# Assignment 2 Control Design

December 23, 2020

Jaya Parmar, MATLAB filename assign_2.m

## 1 Introduction

Our aim is to study the pairing, decoupling of the 2 input 2 output system. We will study the system responses using feedback with PI and LQG controllers in three different scenarios - one without decoupling, another in steady state decoupling and third in dynamic decoupling.

## 2 Method

Our closed loop system transfer function from assignment 1 was below,

$$G_o = [G_{11}\ G_{12}; G_{21}\ G_{22}] \tag{1}$$

where

$$G_{11} = \frac{15}{(50s+1)} \tag{2}$$

$$G_{12} = \frac{(-11)}{10s+1)} \tag{3}$$

$$G_{21} = \frac{-24.5}{(50s+1)} \tag{4}$$

$$G_{22} = \frac{70}{(10s+1)} \tag{5}$$

### 2.1 Pairing

We use Relative Gain Array (RPA) to do the input output pairing. At cross over frequency 0.1rad/s our equation (1) becomes

$$G_{cross} = [15/6\ -11/2; -24.5/6\ 70/2]; \tag{6}$$

Relative Gain Array (RPA) at cross over frequency is found using ' pinv' command

$$R_w = G_{cross}.*pinv(G_{cross}).' \tag{7}$$

$$R_w = \begin{bmatrix} 1.3453 & -0.3453 \\ -0.3453 & 1.3453 \end{bmatrix} \tag{8}$$

Relative Gain Array (RGA) at cross over frequency $\frac{1}{10}$ is almost an identity matrix $R_w$. We disregard the cross coupling between non-diagnol elements since they have negative values. We pair u1 with y1 and u2 with y2 and decouple the MIMO system.

## 2.2 Decoupling

### 2.2.1 Steady-state decoupling s=0

Substituting s=0 in equation (1), we get

$$G_0 = [15\ -11; -24.5\ 70]; \tag{9}$$

with $W_{10} = G_0^{-1}$ and $W_2 = I$

$$G_{0d} = W_2 * G_o * W_{10} \tag{10}$$

where $G_{0d}$ is the steady state decoupled matrix

### 2.2.2 Dynamic decoupling s=0.1i

Substituting s=0.1i in equation (1), we get

$$G_w = [15/(5i+1)\ -11/(1i+1); -24.5/(5i+1)\ 70/(1i+1)]; \tag{11}$$

Taking the real valued matrix from the complex valued matrix $G_w$

$$G_{wr} = real(G_w) \tag{12}$$

with $W_{1w} = G_{wr}^{-1}$ and $W_2 = I$

$$G_{wd} = W_2 * G_o * W_{1w} \tag{13}$$

where $G_{wd}$ is the dynamic decoupled matrix

## 2.3 Feedback controller Fy

Let $F_y = [1\ 0; 0\ 1]$ be the unit feedback controller

1. Case 1 - Feedback without decoupling

$$G_{co} = F_y * G_o/(eye(2) + F_y * G_o); \tag{14}$$

2. Case 2 - Static decoupling matrix

$$F_0 = F_y * W_{10} \tag{15}$$

$$G_{c0} = F_0 * G_o/(eye(2) + F_y * G_o) \tag{16}$$

3. Case 3 - Dynamic decoupling matrix

$$F_w = F_y * W_{1w} \tag{17}$$

$$G_{cw} = F_w * G_o/(eye(2) + F_y * G_o) \tag{18}$$

We plot the three cases with feedback controller $F_y$ in figure 1 in the 'Results' section

## 2.4 PI controller with integration time 10s

Let $PI = \frac{(1+10s)}{10s}$ be the PI controller in the direct path

$$G_{copi} = PI * G_o/(eye(2) + PI * G_o) \tag{19}$$

We plot the closed loop step response of the standard system compared to PI controller in figure 2 in the 'Results' section

## 2.5 LQG controller

We need to make a state space equation from the given transfer functions first for our 2-input 2-ouput system,

$$u = [1\ 0; 0\ 1] \tag{20}$$

$$y1 = [15/(50*s+1) - 11/(10*s+1); -24.5/(50*s+1)70/(10*s+1)] * u \tag{21}$$

$$sys1 = ss(y1) \tag{22}$$

This will give us matrices A, B, C and D for our state space equation
We will introduce process disturbance 'w' in this equation. Let v be random noise

$$v_1 = randn(2,2) \tag{23}$$

It is often desired to have an integrator in the feedback loop, so that the constant disturbances can be eliminated in the steady state.

$$w = \frac{v_1}{s+\delta} \tag{24}$$

where $\delta = 0.001$ i.e. very small

$v_2$ is the white noise causing measurement disturbances.
The state space representation matrices A, B, C and D will now be

$$A = \begin{bmatrix} -0.02 & 0 & 0 & 0 \\ 0 & -0.10 & 0 & 0 \\ 0 & 0 & -0.001 & 0 \\ 0 & 0 & 0 & -0.001 \end{bmatrix} \tag{25}$$

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 4 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{26}$$

$$C = \begin{bmatrix} 0.30 & -0.2750 & 1 & 0 \\ -0.49 & 1.750 & 0 & 1 \end{bmatrix} \tag{27}$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{28}$$

So the new system with process disturbance will be

$$sys2 = ss(A, B, C, D) \tag{29}$$

The lqr command in matlab needs sensor or measurement noise covariance R. Let's choose $R_1 = 1$ ; $R_2 = 0.001$ and $R_12 = 0$

$$R = [R1\ R12; R12\ R2]; \tag{30}$$

Q is the process noise covariance. Let's keep Q1 = 1 and change Q2 which is our penalty matrix since it penalises the input 'u'.

1. Q2 = 1

$$Q2a = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{31}$$

2. Q2 = 0.1

$$Q2b = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \tag{32}$$

3. Q2 = 0.01

$$Q2c = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \tag{33}$$

Feeding this to 'lqr' command in Matlab to find the optimal gain L that minimizes the quadratic cost function. We will use 'lqr' in a function *LQRcontrol* since following steps will be repeated for three different values of Q2.

We will also find the loop gain $G_k$ for the closed loop system; gain and phase margins for $G_k$; sensitivity and complementary sensititivity function. Please refer to function 'LQRcontrol' in Matlab.

In the results section, figure 3 shows the closed loop systems; figure 4 shows the sensitivity functions; figure 5 shows the complementary sensitivity functions and figure 6 shows nyquist diagrams for the three values of Q2.

# 3 Results

Figure 1

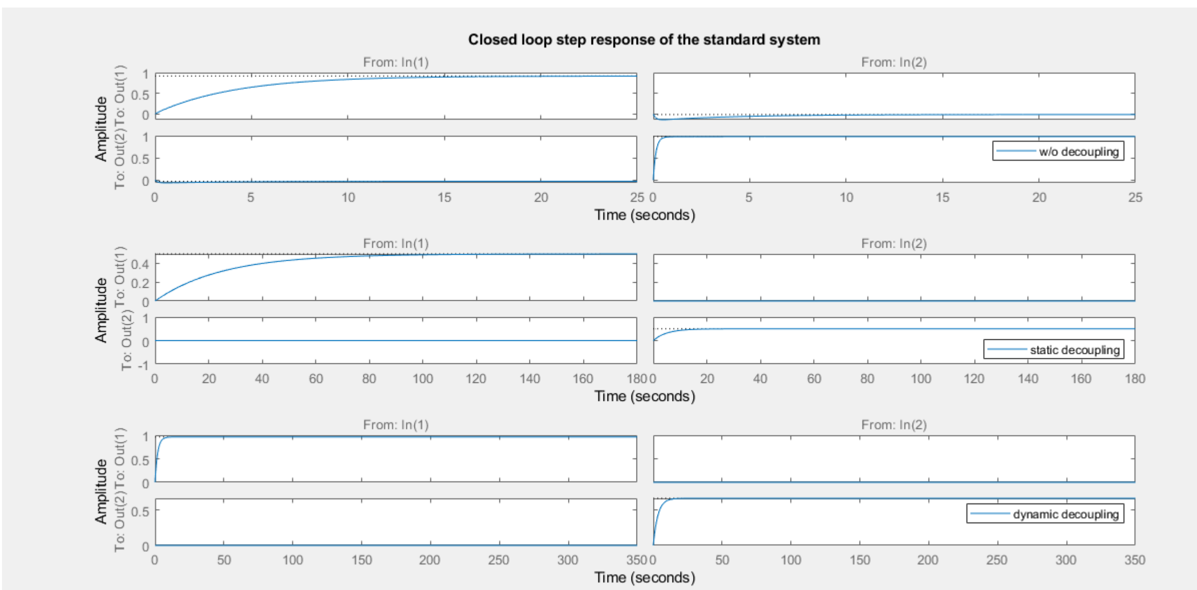



Figure 2

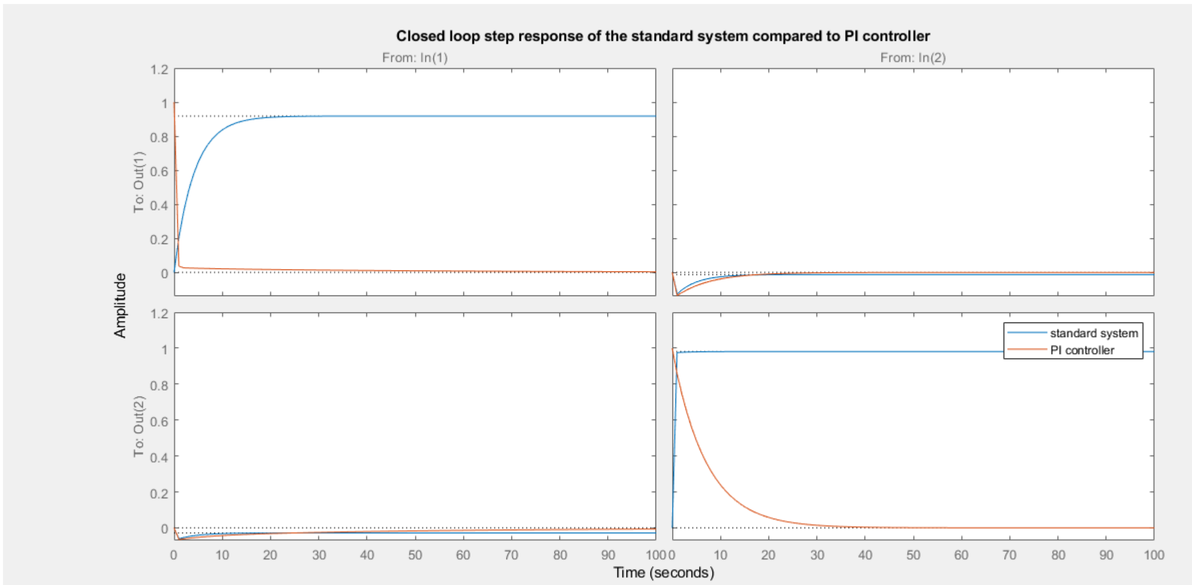



Figure 3

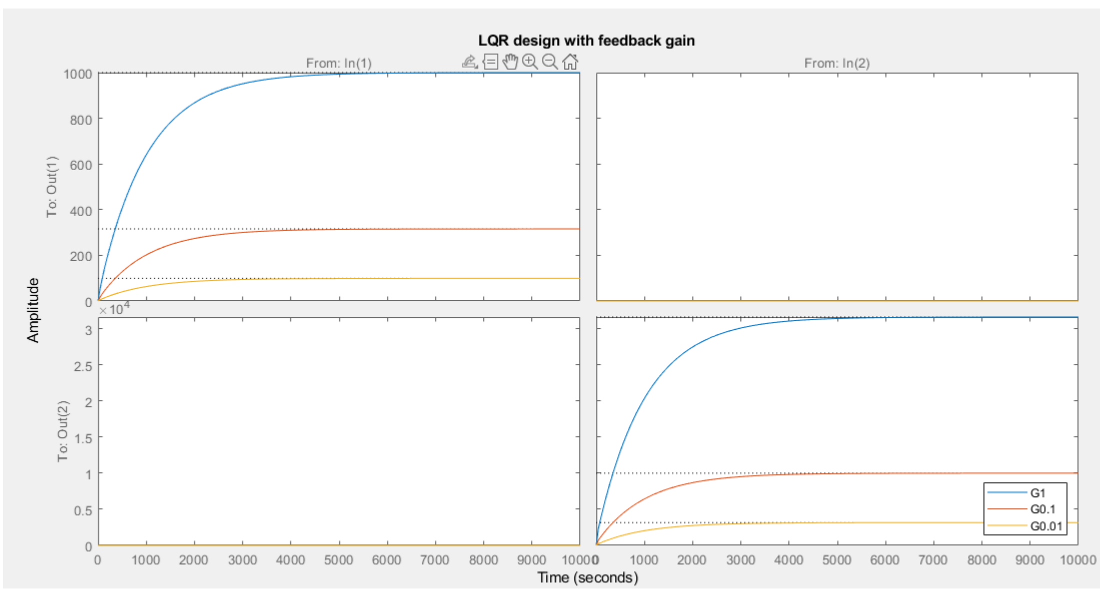



Figure 4

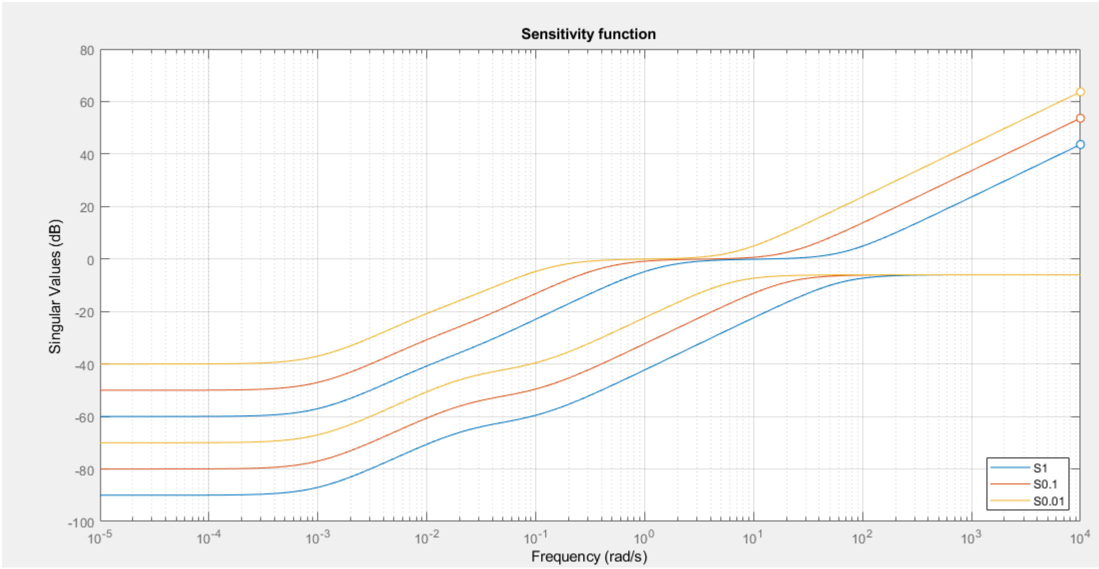



Figure 5

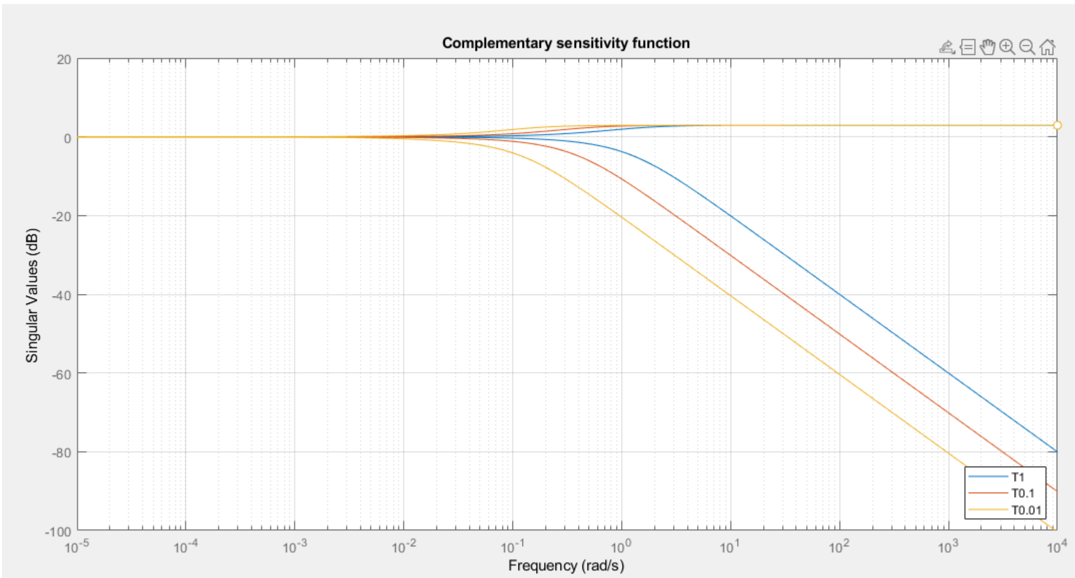
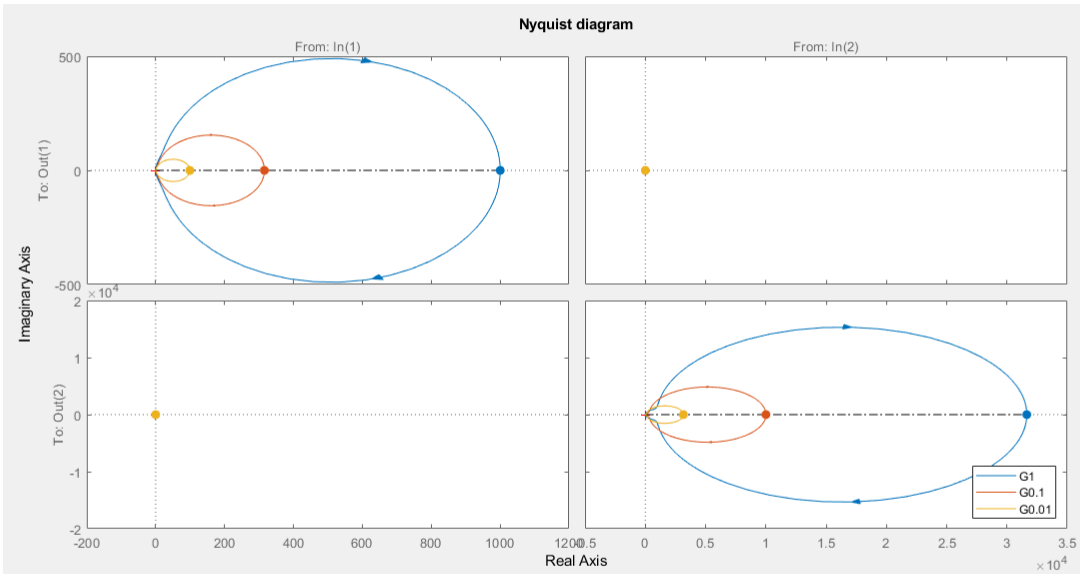



Figure 6




# 4 Discussion

1. **Closed loop step response of the standard system without decoupling compared to static and dynamic decoupling**

   **Standard System (without decoupling)**
   A coupled system is where input ouput interactions exist between a MIMO system. Such a system is difficult to control and therefore we decouple the system. By pairing at the cross over frequency we saw that the decoupling will make the MIMO system behave like two SISO systems with u1 to y1 as SISO 1 and u2 to y2 as SISO 2.

   **Steady state decoupling**
   Using $W_1$ , $W_2$ and static gain $G_0$, we check that $G_{0i}$ is an identity matrix with off diagnol elements zero. This means that we can use $W_1$ , $W_2$ and G to make a transfer function $G_{0d}$ which will decouple the system at s=0.

   **Dynamic decoupling**
   Just like steady state decoupling, we check the gain at frequency 0.1i and get an identity matrix $G_{wi}$. With this confirmation, we can make a transfer function $G_{wd}$ which will decouple the system at s=0.1i.

   We introduce a unity feedback matrix Fy and check for the stability of the standard closed loop system verses the static and dynamic decoupled systems. Figure 1 shows the systems maintain stability in both the SISO systems.

2. **Closed loop step response of the standard system compared to PI controller**
   As seen in Figure 2, the system with PI controller reaches the steady state or the final value of zero very fast.

   Remember that our standard system is a coupled system with multivariable interactions. Despite this, the integral term in the PI controller brings the error to zero. This shows that a PI controller can handle real life control problems

3. **LQG controller with feedback gain**
   Figure 3 shows an LQR design with feedback gain G1,G0.1 and G0.01 for penalty matrices $Q_2 = 1$, $Q_2 = 0.1$ and $Q_2 = 0.01$ respectively. Since $Q_2$ is the penalty on control signal, as the penalty matrix $Q_2$ becomes smaller, the control signal grows bigger. This means that the step response becomes faster. That is why G0.01 is faster than G0.1.Or, G0.1 is faster than G1.

   Figure 4 shows Sensitivity function which is small at lower frequencies. This is achieved by putting an integrator $\delta$ in the feedback loop. At frequency 0.01 rad/sec sensitivity function achieves gain of 1. Loop gain does not decrease as frequency goes to infinity.

   Figure 5 shows the complementary sensitivity function which reduces at 0.01 rad/sec, and gain approaches zero.

   Figure 6 shows the Nyquist curves are not inside the unit circle. The system has a gain margin of infinity and phase margin $90°C$.

   From figure 4,5 and 6 we can say that our LQG design with optimal feedback gives good robustness and sensitivity properties.

# 5 Appendix

```
set(0, 'defaulttextinterpreter','Latex');

s = tf('s');
G11=15/(50*s+1);
G12 = -11/(10*s+1);
```

```
G22 = 70/(10*s+1);
G21 = -24.5/(50*s+1);
Go = [G11 G12;G21 G22]; %open loop transfer function

%Pairing
Gcross = [15/6 -11/2;-24.5/6 70/2];
Rw=Gcross.*pinv(Gcross).';

%Decoupling
W2 = eye(2);
G0 = [15 -11;-24.5 70];
W10 = inv(G0);
G0i = W2*G0*W10;    %Identity matrix
G0d = W2*Go*W10;    %Steady state decoupling

Gw = [15/(5i+1) -11/(1i+1);-24.5/(5i+1) 70/(1i+1)];
Gwr = real(Gw);
W1w = inv(Gwr);
Gwi = W2*Gwr*W1w;       %Identity matrix
Gwd = W2*Go*W1w;        %Dynamic decoupling

%Closed loop step response of the standard system
   without decoupling compared to static and dynamic
   decoupling
Fy = [1 0;0 1];
Gco = Fy*Go/(eye(2)+Fy*Go);
F0 = W10*Fy;
Gc0 = F0*Go/(eye(2)+F0*Go);
Fw = W1w*Fy;
Gcw = Fw*Go/(eye(2)+Fw*Go);

figure(1);
subplot(311);step(Gco);legend('w/o decoupling');title(
   'Closed loop step response of the standard system
   compared to static and dynamic decoupling');
subplot(312);step(Gc0);legend('static decoupling');
   title('');
subplot(313);step(Gcw);legend('dynamic decoupling');
   title('');

%Closed loop step response of the standard system
   compared to PI controller
PI = (1+10*s)/10*s;            %Kp=1 and Ki=0.1
Gcopi = PI*Go/(eye(2)+PI*Go);

figure(2);
```

```
t=(0:1:100)';
step(t,Gco,Gcopi);legend('standard system','PI
    controller');title('Closed loop step response of
    the standard system compared to PI controller');

%LQG controller with feedback gain
u = [1 0;0 1];
y1 = [15/(50*s+1) -11/(10*s+1);-24.5/(50*s+1) 70/(10*s
    +1)]*u;
sys1 = ss(y1);

A = [-0.02 0 0 0;0 -0.1 0 0; 0 0 -0.001 0; 0 0 0
    -0.001];
B = [1 0;0 4;1 0; 0 1];
C = [0.3 -0.275 1 0;-0.49 1.75 0 1];
D = [0 0;0 0];
sys2 = ss(A,B,C,D);         %State space model

R1 = 1;
R2 = 0.001;
R12 = 0;
Q1 = 1;
R = [R1 R12;R12 R2];

Q2a = [1 0 0 0;0 1 0 0; 0 0 1 0; 0 0 0 1];
[Gk1,S1,T1,margin1,L1] = LQRcontrol(sys2,A,B,Q2a,R);

Q2b = [0.1 0 0 0;0 0.1 0 0; 0 0 0.1 0; 0 0 0 0.1];
[Gk2,S2,T2,margin2,L2] = LQRcontrol(sys2,A,B,Q2b,R);

Q2c = [0.01 0 0 0;0 0.01 0 0; 0 0 0.01 0; 0 0 0 0.01];
[Gk3,S3,T3,margin3,L3] = LQRcontrol(sys2,A,B,Q2c,R);

figure(3);
step(Gk1,Gk2,Gk3),legend('G1','G0.1','G0.01','Location
    ','SouthEast');title('LQR design with feedback gain
    ');

figure(4);
sigma(S1,S2,S3);legend('S1','S0.1','S0.01','Location',
    'SouthEast');title('Sensitivity function');

figure(5);
sigma(T1,T2,T3);legend('T1','T0.1','T0.01','Location',
    'SouthEast');title('Complementary sensitivity
    function');
```

```
figure(6);
nyquist(Gk1,Gk2,Gk3);legend('G1','G0.1','G0.01','
    Location','SouthEast');title('Nyquist diagram');

function [Gk,S,T,margins,L] = LQRcontrol(sys,A,B,Q,R)
[L,~,~] = lqr(sys,Q,R);
s = tf('s');
Gk = L*inv(s*eye(4)-A)*B;
margins = allmargin(Gk);
S = 1/(1+Gk);
T = Gk/(1+Gk);
end
```