Automatic Control II

Computer exercise 3

LQG Design

Preparation exercises:
All exercises in Section 3.

Reading instructions: Glad-Ljung, Chapters 6, 8.5 and 9.
Contents

1 Introduction .................................................. 1
2 LQG: Short theoretical background .................. 1
3 Preparation exercises ................................. 2
4 Balancing a stick ......................................... 4
5 The inverted pendulum .................................. 7
   5.1 State feedback: LQ design ......................... 8
   5.2 Optimal state estimation: The Kalman filter ..... 9
1 Introduction

In this computer exercise, controller synthesis with LQG design will be investigated. The synthesis will be performed on two variants of the inverted pendulum. In Section 4 the system is a stick balancing on a finger, and in Section 5 the system is a pendulum mounted to a cart, which is actuated by a DC motor. In Section 4 the linear model and the effects thereupon are emphasized. In Section 5 the use of LQG for a nonlinear system is focused.

To be well prepared for this computer exercise one should have read the Chapters 6, 8.5 and 9 in the course book by Glad-Ljung thoroughly. It may also be beneficial to have read the Chapters 5 and 7.

For the computer exercise, access to Matlab and the Control System Toolbox is necessary. There are also a number of .m-files that are needed.

2 LQG: Short theoretical background

The standard LQG problem is to minimize a quadratic criterion for a linear system. That is, to find the controller that, when applied to the system, minimizes the criterion. The standard model (as in Equation (9.4) in Glad-Ljung) of the system is

$$\dot{x} = Ax + Bu + Nw$$
$$z = Mx$$
$$y = Cx + v$$

where $z$ is the performance signal, $y$ is the measured output signal, and $w$ and $v$ are the white noise-signals with covariance matrices

$$Eww^T = R_1, \quad Evv^T = R_2, \quad Euv^T = R_{12}$$

For simplicity, assume that $w$ and $v$ are independent, so that $R_{12} = 0$.

The control error, i.e., the discrepancy between the performance signal and the reference signal is denoted $e = z - r$. It is of course desirable to keep $e$ as small as possible. The standard formulation of LQG, though, treats the regulation problem, i.e., when $r \equiv 0$. This means that $e = z$. The criterion to be minimized then is

$$V = E \lim_{T \to \infty} \frac{1}{T} \int_0^T (z(t)^TQ_1z(t) + u(t)^TQ_2u(t)) \, dt$$

where $Q_1$ and $Q_2$ are symmetric matrices. $Q_2$ is positive definite and $Q_1$ is positive semidefinite.

When treating the servo problem, when $r \neq 0$, the criterion should be appropriately modified. However, here $r$ will be confined to be piece-wise constant, like set point changes. In this case LQG will give a feedback gain that is the same as for the regulation problem.
The optimal control law has the familiar observer based state feedback structure (see Theorems 9.1 and 9.2 in Glad-Ljung)

\[ u(t) = -L\hat{x}(t) + L_r r(t) \] (6)

where \( \hat{x}(t) \) is the optimal estimate of \( x(t) \), obtained by the Kalman filter for the system (1)–(3). The gain \( L \) depends only on \( Q_1 \) and \( Q_2 \), while the Kalman filter depends only on \( R_1 \) and \( R_2 \) (and \( R_{12} \) if nonzero).

Note that the LQG-controller is optimal only for the idealized situation, that is, when the system is exactly described by (1)–(3) and when \( w \) and \( v \) are white Gaussian noises. This is a very idealized situation. In practical cases one or more of the conditions for the LQG framework may not be fulfilled. In this case study the system is not even a linear model. Hence, it can not be expected that the obtained controller will be optimal. Still, for many systems the LQG approach is useful due to its simplicity and to its ability to generate controllers that behave rather well also in nonidealized situations.

In practice \( R_1 \) and \( R_2 \) are unknown and may, together with \( Q_1 \) and \( Q_2 \), be regarded as design variables. There are some guidelines in how to choose these design matrices.

For simplicity the discussion is confined to diagonal choices of \( Q_1 \) and \( Q_2 \). Each element in these matrices can be interpreted as a penalty of the corresponding signal component. The rule of thumb then is that this signal component will be small in the closed loop simulation if it is penalized considerable. That is, if the corresponding diagonal entry in the diagonal matrices are chosen large. Still, it is stressed that this is only a rule of thumb and that the design procedure should be regarded as an iterative procedure supported by closed loop simulations.

3 Preparation exercises

Consider a system given in the standard state space form:

\[ \dot{x} = Ax + Bu + Nw \] (7)
\[ z = Cx \] (8)
\[ y = z + v \] (9)

The system is controlled with the LQG-controller

\[ \dot{x} = A\hat{x} + Bu + K(y - C\hat{x}) \] (10)
\[ u = -L\hat{x} + L_r r \] (11)

where (10) is the Kalman filter and (11) is the control law. \( r \) is the reference signal.
Exercise 3.1: The state estimation error is $\hat{x} = x - \hat{x}$. What is the dynamics for $\hat{x}$?

Answer:

Exercise 3.2: Give the state space description for the closed loop system. Use

$$\begin{bmatrix} x \\ \hat{x} \end{bmatrix}$$

as state vector.

Answer:

For the next exercise the following identity — regarding block triangular matrices — is useful:

$$\begin{bmatrix} \Gamma & \Phi \\ 0 & \Psi \end{bmatrix}^{-1} = \begin{bmatrix} \Gamma^{-1} & -\Gamma^{-1}\Phi\Psi^{-1} \\ 0 & \Psi^{-1} \end{bmatrix}$$

$\Gamma$ and $\Psi$ are square nonsingular matrices, and $\Phi$ is a matrix of compatible dimension.
Exercise 3.3: The Kalman filter does not affect the dynamics from reference signal to output signal. Show this by computing the transfer function from reference to output. Use the state space description obtained in the previous task.

Answer:

4 Balancing a stick

Example 7.1 in Glad-Ljung concerns the problem of balancing a stick on the finger. For simplicity the situation is idealized to the case where all motion is confined to a vertical plane. Also, the finger is confined to move along a horizontal line, in the $\xi$-direction. The input signal is the acceleration in this direction, i.e., $u = \ddot{\xi}$. The output signal is the angle of the stick with respect to a vertical line (see Figure 7.7 in Glad-Ljung). The second order differential equation (7.32) in Glad-Ljung describes the relation between the input and the output signals. Introducing the state variables $x_1 = y$ and $x_2 = \dot{y}$ yields the nonlinear state space description

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{2g}{l} \sin x_1 - \frac{2}{l} u \cos x_1 \\
y &= x_1
\end{align*}
\]

If the system is linearized around $x = 0$ and $u = 0$, the linear (and hence approximative) state space model is

\[
\begin{align*}
\dot{x} &= \begin{bmatrix} 0 & 1 \\ \frac{2g}{l} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{2}{l} \end{bmatrix} u \\
y &= \begin{bmatrix} 1 & 0 \end{bmatrix} x
\end{align*}
\]

The corresponding transfer function is

\[
G(s) = \frac{-2/l}{s^2 - \frac{2g}{l}}
\]
The system has two poles, in $\pm \sqrt{27} + \frac{1}{2}$, and is thus unstable.

Consider the case when the stick is hit on its top. This can be modeled as an impulse disturbance (remember the analogy between impulses and white noise) on the angle acceleration. Hence it enters the system like the input signal, but with opposite sign. The standard model of the system then is

$$\begin{aligned}
\dot{x} &= \begin{bmatrix} 0 & 1 \\ \frac{1}{2} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -\frac{1}{2} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} w \\
z &= \begin{bmatrix} 1 & 0 \end{bmatrix} x \\
y &= \begin{bmatrix} 1 & 0 \end{bmatrix} x + v
\end{aligned}$$

where $w$ is the process noise, representing the disturbance, and $v$ is the measurement noise.

LQG is applied to design controllers for the system. Set $Q_1 \equiv 1$ and $R_1 \equiv 1$ (this is no restriction, since it is only the ratios, $Q_1/Q_2$ and $R_1/R_2$, that affects the controllers for SISO systems).

First set $R_2 = 10^{-6}$. For $Q_2 = 10^{-2}, 10^{-3}$ and $10^{-4}$, generate controllers with \texttt{lqgstick}. To use this command, type \texttt{reg1=lqgstick(Q1,Q2,R1,R2)}; and so on. Use \texttt{tfstick} to study the transfer functions. Type \texttt{tfstick(reg1,reg2,reg3)}.

**Exercise 4.1:** How does the magnitude of $Q_2$ affect (a) the bandwidth of $G_{ry}(s)$? (b) the bandwidth of $S(s)$? (See Section 6.6 in Glad-Ljung for definitions on bandwidth.)

**Answer:**

Simulate the system with \texttt{simstick} for the different controllers. This is done by writing \texttt{simstick(reg1,reg2,reg3)}. This will simulate a unit step on the reference signal, and a unit impulse disturbance (a hit on the top of the stick). The responses on the output and input signals are shown. It is the linear model that is simulated.
Exercise 4.2: How does the magnitude of $Q_2$ affect (a) the output signal? (b) the input signal?

Answer:

Now let $Q_2 = 10^{-3}$. For $R_2 = 10^{-5}, 10^{-6}$ and $10^{-7}$, generate controllers with lqgstick.

Exercise 4.3: How does the magnitude of $R_2$ affect (a) the bandwidth of $G_{ey}(s)$? (b) the bandwidth of $S(s)$?

Answer:

Simulate the system for the different controllers.
Exercise 4.4: How does the magnitude of $R_2$ affect (a) the output signal? (b) the input signal?

Answer:

NB: Before continuing with the next section, initialize Matlab with the function initpend. Simply type initpend.

5 The inverted pendulum

The system in this Section is essentially the same as in Section 4, but it is slightly more complex. In this case the stick, referred to as the pendulum, is mounted to a cart (vagn) via an articulated shaft (ledad axel). The cart stands upon a track which it can move along. The cart is actuated by a DC motor, whose shaft is connected to a gear. The gear meshes with a toothed rack on the track. Figure 1 shows a sketch of the system. The voltage over the DC motor is the input signal of the system. There are two measured output signals; the position of the cart, $\xi$, and the angle, $\alpha$, of the pendulum with respect to a vertical line. They

![Figure 1: The inverted pendulum.](image-url)
are chosen as the performance signal. Two disturbance signals are considered in the model used here. The first is acting on the input signal, i.e., on the voltage over the DC motor. The second is acting on the torque of the pendulum, like a hit on the top of the pendulum, as in the previous system in Section 4. The disturbances are assumed to be white noises or impulses.

Applying classical mechanics a nonlinear state space model can be derived. A fourth order model with the state vector \( x = [\xi \quad \alpha \quad \dot{\xi} \quad \dot{\alpha}]^T \) is obtained. The model is linearized around the origin, \( x = 0 \) and \( u = 0 \), and the linear standard model

\[
\dot{x} = Ax + Bu + Nw \\
z = Cx \\
y = Cx + v
\]

is obtained. The process noise is \( w = [w_{1u} \quad w_{1a}]^T \), where \( w_{1u} \) is the disturbance on the input signal, and \( w_{1a} \) is the disturbance on the torque of the pendulum. The performance signal is \( z = [\xi \quad \alpha]_T \), and the measured output signal is the performance signal plus the measurement noise, \( v \). The matrices in this specific model are

\[
A = \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & -2.7 & -14 & 0 \\
0 & 29 & 33 & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
0 \\
0 \\
3.2 \\
-7.5
\end{bmatrix}, \quad N = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
3.2 & -4.3 \\
-7.5 & 47
\end{bmatrix}
\]

and

\[
C = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

In the simulations the nonlinear model will be used. The simulations are performed under circumstances that should be as similar to the situation in a real experiment as possible. For instance, the controller is implemented as a sampling controller, the measurements are quantized, and so on. Most important here, though, is that the input signal is bounded; \( u \) must lie within the interval \([-6, 6]\) Volts. This must be taken into account when the controller is designed.

Notice that the controller design is based on the linear model. Also, the transfer functions reflects the effects on the linear model. However, the goal is to get a controller that works on the nonlinear model in the simulations.

5.1 State feedback: LQ design

One method to design the controller is to first find a state feedback gain \( L \) that would work well if pure state feedback was applicable. Then, once such \( L \) is found, the Kalman filter is chosen so that the true but unknown state vector can be replaced with its best estimate in order to compute the control signal\(^1\).

\(^1\)It is of course possible to do it the other way around as well, i.e., to first determine the Kalman filter, and then \( L \), or one could try to determine both simultaneously.
Initially it may be beneficial to focus on one parameter in the tuning procedure. Then, as one has obtained something that roughly works as desired, the other parameters can be taken into account for the fine-tuning of the controller. One aspect is to compare the input and output signals, i.e., the relative size of $Q_2$ compared to $Q_1$. Therefore, set $Q_1 \equiv I$ and vary $Q_2$. Note that, $Q_1$ should be a $2 \times 2$ symmetric positive definite matrix and $Q_2$ should be positive scalar, see(5). The objective is to achieve as fast controller as possible, without saturating the input signal. Generate the controller (the feedback gain $L$) with \texttt{lqpend}, and simulate the system with \texttt{simpend}. In the simulation an impulse disturbance is imposed upon the system at $t = 0.5$ seconds, as if somebody hit the top of the pendulum. The input signal, the voltage over the DC-motor on the cart, must lie in the range $[-6,6]$ Volts. The speed of the system can be judged from the cart position. The cart should return to zero as fast as possible.

Start with $Q_2 = 1$. To see any essential effect in the behavior of the system, $Q_2$ should be changed at least by a factor of ten. Type \texttt{lqpend(Q1, Q2)}.

**Exercise 5.1:** What is your choice for $Q_2$?

**Answer:**

5.2 Optimal state estimation: The Kalman filter

Stick to $Q_1 = I$ and the $Q_2$ obtained above. As for the state feedback control problem discussed above, the Kalman filter depends only on the process and measurement noise intensities, $R_1$ and $R_2$. In practice these are unknown and may be regarded as design parameters. A priori knowledge about the noise intensities can be useful in the tuning procedure.

Once again it is sufficient to consider only one parameter. It is rather natural to compare the intensities of the process noise $R_1$ and the measurement noise $R_2$. Set $R_1 = I$ and $R_2 = rI$, where $r > 0$ is varied. Note that both $R_1$ and $R_2$ are $2 \times 2$ symmetric positive definite matrices. Choosing $r$ small means that the measurement noise is considered small compared to the process noise. The corresponding Kalman filter will then become fast, since it will put more trust in the measurements than in the system dynamics. And, vice versa, a large $r$ will give a slower Kalman filter. In this system the measurement noise is probably small (primarily caused by quantization in the sensors). It is harder to judge the size of the process noise. It includes the disturbances in the system, like the impulse torque on the pendulum (the hit on the top) that occurs
in the simulations. However, it also may represent uncertainties in the model. Particularly in this case where the real system is nonlinear, while the model is linear, this should be taken into account. The system is unstable as well, which invokes the controller to be fast enough (and particularly the state estimator). All together this suggest that $r$ should be chosen quite small. Try some different values on $r$. To see any essential effect in the behavior of the system, $r$ should be changed by a factor of ten or hundred. Generate the controllers with \texttt{lqgpend}. With \texttt{simpend} the nonlinear model is simulated. Find an $r$ for which the simulated nonlinear system is stable and behave well. Type \texttt{reg $r$ = lqgpend(Q1,Q2,R1,R2)} to generate the controller and \texttt{simpend(reg)} to simulate the controller.

\textbf{Exercise 5.2:} What $r$ have you chosen?

\begin{center}
\textbf{Answer:}
\end{center}

Keep $R_1 = I$ and $R_2 = rI$ with the $r$ obtained above. Also keep $Q_1 = I$. Generate new controllers where $Q_2$ is decreased and increased by a factor of ten. Study the transfer functions for these controllers with \texttt{tfpend}. Compare with the working controller obtained above. Note that the transfer functions are matrices in general. The \textit{nonzero singular values} of the corresponding transfer function matrix will be plotted against frequency.

\textbf{Exercise 5.3:} What happens? Is this in agreement with the corresponding results in Section 4?

\begin{center}
\textbf{Answer:}
\end{center}

Reset $Q_2$ to the earlier obtained value, and keep $Q_1 = I$ and $R_1 = I$. Generate controllers with $R_2 = rI$, where $r$ is decreased and increased by a factor of ten. Compare the transfer functions for these controllers with the ones for the working controller.
Exercise 5.4: What happens? Is this in agreement with the corresponding results in Section 4?

Answer:

Now generate a controller for

\[ Q_1 = \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix} \]

with \( Q_2, R_1 \) and \( R_2 \) as for the working controller. Simulate the nonlinear system with this controller and the earlier obtained working controller. Compare the results.

Exercise 5.5: Describe what happens.

Answer: