Automatic Control III, Homework 1

Jonathan Sundell, Philip Ahl, Felix Elmgren, Carl Tysk
Uppsala University
Ångströmslaboratoriet, Uppsala

October 15, 2019

Problem 1, Distillation Column

a)

To find the poles and zeros analytically we start with calculating the minors:

(1x1): $\frac{13}{17s+1}, \frac{-19}{21s+1}, \frac{4}{15s+1}, \frac{7}{12s+1}, \frac{-19}{15s+1}, \frac{5}{12s+1}$

(2x2): $\frac{-255(21s+1)(12s+1)+133(17s+1)(15s+1)}{(17s+1)(15s+1)(21s+1)(12s+1)}, \frac{65(15s+1)-28(17s+1)}{(15s+1)(12s+1)(17s+1)}, \frac{-95(15s+1)^2+76(21s+1)(12s+1)}{(21s+1)(12s+1)(15s+1)^2}$

From this we see that the least common denominator is $(15s+1)^2(12s+1)(17s+1)(21s+1)$ and the poles will be $-\frac{1}{17}, -\frac{1}{15}, -\frac{1}{15}, -\frac{1}{12}, -\frac{1}{21}$. To calculate the same thing in Matlab we use the command *pole()* and get the poles $-\frac{1}{17}, -\frac{1}{15}, -\frac{1}{15}, -\frac{1}{12}, -\frac{1}{12}, -\frac{1}{21}$. The difference here is that Matlab gets an extra pole in $-\frac{1}{12}$ which we assume is because Matlab don't take the least common denominator, but instead add an extra pole and along with that an extra zero (that we can cancel analytically).

Regarding the zeros we don't have any common divisor for the numerators of the largest minors, meaning there is no zeros for the system. The same result was found in Matlab with the command *zeros()*.

b)

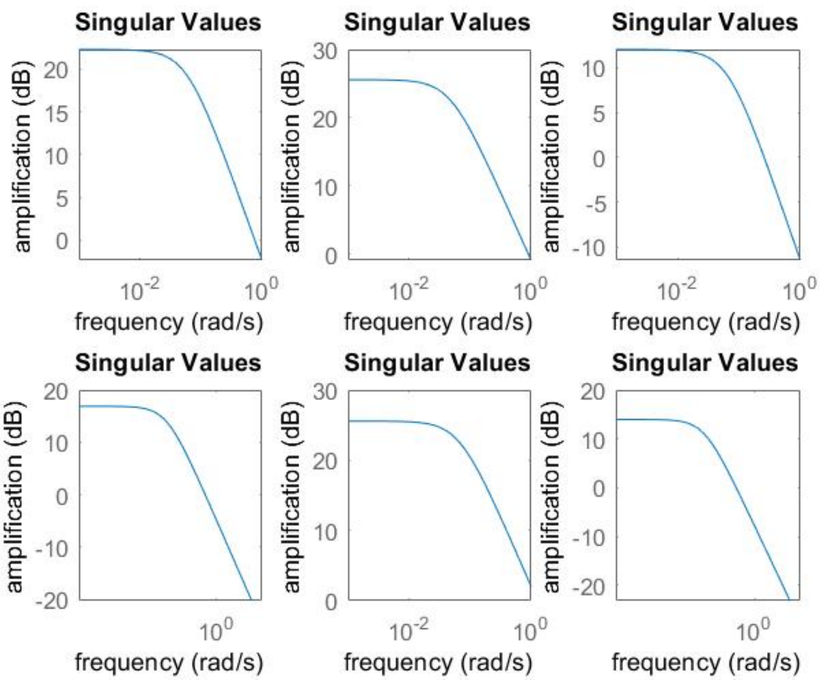

Figure 1: *Amplification of individual elements of G.*

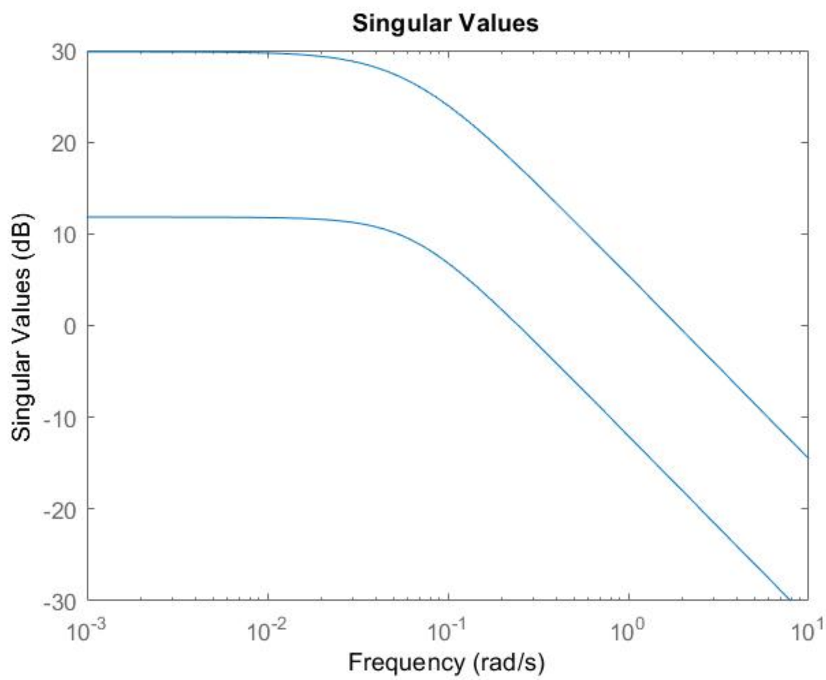

Figure 2: *Amplification of the MIMO system G.*

As seen in Figure 2 30dB is the largest gain for the MIMO system. None of the individual elements reaches this gain. This is assumed to be because of the individual elements may interact with each other for certain frequencies.

Problem 2, Small gain theorem

a)

Find $||f(y)||$:

$$||f(y)||^2 = sup\frac{||f(y)||_2^2}{||y(t)||_2^2} \leq \frac{\int_{-\infty}^{\infty} \frac{1}{9}|y(t)|^2}{\int_{-\infty}^{\infty} |y(t)|^2} dt = \frac{1}{9} \rightarrow ||f(y)|| = \frac{1}{3}. \tag{1}$$

We used the following trick: $|f(y)|^2 \leq |\frac{1}{3}y(t)|^2$.
$G(s) = \frac{s+3}{(s+1)(s+2)}$, Find $||G||$:

$$||G|| = sup|G| = \frac{3}{2} \tag{2}$$

Small gain theorem:

$$||G|| * ||f(y)|| \rightarrow \frac{1}{3} * \frac{3}{2} = \frac{1}{2} \leq 1. \tag{3}$$

Hence the stability of the closed loop system is guaranteed.

b)

No since G(s) has a pole in 0. The real part of the pole has to be negative in order for the system to be stable. The small gain theorem is only applicable when all systems are stable.

Problem 3, RGA and IMC for a heating system

a)

The relative gate array, RGA, matrix is found in Matlab with the equation

$$R = G * (G^{-1})^T \tag{4}$$

At frequency zero, the elements of the RGA matrix are given as

$$R(w = 0) = \begin{bmatrix} 1.1905 & -0.1905 \\ -0.1905 & 1.1905 \end{bmatrix} \tag{5}$$

which suggests there is a cross coupling between T1 and U2 and T2 and U1. The coupling is negative which can lead to instability.

$$R(w = w_c) = \begin{bmatrix} 1.0000 & -2.4876e-5 \\ -2.4876e-5 & 1.0000 \end{bmatrix} \tag{6}$$

Eq.6 shows the RGA matrix evaluated at the cutoff frequency which is $\omega_c = 0.395[rad/s]$, where the diagonal elements are 1, which is desired. The terms in the static RGA matrix shows a strong diagonal connection, hence between T_1 and U_1 and vice versa. Our aim is eliminate this cross coupling so that we can control T_1 with U_1 and T_2 with U_2. Due to the cross coupled terms we cannot use a decentralized controller. Instead we will introduce a decoupled controller. We make the variable change

$$\tilde{y} = W_2 * y, \tilde{u} = W_1^1 * u \tag{7}$$

such that the transfer function from $\tilde{u}$ to $\tilde{y}$, which is

$$\tilde{G} = W_2(s)G(s)W_1(s) \tag{8}$$

becomes as diagonal as possible. With the new signals one can now make a decentralized, hence diagonal, controller. Expressed in its original variables the controller becomes

$$u = -W_1 F_y^{diag} W_2 y \tag{9}$$

which corresponds to the desired decoupled controller. F_y^{diag} is a decentralized controller, if one would neglect the cross coupled terms. This could for example be a PID-controller. Choosing the W matrices is not trivial, the true diagonal system $\tilde{G}$ demands complex valued W matrices, which isn't practically feasible. Hence, a stationary approximation is done at $s = 0$ to get constant real-valued W matrices. With

$$G(0) = \begin{bmatrix} 12.5 & 5 \\ 5 & 12.5 \end{bmatrix} \tag{10}$$

and choosing $W_1 = G(0)^-1$ and $W_2 = I$ gives

$$\tilde{G}(s) = \begin{bmatrix} \frac{0.4762s^3+0.04962s^2+0.001152s+4e-06}{s^4+0.2s^3+0.014s^2+0.0004s+4e-06} & \frac{-0.1905s^3-0.01905s^2-0.000381s}{s^4+0.2s^3+0.014s^2+0.0004s+4e-06} \\ \frac{-0.1905s^3-0.01905s^2-0.000381s-4.235e-22}{s^4+0.2s^3+0.014s^2+0.0004s+4e-06} & \frac{0.4762s^3+0.04962s^2+0.001152s+4e-06}{s^4+0.2s^3+0.014s^2+0.0004s+4e-06} \end{bmatrix} \tag{11}$$

where we are not interested in the cross terms, which will be 0 at frequency 0. The final controller becomes

$$F_y(s) = W_1 F_y^{diag}(s) \tag{12}$$

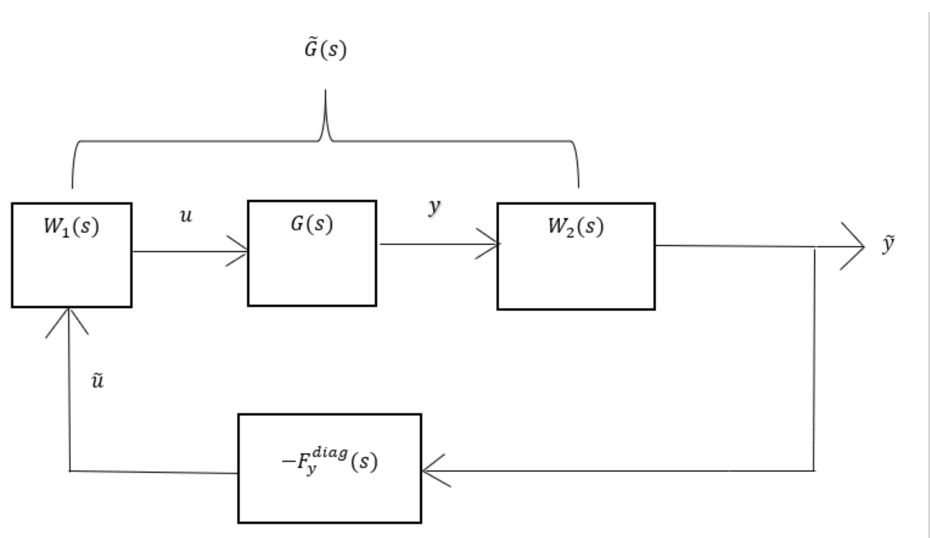

Figure 3: *Block diagram of suggested decoupled controller.*

b)

The transfer function for the heating system is:

$$G(s) = \begin{bmatrix} \frac{5s+0.025}{s^2+0.1s+0.02} & \frac{10^{-2}}{s^2+0.1s+0.02} \\ \frac{10^{-2}}{s^2+0.1s+0.02} & \frac{5s+0.025}{s^2+0.1s+0.02} \end{bmatrix} \tag{13}$$

and since G has more poles than zeros, the inverse of G cannot be realized. To fix this we need to add a factor of $\frac{1}{\lambda s+1}$ to be able to form:

$$Q(s) = \frac{1}{\lambda s + 1} G^{-1} \tag{14}$$

To verify that the model is correct we need to look at the static gain for our closed loop system. The closed loop system is written as:

$$G_c(s) = G(s)Q(s) \tag{15}$$

The static gain was verified to be I using the Matlab command *evalfr()*. To analyze the system further Simulink was used:

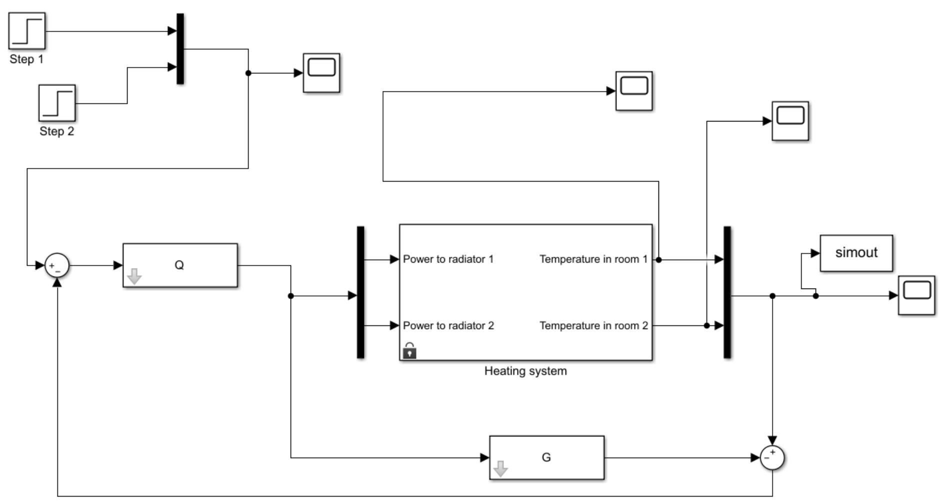

Figure 4: *Simulink setup.*

The goal was to obtain a rise time of 10 ± 2 minutes for a step from 20 to 23, when the reference signal for the other room is kept constant at 20, for each of the rooms. To achieve this λ had to be decided. After some testing the final value of λ was set to be 300. This resulted in a rise time of 600s (10min) for the first room and 662s (11min) for the second room.

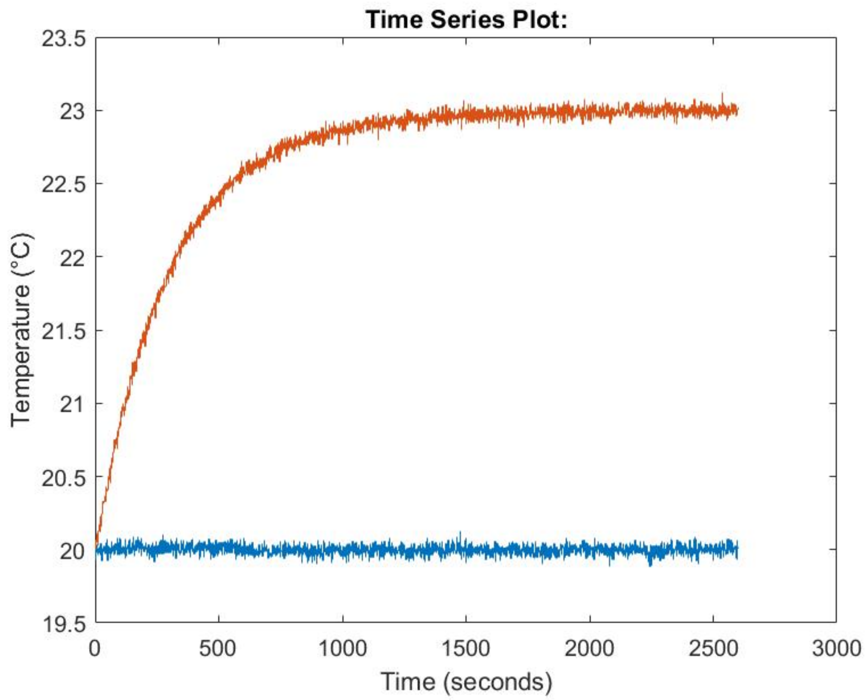

Figure 5: *Temperature of the 2 rooms when the first room is exposed to a step from 20 to 23 while the second rooms reference is kept constant at 20. The rise time for the first room is 600s.*

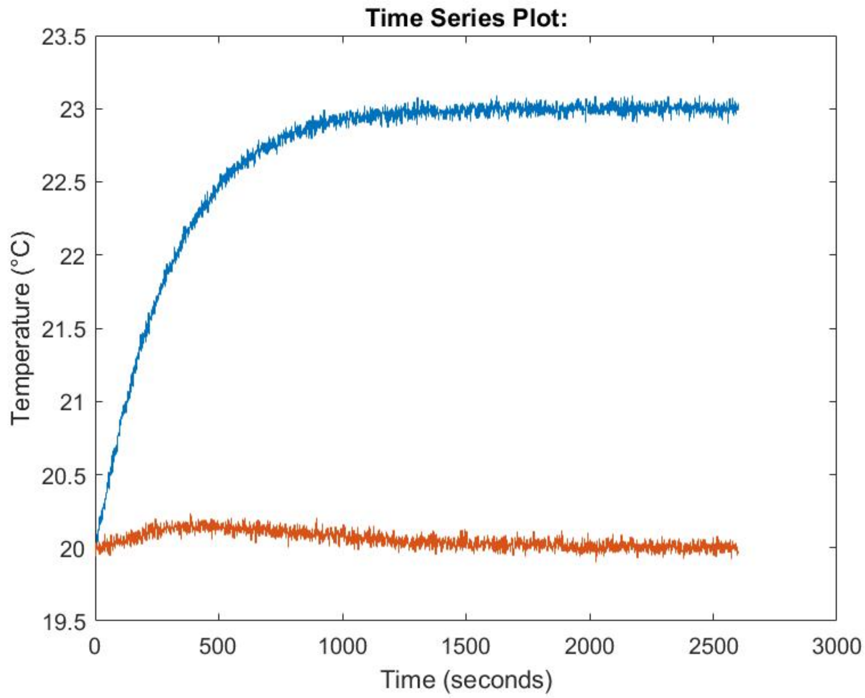

Figure 6: *Temperature of the 2 rooms when the second room is exposed to a step from 20 to 23 while the first rooms reference is kept constant at 20. The rise time for the second room is 662s.*

Problem 4

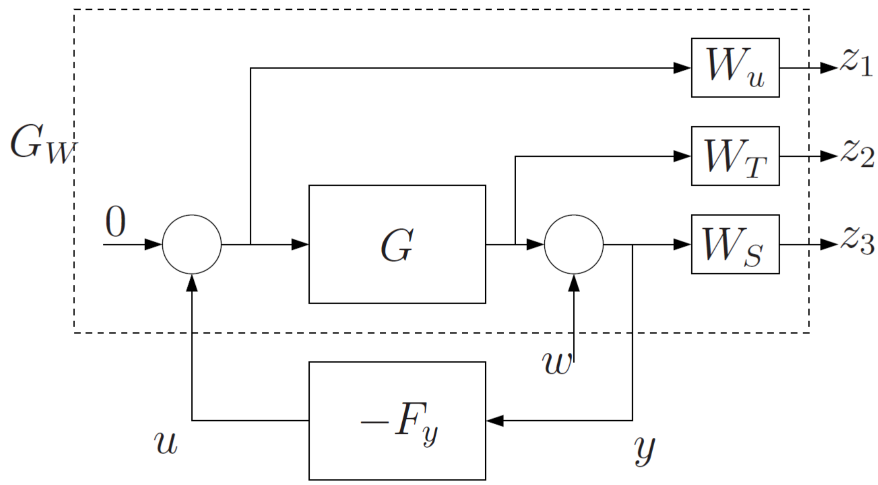

Figure 7: *Block diagram of the system that is to be analyzed*

Figure 7 shows the block diagram of the system. The aim is to find the optimal H_2 controller.

a) Controllable canonical form

From equation 2.17 in Automation control by Glad and Ljung, the controllable canonical form becomes

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \tag{16}$$

$$y(t) = \begin{bmatrix} -0.1 & -1 \end{bmatrix} x(t) \tag{17}$$

The weighting transfer functions are used when the model is extended. To be able to write the system on extended form we need to introduce a new state:

$$x_3 = z_3 = \frac{1}{s + 0.01}(Gu + w), \tag{18}$$

which leads to:

$$\dot{x}_3 + 0.01 x_3 = Gu + w \tag{19}$$

The extended model will then be:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -0.1 & -1 & -0.01 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \tag{20}$$

$$z(t) = \begin{bmatrix} 0 & 0 & 0 \\ -0.1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) \tag{21}$$

$$y(t) = \begin{bmatrix} -0.1 & -1 & 0 \end{bmatrix} x(t) + w \tag{22}$$

We make a technical assumption that:

$$D^T[MD] == \begin{bmatrix} 0 & I \end{bmatrix} \tag{23}$$

We control if the condition is fulfilled:

$$D^T[MD] = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \\ -0.1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

We also need to check that the system is written on innovation form. The system's stability will determine whether a Kalman Filter has to be used or not. If all poles have a negative real part the system is stable.

$$A - KC = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -0.1 & -1 & -0.01 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -0.1 & -2 & -0.01 \end{bmatrix}. \tag{25}$$

Thus the eigenvalues, λ, are given by

$$det(\lambda I - (A - KC)) = det \begin{bmatrix} \lambda & 1 & 0 \\ 1 & \lambda & 0 \\ 0.1 & 2 & \lambda + 0.01 \end{bmatrix} \tag{26}$$

which gives poles in 1, -1 and -0.01. The system is not stable and therefore not on innovation form. To be able to use H_2 regulator a Kalman filter has to be implemented.
The Kalman filter can be calculated using the algebraic Ricatti equation, equation 5.79 in Automation Control by Glad and Ljung

$$AP + PA^t - (PC^T + NR_{12})R_2^{-1}(PC^T + NR_{12})^T + NR_1N^T = 0 \tag{27}$$

$$\rightarrow AP + PA^T - \frac{(PC^T + NR_{12})}{R_2}(PC^T + NR_{12})^T + NR_1N^T = 0 \tag{28}$$

where the covariances are put to 1 because there is only one noise, w, which correlates with itself as the variance of the white noise which we assumed to be 1 for simplicity.

$$\rightarrow AP + PA^T - (PC^T + N) * (PC^T + N)^T + NN^T = 0 \tag{29}$$

Solving this equation in Matlab gives us:

$$P = \begin{bmatrix} 1.6529 & 1.6529 & 0 \\ 1.6529 & 1.6529 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{30}$$

The Kalman gain matrix, K, is then given by

$$K = (PC^T + NR_{12})R_2^{-1} \tag{31}$$

which gives

$$K = \begin{bmatrix} -1.8182 \\ -1.8182 \\ 1 \end{bmatrix} \tag{32}$$

$$\rightarrow \dot{x}(t) = Ax(t) + Bu(t) + K(y(t) - Cx(t)) \tag{33}$$

b) Optimal H_2 controller

The optimal H_2 controller is then given by:

$$F_y(s) = L(sI - A + BB^TS + KC)^{-1}K \tag{34}$$

where

$$L = B^TS \tag{35}$$

we know S solves the following equation

$$0 = A^TS + SA + M^TM - SBB^TS. \tag{36}$$

L then becomes $[2.7250 \quad 3.6105 \quad -0.9740]$.

Solving this in Matlab gives us the controller:

$$F_y(s) = \frac{-12.49s^6 - 131.6s^5 - 630.9s^4 - 1607s^3 - 2709s^2 - 1170s - 86.44}{s^7 + 14.19s^6 + 93.84s^5 + 359s^4 + 838.3s^3 + 1133s^2 + 718.3s + 7.07} \tag{37}$$

When analyzing this controller it could be seen that many poles and zeros lied on top of eachother in the complex plane. This means we don't have a minimal realisation of the system and that we can cancel out multiple poles and zeros. When trying to do this is Matlab the function *minreal()* was used with Matlabs built in tolerance. This didn't improve the high order of the controller so the same procedure was made but now with our own tolerance, set to $1.1 * 10^{-5}$. This made poles cancel out zeros and we got:

$$F_y(s) = \frac{-12.49s^2 - 13.58s - 1.089}{s^3 + 4.735s^2 + 8.956s + 0.08909} \tag{38}$$

c) Bode diagram

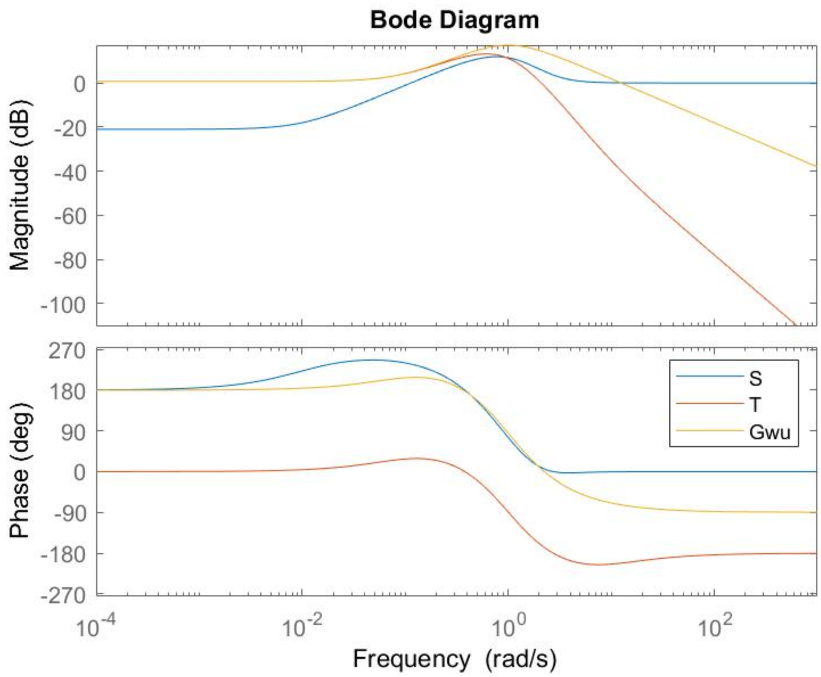

Figure 8: *Bode diagram of the sensitivity function S(s), complementary sensitivity function T(s) and the gain from w to u,* $G_{wu}(s)$.

The sensitivity function S(s) corresponds to process disturbances. This gain is low for low frequencies which the most process disturbances are, hence it looks as expected. The complementary sensitivity function T(s) corresponds to the noise in our system. Since noise often are high frequent we want a low gain for high frequencies. This is also what we see in our case. For $G_{wu}(s)$ we see that the static gain is zero.

d) Sinus waves simulation

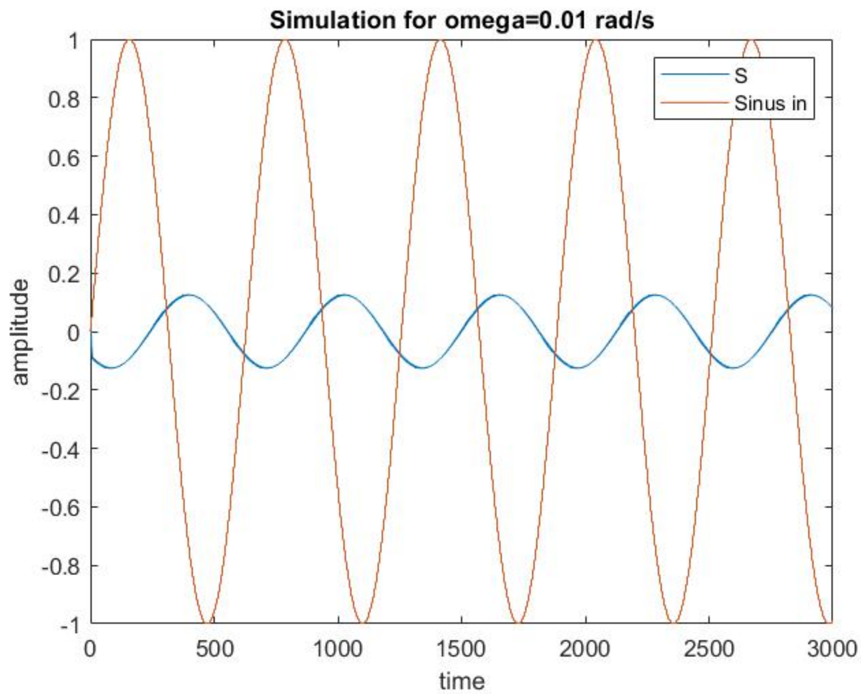

Figure 9: *S(s) plotted together with the sinus signal.*

Figure 9 visualizes that the signal S is damped as it should be when considering how the bode-plot looks. It is also slightly shifted as the phase diagram also shows.

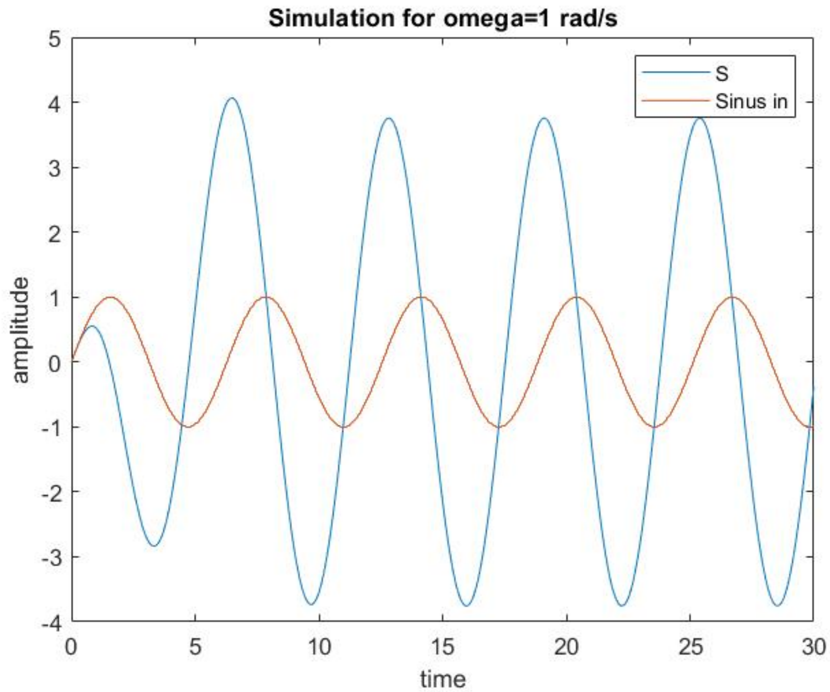

Figure 10: *S(s) plotted together with the sinus signal*

As seen in figure Figure 10 the sensitivity function S is amplified and phase shifted as one would have guessed looking at the bode-plot, Figure 8.

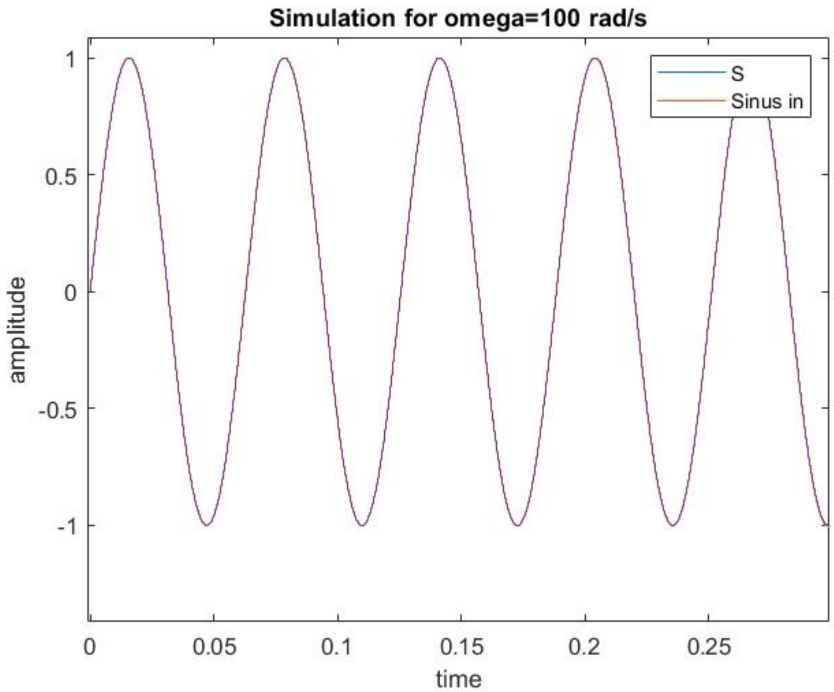

Figure 11: *S(s) plotted together with the sinus signal*

According to the bode plot, Figure 8, the out signal S should have amplitude 1 and phase 0 for large frequencies, which is exactly what can be seen in Figure 11.

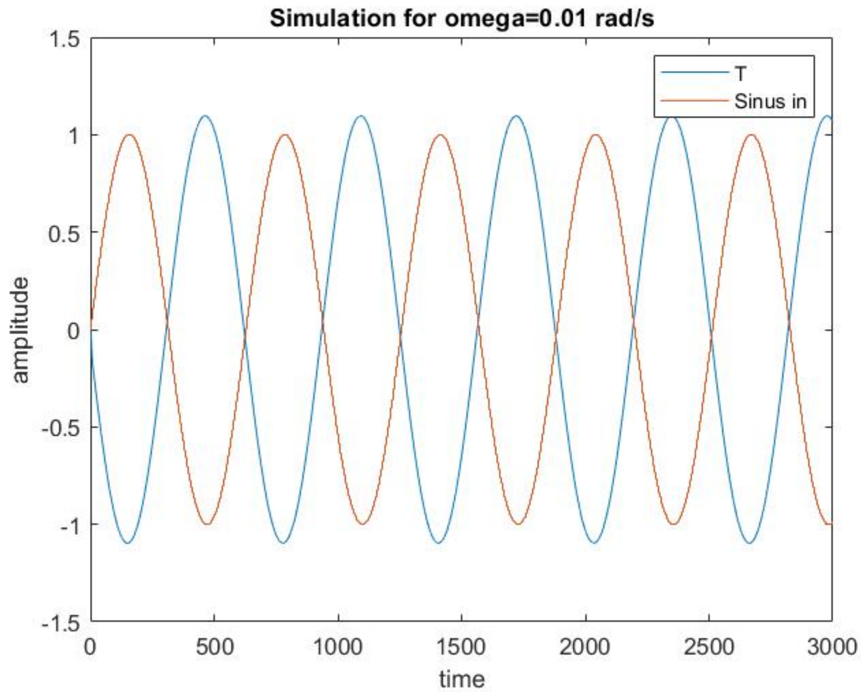

Figure 12: *T(s) plotted together with the sinus signal*

Figure 12 is consistent with the bode plot because the amplitude is the same for the complementary sensitivity function T and the sinus signal, the phase is also correctly shifted.

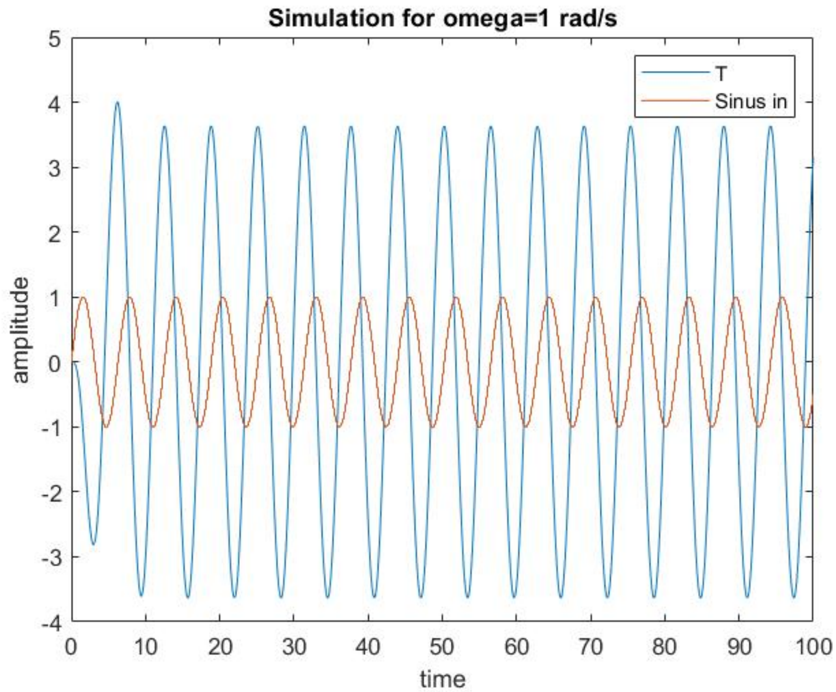

Figure 13: *T(s) plotted together with the sinus signal*

The amplitude and phase of the out signal T that is seen in Figure 13 is in line with what is shown in the bode-plot.

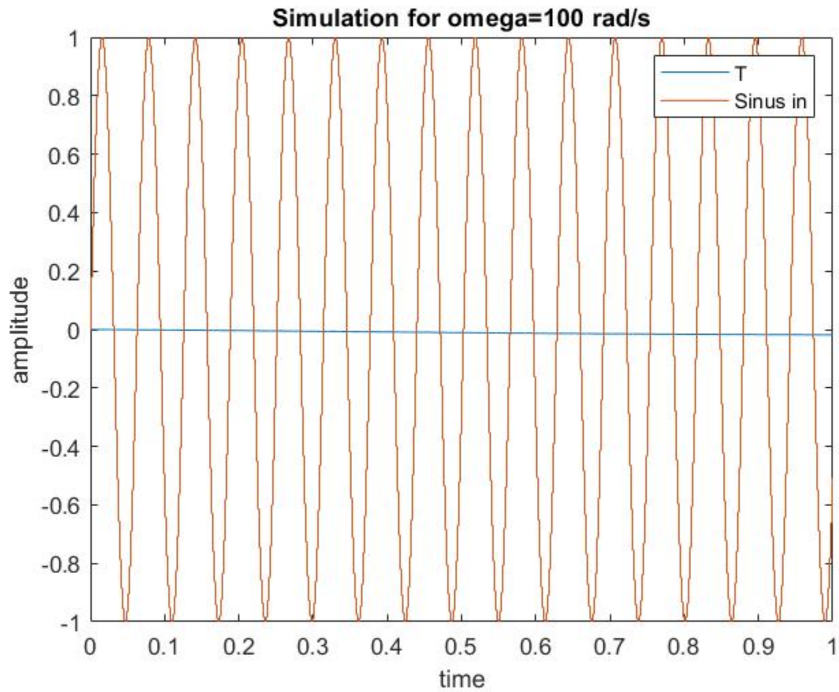

Figure 14: *TT(s) plotted together with the sinus signal*

The out signal in Figure 14 has correct amplitude when compared to the Figure 8. In the bode diagram it is seen that the T signal should be strongly damped for larger frequencies which is precisely what is illustrated in Figure 14.

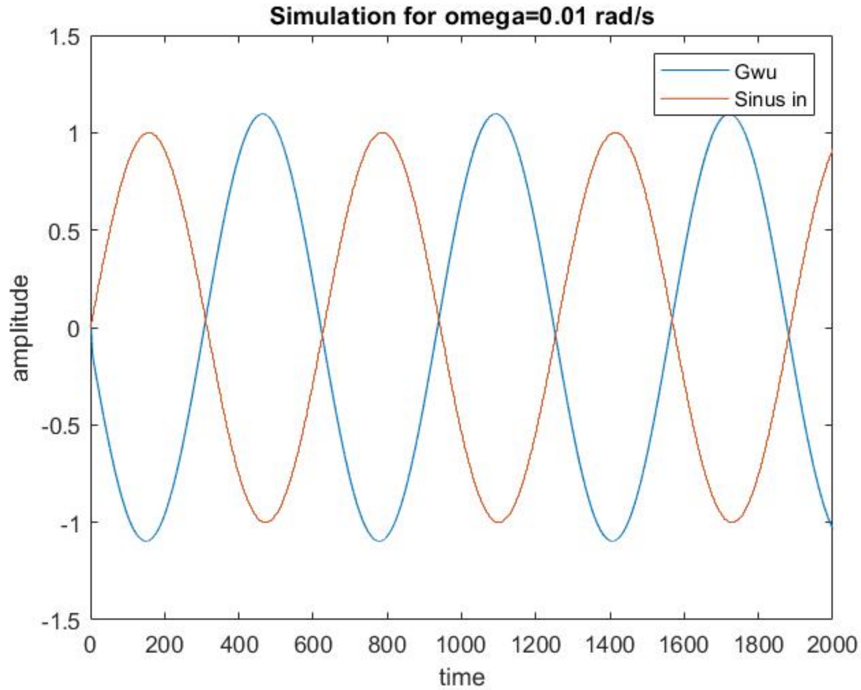

Figure 15: $G_{wu}(s)$ *plotted together with the sinus signal*

The amplitude and phase of the out signal Gwu for small frequencies is consistent with the Figure 8

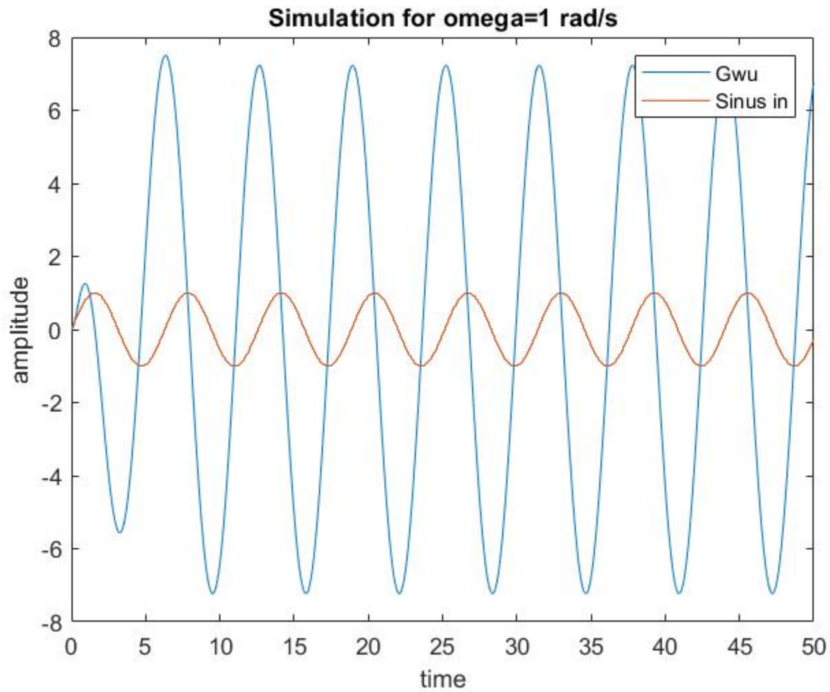

Figure 16: $G_{wu}(s)$ *plotted together with the sinus signal*

As for all the previous figures the amplitude and phase is in line with what can be seen in Figure 8

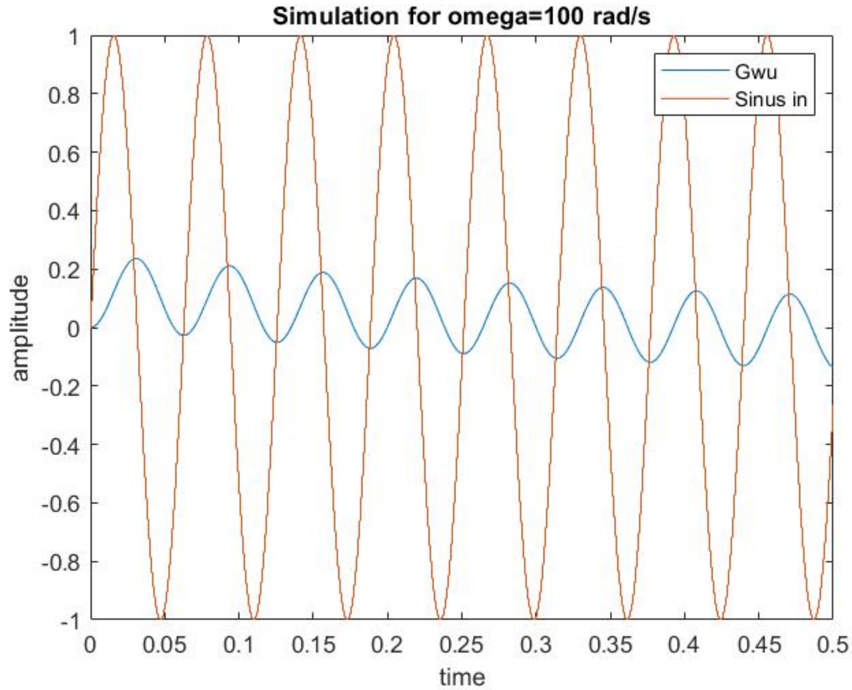

Figure 17: $G_{wu}(s)$ *plotted together with the sinus signal*

Figure 17 visualizes that the bode plot is an accurate description of the system.