A Complex Overview of Modeling and Control of the Rotary Single Inverted Pendulum System

Slavka JADLOVSKA [1], Jan SARNOVSKY [1]

[1]Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Kosice, Letna 9, 042 00 Kosice, Slovak Republic

slavka.jadlovska@tuke.sk, jan.sarnovsky@tuke.sk

Abstract. *The purpose of this paper is to present an in-depth survey of the rotary single inverted pendulum system from a control engineer's point of view. The scope of the survey includes modeling and open-loop analysis of the system as well as design and verification of balancing and swing up controllers which ensure successful stabilization of the pendulum in the unstable upright equilibrium. All relevant tasks and simulation experiments are conducted using the appropriate function blocks, GUI applications and demonstration schemes from a Simulink block library developed by the authors of the paper. The library is called Inverted Pendula Modeling and Control (IPMaC) and offers comprehensive program support for modeling, simulation and control of classical (linear) and rotary inverted pendulum systems.*

Keywords

Automatic model generation, custom Simulink block library, energy-based swing-up methods, linear state-feedback stabilizing control, rotary single inverted pendulum.

1. Introduction

Inverted pendulum systems (IPS) represent a significant class of nonlinear underactuated mechanical systems, well-suited for verification and practice of ideas emerging in control theory and robotics [1]. Stabilization of a pendulum rod in the unstable upright position is considered a benchmark control problem which has been solved by attaching the pendulum to a base that moves in a controlled linear manner (classical or linear IPS) or in a rotary manner in a horizontal plane (rotary IPS).

The Inverted Pendula Modeling and Control (IP-MaC) is a structured Simulink block library which was developed by the authors of this paper and provides complex software support for the analysis and control of both classical and rotary IPS [2]. Strong emphasis is placed on the generalized approach to system modeling [3], [4], allowing the library to handle systems which differ by the number of pendulum links attached to the base, such as single [2], [4], [5] and double [2], [4], [6] IPS.

As an underactuated, unstable and yet controllable system, the rotary single inverted pendulum has been regarded as an attractive testbed system for linear and nonlinear control law verification since Katsuhisa Furuta, Professor at the Tokyo Institute of Technology, introduced it to feedback control community in 1992 [7]. The aim of this paper is to present a thorough overview of this system that covers all topics significant for control engineering. The paper is organized as follows:

- In Section 2, a mathematical model of the system is obtained by automatic model generation, and open-loop responses of the system are evaluated.

- Design and verification of control algorithms that stabilize the pendulum in the unstable upright equilibrium is presented in Section 3 for a pair of model situations which deal with stabilizing the pendulum from around the upright and downward position respectively. The latter situation also involves the design of a swing-up controller.

- Finally, Section 4 concludes the paper with an evaluation of achieved results and some final remarks.

Suitable function blocks and GUI applications from the IPMaC are employed in every described step of the process of modeling and control design. All simulation experiments mentioned in the paper can be run from the Demo Simulations section of the IPMaC which contains links to corresponding simulation schemes.

2. Mathematical Modeling and Simulation of the Rotary Single Inverted Pendulum System

The considered rotary single inverted pendulum system (Fig. 1) is composed of a rigid, homogenous pendulum rod attached to an rigid arm which is free to rotate in a horizontal plane. Since the number of actuators is lower than the number of system links, the system is underactuated:
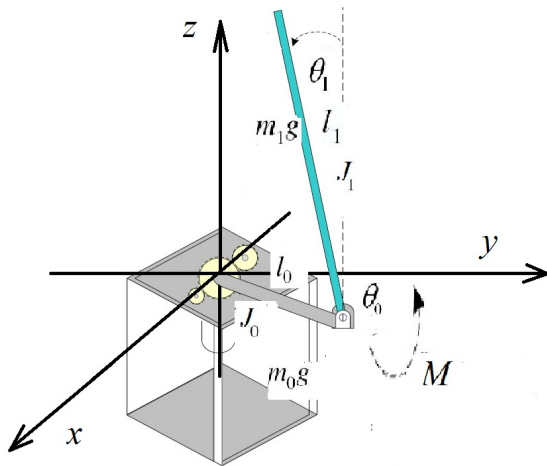

Fig. 1: Rotary single inverted pendulum - scheme and parameter nomenclature.

The only input (the torque $M(t)$ applied on the arm) is used to control the two degrees of freedom of the system: arm angle $\theta_0(t)$ [rad] and pendulum angle $\theta_1(t)$ [rad].

2.1. Automatic Derivation and Evaluation of the Mathematical Model

Manual, step-by-step derivation of differential equations of motion is the traditional approach to modeling of mechanical systems which appears prevalently in accessible works such as [8], [9], [10] despite the fact that it often turns out as laborious and error-prone. As opposed to the traditional approach, the mathematical model of the system analyzed in this paper was generated automatically by the Inverted Pendula Model Equation Derivator (Fig. 2) a MATLAB GUI application which simplified the process of inverted pendula modeling to such a level that the user is only required to select the desired type of system (classical/) and the number of pendulum links. The application was designed as a central component of the IPMaC block library and uses an original procedure of mathematical

model derivation based on the mutual analogy of IPS with a varying number of pendulum links. Via Symbolic Math Toolbox [11], the procedure implements the derivation of Euler-Lagrange equations of motion for a generalized system of n classical or rotary inverted pendula [3], [4], [5]. As it is shown in the preview of the Derivator window, the generated model of the rotary single inverted pendulum system is composed of two second-order nonlinear differential equations of motion which respectively describe the dynamic behavior of the rotary arm and the pendulum. For the purposes of further model analysis as well as control design, it is convenient to rearrange the obtained nonlinear equations into the so-called standard minimal ODE (Ordinary Differential Equation) matrix form [12]:

$$
\begin{aligned}
\mathbf{M}_I(\vec{\theta}(t))\ddot{\vec{\theta}}(t) + \mathbf{N}(\vec{\theta}(t), \dot{\vec{\theta}}(t))\dot{\vec{\theta}}(t) + \\
+ \mathbf{R}(\vec{\theta}(t)) = \vec{V}(t),
\end{aligned} \tag{1}
$$

where $\vec{\theta}(t)$ is the vector of generalized coordinates defined as $\vec{\theta}(t) = (\theta_0(t)\theta_1(t))^T$, $\mathbf{M}_I(\vec{\theta}(t))$ is the inertia matrix, $\mathbf{N}(\vec{\theta}(t), \dot{\vec{\theta}}(t))$ describes the influence of centrifugal and Coriolis forces, $\mathbf{R}(\vec{\theta}(t))$ accounts for gravity forces and $\vec{V}(t)$ is the system's input vector.
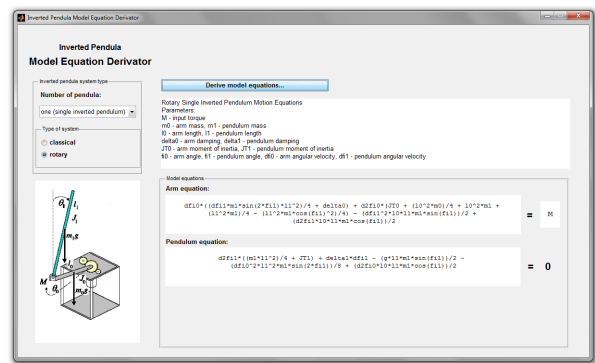

Fig. 2: Automatic derivation of the rotary single inverted pendulum motion equations using Inverted Pendula Model Equation Derivator.

After the described rearrangement of the generated equations of motion, the mathematical model of the system assumed the form (2), see [5], where m_0 and m_1 stand for the masses of the arm and the pendulum, l_0 and l_1 are their respective lengths, δ_0 and δ_1 are the damping constants in the joints of the arm and pendulum, J_0 and J_1 are the moments of inertia of the arm and pendulum with respect to their pivot points, defined in each case as (3), see [13], and $M(t)$ is the input torque applied upon the rotary arm. The generated model will hereafter be referred to as a torque model of a rotary single inverted pendulum system, to distinguish it from a voltage model of the system, which will be presented in section 2.3.

$$\begin{pmatrix} J_0 + m_1 l_0^2 + \frac{1}{4} m_1 l_1^2 sin^2\theta_1(t) & \frac{1}{2} m_1 l_0 l_1 cos\theta_1(t) \\ \frac{1}{2} m_1 l_0 l_1 cos\theta_1(t) & J_1 \end{pmatrix} \begin{pmatrix} \ddot{\theta}_0(t) \\ \ddot{\theta}_1(t) \end{pmatrix} +$$
$$+ \begin{pmatrix} \delta_0 + \frac{1}{4} m_1 l_1^2 \dot{\theta}_1(t) sin2\theta_1(t) & -\frac{1}{2} m_1 l_0 l_1 \dot{\theta}_1(t) sin\theta_1(t) \\ -\frac{1}{8} m_1 l_1^2 \dot{\theta}_0(t) sin2\theta_1(t) & \delta_1 \end{pmatrix} \begin{pmatrix} \dot{\theta}_0(t) \\ \dot{\theta}_1(t) \end{pmatrix} +$$
$$+ \begin{pmatrix} 0 \\ -\frac{1}{2} m_1 g l_1 sin\theta_1(t) \end{pmatrix} = \begin{pmatrix} M(t) \\ 0 \end{pmatrix}, \tag{2}$$

$$J_i = \frac{1}{3} m_i l_i^2, i = 0, 1. \tag{3}$$

A number of differences were observed between the generated mathematical model and models in referenced works. Most of these can be attributed to the initial choice of the coordinate system orientation (right- or left-handed) and the orientation of the angles (clockwise/anticlockwise). In the general procedure implemented by the Derivator, it is assumed that all motion of the system is bound to a standard right-handed three-dimensional coordinate system, and that the rotary motion of the pendulum takes place in a vertical plane that is always perpendicular to both the horizontal arm plane and the arm itself. As a result, the actual plane the pendulum rotates in is different in every instant, which brings additional complexity as well as high accuracy to the generated mathematical model. In referenced works, the pendulum was often assumed to be rotating in a constant plane [8], [9] or the rotary motion of the arm was neglected altogether [10].

2.2. Open-Loop Dynamical Analysis of the Torque Simulation Model

The IPMaC is designed as an open library structured into sublibraries. One of the default components of the Inverted Pendula Models sublibrary of the IPMaC is the Rotary Single Inverted Pendulum (RSIP) block, which implements the mathematical model (2) derived above. The block is equipped with a dynamic block mask [2] which enables the user to edit the numeric values of parameters and initial conditions, to enable or disable the torque input port and to adjust the number of the block's output ports, all with great flexibility (Fig. 3).

The open-loop dynamical behavior of the rotary single inverted pendulum system was verified in a simulation experiment [5] in which the RSIP block was actuated from an initial upright position of the pendulum by a torque impulse of 0,4 Nm lasting 1 s. The numeric parameters of the simulated system were selected to be $m_0 = 0, 5$ kg, $m_1 = 0, 275$ kg, $l_0 = 0, 6$ m, $l_1 = 0, 5$ m, $\delta_0 = 0, 3$ kgs^{-1}, $\delta_1 = 0, 011458$ kgm^{2}s^{-1}. The time behavior of the rotary arm and the freefall and stabiliza-
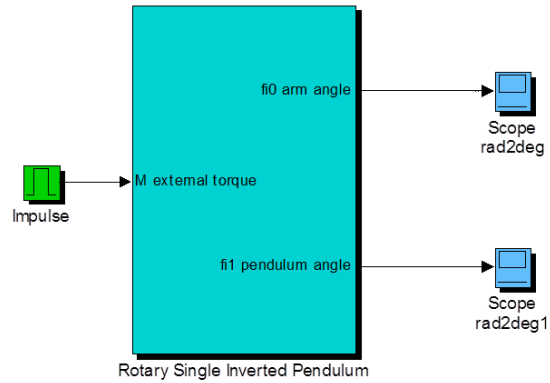

Fig. 3: Open-loop dynamical analysis of the rotary single inverted pendulum (torque model) - simulation setup.
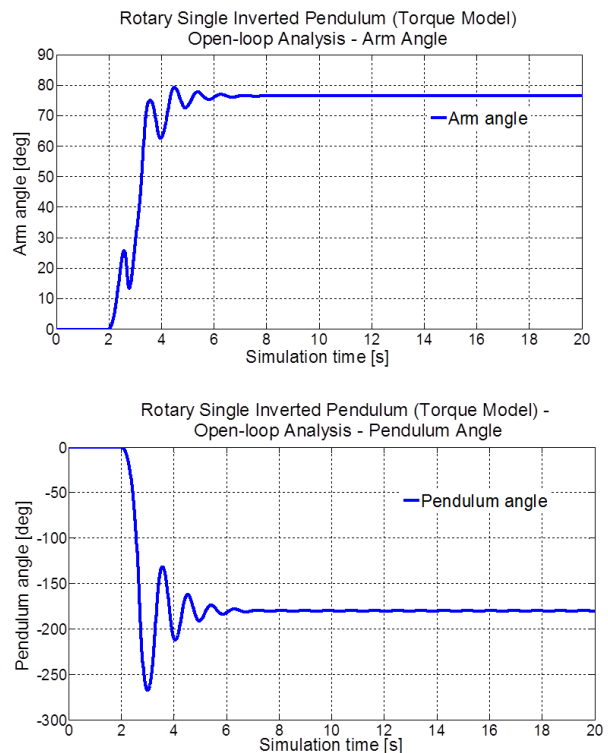

Fig. 4: Rotary inverted pendulum (torque model) - open-loop time behavior of the arm and pendulum angles.
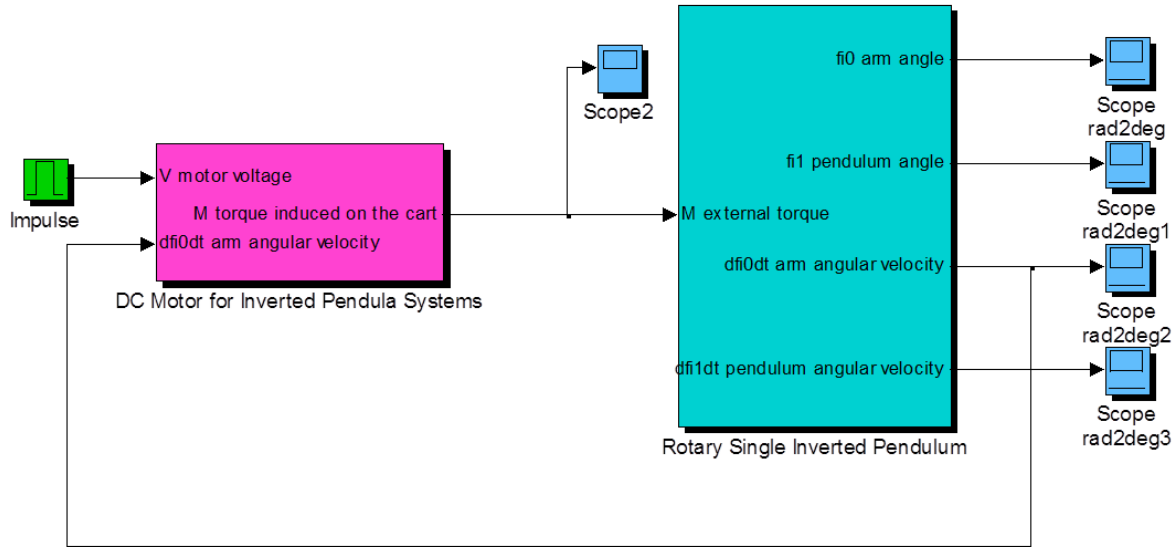
Fig. 5: Open-loop dynamical analysis of the rotary single inverted pendulum (voltage model) - simulation setup.

tion of the pendulum is depicted in Fig. 4. Reasonable behavior of the open-loop response of the RSIP simulation model (damped oscillatory transient state, system reaching the stable equilibrium point with the pendulum pointing downward, visible backward impact of the pendulum on the rotary arm) means that the simulation model based on generated motion equations can be considered accurate enough to serve as a reliable testbed system for the verification of linear and nonlinear control algorithms.

2.3. Open-Loop Dynamical Analysis of the Voltage Simulation Model

As is the case of all IPS, the practical use of the generated rotary single inverted pendulum torque model is limited purely to the simulational environment: we are unable to manually produce a torque which would stabilize the pendulum in a chosen position. Therefore, an electric motor needs to be coupled with the system to act as a controllable source of the torque which actuates both the rotary arm and the attached pendulum [2].

To provide the simulation models of IPS with a model of the most frequently used actuating mechanism, library block DC Motor for Inverted Pendula Systems was included into the Inverted Pendula Motors sublibrary of the IPMaC. The block implements the mathematical model of a brushed direct-current (DC) motor in two alternative forms depending on the type of system (classical/rotary). For rotary IPS, the DC motor is modeled as the following voltage-to-torque conversion relationship, derived in [3]:
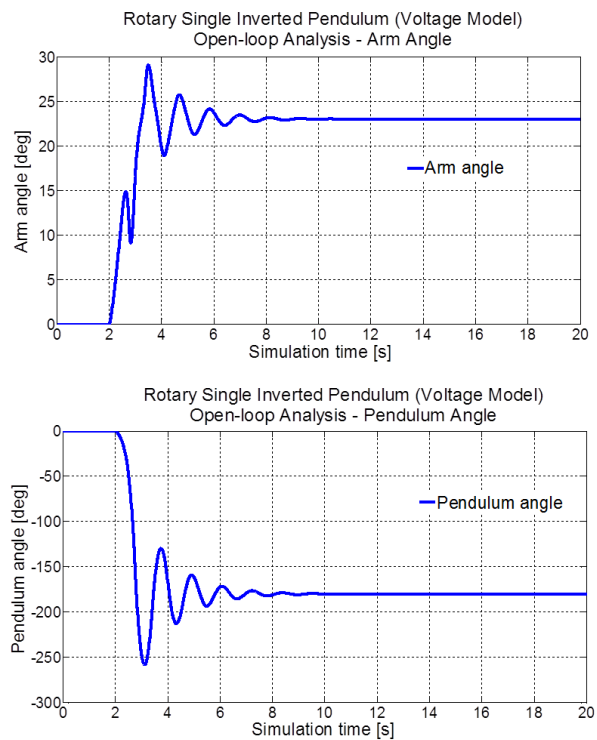

Fig. 6: Rotary single inverted pendulum (*voltage model*) - open-loop time behavior of the arm and pendulum angles.

$$M(t) = \frac{k_m k_g}{R_a} V_a(t) - \frac{k_m^2 k_g^2}{R_a} \dot{\theta}_0(t), \qquad (4)$$

where $V_a(t)$ is the input voltage applied to the motor, k_m is the motor torque constant, equal in value to the

back EMF constant, k_g is the gear ratio, and R_a is the armature resistance. If the DC motor model is appended to the model of the rotary inverted pendulum system (i.e. (4) is substituted into (2)), a voltage model of the system is obtained (Fig. 5). In Fig. 6, the response of the rotary inverted pendulum voltage model to an impulse signal of 1 V lasting 1 s is depicted. The numeric parameters of the DC motor simulation model were borrowed from the motor featured in the series of popular laboratory models of inverted pendulum systems issued by Quanser Academic (SRV02-Series) and are specified in [14].

3. Stabilization of the Rotary Single Inverted Pendulum System in the Unstable Equilibrium

The principal control objective for the considered nonlinear system was defined as stabilization in the unstable equilibrium, i.e. in the vertical upright (inverted) position of the pendulum [5]. Two possible model situations were considered, in which a varying initial position of the pendulum calls for a corresponding control setup. It is sufficient to design a stabilizing feedback controller if the pendulum is kept in the proximity of the upright position (section 3.1). On the other hand, the pendulum in the natural hanging position requires an additional mechanism which swings it up to the upright position before it can be stabilized (section 3.2), as well as a mechanism to switch between the two controllers. The rotary single inverted pendulum can therefore be considered as a suitable testbed system for illustrating hybrid control approaches [15].

3.1. Design of a Stabilizing Controller via State-Feedback Techniques

Let us now suppose that at the beginning of the experiment, the pendulum is held close to the unstable upright equilibrium. The designed stabilizing controller should ensure that the following particular problems will be solved [3], [5], [6]:

- balancing the pendulum around the upright position after an initial deflection (i.e. from nonzero initial conditions),

- compensation of a time-constrained (impulse) or permanent (step) disturbance input signal,

- the arm must be able to track a predefined reference trajectory at any time.

State-feedback control was selected as a principal control technique because simultaneous control of several degrees of freedom can only be ensured if they are all taken into consideration at once (Fig. 7). When close to a chosen equilibrium point, detailed methods and control techniques designed for linear systems can be used to compute a linear state-feedback control law which is subsequently applied to the original nonlinear system in the form of a stabilizing input signal $u_{ST}(t)$.
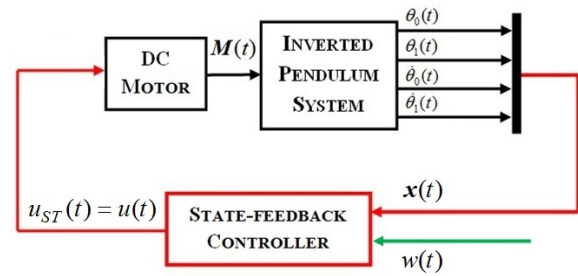

Fig. 7: Basic block diagram of stabilization control of the rotary inverted pendulum system.

1) Automatic Linear Approximation of the Rotary Inverted Pendulum System

To obtain the linear approximation of the original system, which is a necessary prerequisite for linear state-feedback control design, the rotary single inverted pendulum model first had to be expressed by the standard state-space description of a nonlinear SIMO system [4]:

$$\begin{aligned} \vec{x}(t) &= f(\vec{x}(t), u(t), t), \\ \vec{y}(t) &= g(\vec{x}(t), u(t), t). \end{aligned} \tag{5}$$

With the state vector defined as $\vec{x}(t) = (\vec{\theta}(t), \dot{\vec{\theta}}(t))$, the differential vector state equation $f(.)$ was constructed by isolating the 2nd derivative of the vector of generalized coordinates from the standard minimal ODE form (1) of the mathematical model of the system [12]. The algebraic output equation $g(.)$ denotes the m state variables which are assumed to be available through measurement.

The general procedure implemented by the Derivator is based on the assumption that the upright position of the pendulum corresponds to the equilibrium point of $\vec{x}(t) = x_{ST} = 0^T$. If the system input is set to $\vec{u}(t) = u_{ST} = 0$, [2], then the continuous-time state-space description of the linear, time-invariant system which serves as an approximation defined as (5) in the upright equilibrium becomes:

$$\begin{aligned} \vec{x}(t) &= \mathbf{A}\vec{x}(t) + \mathbf{b}u(t), \\ \vec{y}(t) &= \mathbf{C}\vec{x}(t) + \mathbf{d}u(t), \end{aligned} \tag{6}$$

where $\mathbf{A} \in \Re^{4\times4}$ is the state (Jacobian) matrix of the linearized system, $\mathbf{b} \in \Re^{4\times1}$ is the input matrix,

$\mathbf{C} \in \Re^{m \times 4}$ is the output matrix and $\mathbf{d} \in \Re^{m \times 1}$ is the direct feedthrough matrix [16]. The linearized state-space model (6) was obtained from the Inverted Pendula Model Linearizator & Discretizer (Fig. 8), a MATLAB GUI application which generates the continuous-time state-space matrices by expanding (5) into the Taylor series around a given equilibrium point with the higher-order terms neglected (Fig. 8). In case the sampling period constant T_S is provided, the application also returns the matrices of the corresponding discrete-time state-space representation:

$$\begin{aligned} \vec{x}(i+1) &= \mathbf{F}\vec{x}(i) + \mathbf{g}u(i), \\ \vec{y}(i) &= \mathbf{C}\vec{x}(i) + \mathbf{d}u(i), \end{aligned} \qquad (7)$$

where $\mathbf{F} \in \Re^{4 \times 4}$ is the discrete state matrix, $\mathbf{g} \in \Re^{4 \times 1}$ is the discrete input matrix; $\mathbf{C}$ and $\mathbf{d}$ are the same as in the continuous system.
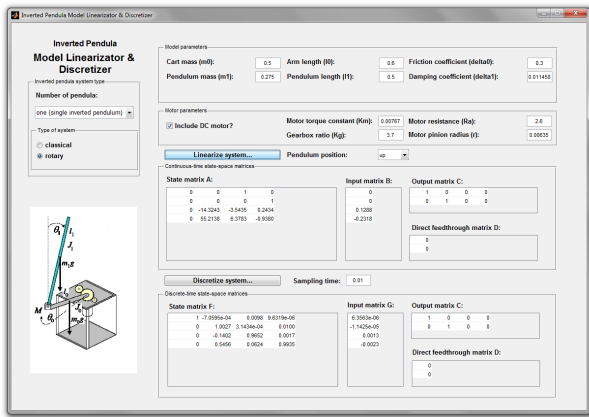

Fig. 8: State-space matrices of the rotary single inverted pendulum via the Inverted Pendula Model Linearizator & Discretizer.

Using the numeric parameters from the open-loop simulations in section 2.2, the following continuous-time state equation matrices of the linearized rotary single inverted pendulum voltage model were generated:

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -14,3243 & -3,5435 & 0,2434 \\ 0 & 55,2138 & 6,3783 & -0,938 \end{pmatrix},$$

$$\mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ 0,1288 \\ -0,2318 \end{pmatrix}.$$

The system eigenvalues were computed to be $(0 \quad 6,404 \quad -9,096 \quad -1,79)$, which implies that the linear approximation of rotary single inverted pendulum in the upper equilibrium is an unstable system with first-degree astatism. The system was next discretized with the sampling period of $T_S = 0,01$ s

and the following discrete-time state equation matrices were obtained:

$$\mathbf{F} = \begin{pmatrix} 1 & -0,0007 & 0,0098 & 0 \\ 0 & 1,0027 & 0,004 & 0,01 \\ 0 & -0,1402 & 0,9652 & 0,0017 \\ 0 & 0,5456 & 0,0624 & 0,9935 \end{pmatrix},$$

$$\mathbf{g} = \begin{pmatrix} 0,000006 \\ -0,000011 \\ 0,0013 \\ -0,0023 \end{pmatrix}.$$

In both cases, the matrices of the output equation are problem-dependent and are specified in later sections.

2) State-Feedback Control with a State Estimator

The Inverted Pendula Control sublibrary of the IPMaC provides complex software support for state-feedback methods of controller design for inverted pendulum systems. It contains several dynamic-masked blocks which were thoroughly described in [2] in terms of their structure, functionality and user interface.

The State-Feedback Controller with Feed-forward Gain (SFCFG) library block evaluates a state-feedback control law which is calculated either from the continuous-time state-space description:

$$\begin{aligned} u(t) &= u_R(t) + u_{ff}(t) + d_u(t) = \\ &- \vec{k}\vec{x}(t) + \vec{k}_{ff}w(t) + d_u(t), \end{aligned} \qquad (8)$$

or from the discrete-time state-space description:

$$\begin{aligned} u(i) &= u_R(i) + u_{ff}(i) + d_u(i) = \\ &- \vec{k}_D x(i) + \vec{k}_{ffD}w(i) + d_u(i), \end{aligned} \qquad (9)$$

assuming that controller output is equivalent to the system output: $u_{ST}(t) = u(t)$. The state-feedback control law is in both cases composed of several functional components [2], [3], [4], [5], [6]. The feedback component $u_R(t)/u_R(i)$ brings the system's state vector $\vec{x}(t)/\vec{x}(i)$ into the origin of the state space, i.e. stabilizes the pendulum in the upright equilibrium and returns the arm to the reference position. The method to determine the feedback gain $\vec{k}/\vec{k}_D$ applied on the full and measurable state vector can be selected from between the pole placement algorithm and the linear quadratic (LQ) optimal control method [5], [16]. The feedforward component $u_{ff}(t)/u_{ff}(i)$ makes the system output $\vec{y}(t)/\vec{y}(i)$ track the reference command $w(t)/w(i)$ by applying feedforward gain $\vec{k}_{ff}/\vec{k}_{ffD}$, which is computed in the form:

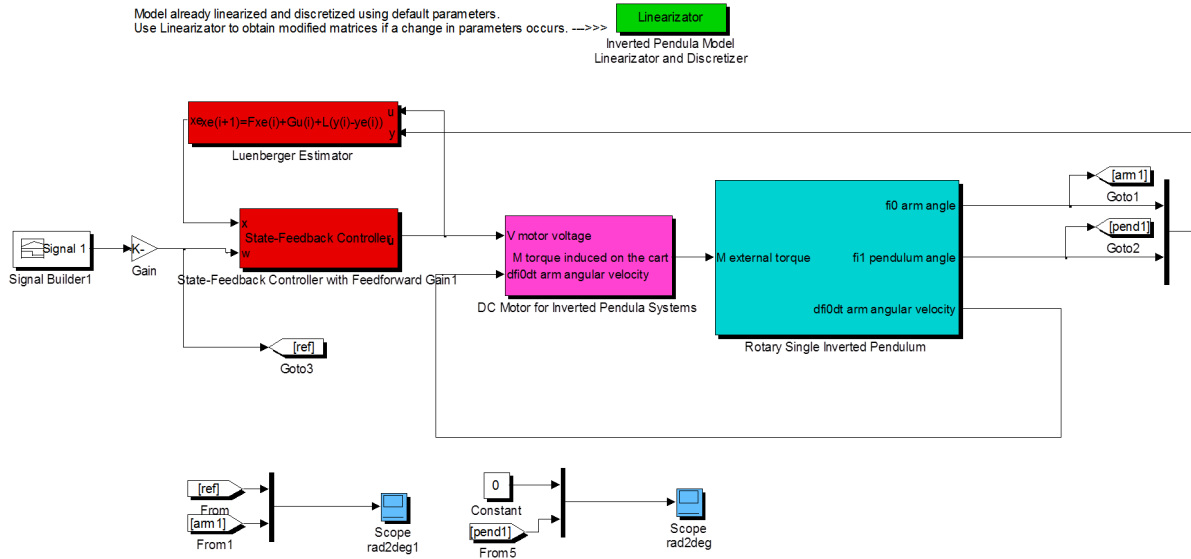$$\vec{k}_{ff} = \frac{-1}{\mathbf{c}_1(\mathbf{A} - \mathbf{b}\vec{k})^{-1}\mathbf{b}}, \qquad (10)$$

Fig. 9: State-feedback control of the rotary single inverted pendulum (*voltage model*) - general simulation setup.

for the continuous-time control law (8), or in the form:

$$\vec{k}_{ffD} = \frac{1}{\mathbf{c}_1(\mathbf{I} - (\mathbf{F} - \mathbf{g}\vec{k}_D))^{-1}\mathbf{g}}, \quad (11)$$

for the discrete-time control law (9), given that $\mathbf{I} = diag(1 \quad 1 \quad 1 \quad 1)$. The artificial output matrix $\mathbf{c}_1 = (1 \quad 0 \quad 0 \quad 0)$ denotes the arm angle as the only state variable for which it makes sense to consider a nonzero reference command. Finally, the unmeasured disturbance input $d_u(t)/d_u(i)$. is considered in both the continuous-time and the discrete-time form of the control law. The dynamic block mask of the SFCFG block allows the user to select a preferred control law, method to compute the feedback gain, and optional constraints. To match a particular control requirement (initial pendulum deflection, compensation of disturbance signal, tracking a reference position of the arm or a combination of the three), the block's appearance may be adjusted by optional enabling or disabling of the reference command input $w(t)/w(i)$ and/or the disturbance input $d_u(t)/d_u(i)$.

Because of measurement limitations, it is often impossible to retrieve the full state-space vector at every time instant [17]. The Luenberger Estimator (LE) block implements a discrete-time linear state observer that provides the controller block with a complete, reconstructed state vector by evaluating a model of the original discrete-time system in the structure:

$$
\begin{aligned}
\vec{\hat{x}}(i+1) &= \mathbf{F}\vec{\hat{x}}(i) + \mathbf{g}u(i) \\
&+ \mathbf{L}(\vec{y}(i) - \mathbf{C}\vec{\hat{x}}(i)), \quad (12)
\end{aligned}
$$

which ensures that the time behavior of estimation error $\vec{\tilde{x}}(i) = \vec{x}(i) - \vec{\hat{x}}(i)$ is independent of input/output signals:

$$\vec{\tilde{x}}(i+1) = (\mathbf{F} - \mathbf{LC})\vec{\tilde{x}}(i), \quad (13)$$

and the estimator gain matrix $\mathbf{L}$ has to be selected in such a way that the approximated state vector $\vec{\hat{x}}(i)$ quickly converges to the actual system state $\vec{x}(i)$.

Figure 10 illustrates the results of applying the state-feedback control law based on both the continuous-time and discrete-time LQR algorithms on the rotary single inverted pendulum voltage model. RSIP, SFCFG and LE blocks were employed in the corresponding simulation scheme (Fig. 9). With the initial conditions set to $\theta_0(0) = \theta_1(0) = 0$, the control objective was to keep the pendulum upright all the time while making the arm rotate for a total of half a circle and stop every quarter-turn to stabilize before returning to its initial position. The weight matrices which occur in both the standard continuous-time LQ functional:

$$J_{LQR}(t) = \frac{1}{2}\int_0^\infty (\vec{x}^T(t)\mathbf{Q}\vec{x}(t) + u_R^T(t)ru_R(t))dt, \quad (14)$$

and the discrete-time LQ functional:

$$J_{LQR}(i) = \sum_{i=0}^{N-1} \vec{x}^T(i)\mathbf{Q}\vec{x}(i) + u_R^T(i)ru_R(i), \quad (15)$$

were set to $\mathbf{Q} = diag(100 \quad 20 \quad 20 \quad 0)$ and $r = 1$. It was next assumed that the arm and pendulum angles would be directly measurable while the velocities would need to be estimated. The state-space matrices of the output equation therefore became

$$\mathbf{C} = \left(\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array}\right), \mathbf{d} = \left(\begin{array}{c} 0 \\ 0 \end{array}\right),$$

while the vector of estimator poles was set to $(0,1 \quad 0,2+0,1i \quad 0,2-0,1i \quad 0,3)$.

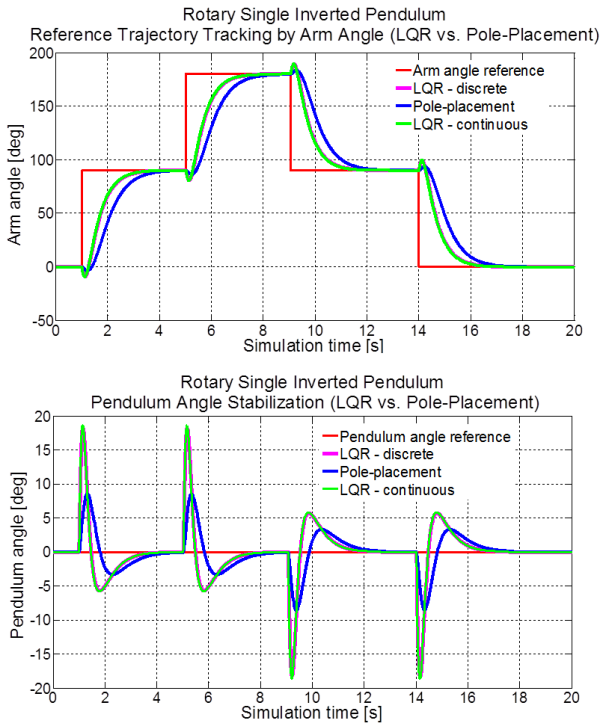**Fig. 10:** Rotary single inverted pendulum - simulation results for LQR control with feedforward gain, compared to pole-placement.

Fig. 11: Rotary single inverted pendulum - evaluation of the influence of the **Q** weight matrix on the performance of the system.

It is obvious that the response of the discrete-time LQR algorithm closely follows that of the continuous-time algorithm, and there is no steady-state error in either case. The simulation results were also compared with those of the continuous-time pole-placement controller: for this case, the vector of desired closed-loop poles was set to $(-2 \quad -3+i \quad -3-i \quad -10)$.

To enhance the performance of the designed LQR controller, optimal weight matrices were sought. After setting $r = 1$, the positive real diagonal elements of the **Q** weight matrix were modified and the influence of each candidate matrix on the overall performance of the system was verified in the discrete-time LQR simulation setup for the rotary inverted pendulum voltage model. The results of two extreme cases of selecting $\mathbf{Q} : \mathbf{Q}_1 = diag(500 \quad 0 \quad 20 \quad 0)$ and $\mathbf{Q}_2 = diag(10 \quad 120 \quad 20 \quad 0)$, are depicted in Fig. 11. For $\mathbf{Q}_1$, the arm quickly reaches the reference position but the pendulum overshoot is quite far from desired. If $\mathbf{Q}_2$ is selected, the pendulum swings are well within the limits, however the rise time of the arm is unacceptably slow. It has therefore been concluded that the optimal time behavior of the arm and the pendulum are two conflicting requirements which cannot be satisfied simultaneously. Any successful tuning of weight matrices must therefore result in a reasonable compromise between the quick rise time of the base and the low overshoot of the pendulum.
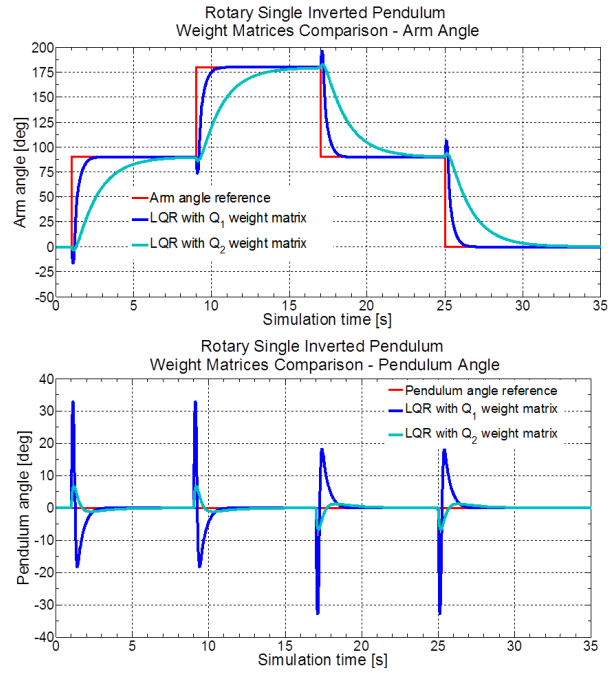
3) State-Feedback Control with Permanent Disturbance Compensation

Applying state-feedback control with feedforward gain on a system is insufficient if the steady-state effect of a permanent disturbance input needs to be compensated. The structure of the State-Feedback Controller with Summator (SFCS) block implements a summator term $v(i)$ which sums up all past error values. This ensures that the system output will track all changes in the reference command and the influence of permanent disturbances will be eliminated. The evaluated control law, presented and derived by Modrlak in [17], is specified as:

$$u(i) = u_R(i) + u_{ff}(i) + u_S(i) =$$
$$- \vec{k}_1 \vec{x}(i) + \vec{k}_2 w(i) + v(i), \qquad (16)$$

where $w(i)$ is the reference command and $\vec{k}_1, \vec{k}_2$ are gain matrices which are computed from the matrix structure:

$$\left(-\vec{k}_1 \quad -\vec{k}_2 \right) =$$
$$\left(-\vec{k}\mathbf{F} \quad -1 - \vec{k}\mathbf{g} \right) \left(\begin{matrix} \mathbf{F} - \mathbf{I} & \mathbf{g} \\ \mathbf{c}_1 & 0 \end{matrix} \right)^{-1}, \quad (17)$$

assuming that $\vec{k}$ (alternatively $\vec{k}_D$)is a standard feedback gain vector equivalent to the one in Section 3.1.2. The layout of the block mask once again allows the user to adjust the number of block input ports to match a specified control requirement and to compute $\vec{k}/\vec{k}_D$ using a preferred method and state-space description [2].
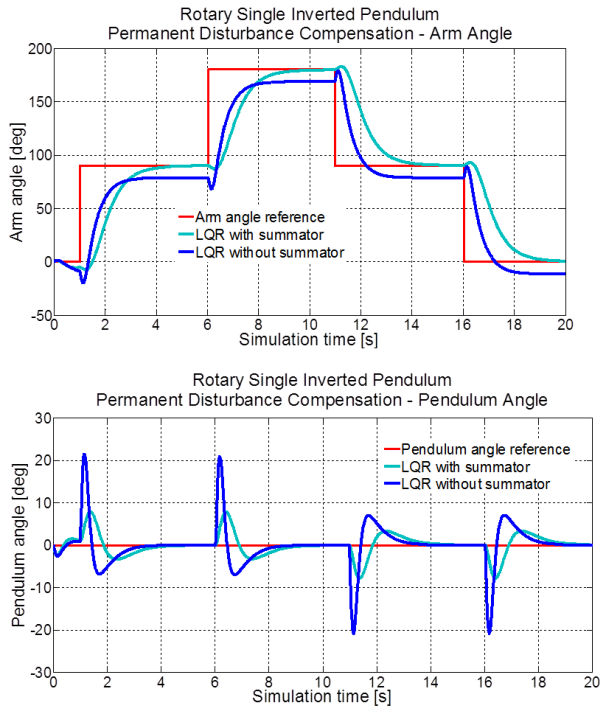
Fig. 12: Rotary single inverted pendulum - simulation results for LQR control with a summator.

The control law (16) was verified for the rotary single inverted pendulum voltage model and the weight matrices of the standard discrete-time LQ functional (15) were set to $\mathbf{Q} = diag(500 \quad 0 \quad 20 \quad 0), r = 1$. The obtained results were compared to those of a conventional discrete-time LQR controller using the same values of weight matrices as in Section 3.1.2. A constant disturbance input (5 V for the first controller, 1 V for the second) was present in both simulations. To compensate for measurement limitations, the LE block was included in both schemes, with the estimator poles set to the same values as before. As it can be seen in Fig. 12, the conventional LQR controller fails to track the reference trajectory without producing steady-state error, but the permanent disturbances are successfully compensated by a LQR algorithm with a summator included in the control structure.

3.2. Design of a Control Strategy for Swing-up and Stabilization of the System

If the initial position of the pendulum is equivalent to the natural hanging position, then the additional control problem of swinging it upwards leads to a hybrid control setup which consists of three basic components, schematically depicted in the block diagram in Fig. 13:

- a swing-up controller, which actuates the arm with an input signal $u_{SW}(t)$ of appropriate direction and magnitude, making the pendulum swing with an increasing amplitude and angular speed until it enters the stabilization zone (balancing region) around the upright position [7],

- a stabilizing (balancing) controller, which maintains the pendulum in the upright position with a stabilizing control signal $u_{ST}(t)$ designed by available feedback control techniques, such as the state-feedback control law based on Jacobian linearization described in section 3.1, or a variety of nonlinear control design approaches [18],

- a transition (switching) mechanism, which intercepts the pendulum when it nears the upright position (i.e. crosses the borderline of the balancing region) and switches to stabilizing control.
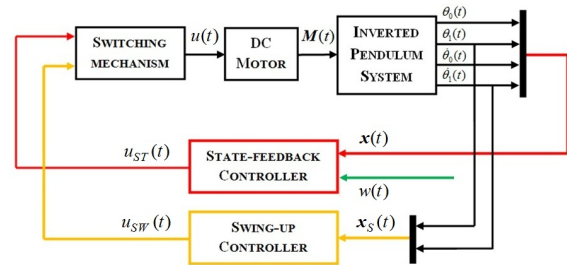

Fig. 13: Block diagram of swing-up and stabilizing control of IPS.

1) Energy-based Methods for Swing-up Control

One of the earliest and most effective approaches to pendulum swing-up is the method based on energy considerations, proposed and explained by Aström, Furuta and Iwase in [19], [20]. The goal is to bring the total mechanical energy W of the pendulum to zero, which corresponds to the upright position. The first derivative of W for the uncontrolled pendulum (2) in the simplified form is given as:

$$
\begin{aligned}
\dot{W} &= \frac{d}{dt}(\frac{1}{2}ml^2\dot{\theta}_1^2(t) + \frac{1}{2}m_1gl_1(cos\theta_1(t) - 1)) = \\
&= J_1\dot{\theta}_1(t)\ddot{\theta}_1(t) - \frac{1}{2}m_1gl_1\dot{\theta}_1(t)sin\theta(t) = \\
&= -\frac{1}{2}m_1l_0l_1\ddot{\theta}_0(t)\dot{\theta}_1(t)cos\theta(t).
\end{aligned}
\tag{18}
$$

Whether the expression in (18) is greater or lower than zero depends directly on the sign of $\dot{\theta}_1(t)(cos\theta_1(t))$. The basic energy-based control law (cosine value controller) therefore becomes:

$$
u_{SW}(t) = -u_m sgn(\dot{\theta}_1(t)(cos\theta_1(t))).
\tag{19}
$$

This control law can be simplified into a law which will be referred to as a zero speed controller [21]:

$$u_{SW}(t) = u_m sgn(\dot{\theta}_1(t)), \qquad (20)$$

or modified into an absolute value control law:

$$u_{SW}(t) = u_m sgn(\dot{\theta}_1(t) |\theta_1(t)|). \qquad (21)$$

The control laws (19), (20), (21), were encapsulated into the structure of Swing-up Controller block which is part of the Inverted Pendula Swing-up sublibrary and allows the user to select a swing-up method, the constraints of the balancing region, and the input magnitude u_m.

The following swing-up / stabilizing control strategy was proposed and verified for the both the torque model and voltage model of the system. As the control objective, the freely hanging pendulum had to be brought into the upright equilibrium and the arm needed to be stabilized at the origin. Initial conditions of the system were therefore set to $\theta_0(0) = -1$ rad $(-57,3$ degrees$)$ and $\theta_1(0) = -\pi$ rad $(-180$ degrees$)$. All three swing-up control laws were tried out and for each control law, the input magnitude was tuned to the highest value which still allows successful swing-up with no oscillations or destabilization. Once the pendulum was approaching the the upright position, control was switched to a balancing controller. State-feedback controller based on discrete-time state-space description was selected and the weight matrices of the discrete-time LQ functional (15), were specified as $Q = diag(100 \quad 20 \quad 20 \quad 0), r = 1$ for each simulation experiment. Switching between the controllers took place when the pendulum was 0,6 rad (34,4 degrees) away from the upright position.

Figure 14 depicts the comparison of the swing-up control laws (19), (20), (21) applied on the torque model of the system. It turned out that although all control laws met the control objective, the rotary inverted pendulum performed at its best for the cosine-value controller - the system's performance was diminished for other controllers, which is caused by long intervals during which energy is taken from the pendulum.

When tuning the input magnitude u_m, the time needed to swing the pendulum to the upright position was discovered to be indirectly proportional to u_m. However, feasible input values could only be selected from a bounded interval. An unlimited increase in magnitude leads to the increase of the pendulum angular velocity, and a too low magnitude may either be entirely unable to bring the pendulum upright, or it needs numerous swings to do so, which results in a displacement of the rotary arm away from the equilibrium point. In both cases, the control mechanism would fail
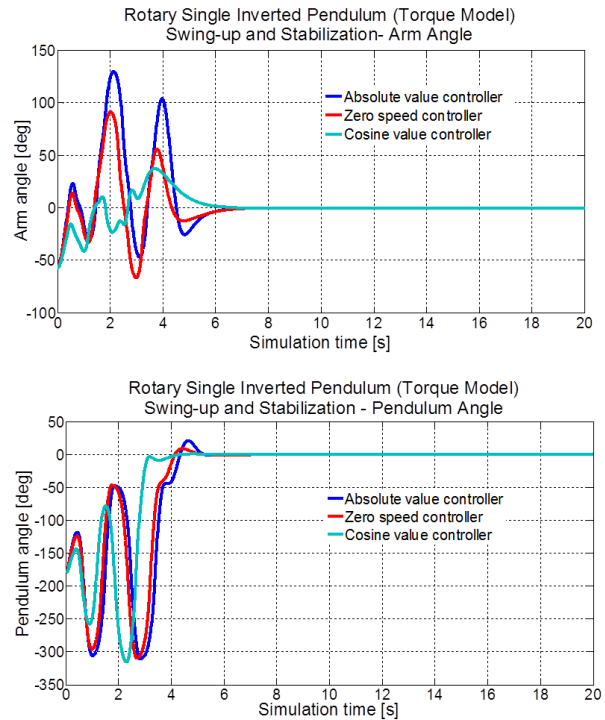

Fig. 14: Swing-up and stabilization of the rotary single inverted pendulum torque model - comparison of methods.
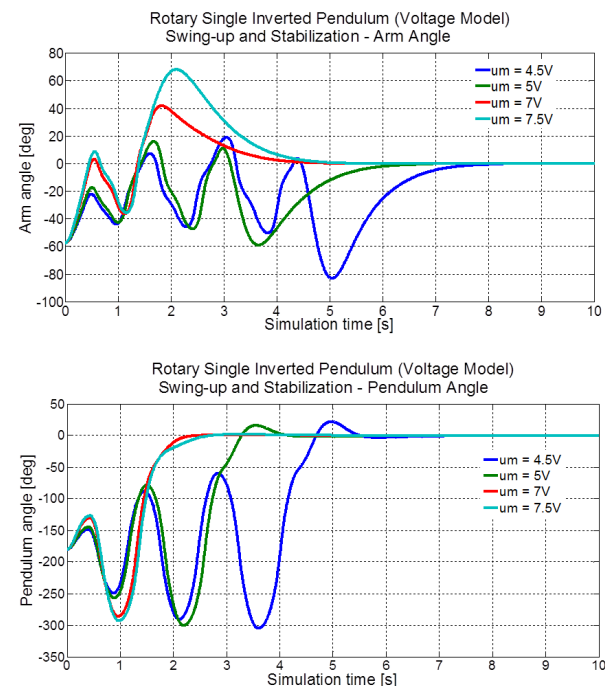

Fig. 15: Swing-up and stabilization of the rotary single inverted pendulum voltage model, comparison of the effect of varying input magnitude.

to stabilize the system, proving that the balancing controller is only effective if the full state of the system is close to the equilibrium point in the moment of inter-

ception. An illustrative experiment which compares the effect of a varying u_m on the rotary single inverted pendulum voltage model, is depicted in Fig. 15.

4.   Conclusion

The purpose of this paper was to present a comprehensive approach to the problem of modeling and control of the rotary single inverted pendulum system. A custom-designed Simulink block library developed by the authors of the paper, Inverted Pendula Modeling and Control (IPMaC), was used as a software framework for all covered issues which included mathematical model derivation, open-loop analysis and linear approximation of the system, as well as followed by design and verification of stabilizing state-feedback control algorithms as well as energy-based swing-up control laws.

The IPMaC block library provided suitable function blocks to support every step of the process of analysis and control design (such as a pre-prepared simulation model of the rotary single inverted pendulum system or a detailed state-feedback controller block), together with several original applications with graphical user interface. One such application was used to derive the mathematical model for the selected inverted pendulum system in the form of symbolic equations of motion; the other performed the automatic linear approximation of the system in a chosen equilibrium point. In these applications, great practical potential of symbolic mathematical software was demonstrated.

With the aid of developed program tools, the paper has presented a set of coherent results together with thorough evaluations. The validity of the nonlinear model generated by the above application was verified by comparing the obtained results to those listed in the resources, and by evaluating the open-loop responses of the simulation model. The outcome of both approaches was highly acceptable: all deviations were justified by the laws of mechanics and it was shown that the dynamic behavior of both the torque and the voltage model is in agreement with the empirically observed facts on pendulum movement. The ability to control the rotary inverted pendulum system with respect to the principal control objective (stabilization in the inverted position) and a variety of secondary issues was demonstrated for all implemented state-feedback control techniques which included pole-placement and the LQR optimal control method; the simulation results also justified the use of the techniques based on a linearized model to control nonlinear systems. To deal with the problem of swinging the pendulum upwards from the natural hanging position, several heuristic swing-up methods based on energy considerations were proposed and compared from multiple viewpoints.

The IPMaC block library enhances the capabilities of the MATLAB/Simulink program environment by providing effective means for analysis and control of an important class of nonlinear mechanical systems. Moreover, the readily available collection of mathematical and simulation models of inverted pendulum systems can be viewed as a testbed model basis for exploring the properties of underactuated nonlinear systems, mechatronic systems, hybrid systems and other concepts. As a meaningful contribution to modeling and control education, the library has been included into the research and teaching activities of the Center of Modern Control Techniques and Industrial Informatics (http://kyb.fei.tuke.sk/laben/) at the Faculty of Electrical Engineering and Informatics, Technical University of Kosice.

Acknowledgment

References

[1] BAKER, G. J. and J. A. BLACKBURN. *The Pendulum: a Case Study in Physics.* New York: Oxford University Press, 2005. ISBN 978-0199557684.

[2] JADLOVSKA, S. and J. SARNOVSKY. An Extended Simulink Library for Modeling and Control of Inverted Pendula Systems. In: *19th Annual Conference Proceedings of the International Scientific Conference - Technical Computing Prague 2011.* Prague: 2011. ISBN 978-80-7080-794-1.

[3] JADLOVSKA, S. Modeling and Optimal Control of Inverted Pendula Systems. Kosice, 2011. pp. 93. Diploma Thesis. Supervisor: prof. Ing. Jan Sarnovsky, CSc. Faculty of Electrical Engineering a Informatics, Technical University of Kosice.

[4] JADLOVSKA, S. and A. JADLOVSKA. Inverted Pendula Simulation and Modeling-a Generalized Approach. In: *Proceedings of the 9th International Scientific Technical Conference on Process Control - RIP 2010.* Kouty nad Desnou: Tribun EU, 2010, pp. C134a-1–11. ISBN 978-80-7399-951-3.

[5] JADLOVSKA, S. and J. SARNOVSKY. A Complex Overview of the Rotary Single Inverted Pendulum System. In: *Proceedings of the 9th International Conference - ELEKTRO 2012*. Rajecke Teplice: IEEE, 2012, pp. 305–310. ISBN 978-1-4673-1178-6. DOI:10.1109/ELEKTRO.2012.6225609.

[6] JADLOVSKA, S. and J. SARNOVSKY. Classical Double Inverted Pendulum - a Complex Overview of a System. In: *Proceedings of the IEEE 10th Jubilee International Symposium on Applied Machine Intelligence and Informatics (SAMI 2012)*. Helany: IEEE, 2012, pp. 103–108. ISBN 978-1-4577-0195-5.

[7] FURUTA, K., M. YAMAKITA and S. KOBAYASHI. Swing Up Control of Inverted Pendulum. In: *Proceeding of the International Conference on Industrial Electronics, Control and Instrumentation (IECON'91)*. Kobe: IEEE, 1991, vol. 3, pp. 2193–2198. ISBN 0-87942-688-8. DOI: 10.1109/IECON.1991.239008.

[8] ERNEST, P. and P. HORACEK. Algorithms for Control of a Rotating Pendulum. In: *Proceeding of the 19th IEEE Mediterranean Conference on Control and Automation (MED '11)*. Athens: National Technical University, 2011. ISBN 960-87706-0-2.

[9] Kats, C. J. A. *Nonlinear control of a Furuta Rotary Inverted Pendulum*. DCT Report, no. 2004:69.

[10] REX Controls s. r. o.: *Rotary Inverted Pendulum FPM-210/211. User Manual*, 2010.

[11] Symbolic Math Toolbox User's Guide [online]. 2013. Available at: http://www.mathworks.com/help/pdf_doc/symbolic/symbolic_tb.pdf.

[12] BOGDANOV, A. *Optimal Control of a Double Inverted Pendulum on the Cart*, Technical Report CSE-04-006, OGI School of Science and Engineering, OHSU, 2004.

[13] HAJKO, V. and J. DANIEL-SZABO. *Fundamentals of Physics*. Bratislava: VEDA - Slovak Academy of Sciences, 1980.

[14] Quanser Academic, Rotary Motion Servo Plant: SRV02, User Manual, no. 700, Rev. 2.3.

[15] JADLOVSKA, S. and J. SARNOVSKY. Swing-up and Stabilizing Control of Classical and Rotary Inverted Pendulum Systems. In: *Proceeding of the 12th Scientific Conference of Young Researchers (SCYR 2012)*. Herlany: The Technical University of Kosice 2012, pp. 38–41, ISBN 978-80-553-0943-9.

[16] SARNOVSKY, J., A. JADLOVSKA and P. KICA. *Optimal and Adaptive Systems Theory*. Kosice: ELFA s r.o., 2005. ISBN 80-8086-020-3.

[17] MODRLAK, O. *Automatic Control Theory, vol. 1 - Basic State-Space Analysis and Synthesis (Study Materials)*. Liberec: Technical University of Liberec, 2002.

[18] KHALIL, H.K. *Nonlinear Systems*, 3nd ed. New York: Macmillan, 2001. ISBN 978-0130673893.

[19] ASTROM, K.J. and K. Furuta. Swinging up a pendulum by energy control. In: *Proceeding of the 13th IFAC World Congress*. San Francisco: Oxford, 1996, pp. 1–6. ISBN 9780080429243.

[20] FURUTA, K. and M. IWASE. Swing-up time analysis of pendulum. *Bulletin of the Polish Acknowledgement of Sciences-Technical Sciences*. 2004, vol. 52, no. 3, pp. 153–163. ISSN 0239-7528.

[21] BUGEJA, M. Non-linear Swing-up and Stabilizing Control of an Inverted Pendulum System. In: *The IEEE Region 8 EUROCON 2003 Computer as a tool (EUROCON'03)*. Ljubljana: IEEE, 2003, vol. 2, pp. 437–441. ISBN 0-7803-7763-X. DOI: 10.1109/EURCON.2003.1248235.

About Authors

Slavka JADLOVSKA was born on June 25th, 1988 in Kosice, Slovak Republic. In 2011, she graduated (M.Sc.) with distinction at the Department of Cybernetics and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University in Kosice. She is currently pursuing her Ph.D. degree at the same department. Her dissertation research is focused on hybrid methods of control for nonlinear dynamic systems. In addition, she has been researching issues related with modeling of underactuated mechatronic systems, optimal and predictive control design for hybrid systems, and the advanced application of MATLAB toolboxes in modeling and control. So far, she has authored several scientific articles published in international conference proceedings and in a SCOPUS-cited journal.

Jan SARNOVSKY was born on March 28th, 1945. He obtained his M.Sc. degree in Technical Cybernetics on the Faculty of Electrical Engineering of the Slovak Technical University Bratislava in 1968. He defended his dissertation thesis in the field of Automation and Control in 1980 at the same University; his thesis title was "Control of Large Scale Systems Using Hybrid Computer Systems". Since 1969 he has been working at Faculty of Electrical

Engineering and Informatics Technical University in Kosice as a Associate Assistant, since 1980 as Associate Professor and since 1993 as a Professor. Since 1985 he has been working as a tutor at the Department of Cybernetics and Artificial Intelligence. His scientific research focuses on the large scale systems and multiagent systems in control of the large scale systems. He has authored a number of scientific articles and contributions published in the journals and international conference proceedings. He is also an author of several monographs.