EE4307 Control System Design and Simulation

AY2021/2022 Semester 2

Continuous Assignment

Philippe Andrin Brigger
A0248834U

Contents

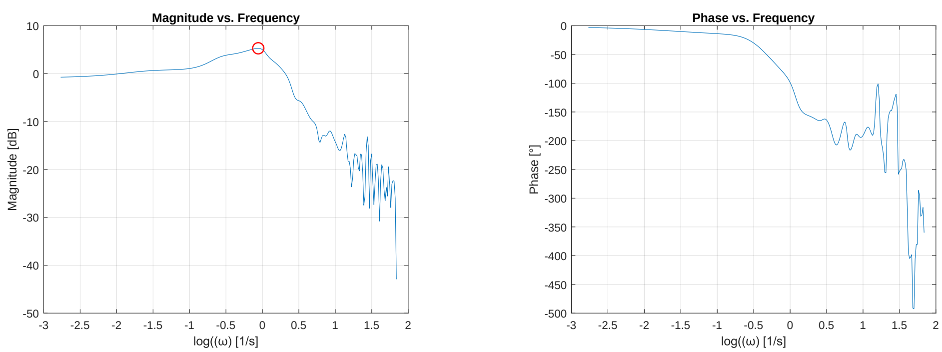
1 Introduction

The idea of the assignment was to directly apply the concepts which were learned during the lectures. This active learning approach was very useful to help understand how the concepts can be used in practice. This report shows and briefly discusses the results of the assignment.

2 Class Activities

2.1 A. Model Order Estimation

The model order of a system can be estimated by plotting the magnitude and phase frequency response. The number of peaks in the magnitude frequency response indicates what the order of the system may be. Each peak suggests an increase in the model order of two. To estimate the data, the data set *"ee4307_freq.mat"* was used.



(a) System magnitude vs. frequency response.

(b) System phase vs. frequency response.

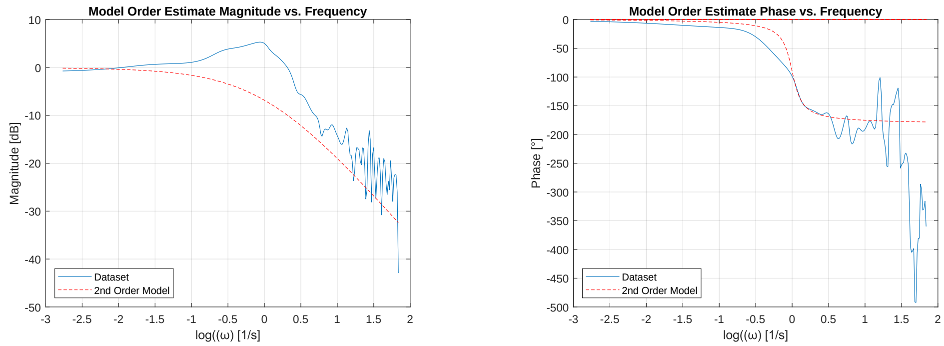
Figure 1: A model order estimate can be found by plotting the magnitude and phase frequency response of recorded data. The red circle indicates the only peak that was identified.

Since there was only one peak as seen in fig. 1, it was assumed that the system is of second order.
As fig. 2 shows, the system response is very similar to the frequency response of a second order reference model. Here, the second order reference model $H(s)$ had a natural frequency of $\omega_n = 1rad/s$ and a damping coefficient of $\zeta_m = 0.1$.

$$H(s) = \frac{1}{s^2 + 2\zeta_m \omega_m^2 s + \omega_m^2}$$

2.2 B. Delay Estimation

The delay of a system can be estimated by either plotting the impulse response, by plotting the cross correlation R_{yu} between the output data y and the input data u or by using the system identification toolbox function *delayest*.
Figure 3 suggests that the system doesn't have a large delay.
Figure 4 suggests that the system either has a delay of three or four samples. Running the command $delay = delayest(iddata(y, u, T))$ estimated that the delay of the system is two.



(a) System magnitude vs. frequency response.

(b) System phase vs. frequency response.

Figure 2: An arbitrary second order reference model has a similar magnitude and phase response as the original system from the dataset.

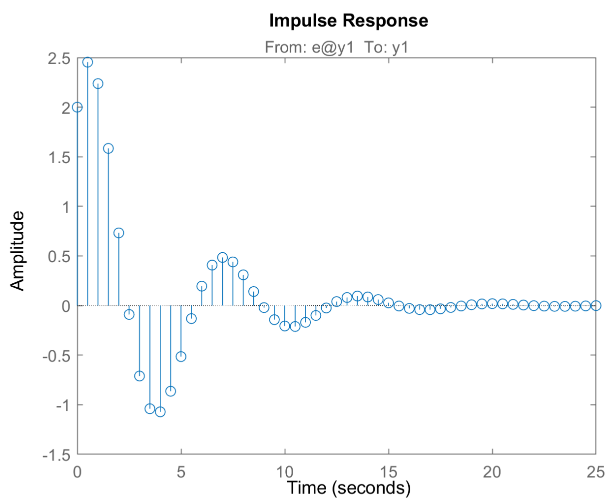


Figure 3: Delay estimation with impulse response from output data and sampling time.

2.3 C. Model Identification

The four model identification methods *arx*, *armax*, *bj*, and *oe* were used to fit a model onto the actual data. For each method, the delay was estimated as three samples, the order of the numerator was one while the order of the denominator was two. All the other polynomials were initialized as order two polynomials.
As can be seen in fig. 5, the OE method matches the real dataset best. On the other hand, the ARX method does not provide satisfying results. BJ and ARMAX provide good results for the filtered data set too.
The parameters of the identified models are shown in the subsequent table. For all of the models a delay of three samples was assumed.

Polynomial Coefficients					
	$A(q^{-1})$	$B(q^{-1})$	$C(q^{-1})$	$D(q^{-1})$	$F(q^{-1})$
ARX	[1, -0.678, 0.002]	0.128	-	-	-
ARMAX	[1, -1.509, 0.754]	0.159	[1, -1.079, 0.386]	-	-
BJ	-	0.169	[1, -1.211, 0.354]	[1, -1.618, 0.748]	[1, -1.418, 0.698]
OE	[1, -1.429, 0.725]	0.226	-	-	-

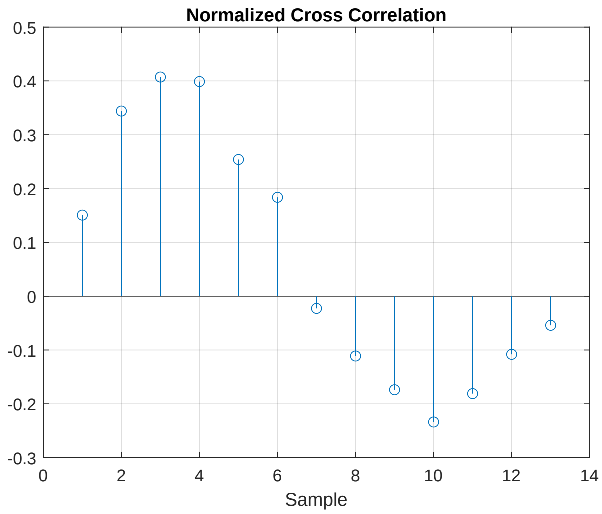
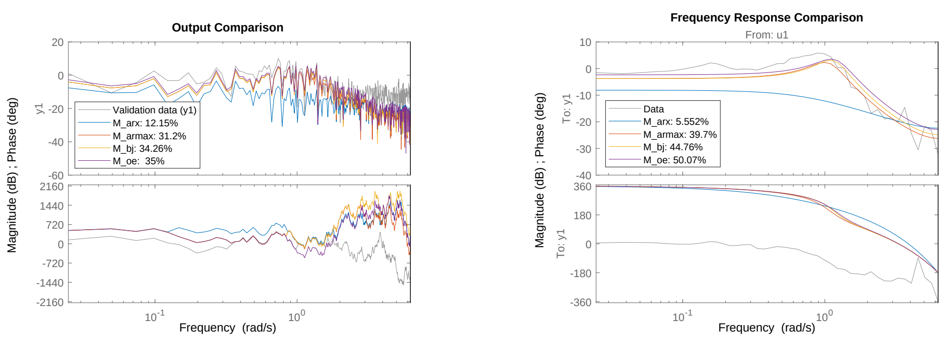


Figure 4: Normalized cross correlation between the input and output data.



(a) Noisy frequency response of identified models with different model identification methods.

(b) Smooth frequency response of identified models with different model identification methods.

Figure 5: Different model identification methods produce different models. The model identification method strongly influences how well the actual data is matched.

2.4 D. Model Validation

To validate the models, the residual mean and variance between the model fit and the dataset can be calculated. Plotting this information in a histogram plot shows how well a model fits the actual data.
The residual correlation plots show that the BJ and $ARMAX$ model fits have the least correlation between input and output data points of different time steps which is also what is desired from a model fit.
The histogram plots in fig. 7 show that the ARX model has a large variance. This is undesired as it means that there are many data points where the error between the fitted model output and actual data is large. Among the other three models, BJ and $ARMAX$ have the smallest variance and the errors are all close to zero.

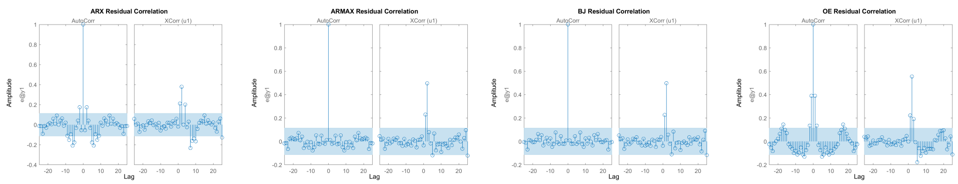


Figure 6: Residual correlations of different model identification methods.

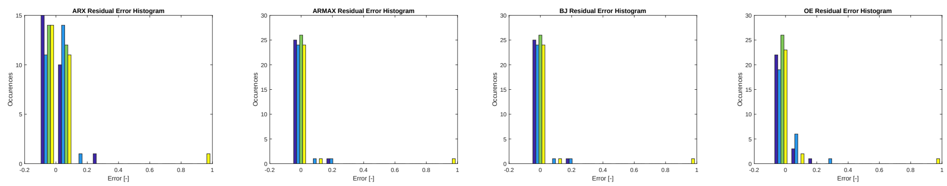


Figure 7: Histograms of the different model identification methods. The histograms give an impression of what the residual mean and variance of a model is.

2.5 E. Optimize ARX Model

An alternative approach to fit an ARX model to a dataset is to use the system identification toolbox command *struc*. Although this command is very handy, the model fit is usually of high order. This bears the risk of overfitting the dataset. This means that although the model fits the dataset very well, it will not perform well for different inputs and thus only represents the recorded dataset well but not the actual system. Figure 8 shows that the command fits a 10^{th} order system to the dataset versus the second order system which was manually identified.

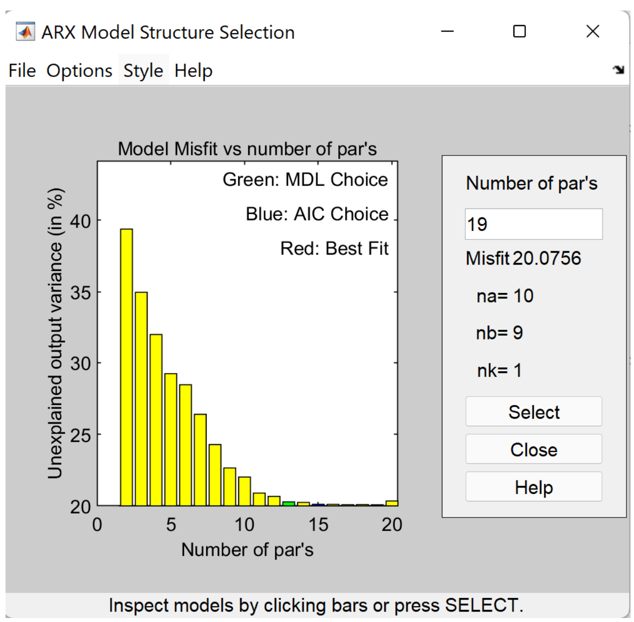


Figure 8: High order ARX model fit using *selstruc*

3 Conclusion

This assignment was a great way to apply the theoretical tools we learned in the lectures in practice. It became clear that system identification is as much an art as a science. Experience plays a major role when it comes to choosing the best model identification method as well as deciding which model order should be fit to the data. Software tools such as Matlab are very convenient and easy to use when it comes to fitting a model to a dataset. However, one can quickly get lost and forget what the goal of a certain identification step is. Using automated functions also bears the risk of overfitting a model. I appreciate the strong theoretical foundation we received in the lectures as this helped me understand what the outcome of the different identification steps should be and also what their implications are. I enjoyed working on this assignment.

A Appendix

Here is the code that was used to generate all the plots in this report.

A. Model Order Estimation

The model order of a system can be estimated by plotting the magnitude and phase frequency response. The number of peaks in the magnitude frequency response indicates what the order of the system may be.

```
% Clean up
close all; clc; clear all;

load("ee4307_freq.mat"); % Load data for model order estimation

log_w = log(w); % Take logarithm of frequency data

mag_dB = mag2db(mag); % Convert magnitude to dB

figure(1);
plot(log_w,mag_dB)
grid on
title('Magnitude vs. Frequency')
ylabel('Magnitude [dB]')
xlabel('log((ω) [1/s]')
hold on;
[max_val, index] = max(mag_dB);
plot(log_w(index),mag_dB(index),'or','MarkerSize',10,'LineWidth',1.1)
hold off;
saveas(gcf,'A_mag_freq.pdf')

figure(2);
plot(log_w,ph)
grid on
title('Phase vs. Frequency')
ylabel('Phase [°]')
xlabel('log((ω) [1/s]')
saveas(gcf,['A_phase_freq.pdf'])
```

As there is one peak in the magnitude vs. frequency response, it can be assumed that the system is a second order system. This can be verified by plotting the magnitude and phase frequency response of an ideal second order system.

```
% 2nd order reference model parameters
wm = 1; % Natural frequency
zetam = 0.1; % Damping coefficient
% Transfer Function
s = tf('s')
H = 1/(s^2+2*zetam*wm^2*s+wm^2)
figure(3)
bode(H)
Hw = mag2db(1./(w.^2+2*zetam*wm^2.*w+wm^2));
top =  (2.*w./wm*zetam);
bot = (1-(w./wm).^2);
Phi = zeros(length(w));
for i=1:length(Phi)
    Phi(i) = -atan2(top(i),bot(i))*180/pi;
end

figure(1);
plot(log_w,mag_dB)
grid on
hold on;
[max_val, index] = max(mag_dB);
plot(log_w,Hw,'--r')
title('Model Order Estimate Magnitude vs. Frequency')
ylabel('Magnitude [dB]')
xlabel('log((ω) [1/s]')
legend('Dataset','2nd Order Model','Location','southwest')
hold off;
saveas(gcf,'A_order_estimate_mag_freq.pdf')
figure(2);
plot(log_w,ph)
grid on;
hold on;
plot(log_w,Phi,'--r')
title('Model Order Estimate Phase vs. Frequency')
ylabel('Phase [°]')
xlabel('log((ω) [1/s]')
legend('Dataset','2nd Order Model','Location','southwest')
saveas(gcf,['A_order_estimate_phase_freq.pdf'])
```

B. Delay Estimation

Estimate delay with impulse response, cross correlation of input and output data or system identification toolbox command delayest.

```
% Clean up
close all; clear all; clc;

% Estimate delay cross correlation of input and output data
load("ee4307_io.mat")

figure(1);
[c,lags] = xcorr(y,u,'normalized');
stem(lags(513:525),c(513:525));
title('Normalized Cross Correlation')
xlabel('Sample')
grid on
saveas(gcf,'u_y_cros_corr.pdf')
```

```
% Estimate delay with system identification toolbox command delayest
delay = delayest(iddata(y,u,T))

% Estimate delay with impulse response
load('ee4307_impulse.mat')
sys = n4sid(iddata(y_impulse,[],T))
impulse(sys)
```

C. Model Identification

The system identifiaction toolbox has four

```
% Clean up
clear all; clc; close all;
load("ee4307_io.mat")
Z = iddata(y,u,T);

% Identification models
% ARX
M_arx = arx(Z,'na',2,'nb',1,'nk',3);
[A_arx, B_arx, C_arx, D_arx, F_arx] = polydata(M_arx);

% ARMAX
M_armax = armax(Z,'na',2,'nb',1,'nc',2,'nk',3);
[A_armax, B_armax, C_armax, D_armax, F_armax] = polydata(M_armax)

% BJ
M_bj = bj(Z,'nb',1,'nc',2,'nd',2,'nf',2,'nk',3);
[A_bj, B_bj, C_bj, D_bj, F_bj] = polydata(M_bj);

% OE
M_oe= oe(Z,'nb',1,'nf',2,'nk',3)
[A_oe, B_oe, C_oe, D_oe, F_oe] = polydata(M_oe)

figure(1)
compare(fft(Z),M_arx,M_armax,M_bj,M_oe)
L = findobj(gcf,'type','legend');
L.Location = 'southwest';

figure(2)
compare(spafdr(Z),M_arx,M_armax,M_bj,M_oe)
L = findobj(gcf,'type','legend');
L.Location = 'southwest';
```

D. Model Validation

Compare residual mean and variances of different model identification methods to validate which method works best for data set.

```
[E_ARX, R1] = resid(M_arx, Z);
[E_ARMAX, R2] = resid(M_armax, Z);
[E_BJ, R3] = resid(M_bj, Z);
[E_OE, R4] = resid(M_oe, Z);

resid(M_arx, Z);
title('ARX Residual Correlation')
resid(M_armax, Z);
title('ARMAX Residual Correlation')
resid(M_bj, Z);
title('BJ Residual Correlation')
resid(M_oe, Z);
title('OE Residual Correlation')

hist(R1);
title('ARX Residual Error Histogram')
xlabel('Error [-]')
ylabel('Occurences')
hist(R2);
title('ARMAX Residual Error Histogram')
xlabel('Error [-]')
ylabel('Occurences')
hist(R3);
title('BJ Residual Error Histogram')
xlabel('Error [-]')
ylabel('Occurences')
hist(R4);
title('OE Residual Error Histogram')
xlabel('Error [-]')
ylabel('Occurences')
```

E. Optimize ARX model

```
load('ee4307_io.mat')
load('ee4307_val.mat')
Z_val = iddata(y_val,u_val,T)
Z = iddata(y,u,T)
NN = struc(1:10,1:10,1:5)
V = arxstruc(Z, Z_val, NN);
figure(1)
selstruc(V);
```