hat{J}_p$	$4.252 \cdot 10^{-4}$	$kg \cdot m^2$
m_p	$18 \cdot 10^{-3}$	kg
g	9.91	m/s^2
l_p	0.17	m
K	$1.9 \cdot 10^{-3} \operatorname{sign}(\cos\alpha)$	kg/m^2

cost is positive definite, terminal set $\mathcal{X}_f$ is invariant under the local control law $v(x_k)$, all state and input constraints are satisfied in $\mathcal{X}_f$, the terminal cost is a continuous Lyapunov function in the terminal set $\mathcal{X}_f$ and it decreases along the closed loop trajectories, then the closed-loop system under the MPC control law is asymptotically stable and the terminal constraint set provides recursive feasibility [6].

The common choice for the terminal controller is the unconstrained DLQR control law, the terminal weight P is chosen as the corresponding solution of the discrete-time algebraic Riccati equation, and the terminal set $\mathcal{X}_f$ is chosen as the maximum invariant set for the closed-loop system under the terminal control law. Even under the assumption that the mathematical model of the system dynamics is linear and all state and input constraints are linear, the resulting MPC control law is generally nonlinear. This can be easily seen if the optimization problem (11) is solved as a multi-parametric quadratic program (mp-QP) for all possible values of initial state x_0. The optimizer $u_0^\star(x_0)$ expressed as an explicit function of the initial state x_0 is a continuous piecewise affine function defined on polyhedra [7].

D. Sliding Mode Controller

For sliding mode (SM) controller synthesis, a simplified nonlinear model of the Furuta pendulum is used as described in [8]:

$$\ddot{\theta} = -a_m\dot{\theta} + K_p u, \tag{12}$$

$$\ddot{\alpha} = -\frac{C}{\hat{J}_p}\dot{\alpha} + \frac{m_p g l_p}{2\hat{J}_p}\sin(\alpha) - \frac{K}{\hat{J}_p}a_m\dot{\theta} + \frac{K}{\hat{J}_p}K_p u. \tag{13}$$

Equation (12) represents the dynamics of the DC motor which rotates the pendulum base while (13) represents the dynamics of the inverted pendulum itself.

The constant a_m includes back-EMF and friction effects. K_p includes the effect of the input voltage on the angular acceleration of the base. C represents the pendulum friction, m_p the pendulum's mass, $\hat{J}_p$ is the pendulum moment of inertia calculated at its centre of mass, and g is the gravitation constant while K is a constant that has a positive value for the inverted and a negative value for non-inverted position. The control input u still denotes the DC voltage applied to the motor. The values of all parameters of the simplified nonlinear pendulum model are reported in Table II.

In order to achieve control of both angles θ and α using a single sliding surface the following transformation needs to be applied [8]:

$$y = \theta - \frac{\hat{J}_p}{K}\alpha, \tag{14}$$

$$x = \dot{y} - \frac{C}{K}\alpha. \tag{15}$$

Equations (14) and (15) transform the system to the following formulation:

$$\dot{x} = -\frac{m_p g l_p}{2K}\sin(\alpha), \tag{16}$$

$$\dot{y} = x + \frac{C}{K}\alpha. \tag{17}$$

The design of the sliding mode controller from the equations above is described in detail in [8]. The time derivative of the sliding surface has the form:

$$\dot{s} = \frac{KK_p}{\hat{J}_p}\cos(\alpha)u + \psi(x, y, \alpha, \dot{\alpha}), \tag{18}$$

where ψ is a nonlinear function of the system states. The sliding mode controller is stable if the sliding surface and its derivative have opposing signs:

$$\dot{s} = -\eta\operatorname{sign}(s). \tag{19}$$

The final control signal for sliding mode control as proposed in [8] has the following form:

$$u = \frac{-\hat{J}_p}{KK_p\cos\alpha}(\psi + \eta\operatorname{sign}(s)). \tag{20}$$

In order to reduce chattering in the control signal applied to the experimental setup, equation (19) is replaced with:

$$\dot{s} = -\eta\tanh(s/2). \tag{21}$$

IV. EXPERIMENTAL RESULTS

All controllers discussed in Section III have been tested on the laboratory experimental setup described in Section II. The choice of controller parameters for each controller is briefly discussed first and then the experimental results are presented and analyzed.

A. PD regulator

PD regulator gains were determined by a pole placement method as follows: $K_{p,\theta} = -1$, $K_{p,\alpha} = 12$, $K_{d,\theta} = -0.5$, $K_{d,\alpha} = 1.5$. With such choice of controller gains all closed-loop poles are located in the left half-plane of the complex plane (-69.0423, -6.8722,$-1.8595 \pm 3.5848j$) and the closed-loop system is stable. T_f was set to 20 ms.

B. DLQR

The continuous-time linear model (4) was discretized using a zero-order hold (ZOH) discretization with a sampling time of $T_d = 10$ ms. The weighing matrices Q and R were chosen as $Q = \mathrm{diag}\,(40, 40, 1, 1)$ (where diag denotes a diagonal matrix with diagonal elements listed in brackets) and $R = 30$. The optimal gains were computed by solving the discrete algebraic Riccati equation and the values obtained are $K_{\mathrm{DLQR}} = [-0.9592,\ 10.3582,\ -0.5756,\ 1.2128]$.

C. MPC

The MPC optimization problem (11) was formulated using the same discretized linearized model that was used to design a DLQR controller in the previous subsection. The same DLQR controller was used as a terminal controller for the MPC, the corresponding solution of the Riccati equation was chosen as the terminal cost function weight P, and the terminal set $\mathcal{X}_f$ was chosen as the maximum invariant set for the closed-loop system under the terminal control law, as previously discussed in Section III. Prediction horizon length was set to $N = 20$. Control input was constrained to $-3 \leq u \leq 3$. The described MPC algorithm was implemented in Simulink using Matlab's MPC Toolbox.

D. SM controller

The following controller parameters for the sliding-mode controller were used: $\lambda = 900$ and $\eta = 80$ [8]. Such choice ensures that the control input signal is between its maximum and minimum values ± 3 V.

E. Experiments

To test all designed controllers, the following reference for the angle θ was used: for the first 12 seconds the reference θ_r remains at zero, then for the next 12 second it switches to 60 degrees, and finally for the last 12 seconds it goes back to zero. The pendulum rod is manually lifted to the upright position in the first 5 seconds of the experiment. Fig. 3 shows: (a) the response of the angle θ (upper sub-figure), (b) the response of the angle α (sub-figure in the middle), and (c) the applied control signal u (lower sub-figure).

From Fig. 3 it can be seen that all designed controllers are successful to a certain degree at controlling both angles α and θ. They all however have a certain drift in angle θ (Fig. 3a). For the SM controller this drift is smaller than for other controllers because it was derived using a nonlinear model while other controllers were designed using a linearized model. However, the drift still exists for the SM controller and may be explained by simplified modelling of the pendulum, which did not include all nonlinear effects present on the real pendulum.

The response of angle α is also the best in case of the SM controller where the oscillations around the upright position were the smallest (less than 1 degree). For all other controllers these oscillations were slightly larger (around 5 degrees on average) with jumps to almost 15 degrees in time instances when sudden change of the reference for angle θ occurs which is accompanied by a corresponding jump in the control signal (Fig. 3c). This jump in the control signal is easy to understand in the case of the PD regulator: a sudden change of the reference signal produces an impulse in the derivative component of the PD regulator.

The control signal is the most active in the case of the SM controller, where chattering effect can be noticed, which causes fast switching behavior (even if the $\tanh$ function is used instead of sign function in (19)). For all other controllers the control input signal remained mostly between -1 and $+1$ volts, except at time instances when the reference changes, as discussed previously.

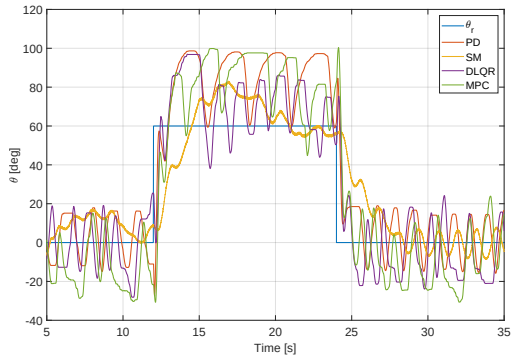
The relatively poor tracking of angle θ for PD, DLQR and MPC controllers can be explained by a nonlinear effect in the system which was not taken into account. The nonlinear effect in question is the static friction of the motor. Since the motor is controlled by the voltage, this means that the motor will not react if the supplied voltage is too small (e.g. smaller than 0.5 V), which manifests as a dead-zone at the system's input. The easiest way to compensate the dead-zone in this case is to add a small offset to the control signal (e.g. $+0.4$ V if $u > 0$ and -0.4 V if $u < 0$) to overcome the dead-zone.

Since the control signal normally oscillates around zero (to keep the pendulum rod in the upright position), the described addition of an offset to the control signal is expected to introduce a higher frequency component signal at the system's input (due to frequent changes between plus and minus offsets), which is similar in effect to adding a dither to a control signal. However, the average value of such modified control signal should be similar to the original control signal so the behavior of the system should be closer to the desired behavior.

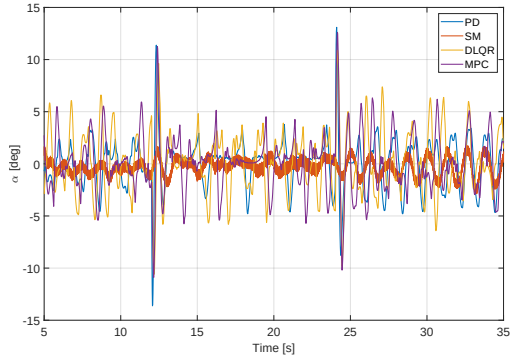
Fig. 4 shows the response of the system when the proposed static friction compensation scheme is employed. From Fig. 4a and Fig. 4b it is evident that PD, DLQR and MPC perform much better than before. As expected, this was paid by a much more active control signal as shown in Fig. 4c, where higher frequency oscillations were introduced to overcome the dead-zone. These higher frequency oscillations propagated from the input to the output of the system, as can be seen from the responses of output angles θ and α.

V. Conclusion

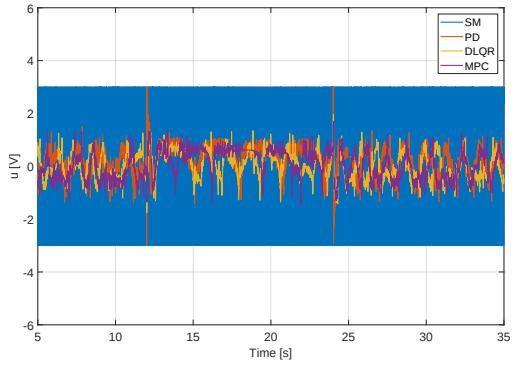
Several control methods for stabilizing a rotary inverted pendulum were described and tested on a real experimental setup. All tested controllers were successful at keeping the pendulum rod in the upright position and at the same time tracking the reference for the pendulum arm. Controllers derived from a linearized model were more sensitive to unmodeled nonlinear effects such as the input dead-zone. However, with a simple dead-zone compensa-



(a) Reference tracking for angle θ.

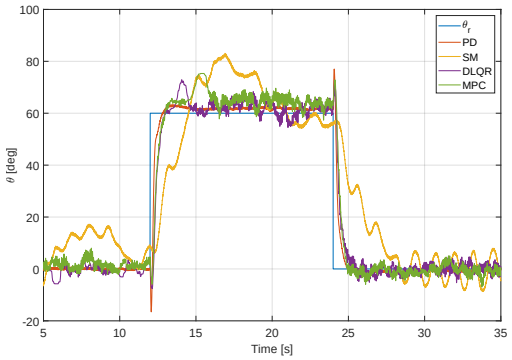


(b) Reference tracking for angle α.

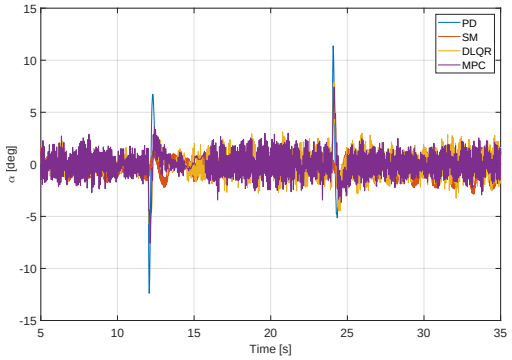


(c) Control input signal u.

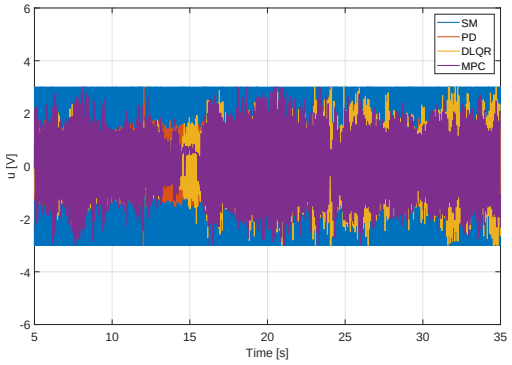
Fig. 3: Experimental results for all tested controllers (without static friction compensation).



(a) Reference tracking for angle θ.



(b) Reference tracking for angle α.



(c) Control input signal u.

Fig. 4: Experimental results for all tested controllers (with static friction compensation).

tion scheme the response of the system was significantly improved. Hence, in this case, adding a suitable static friction compensation has a larger impact on the closed-loop behavior than selecting different control solutions. MPC was designed to emulate the infinite horizon DLQR and PD gains were also very similar to DLQR gains, which is why they performed similarly. Advantages of MPC would be more obvious if the system was driven closer to the constraints or if reference signal was explicitly taken into account on the prediction horizon in the optimization problem formulation.

REFERENCES

[1] K. Furuta, M. Yamakita, and S. Kobayashi, "Swing-up control of inverted pendulum using pseudo-state feedback," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 206, pp. 263 – 269, 1992. [Online]. Available: https://api.semanticscholar.org/CorpusID:109703709

[2] B. Cazzolato and Z. Prime, "On the dynamics of the furuta pendulum," *Journal of Control Science and Engineering*, 2011.

[3] M. Svec, S. Iles, and J. Matusko, "Sliding mode control of custom built rotary inverted pendulum," in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 943–947.

[4] K. Ogata, *Modern Control Engineering*, 4th ed. USA: Prentice Hall PTR, 2001.

[5] N. Lehtomaki, N. Sandell, and M. Athans, "Robustness results in linear-quadratic gaussian based multivariable control designs," *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 75–93, 1981.

[6] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill Publishing, 2017.

[7] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[8] V. Utkin, J. Guldner, and J. Shi, *Sliding Mode Control in Electromechanical Systems*. CRC press, 2009, vol. 1.