Automatic control III - Homework assignment 2

Andreas Fredlund Hannes Larsson Elliot Rask

October 11, 2017

1 Problem I

a)

Since we have $W_S = \frac{1}{s}$ we put a strong limitation on the sensitivity function. The sensitivity function tells us about how errors in the model turns into an error the output signal, and with our choice of W_S we punish S hard for small frequencies which makes the error tend to zero for the static case which is what we get out of integral action.

b)

We have that

$$G(s) = \frac{-(0.1s+1)}{s^2-1} \tag{1}$$

this becomes, in controllable, canonical state space form:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \tag{2}$$

$$y(t) = \begin{bmatrix} -0.1 & -1 \end{bmatrix} x(t). \tag{3}$$

With our wanted weighting of transfer functions as

$$W_S(s) = \frac{1}{s},\ W_T(s) = 1,\ Wu(s) = 1 \tag{4}$$

we add a third state

$$x_3 = \frac{1}{s}(Gu + n) = \frac{1}{s}y \tag{5}$$

and the extended state space form becomes:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -0.1 & -1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} n(t) \tag{6}$$

$$\tilde{z}(t) = \begin{bmatrix} 0 & 0 & 0 \\ -0.1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) \tag{7}$$

$$y(t) = \begin{bmatrix} -0.1 & -1 & 0 \end{bmatrix} x(t) + n(t). \tag{8}$$

with

$$D^T \begin{bmatrix} M & D \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \\ -0.1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

which is what we want. To ensure innovation form we check if $A - NC$ is a stable matrix.

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -0.1 & -1 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} -0.1 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{10}$$

which has the eigenvalues

$$\begin{cases} \lambda_1 = -1 \\ \lambda_2 = 1 \\ \lambda_3 = 0 \end{cases} \tag{11}$$

which means that we have an eigenvalue in the right half plane and our system is not in innovation form. We will have to compute the Kalman filter for the system.

c)

We will try to determine the optimal $\mathcal{H}_2$ controller with the system listed above, using the Kalman filter.
The regulator that we are searching for can be found using:

$$u = -L\hat{x} \tag{12}$$

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{13}$$

where K is given by

$$K = (PC^T + NR_{12})R_2^{-1} \tag{14}$$

where P is the solution to the matrix equation

$$AP + PA^T - (PC^T + NR_{12})R_2^{-1}(PC^T + NR_{12})^T + NR_1N^T = 0. \tag{15}$$

L is given by

$$L = B^TS \tag{16}$$

where S is the solution to the matrix equation

$$0 = A^TS + SA + M^TM - SBB^TS \tag{17}$$

Here $A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, $N = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $C = \begin{bmatrix} -0.1 & -1 & \epsilon \end{bmatrix}$, and $R = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ where ϵ is a small constant added to move the eigenvalue in 0 in $A-NC$ away from the border. Here, we chose $\epsilon = 10^{-3}$. Solving these equations in Matlab gives us $K = \begin{bmatrix} -1.8182 \\ -1.8182 \\ 1 \end{bmatrix}$ and $L = \begin{bmatrix} 2.7349 & 3.6349 & -1 \end{bmatrix}$.

The controller is given by

$$F_y(s) = L(sI - A + BB^TS + KC)^{-1}K = \frac{-(12.58s^2 + 13.58s + 1)}{s^3 + 4.735s^2 + 8.947s} \tag{18}$$

which contains integral action.

d)

$W_S(s)S(s)$ should be large for small frequencies and small for large frequencies as we have a factor of $\frac{1}{s}$ in W_S. $W_T T(s)$ should be ~ 0 dB for small frequencies since S is small and $T = 1 - S$. For high frequencies $W_T T(s)$ should be small. $W_u G_{wu}(s)$ should have gain 1 (0 dB) for small frequencies and should supress noise on the input signal for high frequencies. The Bode diagrams for $W_S(s)S(s)$, $W_T(s)T(s)$ and $W_u G_{wu}(s)$ are presented in Figure 1, 2 and 3 which looks as expected.

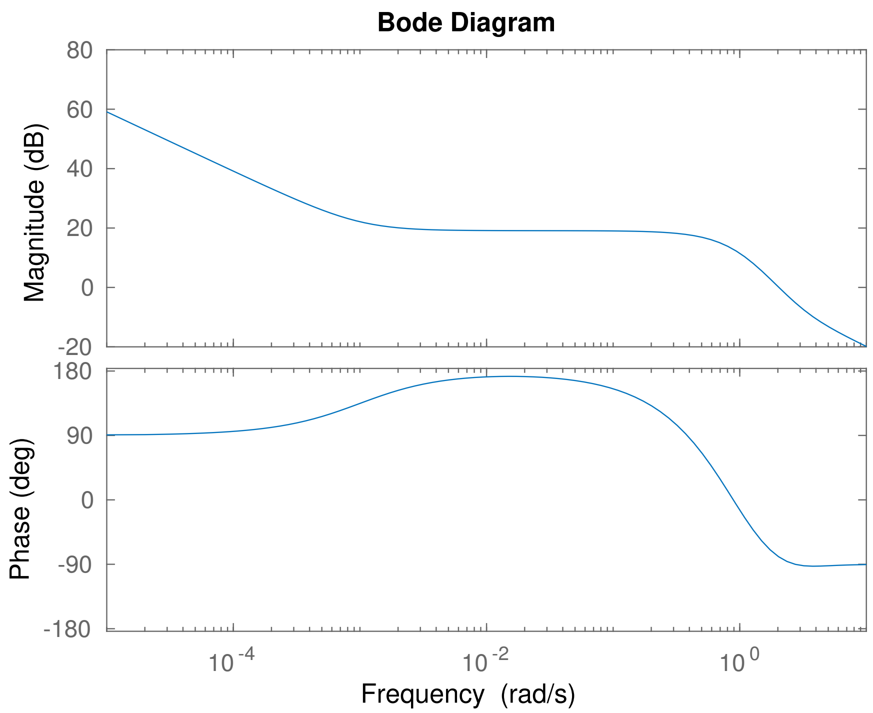

Figure 1: Bode diagram of $W_S(s)S(s)$

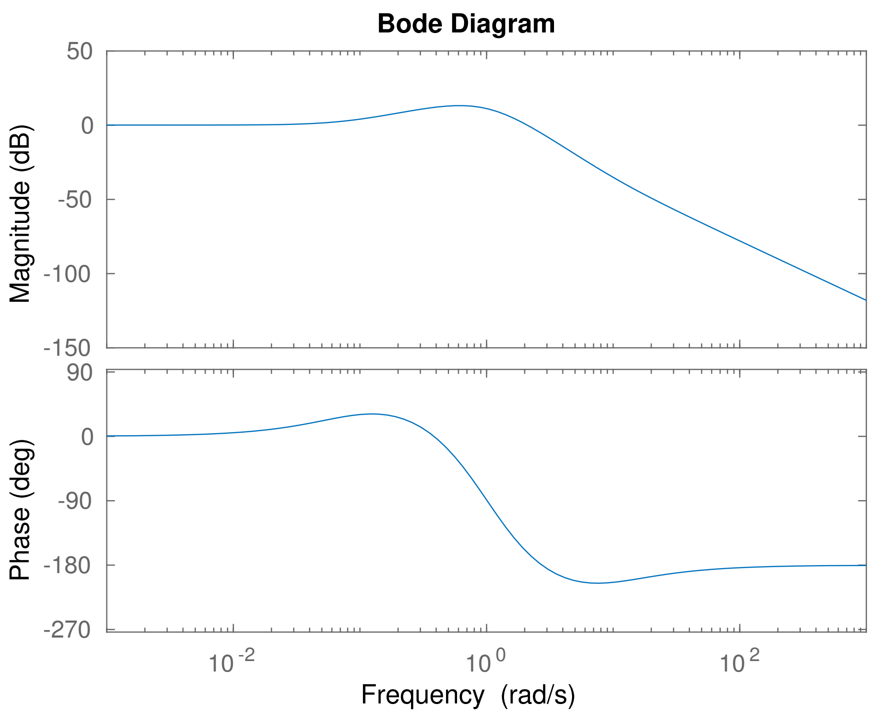

Figure 2: Bode diagram of $W_T(s)T(s)$

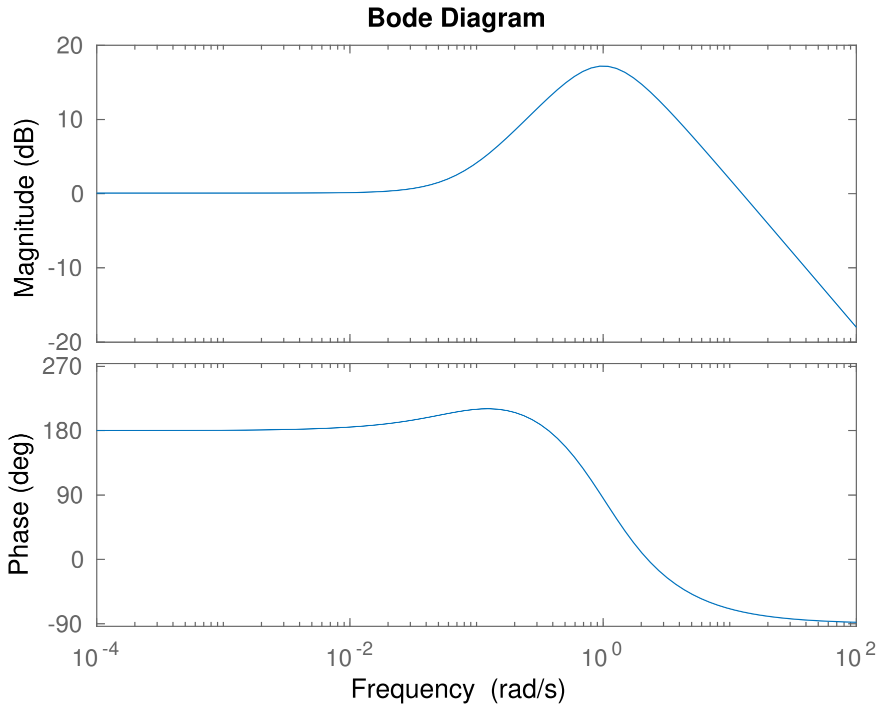

Figure 3: Bode diagram of $W_u(s)G_{wu}(s)$

e)

In figures 4 to 6 the step responses for S, T and G_{wu} can be seen. Since we used $W_S = \frac{1}{s}$ we expect S to go to zero for small s. Since S goes to zero we also expect T to go to 1 for small s which is the case. We can also compare the step responses to the bode diagrams in figures 1 to 2. Looking at the complementary sensitivity function T one can see that for time ~ 0 s T equals zero and for time $\rightarrow \infty$ $T \rightarrow 1$. This matches the bode diagram in Figure 2 where the magnitude decreases with increasing frequency and starts at 0 dB for low frequencies. We also have a phase shift of over 180 degrees which explains why the amplitude goes down and then up. Now, looking at the sensitivity function S one can see that it matches with the complementary sensitivity function T since $T + S = 1$. When time $\rightarrow \infty$ $G_{wu} \rightarrow -1$ which corresponds to low frequencies where the magnitude is 0 dB and the phase is $180°$. When time ~ 0 s G_{wu} equals zero which corresponds to decreasing magnitude with increasing frequency which is the case in Figure 3.

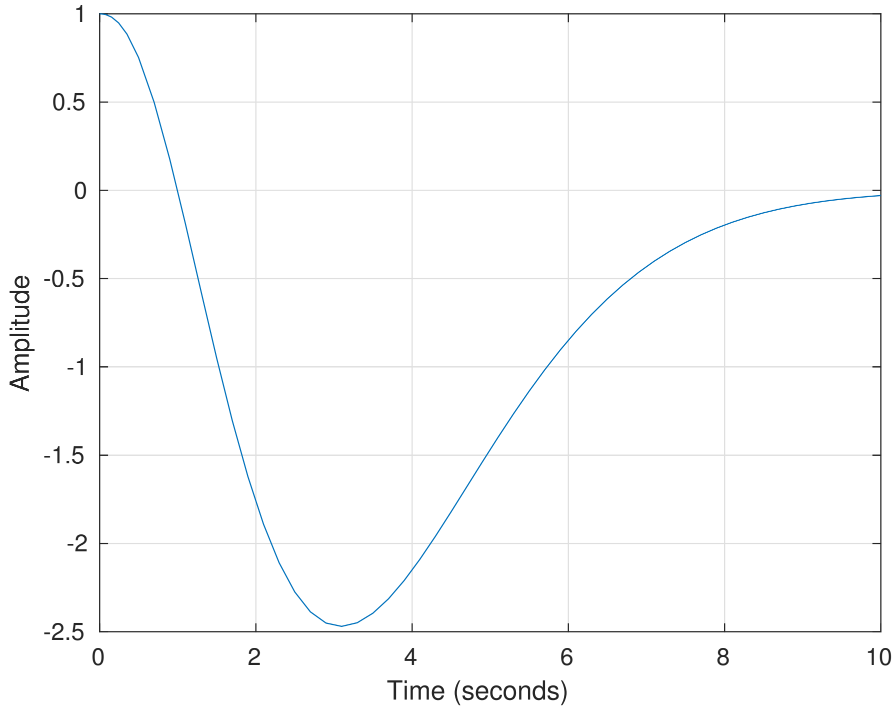

Figure 4: Step response for the sensitivity function S.

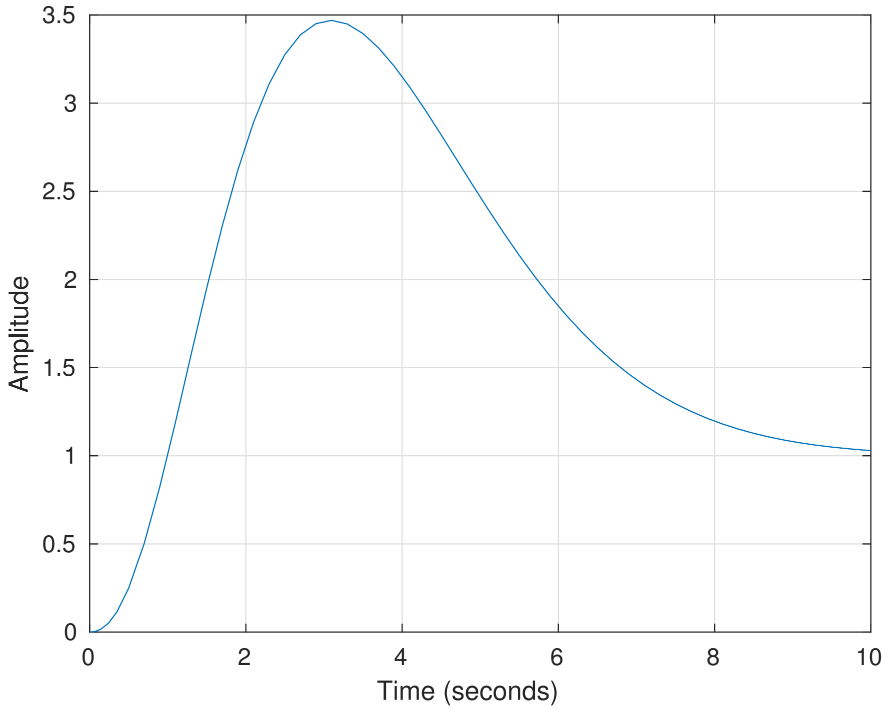

Figure 5: Step response for the complementary sensitivity function T.

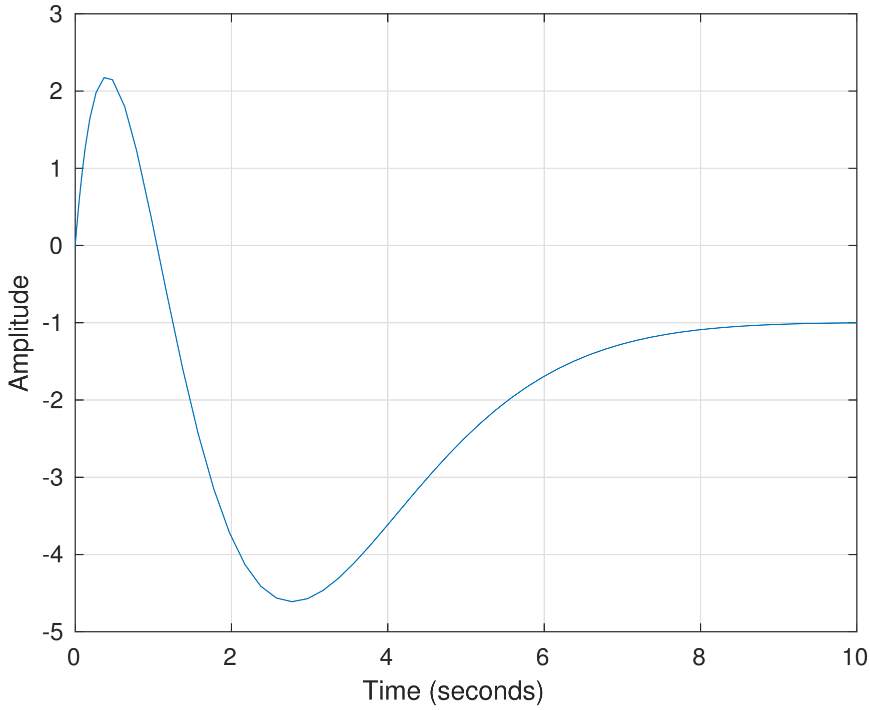

Figure 6: Step response for the transfer function from the noise to the input signal G_{wu}.

2 Problem II

a)

The system

$$\dot{x}(t) = rx(t), \tag{19}$$

or

$$\dot{x}(t) = f(x,t), \tag{20}$$

where $f(x,t) = rx(t)$, $r > 0, x(t) \in \mathbb{R}$ *and* $t \in \mathbb{R}_+$, is a linear system.

Proof.

$$\begin{aligned} f(x_1 + x_2, t) &= f(x_1, t) + f(x_2, t) \\ \alpha f(x,t) &= f(\alpha x, t). \end{aligned}$$

□

The actual calculations for this system are so simple that they are omitted here. This is a first order system since we only have the first derivative.

b)

The equilibrias of the system are given by

$$\dot{x}(t) = 0 \Rightarrow rx(t) = 0 \Rightarrow x(t) = 0.$$

Since the system is linear, it is stable if the eigenvalues lie in the left half plane, here our eigenvalue is $r > 0$ which lies in the right half plane $\Rightarrow$ instability.

c)

The solution for our system with $x(0) = x_0$ is

$$x(t) = x_0 e^{rt}$$

so

$$\lim_{t\to\infty} x(t) = \infty$$

d)

We now have the model

$$\dot{x}(t) = rx(t)\big(1 - \frac{x(t)}{K}\big) = rx(t) - r\frac{x^2(t)}{K},$$

where $K > 0$ and everything else is as in the previous model.
The system is not linear.

Proof.

$$f(x_1 + x_2, t) = rx_1 + rx_2 + \frac{x_1^2}{K} + \frac{x_2^2}{K} + 2\frac{x_1 x_2}{K}$$
$$f(x_1, t) + f(x_2, t) = rx_1 + rx_2 + \frac{x_1^2}{K} + \frac{x_2^2}{K}$$

$$\therefore f(x_1 + x_2, t) \neq f(x_1, t) + f(x_2, t) \Rightarrow \text{the system is not linear.}$$

□

The system is of order 1 since we only have one derivative.

e)

The equilibriums are given by

$$x^2(t) - Kx(t) = 0 \Rightarrow \begin{cases} x_1 = 0 \\ x_2 = K \end{cases}$$

We linearise this by taylor expanding $f(x,t) = rx(t) - r\frac{x^2(t)}{K}$.
The taylor expansion around a stationary point x^* is

$$f(x^* + \epsilon) = f'(x^*)\epsilon + \mathcal{O}(\epsilon^2)$$

so we only need to look at the derivative of f at x_1 and x_2.

$$f'(x) = r - 2r\frac{x}{K}$$

so

$$\begin{aligned} f'(x_1) = f'(0) &= r > 0 \\ f'(x_2) = f'(K) &= r - 2r = -r < 0, \end{aligned}$$

So we see that the $x_1 = 0$ is unstable and $x_2 = K$ is stable.

f)

If

$$x(t) = \frac{Kx_0e^{rt}}{K + x_0(e^{rt} - 1)} = Kx_0e^{rt}(K + x_0(e^{rt} - 1))^{-1}$$

then $\dot{x} = f(x,t)$.

Proof.

$$\begin{aligned} \dot{x} = rKx_0e^{rt}(K + x_0(e^{rt} - 1)^{-1} - Kx_0e^{rt}(K + x_0(e^{rt} - 1))^{-2}x_0re^{rt} = \\ = \frac{rKx_0e^{rt}}{K + x_0(e^{rt} - 1)} - \frac{Kx_0^2re^{2rt}}{(K + x_0(e^{rt} - 1))^2} \\ f(x,t) = \frac{rKx_0e^{rt}}{K^2 + x_0(e^{rt} - 1)} - \frac{K^2x_0^2re^{2rt}}{K(K + x_0(e^{rt} - 1))^2} \Rightarrow \\ \dot{x} = f(x,t) \end{aligned}$$

□

We can also see that

$$\lim_{t\to\infty} x(t) = K,$$

so x goes to out stable stationary point as the time goes to infinity. In c) the number of people just grew larger and larger and diverged, this model is more realistic where there is a maximum number of people.

g)

Simulation of the system for different values of the parameters are presented in Figure 7, 8 and 9.

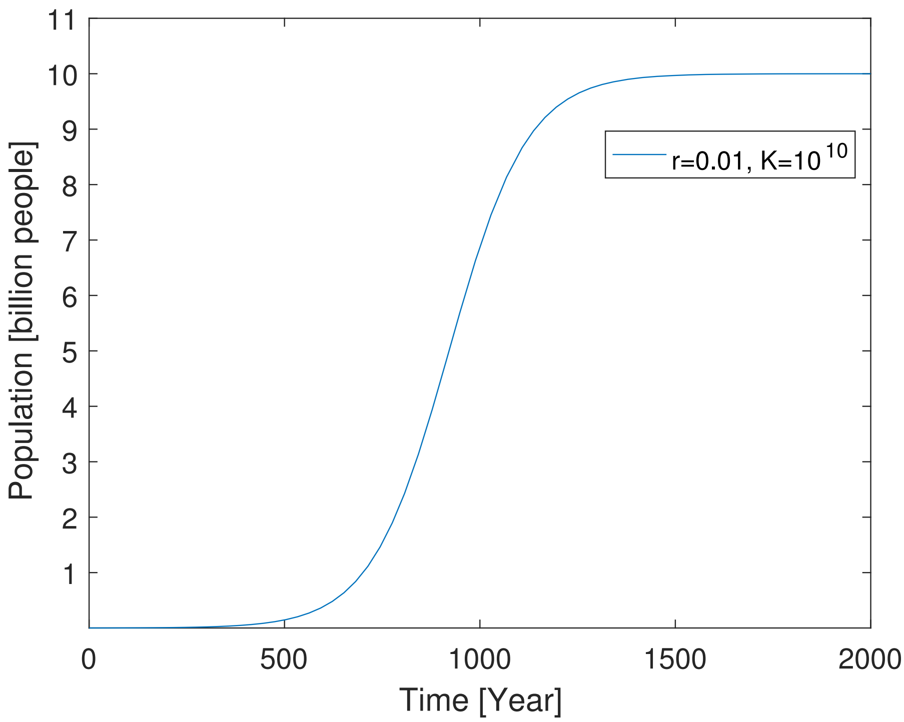

Figure 7: Population starting at 10^6 with r=0.01 and $K = 10^{10}$

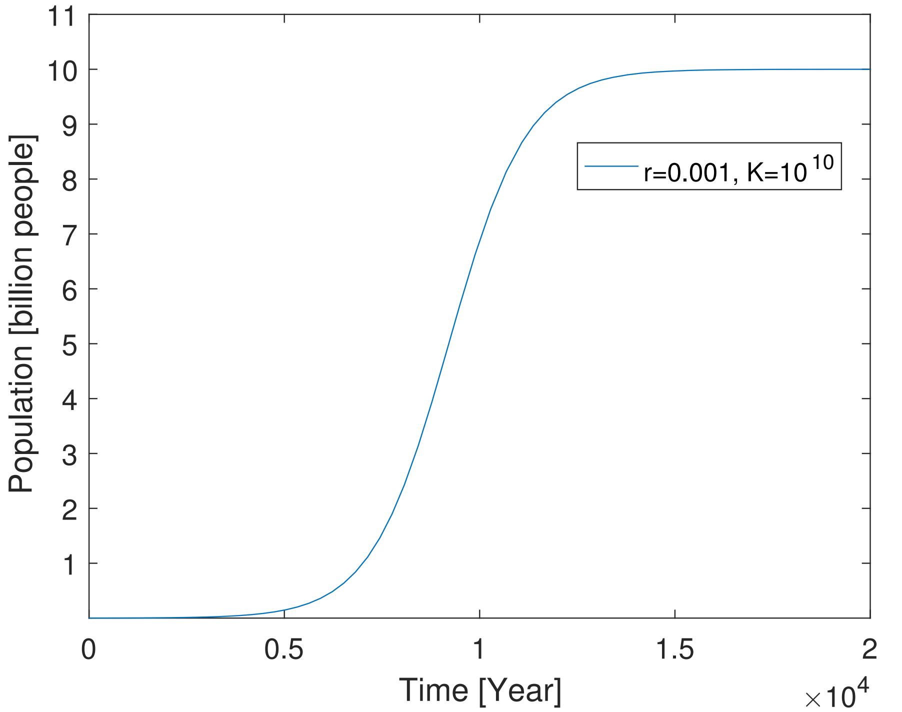

Figure 8: Population starting at 10^6 with r=0.001 and $K = 10^{10}$

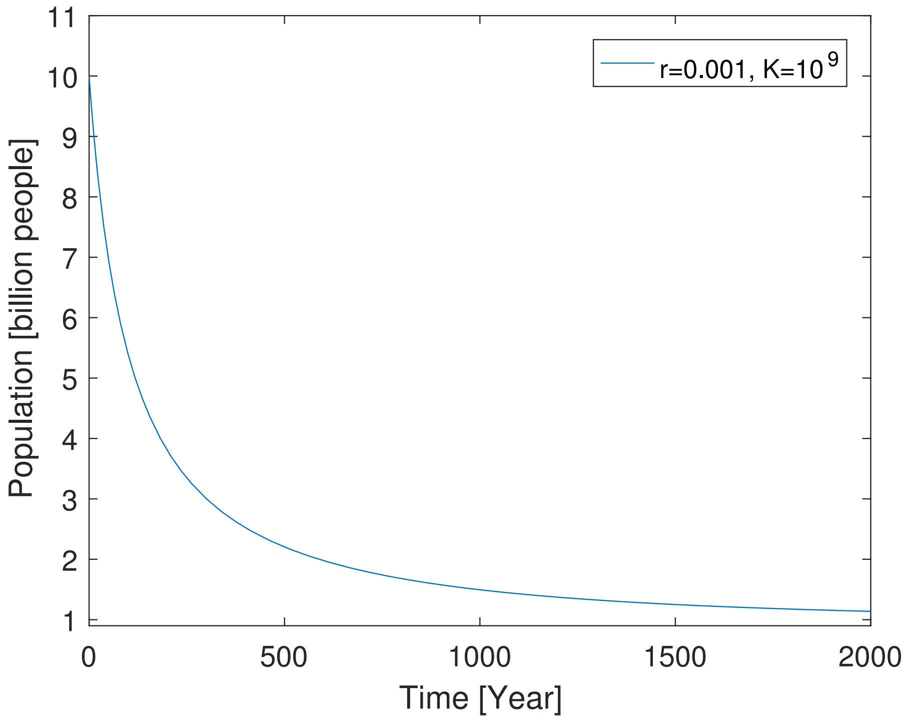

Figure 9: Population starting at 10^{10} with r=0.01 and $K = 10^9$

3 Problem III

a)

The order of the regulator is equal to the extended system's, seen in the block diagram for the task, order, which in this case is: $9+1+1+5+2+0+7 = 25$. Not using model reduction, the model's order is 30 and the order becomes: $30+5+2+0+7 = 46$.

b)

Since all the elements in the steady-state RGA matrix are positive and the $_{12}$ and $_{21}$ elements in the RGA matrix for the crossover-region are close to 1 this appear to be suitable for a decentralized controller and we choose to pair $y_1 \rightarrow u_2$ and $y_2 \rightarrow u_1$.

c)

Pairing the signals looking at the RGA for the crossover frequency, which is quite near the identity matrix, would lead to negative values in the steady state RGA matrix. This means that the pair is not suitable for a decentralized controller.
Static decoupling can be used if $G(0)$ is invertible.

A Appendix

```
%homework2
clear all
close all
clc

s = tf('s');
G = -(0.1*s+1)/(s*s-1);

A = [0 1 0; 1 0 0; -0.1 -1 0];
B = [1 0 0]';
N = [0 0 1]';
C = [-0.1 -1 1e-3];
Creal = [-0.1 -1 0];
R = [1 1; 1 1];
M = [0 0 0; -.1 -1 0; 0 0 1];

matris = A-N*C;

eig(matris)

%kalman
P = care(A', C', N*N', 1, N);
K = (P*C'+N); %R12 = R2 = 1

%LQG (or H2 here)
S = care(A,B,M'*M);
L = B'*S;

%create our Fy
system = ss(A-K*Creal-B*L,K, -L,0);
Fy = -tf(system);

%plot stuff
Ws = 1/s;
Wt = 1;
Wu = 1;
Ss = inv(1 + G*Fy);
T = 1 - Ss;
Gwu = -Fy*Ss;

bode(Ws*Ss);
figure
bode(Wt*T);
figure
```

```
bode(Wu*Gwu);
```