Amirkabir University of Technology
(Tehran Polytechnic)

Electrical Engineering Faculty

Control Department

MSc Program

Assignment 01

through

System Identification

Course

by

Mohammad Azimi - 402123100

mohammadazimi2000@aut.ac.ir

Course Lecturer

Dr. Mehdi Karrari

I.Question 1

Code Section 01 – loading data

We read the set of data using the *load* command.

Code Section 02 – Visualizing Identification Data

The first 100 samples are utilized for system identification. To do so, the dataset must be sliced from 1 to 100 samples. Figure.1 illustrates the corresponding input and output values of the dataset to the mentioned slice of data.

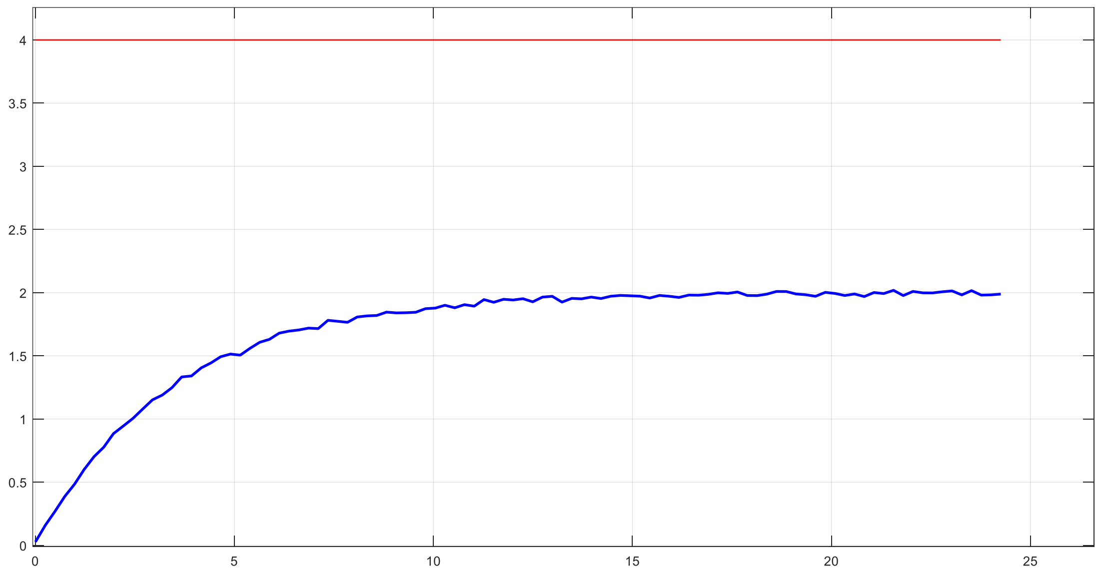


Figure 1

Obviously, the input signal is not a unit step signal. Accordingly, the output values of the system are required to be normalized.

Code Section 03 – Normalizing Identification Data

In this section, all of the samples used for system identification are normalized and scaled to the size of the unit step input signal as the reference. The normalized input and output values are illustrated in Figure.2.

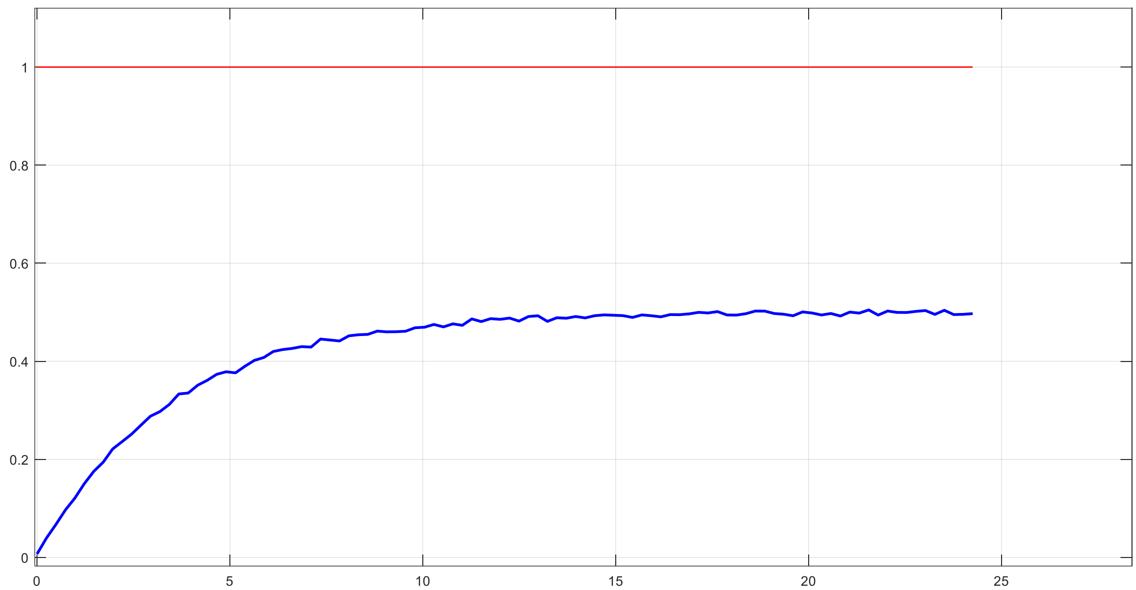


Figure 2

Code Section 04 – Parameters Estimation (K & Tau)

Since K parameter represents the final value of the system response, we have used the very last 10 samples of the output values to estimate the K parameter. Calculating the mean of the values results into the following K parameter.

$$K = 0.49898$$

In addition, the τ parameter is defined as the Time Constant of the system and is equal to the time that the system response reaches to about 63% of the its final value. Therefore, we have:

$$\tau = 3.552\ (seconds)$$

Code Section 05 – Forming the Transfer Function

Now we can simply form the transfer function of the system using the estimated parameters by the *tf* command. As a result:

$$G(s) = \frac{0.49898}{3.552s + 1}$$

Figure.3 is showing the response values that we used to identify the system and the estimated system response the same input signal. The result increases the hopes that the system would have a good performance under other input signals.

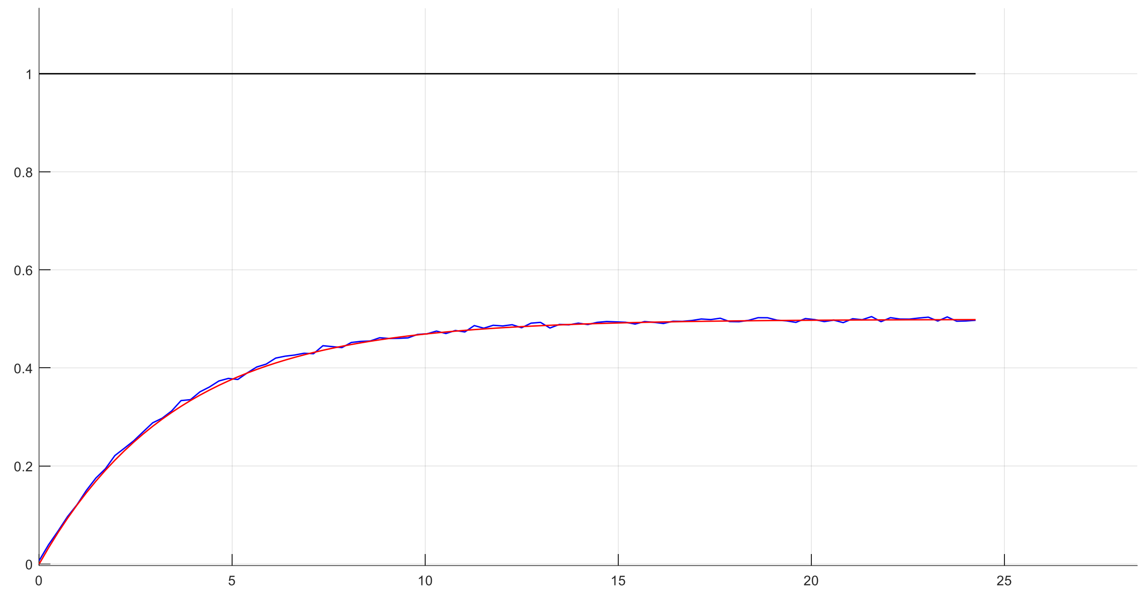


Figure 3

Code Section 06 – Model Evaluation (Graphically)

To evaluate the estimated model, we can compare real output values of the system with the output values of the estimated transfer function. This is illustrated in Figure.4 graphically and it shows that the system is performing well as expected on different input signals.

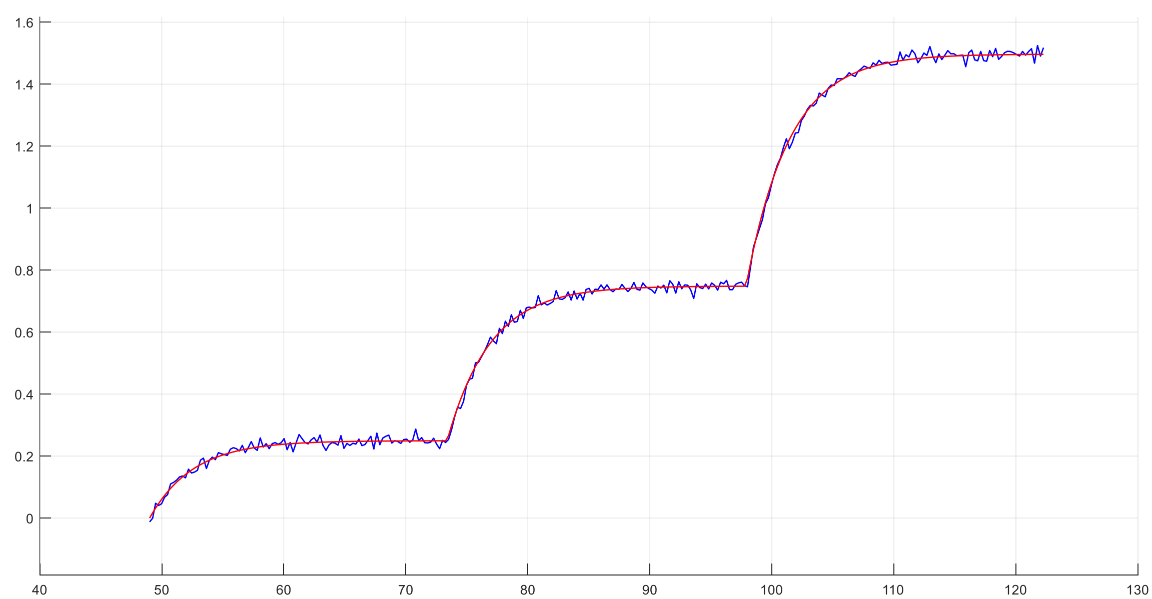


Figure 4

Code Section 07 – Model Evaluation (SSE, MSE & R2 Metrics)

Also, we can use SSE, MSE and R2 metrics to evaluate our model. This section of the code calculates SSE, MSE and R2 parameters to for evaluation. First, the error of the estimated system is calculated comparing to the real system output values. SSE is the sum of squared errors and MSE is the mean of the squared errors. Also, R2 formula is reported in the assignment appendix file too. These parameters results are as follows:

$$SSE = 0.0526920$$

$$MSE = 0.0001756$$

$$R2 = 0.9992976$$

Note that the lower SSE and MSE values are better. On the other hand, the closer value to 1 for R2 parameter shows the better quality of the system. It is worth mentioning that the estimated transfer function of the system is performing well since the R2 parameter is significantly greater than 0.90 and is so close to 1.

Code Section 08 – Report Display

Run the whole *main.m* file to have the following report printed on your command prompt.

```
===========================================================
Mohammad Azimi - 402123100 - Question 01
===========================================================
--------------The System Identification Report-------------
------> K   : 0.499
------> Tau : 3.552
------> G(s):    0.49898 /    3.5525 s + 1
----------------Model Evaaluation Report-------------------
------> SSE : 0.0526920
------> MSE : 0.0001756
------> R2  : 0.9992976
===========================================================
```

II.Question 2

Code Section 01 – loading data

We read the set of data using the *load* command.

Code Section 02 – Visualizing Identification Data

The first 100 samples are utilized for system identification. To do so, the dataset must be sliced from 1 to 100 samples. Figure.5 illustrates the corresponding input and output values of the dataset to the mentioned slice of data.

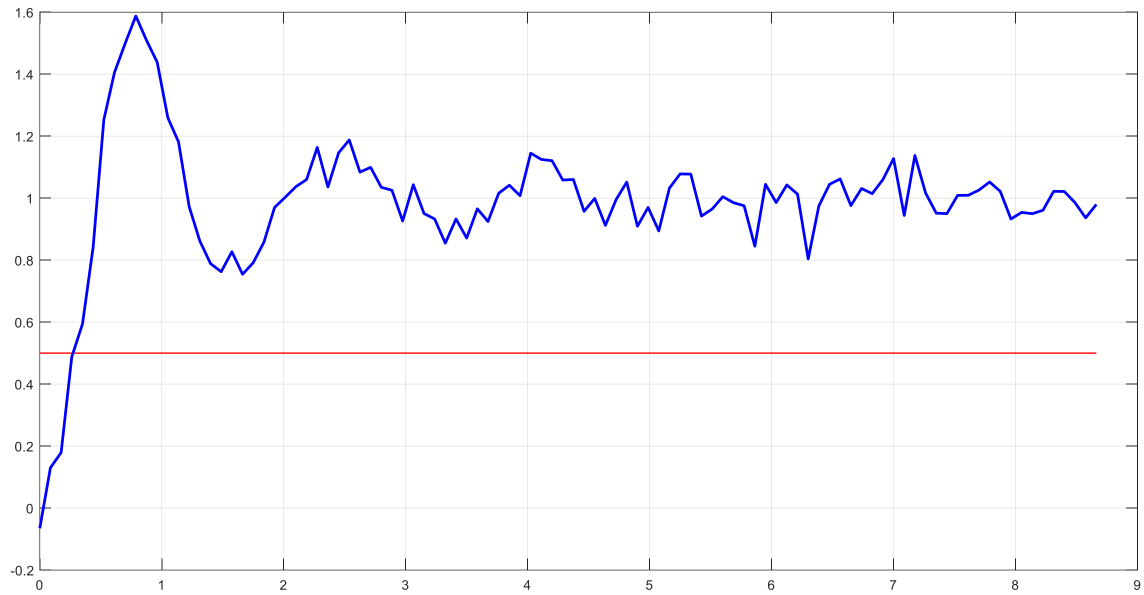


Figure 5

Obviously, the input signal is not a unit step signal. Accordingly, the output values of the system are required to be normalized.

Code Section 03 – Normalizing Identification Data

In this section, all of the samples used for system identification are normalized and scaled to the size of the unit step input signal as the reference. The normalized input and output values are illustrated in Figure.6.

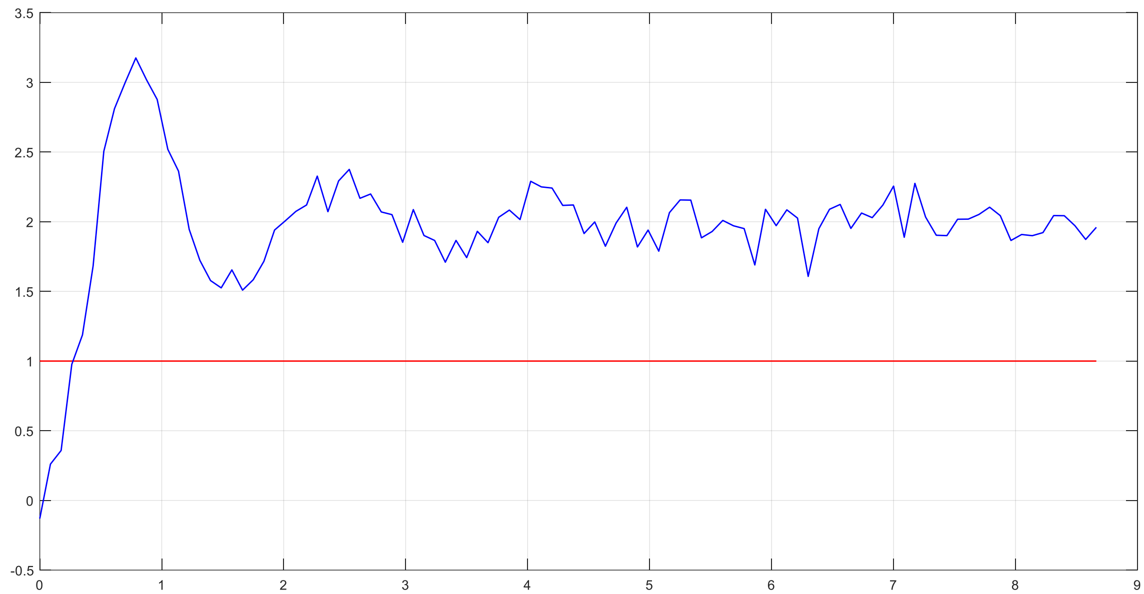


Figure 6

Code Section 04 – Parameters Estimation (K, T, M thus ω_n and ξ)

Since K parameter represents the final value of the system response, we have used the very last 10 samples of the output values to estimate the K parameter. Calculating the mean of the values results into the following K parameter.

$$K = 1.9704$$

To move forward, we need to estimate some more parameters. T is the time interval between the peak response values (see Figure.7).

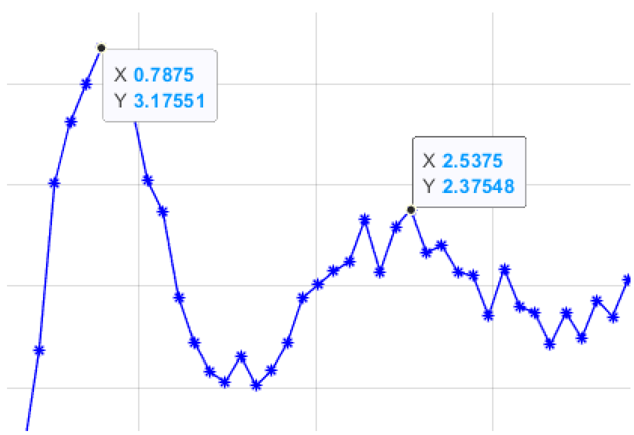


Figure 7

In our case, T could be calculated as following.

$$T = 2.53 - 0.78 = 1.75$$

Also, M is calculated with the equation below:

$$k(1+M) = 3.1755 \implies M = 0.5733$$

Hence, ω_n and ξ are derived as;

$$\omega_n = \frac{2}{T}[\pi^2 + (lnM)^2]^{\frac{1}{2}} = 3.6462$$

$$\xi = \frac{-lnM}{[\pi^2 + (lnM)^2]^{\frac{1}{2}}} = 0.1744$$

Code Section 05 – Forming the Transfer Function

Now we can simply form the transfer function of the system using the estimated parameters by the *tf* command. As a result:

$$G(s) = \frac{26.2}{s^2 + 1.27s + 13.29}$$

Figure.8 is showing the response values that we used to identify the system and the estimated system response the same input signal. The result increases the hopes that the system would have a good performance under other input signals.

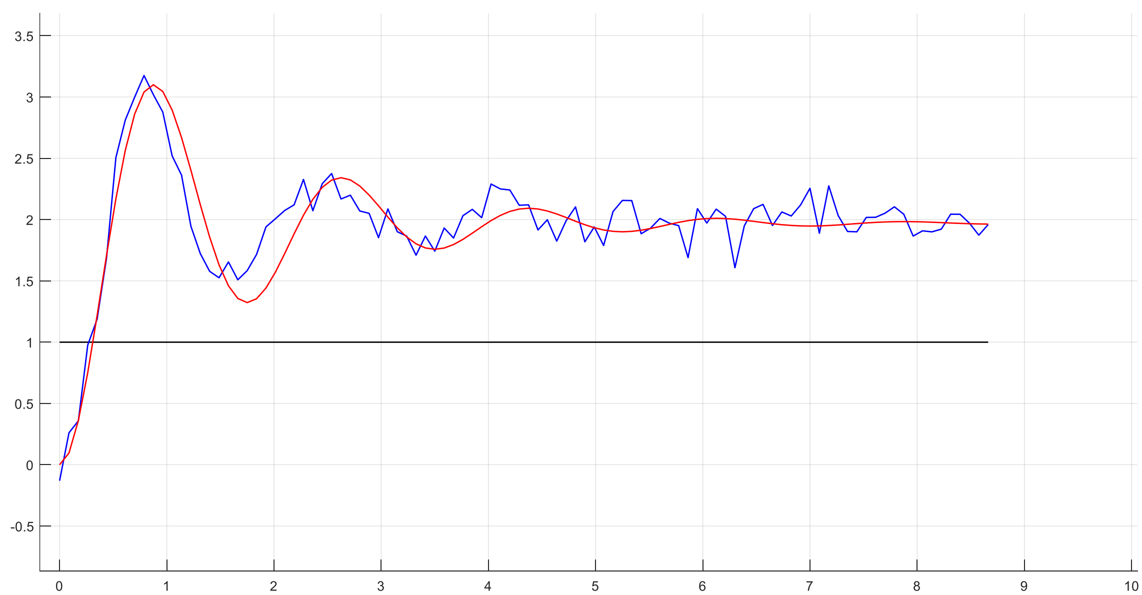


Figure 8

Code Section 06 – Model Evaluation (Graphically)

To evaluate the estimated model, we can compare real output values of the system with the output values of the estimated transfer function. This is illustrated in Figure.9 graphically and it shows that the system is performing well as expected on different input signals.

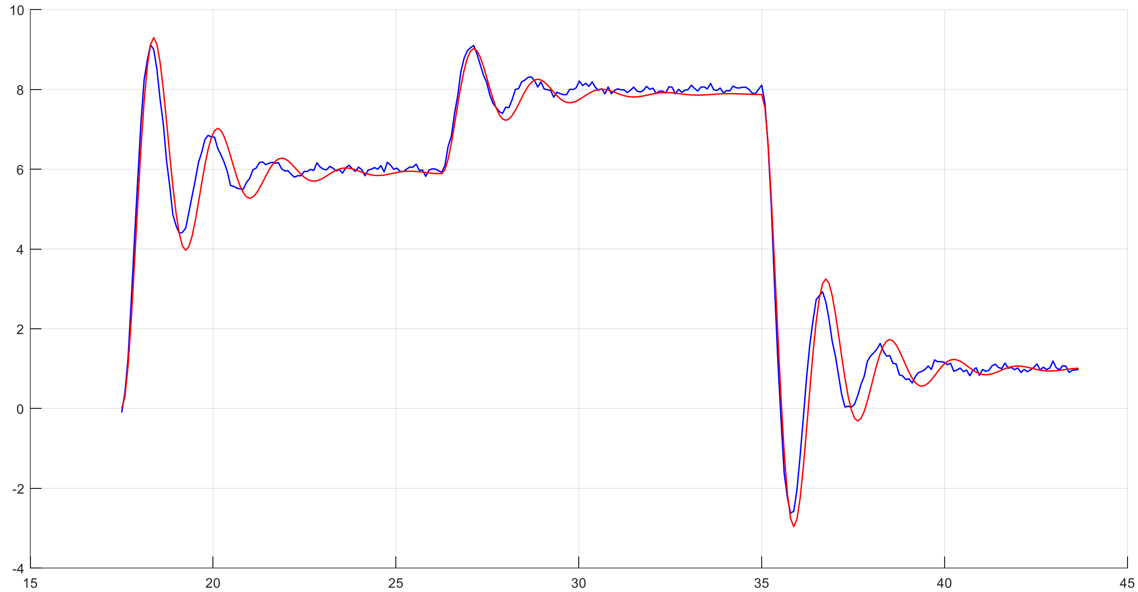


Figure 9

Code Section 07 – Model Evaluation (SSE, MSE & R2 Metrics)

Also, we can use SSE, MSE and R2 metrics to evaluate our model. This section of the code calculates SSE, MSE and R2 parameters to for evaluation. First, the error of the estimated system is calculated comparing to the real system output values. SSE is the sum of squared errors and MSE is the mean of the squared errors. Also, R2 formula is reported in the assignment appendix file too. These parameters results are as follows:

$$SSE = 43.4261$$

$$MSE = 0.14475$$

$$R2 = 0.984901$$

Note that the lower SSE and MSE values are better. On the other hand, the closer value to 1 for R2 parameter shows the better quality of the system. It is worth mentioning that the estimated transfer function of the system is performing well since the R2 parameter is significantly greater than 0.90 and is so close to 1.

Code Section 08 – Report Display

Run the whole *main.m* file to have the following report printed on your command prompt.

```
=============================================================
Mohammad Azimi - 402123100 - Question 02
=============================================================
--------------The System Identification Report-------------
------> K  : 1.970
------> T  : 1.750
------> M  : 0.573
------> Wn : 3.646
------> Z  : 0.174
------> G(s):    26.1961 /    s^2 + 1.2716 s + 13.2951
----------------Model Evaaluation Report-------------------
------> SSE : 43.4261090
------> MSE : 0.1447537
------> R2  : 0.9849019
=============================================================
```

III. Question 3

Given the step response data through the ***t.mat*** and ***y.mat*** and the following plot in Figure.10, one can identify the system as either a first-order or second-order system. Note that the given step response is noisy. Moreover, the response in demonstrating that the system is coming with a noticeable delay which must be considered for identification.

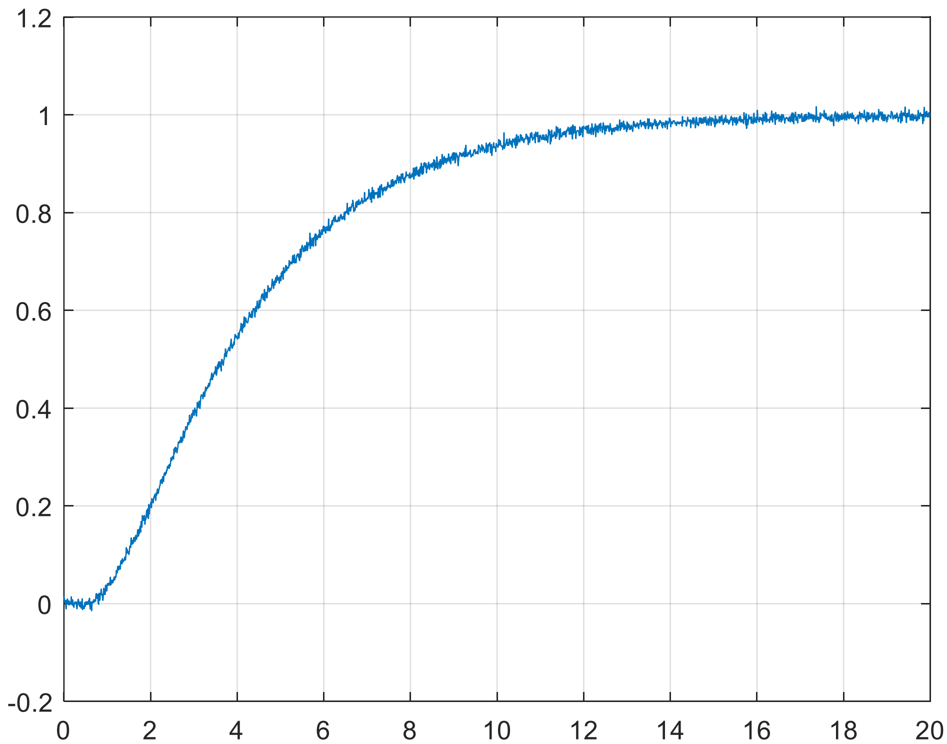


Figure 10 - given step response for Q3

a) 1st-Order Identification

Considering the general transfer function in (1) for 1st-order systems, we need to analyze the available step response regarding the τ factor.

$$G(s) = \frac{a}{s+a} = \frac{1}{1+\tau s} \quad ; \quad \tau = \frac{1}{a} \quad \xrightarrow{\text{Considering the Delay Factor}} G(s) = \frac{ae^{-t_0 s}}{s+a} = \frac{1}{1+\tau s}e^{-t_0 s} \quad (1)$$

τ is defined as the Time Constant in 1st-order systems in which the step response reaches out to almost 63% of the final response value. Since the available step response is noisy and also exposing a delay factor, we may need a second metric to increase our modeling accuracy. For this, 2τ is defined to provide 84% of the response final value. Furthermore, 3τ is 95% and 4τ, 98%.

i) Check for the Delay Factor

Let's zoom in the plot on the very beginning area to estimate the delayed time value. Check out Figure.11.

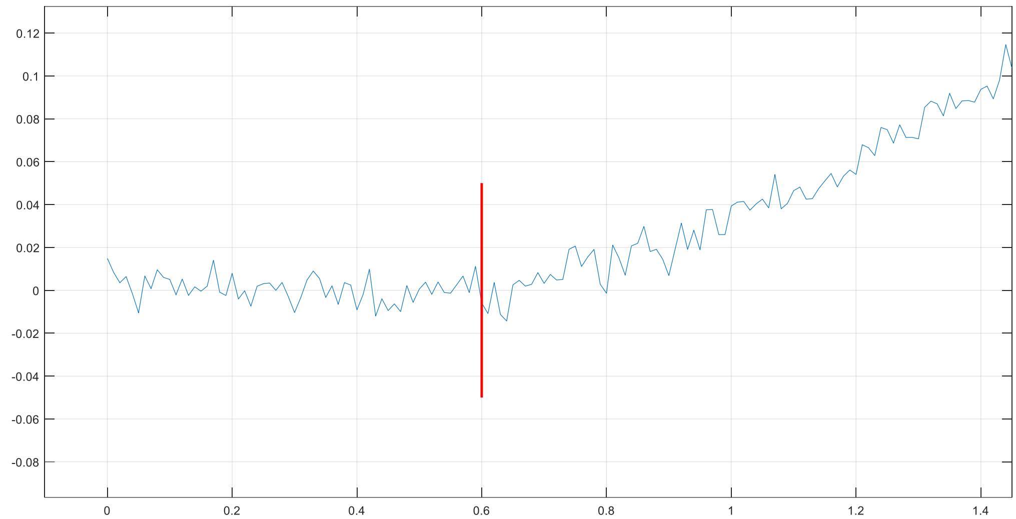


Figure 11 - Estimating the Delay value of the response

The red line in Figure.11 may present a satisfying delay time value in the response on the very first sight. Accordingly, we will have $\underline{t_0 = 0.6}$s.

ii) Check for the Time Constant

Firstly, time of [0.63∗ the response final value] must be estimated. Since the step response is reaching to the final value of 1, we are looking for the time that the system output is 0.63. Once again, let's zoom in the plot to discover where the response is being around 0.63.

Figure.12 is almost declaring that the response value is 0.63 at about 4.6s. The corresponding time constant value may satisfy our model or not. We could give it a try.

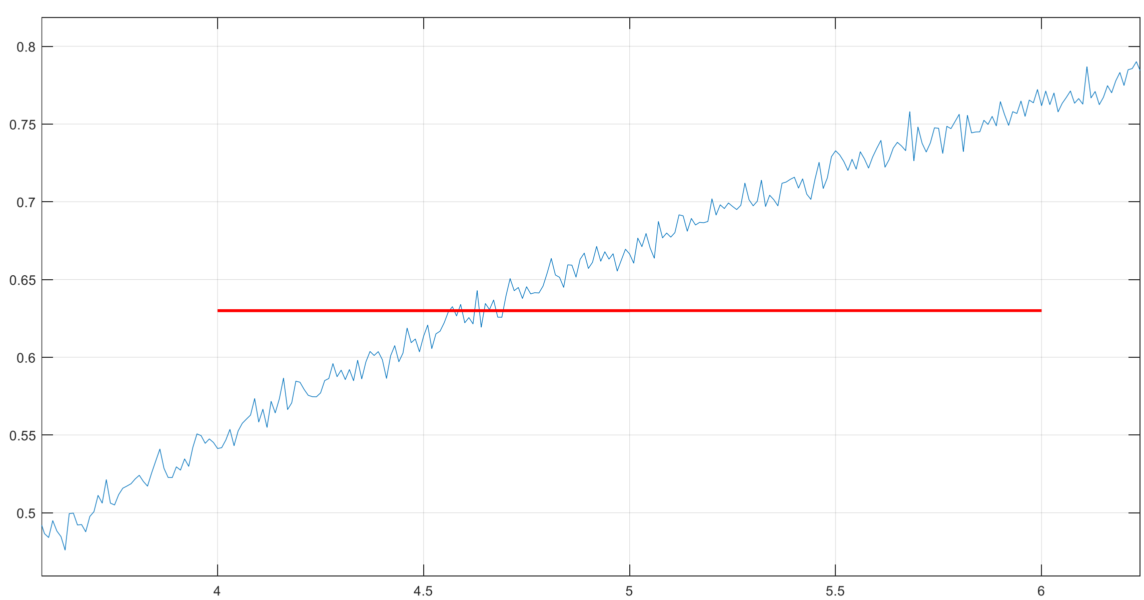


Figure 12 - Estimating the Time Constant value

iii) Forming the Transfer Function

Up to now, we have the following values for the required parameters.

$t_0 = 0.6$s

$\tau + t_0 = 4.6$s

Hence:

$\tau = 4.0$s

Therefore, the transfer function would be:

$$G(s) = \frac{1}{4s+1} e^{-0.6s}$$

The step response for the estimated transfer function is shown in Figure.13.

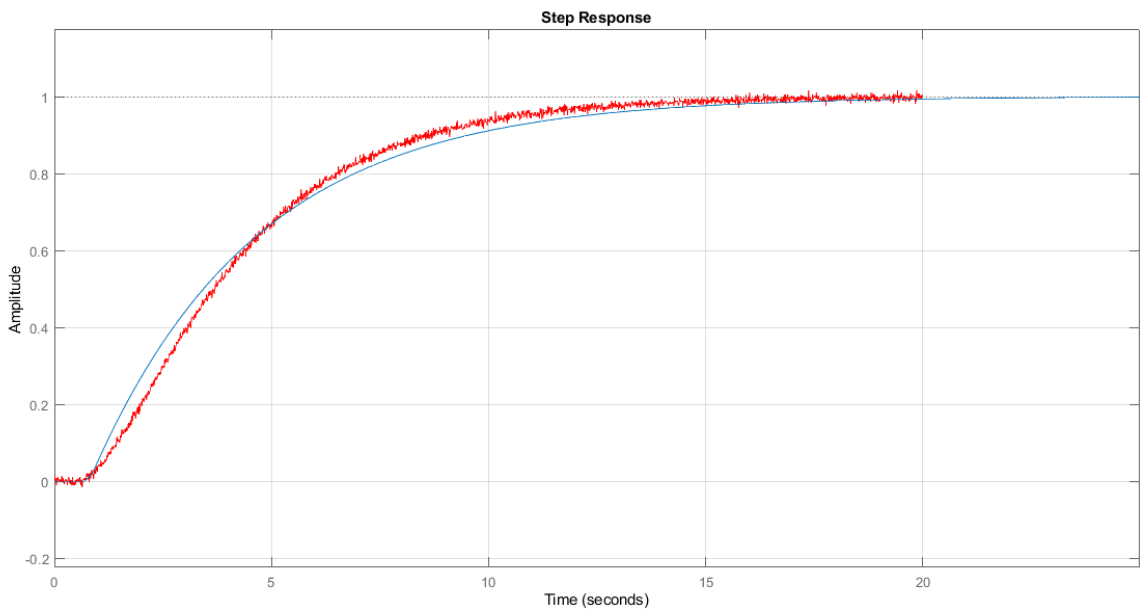


Figure 13

Figure.13 demonstrates that the system could not be modeled well by a 1st-order transfer function. Therefore, it is expected to have a better result in case of estimating the system with a 2nd-order structure.

b) 2nd-Order Identification

Considering the general transfer function in (2) for 2nd-order systems, we need to analyze the available step response regarding the ξ and ω_n.

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \qquad (2)$$

Since there is no overshoot in the transient part of the response, it is guaranteed that the system is not underdamped. and it could be said to be either overdamped or critically damped. This means that ξ is greater or equal to 1. Figure.14 shows the step response form of a 2nd-order system in case that it is an overdamped system.

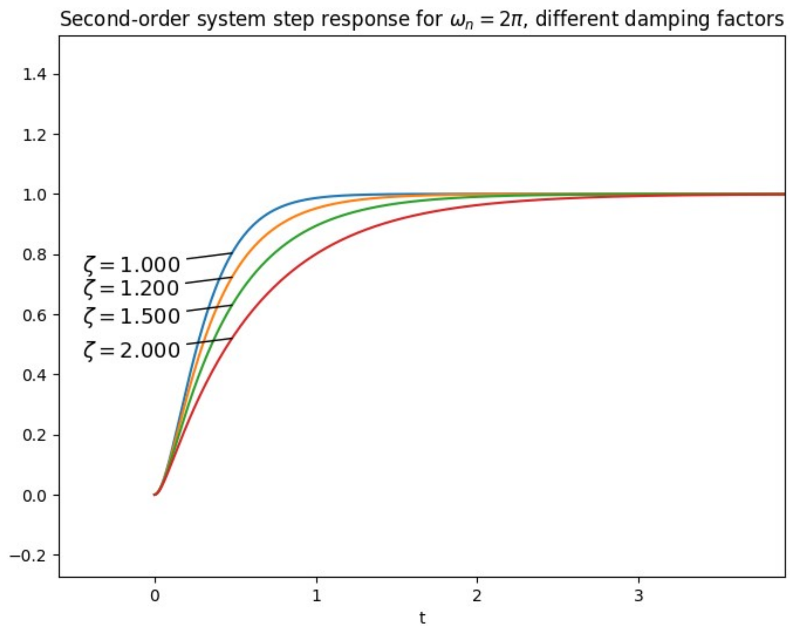


Figure 14

approach 1

The wise way to identify this system as second order is to use the previous section estimated transfer function, find its pole and place a second pole for the system over than at least 10 times of the first pole place. In this case, the transfer function is more likely to behave based on the dominant pole of the system, although the system is having two poles. Since this method is not of my favorites, I intend to try the second approach.

approach 2

Considering that there is not a specific set of criteria for estimating parameters of an overdamping or critically damping system, the trial and error method is used to estimate the transfer function. After trying some cases, it appears to have the system perfectly estimated under the following circumstances.

$$\xi = 1.2$$

$$\omega_n = 0.6$$

Hence, the transfer function will be:

$$G(s) = \frac{0.36\, e^{-0.6s}}{s^2 + 1.44s + 0.36}$$

And the estimated transfer function step response comes along the actual system response in Figure.15. Clearly, the identified system has a good performance to represent the actual system. Note that there is still a 0.6s of delay in the transfer function just like the 1st-order estimation.

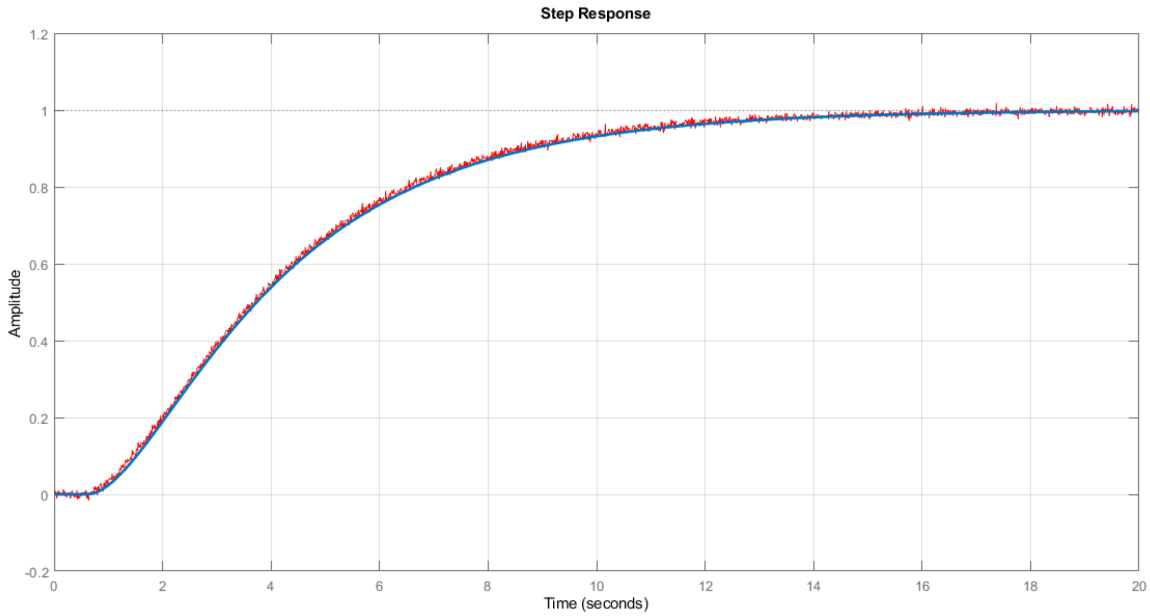


Figure 15

IV.Question 4

Given the step response data through the ***t.mat*** and ***y.mat*** and the following plot in Figure.16, one can identify the system as either a first-order or second-order system. Note that the given step response is noisy.

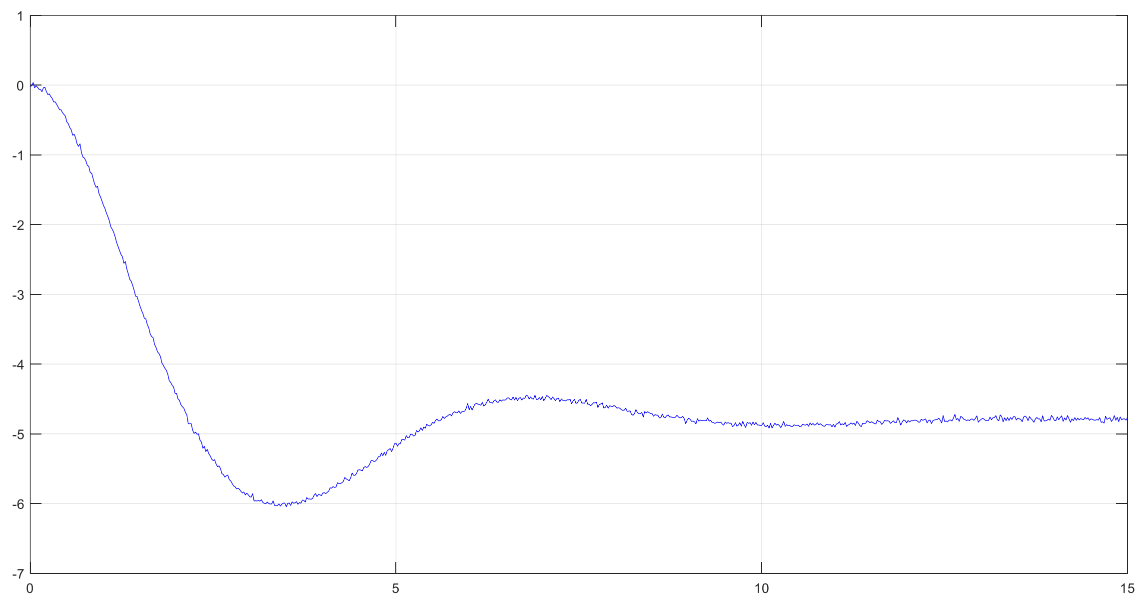


Figure 16

There is this weird situation of the system response that it is upside-down. The idea we may use to overcome this condition is to multiply the response values by -1 to flip the plot. Then, however, identify the transfer function and at the end, multiply the estimated system by -1 to compensate the very first reverse we applied on the data. Taking this scenario, we have the plot in Figure.17.

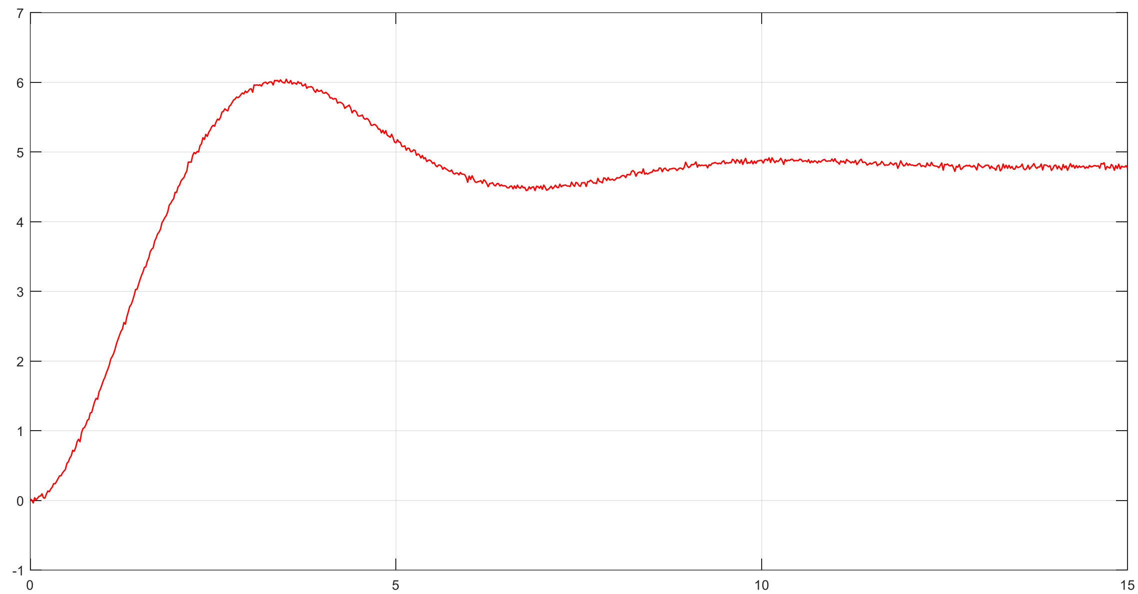


Figure 17

Despite the previous exercise step response, here the response is definitely representing an underdamping system. So we are able to utilize the common relations to extract the system required parameters. Firstly, K is measured by following the final value of the response. Taking the average value of the very last 50 samples. Accordingly: `K = mean(y_reversed(700:751));`

then: $K = 4.7898$

To estimate ξ and ω_n parameters, we could look for the peak time. Zooming in the plot and trying to measure the peak time (Figure.18) would lead us to:

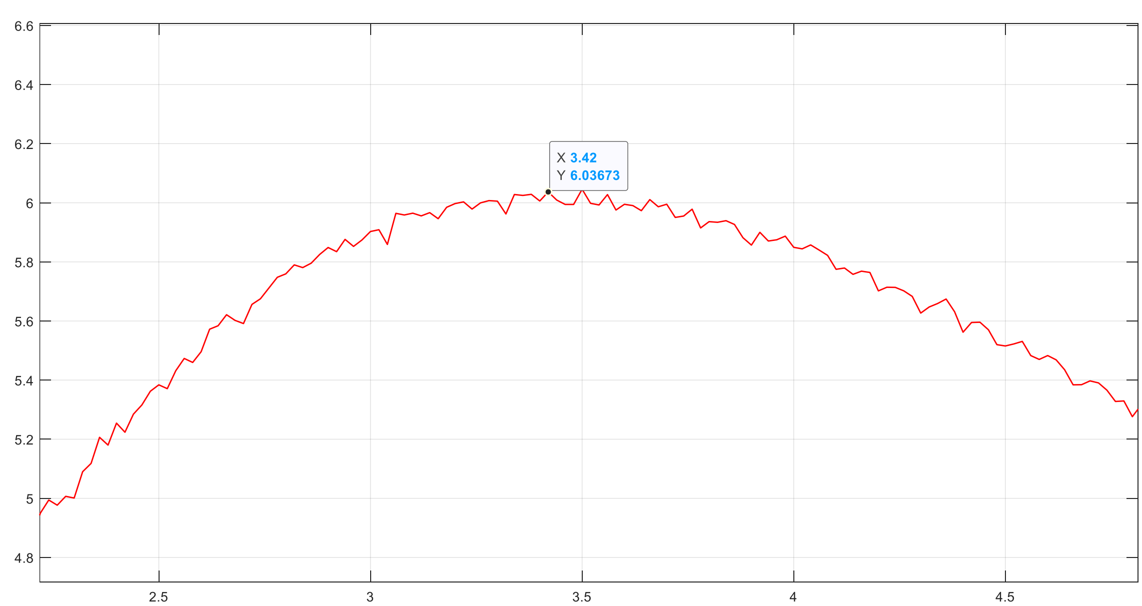


Figure 18

Therefore, we get:

$$t_p = 3.42s \longrightarrow t_p = \frac{\pi}{\omega_d} \Longrightarrow \omega_d = 0.9186 \quad ; \quad \omega_d = \omega_n\sqrt{1-\xi^2}$$

Also, the rise time of the system would be so helpful. Check out the attached MATLAB script for the piece of code used to calculate the rise time programmatically.

$$t_r = 2.17s = \frac{\pi - \phi}{\omega_d} \Longrightarrow \phi = 1.1482 = cos^{-1}(\xi) \Longrightarrow \xi = 0.4101$$

$$\longrightarrow \omega_d = \omega_d\sqrt{1-\xi^2} \Longrightarrow \omega_n = 1.0072$$

Finally, the transfer function would be:

$$\widehat{G(s)} = \frac{4.859}{s^2 + 0.8261s + 1.014}$$

Comparing the actual and estimated systems step responses, in Figure.19, proves that the identified transfer function is fitting really well with the actual system. The last but not the least, remember that we need to apply a -1 multiplication to the transfer function so match the actual system. Thus:

$$G(s) = \frac{-4.859}{s^2 + 0.8261s + 1.014}$$

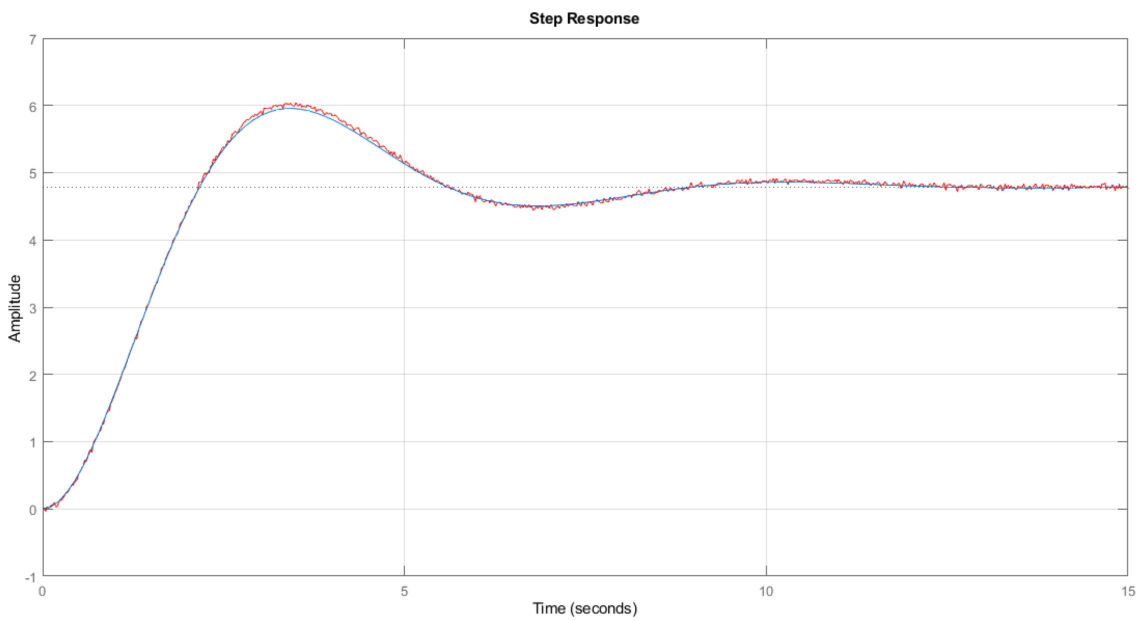


Figure 19

V.Question 5

Having loaded the data, we split them into two sets. One for identifying the system and one for evaluating the identified system. To see the slicing form of the data for the first set, you can refer to code section 02 within the corresponding *main.m* script. Considering that the system had not a zero initial state while it had been sampling, the procedure of system identification based on the impulse response is as follows. All the explained process is available in code section 03.

1. Calculate the y_0 and y_{ss} of the response by taking the average of the very last 10 samples of the response.

$$y_{ss} = y_0 = 0.0490$$

2. Extract the y_{max} value and the time at which it is taking place.

$$y_{max} = 0.2418 \qquad t_1 = 1.24s$$

3. Find the response value of $y_0 + 0.368(y_{max} - y_0)$ and the time at which it is taking place.

$$y_0 + 0.368(y_{max} - y_0) = 0.1199 \qquad t_2 = 1.78s$$

4. Calculate K and τ using the following equations.

$$K = \frac{y_{ss}}{u_{ss}} = 0.979 \qquad \tau = t_2 - t_1 = 0.54$$

Now, we form the transfer function and check its response over the same input along the actual system response (Figure.20). The result seems to be promising; however, we move forward with it to evaluate the estimated model.

$$G(s) = \frac{0.09794}{0.54s + 1}$$

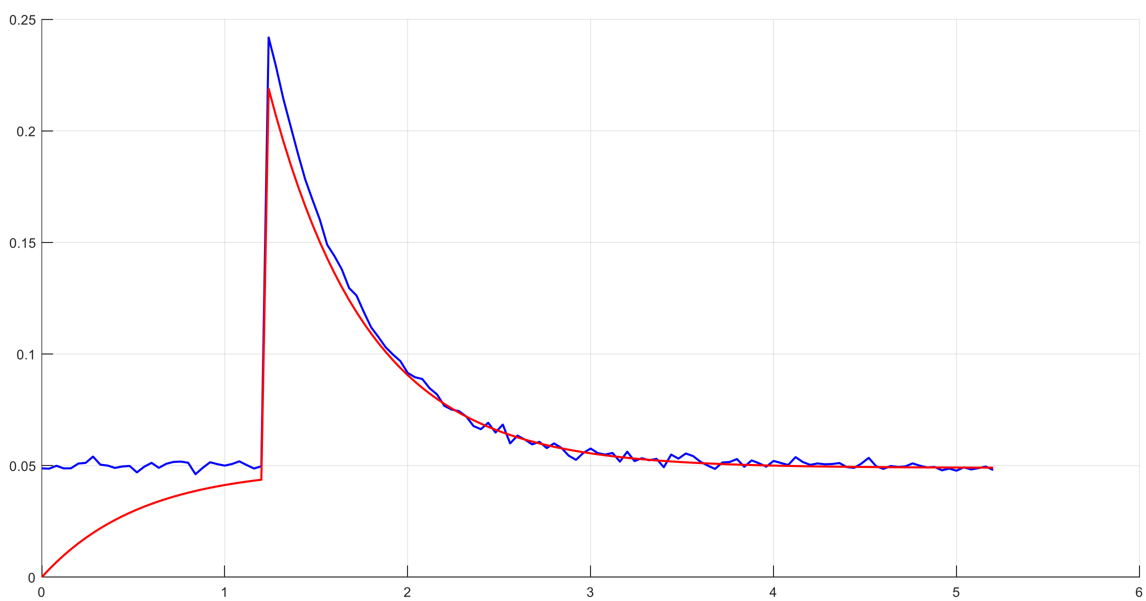


Figure 20

Code section 05 provides a graphical view of the estimated and actual system response of which the results are depicted in Figure.21. This is obtained using the second set of sliced data and as expected, the estimated system seems to work fine.

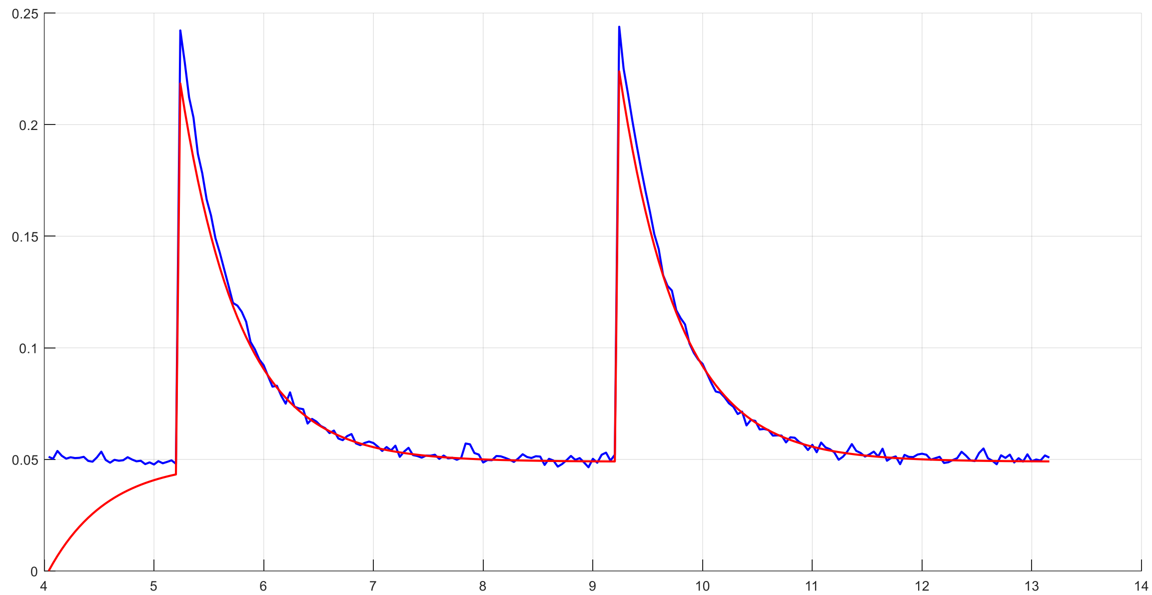


Figure 21

It is the numeric analytics that always talks. Code section 06 provides the code to calculate SSE, MSE and R^2 metrics to evaluate the identified model. You can easily run the *main.m* script to have the result printed on your console. However, the report is as:

$$SSE = 0.0235667 \qquad MSE = 0.0001029 \qquad R2 = 0.9394065$$

To conclude, having the R^2 parameter as 0.93 which is greater than 0.90 proves that the identified system is capable of representing the actual system perfectly.

```
===========================================================
Mohammad Azimi - 402123100 - Question 05
===========================================================
--------------The System Identification Report-------------
------> K   : 0.098
------> Tau : 0.540
------> G(s):    0.097936 /    0.54 s + 1
----------------Model Evaaluation Report-------------------
------> SSE : 0.0235667
------> MSE : 0.0001029
------> R2  : 0.9394065
===========================================================
```

VI.Question 6

Figure.22 shows the given data for identification stage. The existing oscillation suggests that the system is underdamped. The first step to take is determining u_{ss}, u_0, y_{ss} and y_0 values which is done in code section 03.

$$u_{ss} = 0.5 \qquad u_0 = 0.5 \qquad y_{ss} = 0.9965 \qquad y_0 = 0.9965$$

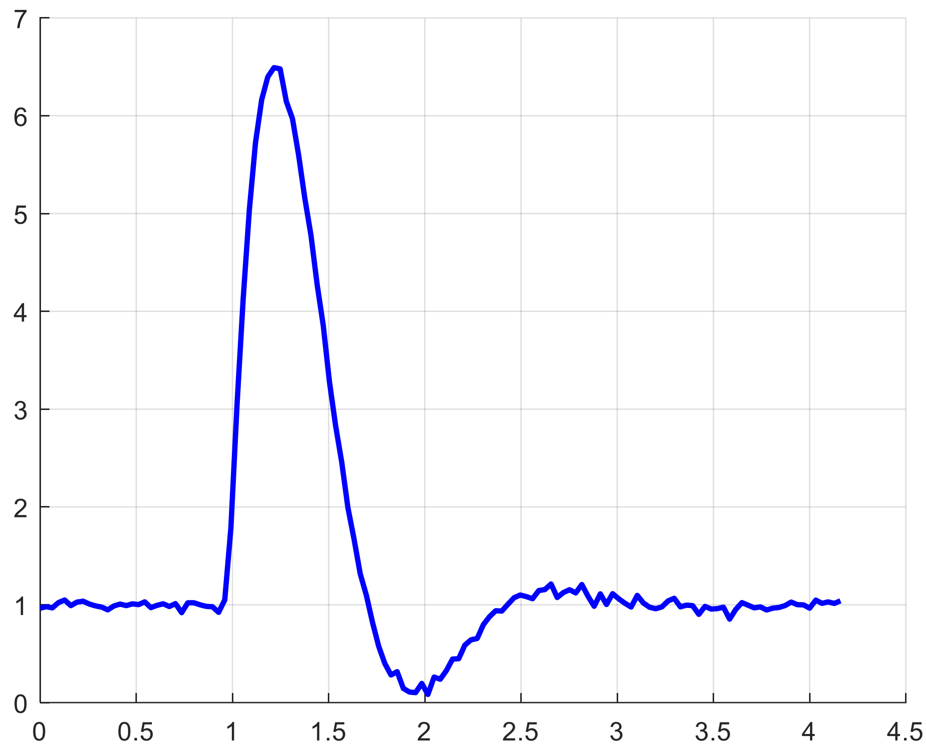
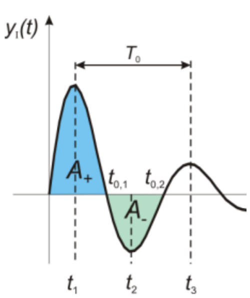


Figure 22

The next target is *M* for which we must calculate the areas under oscillation halves. Three time values are marked in Figure.23 which represent the time intervals of oscillation parts. These values are named as:

$$t_{01} = 0.9280 \qquad t_{02} = 1.6960 \qquad t_{03} = 2.4320$$

Continuing with code section 03, you can find the code for measuring the areas within the intervals. The measured areas are as follows and accordingly, *M* is calculated.



$$\begin{Bmatrix} A_+ = 0.0032 \\ A_- = 0.2881 \end{Bmatrix} \quad \Longrightarrow M = \frac{A_-}{A_+} = 0.899$$

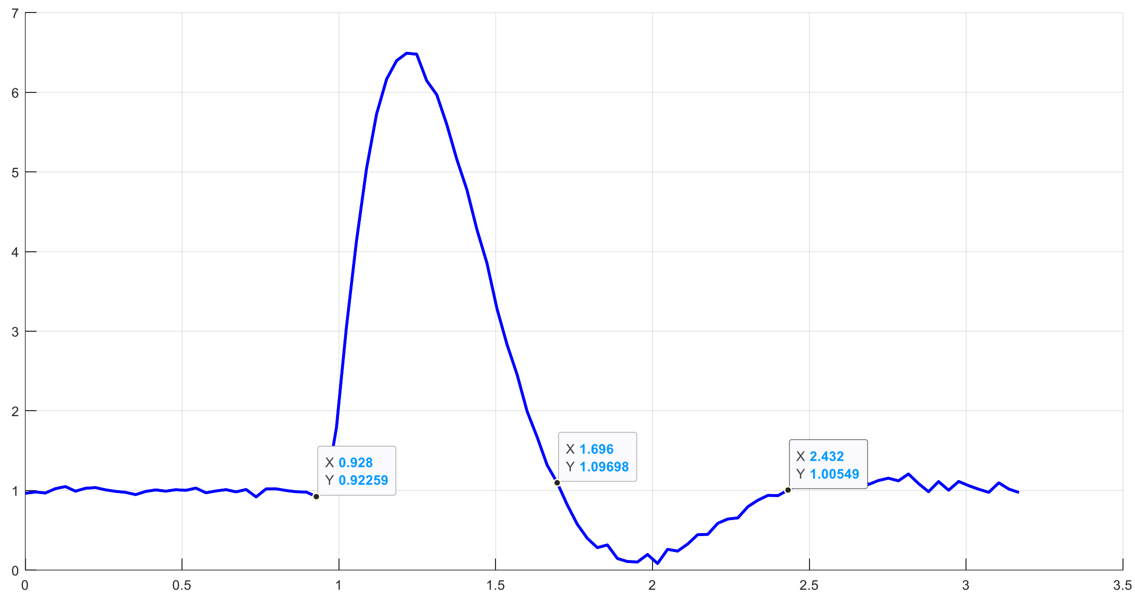


Figure 23

Having calculated *M*, it is easy to obtain ξ using the equation: $\xi = \frac{-lnM}{[\pi^2+(lnM)^2]^{\frac{1}{2}}} = 0.6086$

Now, t_1, t_2 and t_3 times must be located as in Figure.24.

$$t_1 = 1.2160 \qquad t_2 = 2.0160 \qquad t_3 = 2.6560$$

$$\longrightarrow T_0 = t_3 - t_1 = 1.44s \qquad \longrightarrow \omega_n = \frac{2\pi}{T_0\sqrt{1-\xi^2}} = 5.4989$$

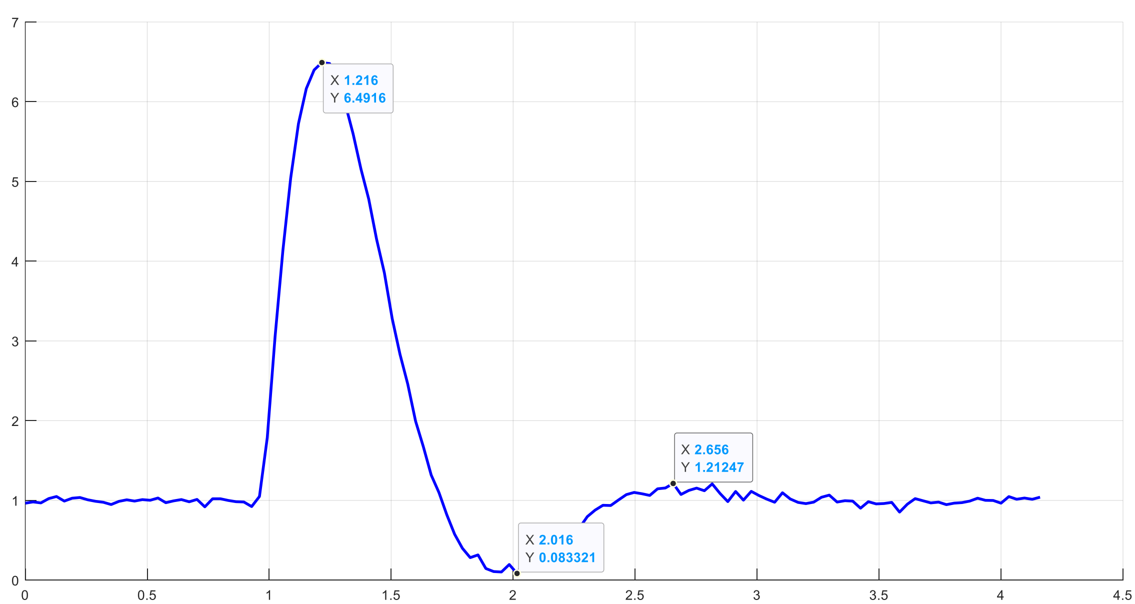


Figure 24

Finally, K is derived by a simple division of $K = \frac{y_{ss}}{u_{ss}} = 1.9929.$

After all, it's time to move into code section 05 and form the transfer function and check out the response of the estimated system which is depicted in Figure.25.

$$G(s) = \frac{60.26}{s^2 + 6.693s + 30.24}$$

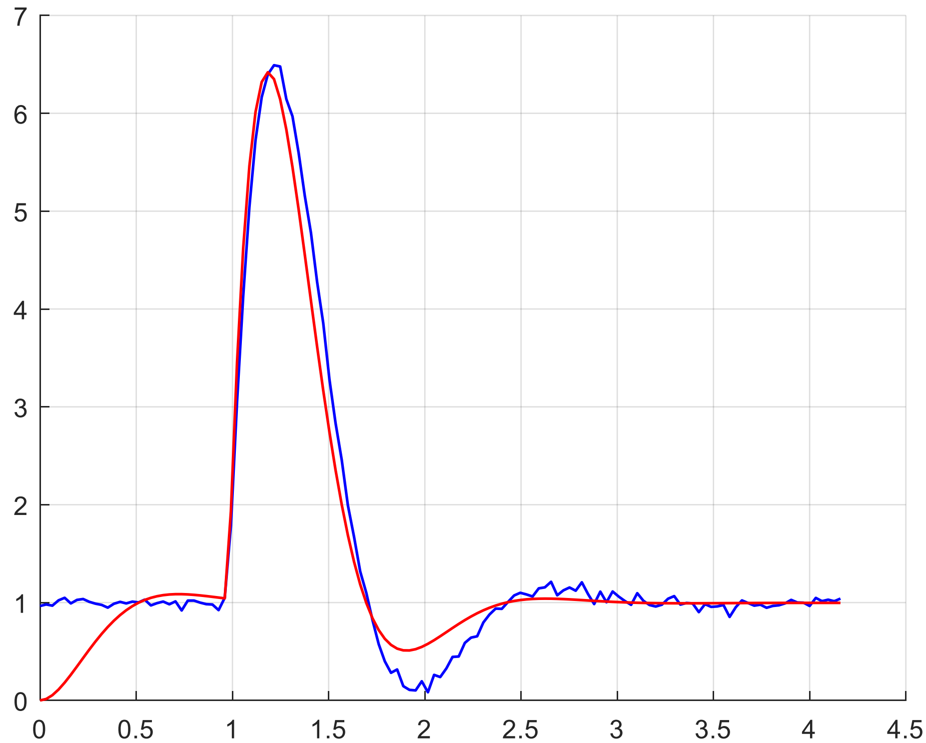


Figure 25

Response magnitude seems to be fine but the undershoot value would bring up some concerns for the accuracy of the system. Let's keep hopes up and visualize two more input impulses and see the results. Also most importantly, it's the numerical analysis that decides whether the estimation is working fine or not.

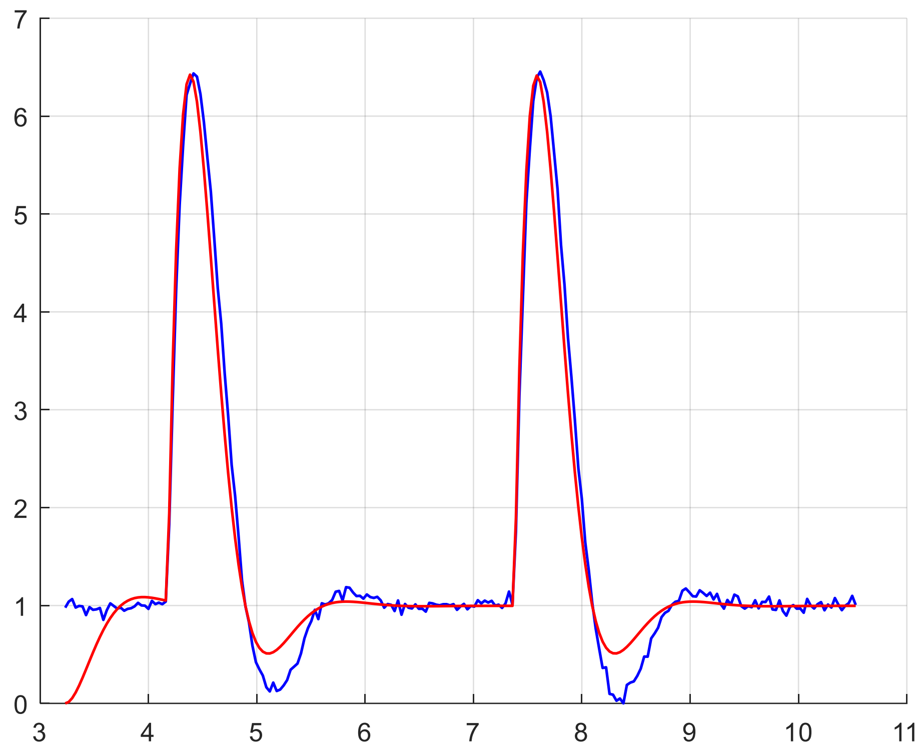


Figure 26

Figure.26 is similar to the first input signal results too. So let's skip to numerical evaluation. SSE, MSE and R2 parameters of the estimated transfer function are reported as follows:

$$SSE = 19.0906$$

$$MSE = 0.0833$$

$$R2 = 0.9658$$

Fortunately, the R^2 metric is stating a good performance from the estimated system in spite of the concerns we had based on out visualization. So the identified model could be used to represent the actual model.

```
============================================================
Mohammad Azimi - 402123100 - Question 06
============================================================
--------------The System Identification Report-------------
------> K  : 1.993
------> M  : 0.090
------> Wn : 5.499
------> Z  : 0.609
------> G(s):    60.2614 /    s^2 + 6.693 s + 30.2378
----------------Model Evaaluation Report-------------------
------> SSE : 19.0906177
------> MSE : 0.0833651
------> R2  : 0.9658781
============================================================
```

VII. Question 7

Being supposed to use the Prony method to identify a system, we enter the given step response values in MATLAB as an array. Since Prony uses impulse response to estimate the system model, it is required to convert the step response to impulse response. To do so, the following equation is utilized. Figure.27 and Figure.28 are illustrating the step and the impulse response of the system.

$$h_t \cong \frac{S_t - S_{t-1}}{T}$$

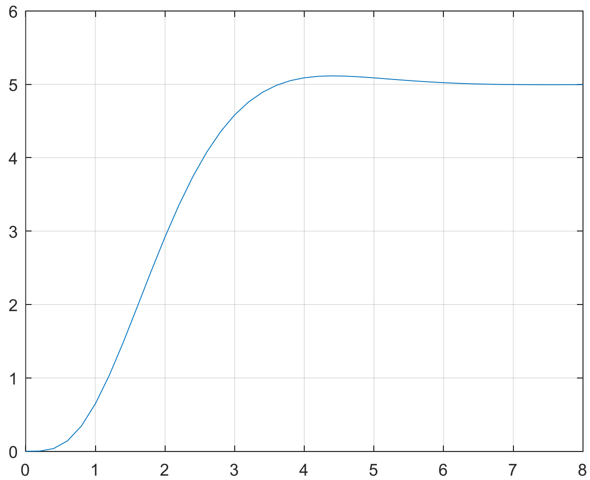


Figure 27

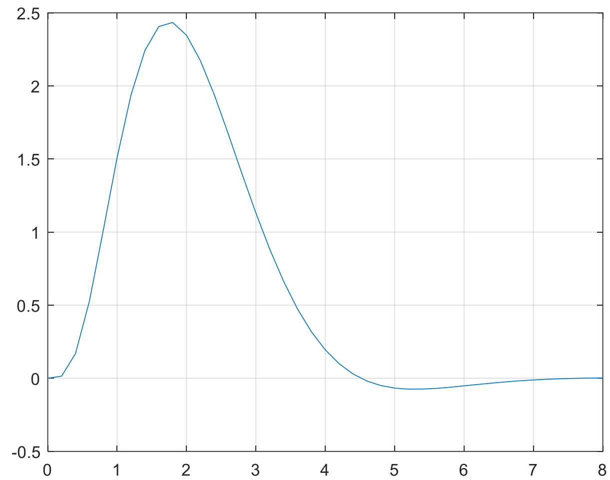


Figure 28

The first step in this method is defining a system degree. In our case, let's assume that the system degree is 15. So we take $n = 15$. Also, we already know the samples count which is $N = 41$. Secondly, the equation

$$Da = Y$$

must be formed and solved.

Before moving forward, let's try to find the real degree of the system using pseudo eigenvalues of the matrix D. The set of $D^T D$ matrix eigenvalues includes 4 nonzero positive values which represents the real order of the system. Hence we take $n = 4$ and form again the

$$Da = Y$$

and solve it for a.

In this equation, D and Y are known and a is the variable for which the equation needs to be solved. Since D is not a square matrix, we need to use the pseudo inverse approach. Therefore, we have:

$$a = D^+ \times Y = \begin{bmatrix} -a_1 \\ -a_2 \\ \vdots \\ -a_n \end{bmatrix}$$

where

$$D^{+} = (D^{T}D)^{-1}D^{T}Y$$

having measure a, the characteristic equation of the discrete system is formed as:

$$z^{n} + a_1 z^{n-1} + \cdots + a_n = 0$$

of which the roots are the discrete system poles recalled as z_i. In our case, it goes like:

$$a = \begin{bmatrix} 2.8852 \\ -3.1540 \\ 1.5503 \\ 0.2906 \end{bmatrix} \Rightarrow a_1 = -2.8852\ ;\ a_2 = 3.1540\ ;\ a_3 = -1.5503\ ;\ a_4 = 0.2906$$

Accordingly:

$$z^4 - 2.8852z^3 + 3.1540z^2 - 1.5503z + 0.2906 = 0$$

That leads us to discrete system poles:

$$\begin{Bmatrix} z_1 = 0.8230 + 0.1433i \\ z_2 = 0.8230 - 0.1433i \\ z_3 = 0.6196 + 0.1804i \\ z_4 = 0.6196 - 0.1804i \end{Bmatrix}$$

Now that we are capable of calculating $a_i$and z_i values, the equation $ZB = Y$ in which the known variable is *B* could be solved too. Once again, pseudo inverse approach is the way to solve this equation for *B*. This process would lead to:

$$\begin{Bmatrix} B_1 = -5.8544 + 6.6543i \\ B_2 = -5.8544 - 6.6543i \\ B_3 = +5.8544 + 1.3559i \\ B_4 = +5.8544 - 1.3559i \end{Bmatrix}$$

At this point, we seem to have any information required to set the transfer function of the system up. Considering the relation between discrete and continuous poles as $p_i = -\frac{\ln(z_i)}{T}$, the continuous transfer function is obtained as:

$$G(s) = \frac{B_1}{s + p_1} + \frac{B_2}{s + p_2} + \frac{B_3}{s + p_3} + \frac{B_4}{s + p_4}$$

Finally, the transfer function appears as following:

```
(1.22e-05+2.442e-15i) s^3 + (0.2019+7.105e-14i) s^2 - (4.321+5.329e-15i) s + (52.16+7.461e-14i)
------------------------------------------------------------------------------------------------
                         s^4 + 6.179 s^3 + 16.23 s^2 + 19.03 s + 10.56
```

The estimated transfer function unit step response is depicted in Figure.29 along the given one of the actual system in which it is following the actual step response perfectly.

Furthermore, SSE, MSE and R^2 metrics are approving the estimated model as they are reporting the following values.

$$SSE = 0.4858$$

$$MSE = 0.0118$$

$$R2 = 0.9964$$

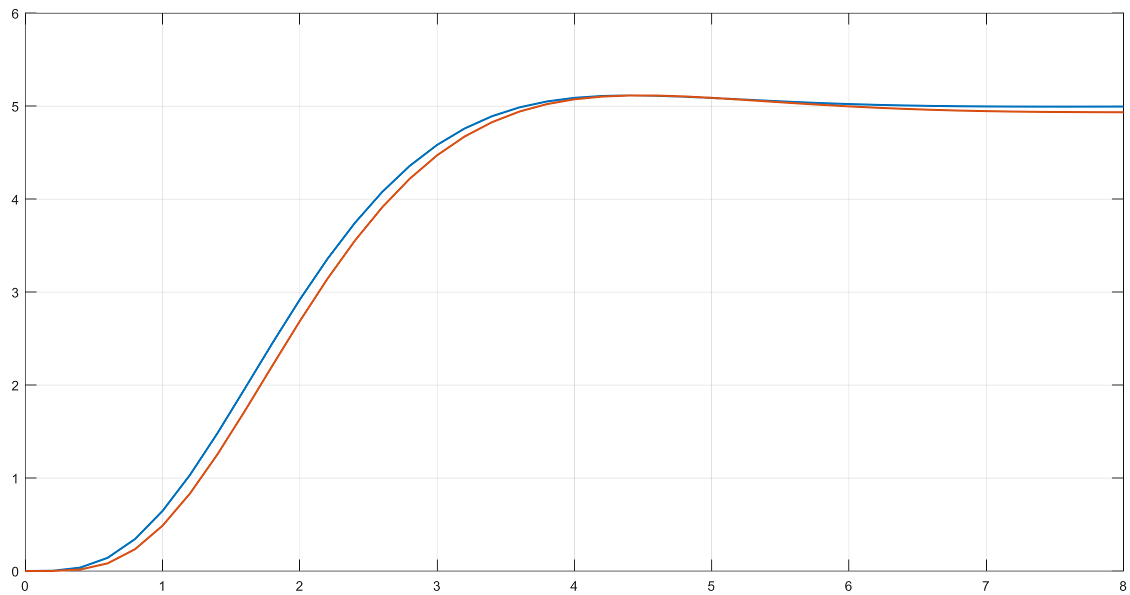


Figure 29

VIII.Question 8

Since the Bode diagram is having a 40db magnitude at 0.01 frequency, the system gain is derived as following.

$$20\log(k) = 40 \Longrightarrow k = 10^2 = 100$$

Also, the magnitude diagram is having a -20db/decade slope meaning that the system is of type 1. So up to here, the transfer function is something like

$$G(s) = \frac{k(\,something\,)}{s\,(\,somthing\,)}$$

Going on with the magnitude diagram, the slope is increased to -40db/decade, meaning that there is a minimum-phase pole at that very frequency which is 1. Surprisingly, the phase diagram is decreasing 270 degrees at the recalled frequency which can only happen if there are three either minimum-phase pole or non-minimum-phase zero. Since the magnitude slope is changing only -20db/decade, it could be concluded that there are two minimum-phase poles and one non-minimum-phase zero at ω=1. Thus, we have:

$$G(s) = 100 \times \frac{(s-1)}{s\,(s+1)(s+1)}$$

The Bode diagram of the transfer function is depicted in Figure.30.

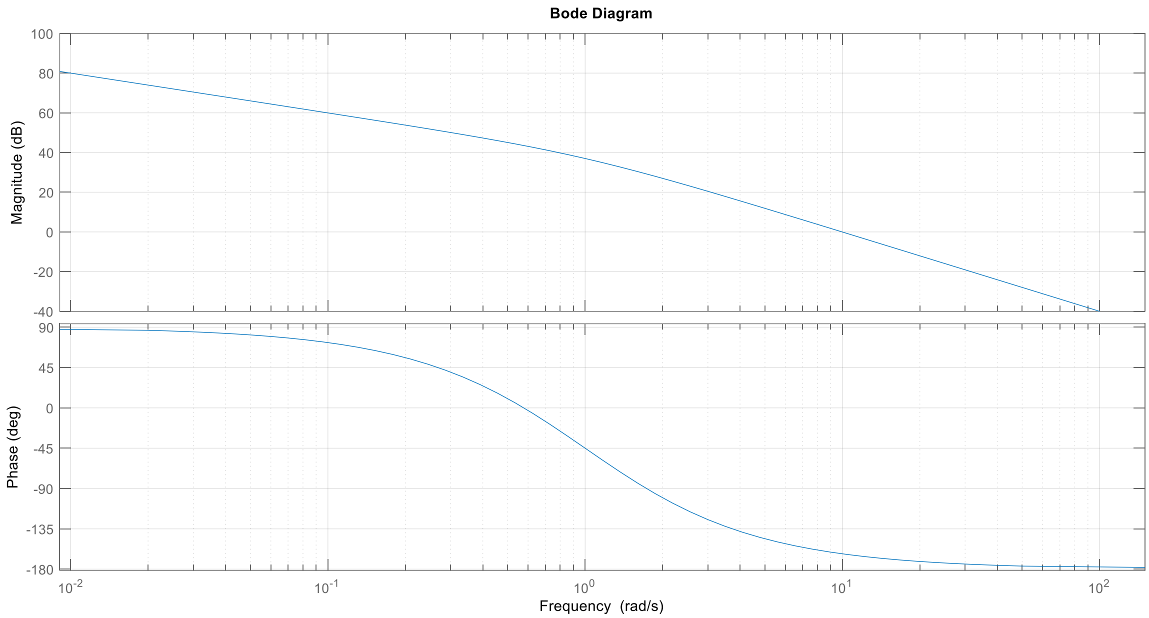


Figure 30

Note that the magnitude value of the diagram at ω=0.01 is 40db in the given actual Bode diagram while our estimated model is 80db at that very frequency. With taking some trial and error steps, the closest gain value (*K*) that matched the most to the starting value of the given magnitude plot appears to happen at almost *K*=1.