Manual for 2-DOF Inverted Pendulum *

Laboratory Courseware
Department of Signals and Systems,
Chalmers University of Technology
August 24, 2016

*Peixi Gong (MPSYS student), Balázs Kulcsár (kulcsar@chalmers.se) Automatic Control Group

Figure 1: Experimental lab steup, Quanser 2DOF Inverted Pendulum, [3]

1 Introduction

The inverted pendulum is a special, up-side-down positioned (regular) pendulum, see Fig. 1. In most of the case a rigid rod is mounted in a small cart/manipulator via a pivot. Since, the inverted pendulum is an unstable physical system, the task with such system is to balance (move the manipulator) in order to keep the rod in its upright position. The laboratory device is a 2-degree of freedom pendulum where the rod can turn in 2 orthogonal directions. Instead of a cart, this system has two robot arm manipulators, to move the pivot position and keep the rod in its upright position.

This courseware and lab experiment contributes to the understanding of stabilizing systems with linear optimal control theory. Did you know rocket launching and landing or segways are inverted pendulum like problems, see Fig. 2?

Figure 2: Rocket and segway. Source: Wikipedia

This syllabus is designed for students who are taking the course *Linear control system design(SSY285)* at Chalmers University of Technology, Gothenburg, Sweden and are selecting the 2-DOF Inverted Pendulum as an option for the lab session. The information contained within this paper has two main parts; (*i*) mandatory a-priori preparation, (*ii*) real time experiment. Students are supposed to read this paper carefully as well as perform all the exercises before attending the session.

The layout of the labware is the following; on the basis of first principal formalism, a linear time invariant model is derived and analyzed for the inverted pendulum. This LTI model will be used to develop optimal state estimator and controller for implementation.

2 Equipment

The equipment used is a 2-DOF Inverted Pendulum (Quanser) with four-links and two robot manipulators. The goal is to implement linear stationary Kalman Filter, Linear Quadratic Regulator, and Linear Quadratic Gaussian controller. Part of this labmanual is taken from [3].

As you can see from Figure 3, the servo x (left hand side servo) controls the pendulum in direction x while the servo y (right hand side servo) controls the pendulum in direction y. The rod can tilt in direction x and y too. Links that are directly connected to the servos are called rotary arms. The *home position* of the 2-DOF Inverted Pendulum is represented by the schematic draw illustrated in Figure 4. This position is defined as $\theta_x = \theta_y = 0$, and rod angle $\theta_x = \theta_y = 0$. Angles increase positively in counter-clockwise(CCW), the servos (and thus the arm) turn CCW when the servo control voltage is positive [3]. The inverted pendulum angle α_x and α_y is zero when it is perfectly in the upright position and increases positively CCW [3]. The home position is selected and used as an operating point for the rest of the studies.

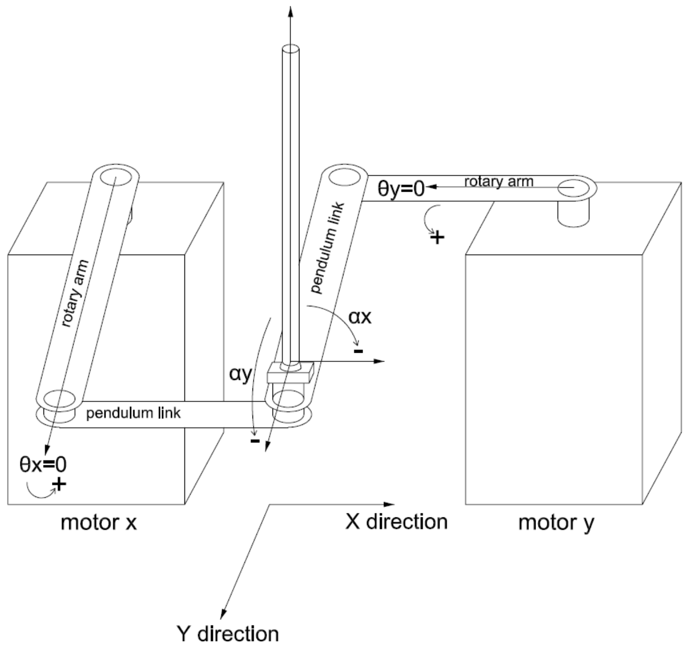

Figure 3: 2-DOF Inverted Pendulum.

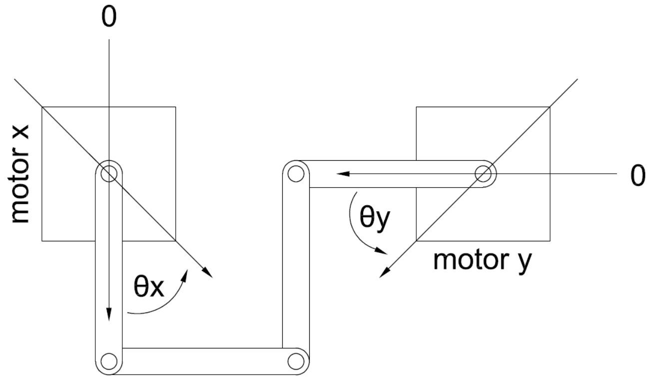

Figure 4: 2-DOF Robot HOME Position.

3 First principal pendulum model

If the rod does not deviate much from its home position, the coupling in the dynamics of two directions is negligible. Hence, the 2-DOF Inverted Pendulum can be modelled as the combination of two identical, but 1-DOF Inverted Pendulum. One pendulum moves in x direction while the other in y direction. In the sequel we will use this simplification.

First principal pendulum model can be obtained by finding the equations of motion for robot manipulators with multiple joints and rod displacement. We suggest to use the Euler-Lagrange method to derive the dynamics equations (*Modeling and control of mechatronic systems(SSY156)*).

Exercise 1(optional) Formulate and derive the equations of motion based on Euler-Lagrange method.

Answer

3.1 Nonlinear Equations of Motion

The nonlinear equations of motion for 1-DOF Rotary Pendulum are given

$$
\begin{aligned}
\ddot{\theta}(t) = f_3(t) = & \frac{-\frac{1}{2}((4M_pL_p^2\alpha(t)\dot{\theta}(t)\dot{\alpha}(t) - 8C_oV_m(t) + 8D_r\dot{\theta}(t))J_p + M_p^2L_p^4\alpha(t)\dot{\theta}(t)\dot{\alpha}(t))}{((4J_r + 4M_pL_r^2)J_p + M_pL_p^2J_r)} \\
& - \frac{-\frac{1}{2}((M_p^2L_p^3L_r\dot{\theta}^2(t) + 20M_p^2L_p^2L_rg)\alpha(t) - 2M_pL_p^2D_r\dot{\theta}(t) + 2M_pL_p^2C_oV_m(t))}{((4J_r + 4M_pL_r^2)J_p + M_pL_p^2J_r)}
\end{aligned} \tag{1}
$$

$$
\begin{aligned}
\ddot{\alpha}(t) = f_4(t) = & \frac{((M_p^2L_p^2L_r^2 + M_pL_p^2J_r)\dot{\theta}^2(t) + 20J_rM_pL_pg + 20M_p^2L_r^2L_pg)\alpha(t)}{((4J_r + 4M_pL_r^2)J_p + M_pL_p^2J_r)} \\
& + \frac{2M_pL_rL_pC_oV_m(t) - 2M_pL_rL_pD_r\dot{\theta}(t) - M_p^2L_p^3L_r\alpha(t)\dot{\theta}(t)\dot{\alpha}(t)}{((4J_r + 4M_pL_r^2)J_p + M_pL_p^2J_r)}
\end{aligned} \tag{2}
$$

Parameters that are used in the equations are listed in Table 1.

Parameters	**Description**	**Value**
M_p	Pendulum mass with T-fitting	0.1270 kg
L_r	Length of rotary arm	0.1270 m
L_p	Length of the pendulum(w/T-fitting)	0.3111 m
J_r	Equivalent inertia with the 4-bar linkage	0.0083 $kg.m^2$
J_p	Pendulum inertia around CoG	0.0012 $kg.m^2$
D_r	Arm viscous damping coefficient	0.0690 $N.m.s/rad$
C_o	Voltage convert coefficient	0.1285 $N.m.s/rad$
g	Gravitational constant	0.981 $kg.s^2$

Table 1: Parameter values

$V_m(t)$ is the control input signal (voltage to the servo motor). As about the model outputs, there are four encoders that separately measure $\theta_x(t)$, $\theta_y(t)$ and $\alpha_x(t)$, $\alpha_y(t)$. The above nonlinear equation can be linearized around the upright, home position. That position is an operating point for the linearization. Note, all variable are zero at the home position.

3.2 Linear Time-invariant State-space Model

The standard form for a dynamic model, suitable for simulation and controller design, is the state-space form

$$\dot{x}(t) = A_\delta x(t) + B_\delta u(t) \tag{3}$$

and

$$y(t) = C_\delta x(t) + D_\delta u(t), \tag{4}$$

where $x(t)$ is the state, $u(t)$ is the control input, A_δ, B_δ, C_δ, and D_δ are state-space matrices. For 1-DOF Rotary Pendulum, the linear states are defined

$$x(t) = [\delta\theta(t)\ \delta\alpha(t)\ \delta\dot{\theta}(t)\ \delta\dot{\alpha}(t)]^T, \tag{5}$$

where $\delta\cdot$ refers to the fact that small variables change around the known operating point. Only angles can be sensed by encoders, hence, the states $x_1(t)$ and $x_2(t)$ are measured and the outputs of 1-DOF Rotary Pendulum are obtained as

$$y(t) = [\delta\theta(t)\ \delta\alpha(t)]^T = [x_1(t)\ x_2(t)]^T. \tag{6}$$

The control input becomes

$$u(t) = \delta V_m(t). \tag{7}$$

According to eqs.(1) − (2), the system state equations are given by

$$\frac{\mathrm{d}\theta(t)}{\mathrm{d}t} = \dot{\theta}(t) = f_1(t) \tag{8}$$
$$\frac{\mathrm{d}\alpha(t)}{\mathrm{d}t} = \dot{\alpha}(t) = f_2(t) \tag{9}$$
$$\frac{\mathrm{d}\dot{\theta}(t)}{\mathrm{d}t} = f_3(t) \tag{10}$$
$$\frac{\mathrm{d}\dot{\alpha}(t)}{\mathrm{d}t} = f_4(t) \tag{11}$$

After determining the operating point, linearization results in matrices A_δ and B_δ as

$$A_\delta = \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\right|_{(\theta=\alpha=\dot{\theta}=\dot{\alpha}=0)}$$

$$B_\delta = \left.\begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix}\right|_{(\theta=\alpha=\dot{\theta}=\dot{\alpha}=0)}$$

For C_δ and D_δ, the measured states are $\delta\theta(t)$ and $\delta\alpha(t)$, meanwhile, the control input does not influence the measurement, so

$$C_\delta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad D_\delta = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{12}$$

Exercise 2 Linearize the nonlinear model with the parameter values in Table 1 around the operating point. Find the numerical values for the matrices A_δ and B_δ.

Answer

3.3 Angular tracking error

To minimize the servo to drift away from the home position while balancing the pendulum (this is called position tracking on θ), let us augment the LTI model's state-space. Hence, an integrator can be added by expanding the meaning of the state as,

$$x \triangleq [x(t) \;\; z(t)]^T, \tag{13}$$

where $z(t) = \int (\delta\theta(t) - \delta\theta_{ref})dt$ with $\delta\theta_{ref} = 0$, i.e. to track the cumulative error of $\delta\theta$ from its home position. The new state-space matrices are redefined as (A, B, C, D).

Exercise 3 Expand the linear model with the above angular postion error and find the numerical values for the augmented system as (A, B, C, D).

Answer

4 Open Loop System Analysis

4.1 Stability

Stability of a state space model can be analyzed by the eigenvalues of the matrix A.

Exercise 4 Is the system A, B, C, D stable? Find the zeros of the system. What can you conclude out of the zeros? Check both systems $(A_\delta, B_\delta, C_\delta, D_\delta)$ and (A, B, C, D). Does the inclusion of extra state change stability property? Are the above models asymptotically stable?

Answer

4.2 Observability

If by only measuring the output $y(t)$ and input $u(t)$, the states of the system are determinable at any time, then the system is called observable. Observability can be checked by alternative ways.

Exercise 5 Find the observability matrix (Kalman rank test), check if the system is observable. Check both systems $(A_\delta, B_\delta, C_\delta, D_\delta)$ and (A, B, C, D). Does the inclusion of extra state change observability property? **Answer**

Exercise 6(optional) Check observability by means of PHB (Popov-Hautus-Belevich) test. **Answer**

4.3 Controllability

If the control input, $u(t)$, of a system can take each state variable, $x(t)$ from any non-zero initial state to the origin of the state coordinate frame under finite time, the LTI system is called full state controllable. To check controllability, we can use different tests.

Exercise 7 Find the controllability matrix(Kalman ran test), check if the system is controllable. Check both systems $(A_\delta, B_\delta, C_\delta, D_\delta)$ and (A, B, C, D). Does the inclusion of extra state change the property?

Answer

Exercise 8(optional) Check controllability by means of PHB(Popov-Hautus-Belevich) test.

Answer

5 Controller Synthesis

5.1 State feedback control design

A feedback control loop is designed to stabilize the pendulum at its home position. The feedback policy with a possible time varying reference signal $(r(t))$ can be written as,

$$u(t) = K(r(t) - x(t)) = -Kx(t), \tag{14}$$

the closed-loop state-space equation becomes

$$\dot{x}(t) = Ax(t) + B(-Kx(t)) = (A - BK)x(t), \tag{15}$$

given the fact that $r(t)$ is zero (home position referencing). K can be obtained in different ways, for example, pole placement design or linear quadratic optimization algorithm.

5.2 Linear Quadratic Regulator

Stationary LQR optimization method can be used to find a feedback control gain as $\bar{K}$. Here, the task is to find a control input $u(t)$ that minimizes the cost function

$$J = \frac{1}{2}\int_0^\infty (x(t)^T Q_x x(t) + u(t)^T Q_u u(t))dt. \tag{16}$$

where Q_x and Q_u are penalty weightings, $Q_x \in \mathbb{R}^{5\times 5}$, $Q_u \in \mathbb{R}^{1\times 1}$. By solving the Control Algebraic Ricatti Equation offline we may find solution to the problem of $\min_{u(t)} J(u)$. Based on the solution matrix of the Ricatti equation, $\bar{K}$ can be obtained.

Exercise 9 Given the fact that angles can reach at most 20 degree, the angular velocities 20 degree/sec, and the control input voltage can change at most ± 10 V, find reasonable values for Q_x (diagonal) and Q_u by means of Bryson rule. Apply an expensive LQR control policy. Find the gain $\bar{K}$.

Answer

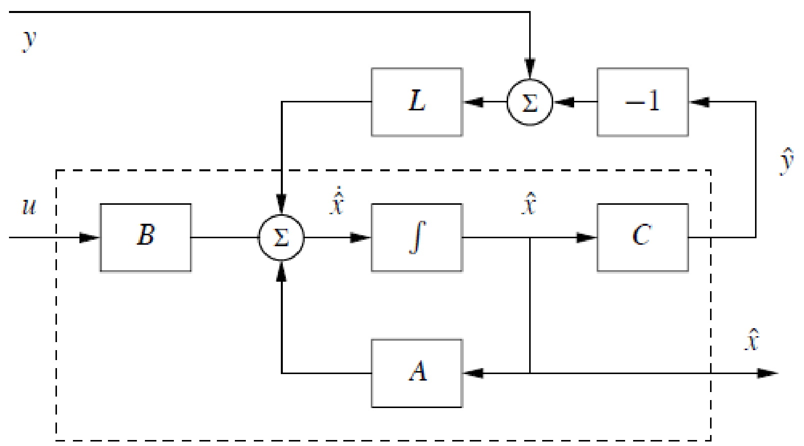

Figure 5: Block diagram of the observer.

6 State Estimator

6.1 State observer design

In order to accomplish the control task, information of all states is necessary. Unfortunately, this inverted pendulum device can not measure neither the rotary arm velocity (state x_3) nor the pendulum link velocity (state x_4). Therefore, we apply state observer and reconstruct the missing state information for $\bar{K}$. Block diagram of the structure of an observer is shown in Figure 5. The observer takes the signals y and u as inputs and produces estimated state vector as $\hat{x}(t)$. Notice that the observer contains a copy of the process model(Luenberger observer) that is driven by $y(t) - \hat{y}(t)$ through the observer gain L.

Remark 1. *Remember, when designing an observer, do not estimate $z(t)$, the cumulative tracking error (it is measured). Although in the sequel a full order state observer is designed, a reduced order one would suffice, since the angles are directly and almost noise-free measured.*

The equation of observer is given by

$$\dot{\hat{x}}(t) = A_\delta \hat{x}(t) + B_\delta u(t) + L(y(t) - C_\delta \hat{x}(t)) \tag{17}$$

6.2 Stationary Kalman filter design

There are ways to determine the observer gain, an optimal way is to use Kalman Filter algorithm. Suppose to have the noise corrupted continuous-time plant given by,

$$\begin{aligned} \dot{x}(t) &= A_\delta x(t) + B_\delta u(t) + Gw(t) \\ y(t) &= C_\delta x(t) + Hv(t) \end{aligned} \tag{18}$$

with known input $u(t)$ and $y(t)$, but unknown and uncorrelated white process $w \sim \mathbb{N}(0, Q_w)$, and measurement noise $v \sim \mathbb{N}(0, Q_v)$ satisfying

$$G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{19}$$

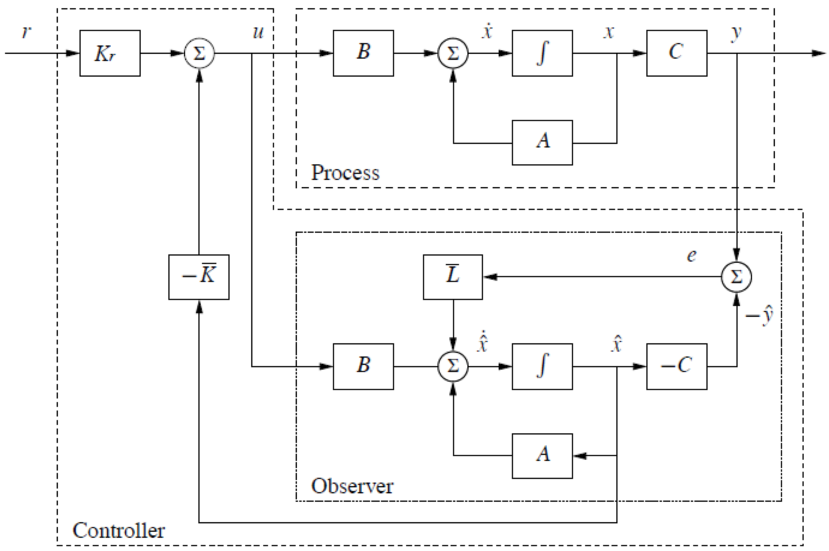

Figure 6: Block diagram of an observer-based control system.

construct a state estimate $\hat{x}(t)$ that minimizes the steady-state error covariance

$$\bar{P} = \lim_{t\to\infty} E(\{x(t) - \hat{x}(t)\}\{x(t) - \hat{x}(t)\}^T). \tag{20}$$

The optimal solution is the Kalman filter with equations

$$\dot{\hat{x}}(t) = A_\delta \hat{x}(t) + B_\delta u(t) + \bar{L}(y(t) - C_\delta \hat{x}(t)) \tag{21}$$

$$\hat{y}(t) = C_\delta \hat{x}(t). \tag{22}$$

The filter gain $\bar{L}$ is determined by solving the Filter Algebraic Riccati Equation(FARE).

Exercise 10 Define $Q_w = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$ and $Q_v = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, find the Kalman Filter gain $\bar{L}$.

Answer

6.3 LQG controller design

Based on the principle of separation, $\bar{K}$ and $\bar{L}$ can independently be synthesised. The interconnection of the state observer and the optimal control gain results in an output feedback controller, in our case called Linear Quadratic Gaussian controller, see the block diagram in Figure 6. The observer uses the measured output $y(t)$ and the input $u(t)$ to construct an estimation of the states $\hat{x}(t)$. This estimated states is then gained up with $\bar{K}$ for the $u(t)$.

7 Closed-loop simulation

In this section you will use the Simulink diagram shown in Figure 5. to simulate the closed-loop behaviour of the 1-DOF Rotary Inverted Pendulum system. Recall that the 2-DOF Inverted Pendulum is modelled as two independent and identical Rotary Pendulum systems.

Exercise 11

1. Start Matlab, direct to folder **2DOFIP**.
2. Open **Setup_sim.m**. By means of symbolic tools, derive from the nonlinear pendulum model its LTI model around the home position. Second, by using the model parameters, find the numerical values for the state-space matrices $(A_\delta, B_\delta, C_\delta, D_\delta)$ (original matrices, not the error-augmented ones). This state-space is used for simulating the plant and Kalman Filter design.
3. *Augment the state−space model matrices* with tracking error integration, to obtain (A,B, C,D). This state-space is used for computing stationary LQR gain (but not the Kalman gain).
4. *Calculate the LQR control gain* by using (A, B, C, D). Define Q_x diagonal and Q_u. Apply the same values as in **Exercise 9**.
5. *Calculate the stationary Kalman Filter gain* by using the non-augmented model $(A_\delta, B_\delta,$ $C_\delta, D_\delta)$. Apply the same Q_w, Q_v values as in **Exercise 10**.
6. Run the script **setup_sim.m**.
7. Open Simulink file **Simulation.mdl**.
8. Plug your LQR gain in *LQR gain* block (be careful with the sign), and the Kalman Filter gain in *Kalman Filter gain* block. Enter the augment matrices in the corresponding model blocks, remember to use the non-augmented system matrices in the observer design. In subsystem *measured states scope* you can check the measured states, in subsystem *estimated states scope* you can check the states which are estimated by observer.
9. Run *SRV02+SIP linear model.* As 'initial conditions' for the state space model apply e.g. $pi/180 * [-5, 2, 1, 1, 0.3]$. This vector will simulate the small deviations from operating point at the beginning. You can try reasonable different initial values. Try different disturbance inputs added to the plant input. Put down your observation.

Answer

Exercise 11 (optional)

1. In Simulink, create a nonlinear block for the pendulum model. Replace the linear pendulum model with a nonlinear one then. Remember to compute the tracking error separately. The integrators initial conditions should be set to small, but close to the home position values. Couple and run the nonlinear model with the above computed LQG controller. Put down your observation.

Answer

PLEASE BRING ALONG YOUR MATLAB SCRIPT AND SIMULINK FILE TO THE LABSESSION.

8 Laboratory Session

8.1 Attentions

- Be aware, the inverted pendulum device is quite sensitive, work gently with it.
- Before you run your code, ask permission to execute it from the teaching assistant!
- Before turning on the voltage amplifier device, ensure the voltage gains of both amplifier ports are set to 1×(Figure 7), or the pendulum device will be damaged!
- If there comes error about '*quarc_comm*' when running the experiment with Simulink, go to *SRV02+2DIP-E*→ *HIL-Initialize*, set *Board type* to *q8_usb*. If the problem still persists, select QUARC|Set default options.
- After long time running, the 6 thumb screws(Figure 7) on the pendulum might be loose and have some bad impact, ensure the thumb screws are properly adjusted to be tense.

8.2 Experiment

The **Experiment.mdl** in Simulink as shown in Figure 8 is used to balance the 2-DOF Inverted Pendulum to keep the rod in the upright position. The 2-DOF Inverted Pendulum contains **QUARC**® blocks that interface with the DC motor and the angular sensors of the 2-DOF Inverted Pendulum system.

As you can see from Figure 8, the direction x and y are separately controlled by two identical LQG controllers, due to the fact of link coupling around the home position.

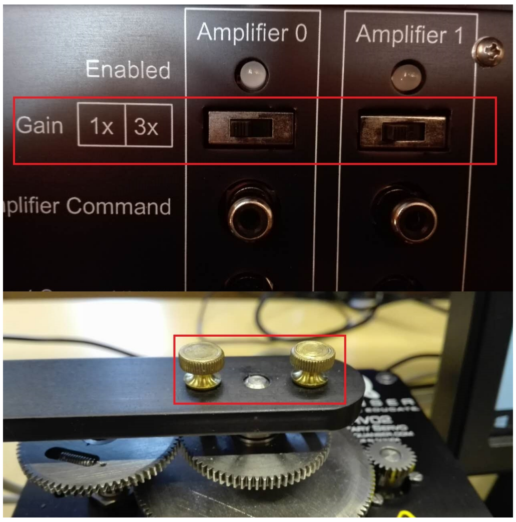

Figure 7: Voltage Amplifier Gain.

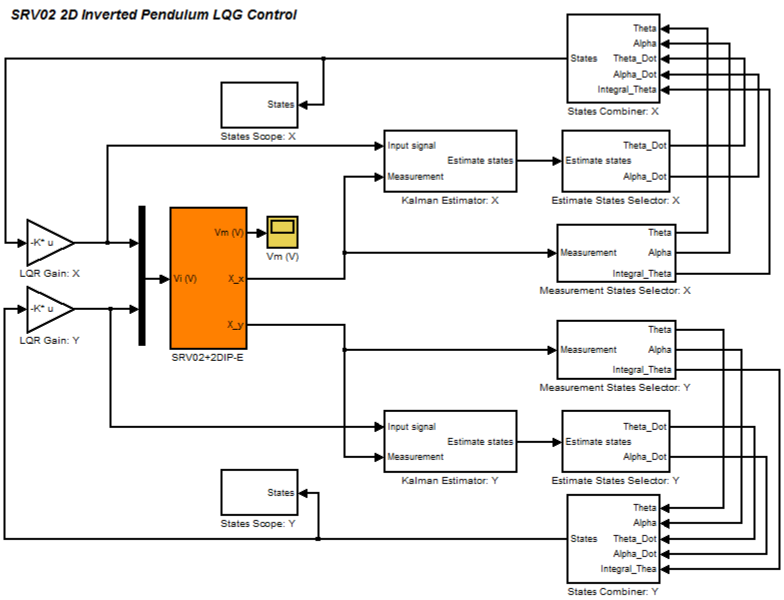

Figure 8: Simulink model used with **QUARC®** to run controller on the 2-DOF Inverted Pendulum.

Experiment

1. Open the script **setup_exp.m**.
2. Copy your settings from **setup_sim.m** to the corresponding sections in this script.
3. Run the script.
4. Open Simulink file **Experiment.mdl**.
5. Plug in your setting values in the related blocks.
6. Open the scopes in subsystems *states scopes:X*, *states scopes:Y* and the motor input voltage scope.

7. In the Simulink diagram, run QUARC|Build.

8. Manually bring the pendulum to the upright vertical position and hold it by the uppermost tip (home position). Make sure that the pendulum is centered along both axes and is motionless.

9. To start the controller, click on the *Connect to target* button and then the *Start* on the Simulink toolbar (or select QUARC|Start from the menu). Gently release the pendulum once you feel the servo begin to stabilize the pendulum.

10. Inspect the movement of the pendulum, and check the scopes.

8.3 Task for experimenting with the inverted pendulum

Task 1

1.1 Select the following weighting functions for optimization

$$Q_x = diag([500\,500\,1\,1\,1]),\ Q_u = 0.2, Q_w = 50 \cdot diag([1, 1]),\ Q_v = 0.01 \cdot diag([1,\ 1])$$

Calculate the corresponding servo LQR $\bar{K}$ and Kalman filter gain L based on the model developed in the preparation phase. Run the implementation, start the experiment from the home position.

1.2 Try to release the pendulum from a different initial position than the home one. Try different alternatives. Is the previously computed LQG controller still stabilizing? Put down your experience and the reason why it behaved so.

1.3 Release your pendulum form the home position again with the previosuly computed LQG controller. While the controller is working, gently push the rod to deviate it from its stable position. Does the controller stabilize and pull it back to the reference position? Put down your observation.

1.4 Now, modify the value for Q_u, first use the value 0.05 and then 0.5. What is the main difference in their closed-loop behaviour? Can you connect that to linear optimal control theory?

1.5 Gently push the rod to deviate it from its stable position. Does the controller stabilize and pull it back to the reference position? What is the difference in between the two controllers? Write down your conclusion.

1.6 Implement an LQG controller where the cost functional is free of speed penalization. Change the weightings Q_u, Q_x accordingly with $Q_w = 50 \cdot diag([1, 1])$, $Q_v = 0.01 \cdot diag([1,\ 1])$.

1.7 (optional) Select $Q_x = diag([500\,500\,1\,1\,0])$ and do the LQG design. Implement the controller. Write down your experiences.

Answer

Task 2

2.1 Modify the Simulink block by adding scopes to compare the measured and the reconstructed angular position errors (α, θ both x and y dimensions). With weightings,

$$Q_x = diag([500\,500\,1\,1\,1]),\ Q_u = 0.5, Q_w = 50 \cdot diag([1, 1]),\ Q_v = 0.01 \cdot diag([1,\ 1])$$

implement the LQG controller. Did you get small or large the values as errors? Change Q_u to 0.05. Does it help to reduce the error between the observation and the reconstruction?

2.2 By only modifying the Kalman filter gain, we will influence the error. Given the following range of weights, $Q_w = q_w \cdot diag([1, 1])$, $Q_v = q_v \cdot diag([1,\ 1])$ with $q_w = 50 \ldots 300$ and $q_v = 0.01$, which extremal weighing strategy earns you the least error? Did you experienced unstable behaivour/very bad performance? If yes what may be the reason for that? What would happen if q_w would increase to infinity do you think?

2.3 Select the q_w that returns with the smallest angle error. Analyze it with disturbances. Gently push the rod to deviate it from its stable position. Does the controller stabilize and pull it back to the reference position, how the error changes in the transient phase? Comment on that.

2.4 We will change q_v this time. Given the following range of weights, $Q_w = q_w \cdot diag([1, 1])$, $Q_v = q_v \cdot diag([1,\ 1])$ with $q_w = 50$ and $q_v = 0.001 \ldots 0.1$, which extremal weighing strategy earns you the least error? Did you experienced unstable behaivour/very bad performance? If yes what may be the reason for that? What would happen if q_v would increase even further?

2.5 Take the least error providing q_w and q_v, implement the LQG controller. Write down your experinces

Answer

References

[1] O Wigström, *3-DOF Helicopter Laboratory Session*, laboratory courseware and manual, Chalmers University of Technology, Göteborg, Sweden, 2014.

[2] T Wik, *LQG control of a double-tank*, laboratory courseware and manual, Chalmers University of Technology, Göteborg, Sweden, 2004.

[3] J Apkarian, H Lacheray, M Lévis, *LABORATORY GUIDE: 2 DOF Inverted Pendulum Experiment for MATLAB®Simulink®Users*, Quanser, 2012.

[4] B Kulcsar, *Lecture notes for Linear Control System Design, SSY285*, Chalmers University of Technology, Göteborg, 2013-16.