System Identification

227-0689-00L

Final examination: instructions

Due: 17:00 on Friday, December 23^{rd}, 2022

Overview

The final exam is composed of THREE problems. To solve these problems you need to write three separate MATLAB functions: one for Problem 1, one for Problem 2, and one for Problem 3. Each function will analyze the data associated with the corresponding problem.

This work must be done individually. You may not discuss these problems with anyone else.

Your functions will be submitted through Moodle. In the submission, you must also include a scan of the signed declaration of originality form. In the form, the title of work should be specified as `System Identification final exam` and the **scanned** document should be named as `HS2022_SysID_final_DO_LegiNumber.pdf`, where "LegiNumber" must be your legi-number, without any dash, slash, backslash, etc., e.g., `HS2022_SysID_final_DO_12345678.pdf`.

If you do not have a legi-number, before the due date of the exam, you should send an email to `sysid@ee.ethz.ch` and ask for a temporary number to be assigned.

The Moodle forums are closed during the final examination. If you encounter any problems with the examination, please send an email to `sysid@ee.ethz.ch` or Prof. Roy Smith.

Your submission must be submitted to Moodle through the link

`https://moodle-app2.let.ethz.ch/mod/assign/view.php?id=787736`

by the due date and time given above. The submission should contain exactly 4 separate files (3 MATLAB m-files and a PDF).

Your grade for the final will be evaluated based on your performance in the three problems weighted in the following way: 30% for Problem 1, 35% for Problem 2, and 35% for Problem 3.

Downloadable data

The required files for the individual parts will be provided through the Moodle platform under the "Final Matlab project" section. Here is the list of uploaded files:

File	Problem	Summary description
`HS2022_SysID_final_p1_GenerateData.p`	Problem 1	Generate the problem data
`HS2022_SysID_final_p1_12345678.m`	Problem 1	Solution template
`HS2022_SysID_final_p1_check.p`	Problem 1	Code compliance check function
`HS2022_SysID_final_p2_GenerateData.p`	Problem 2	Generate the problem data
`HS2022_SysID_final_p2_12345678.m`	Problem 2	Solution template
`HS2022_SysID_final_p2_check.p`	Problem 2	Code compliance check function
`HS2022_SysID_final_p3_GenerateData.p`	Problem 3	Generate the problem data
`HS2022_SysID_final_p3_12345678.m`	Problem 3	Solution template
`HS2022_SysID_final_p3_check.p`	Problem 3	Code compliance check function
`HS2022_SysID_final_DO_12345678.pdf`	-	Declaration of originality

MATLAB function requirements

Each problem has a solution template (named `HS2022_SysID_final_p?_12345678.m`). Replace the string `12345678` in the solution template with your legi-number (using only the digits) and use it as the starting point for your solution. You will also have to make the same change on the first line of the solution function. It is important that you do not change the name or the order of the output variables on the first line. You will see that the solution template calls a function called `HS2022_SysID_final_p?_GenerateData.p` to generate the data for your analyses. When running your solutions, please avoid using 'Run section' (or Ctrl + Enter). This is because your legi-number is extracted from the filename but 'Run section' doesn't preserve the filename.

Each problem also has a compliance check function (named `HS2022_SysID_final_p?_check.p`) which can be used to check if your functions meet the minimal requirements for submission. Running this function, with your legi-number as the input argument, will perform this check on the script matching the solution naming format in your current directory. **Do NOT submit any code that has not been successfully checked by this function.**

The description of the individual problems (given on the following pages) will specify the variables that must be returned by your functions. You will also be asked to answer certain questions and explain your results and the choices you are making. These explanations must be given in the command window of MATLAB, using the `disp` function. Explanations written in the comment of your functions will NOT be graded. Please keep your explanation concise (around 2-4 sentences).

You will be graded based on:

- The correctness and accuracy of the returned variables. We will compare your outputs with the correct outputs obtained for your individual experiment data using the methods specified in the task description.

- The quality, clarity, and correctness of your explanations in the command window and plots required by the task description (if any).

These two aspects will have approximately equal weights in your grade. We will NOT grade your code directly.

The following are general considerations for the behaviour of your code. They apply to each of the problems.

- You may only use functions from MATLAB and the Control Systems Toolbox for Problems 1 and 2. For Problem 3, you can also use functions from the Robust Control Toolbox. Functions from other toolboxes are not permitted. This is checked by the compliance check function.
- Your functions must perform the calculations requested on the data generated by the data generation function. If the solution to the problem requires that you make a choice of parameter, this may be hard-coded into your function. You should explain in the command window why and how you made any such choice.
- The script should run unattended. Do not use the commands `pause`, `clear`, `clc`, `clf`, or `close` within your solution functions. Do not define any global variables in your functions or subfunctions. If you skip one part of a problem, please keep the dummy variable as in the solution template.
- If you write any custom functions, include them in the solution file after the main function. This means that MATLAB treats them as subfunctions and calls them instead of any other functions in the path that have the same name.
- For any figures requested in the problem, use the command `figure(xxx)`, where `xxx` is specified in the problem, to ensure that all figures remain visible on the screen after your code terminates.
- Each MATLAB figure can only contain a single plot, except for Bode plots which can use the subplot function to display gain and phase separately.
- Specify all axis limits via the MATLAB command `axis`. Auto-scaling must be avoided as it works differently on different machines. If your plot uses a legend, use the `'Location'` flag in the `legend` command to make sure that relevant data are not covered.

Before submitting your functions, restart MATLAB and test them by running the compliance check function for each problem in a directory with no other files.

Problem 1 (Weight: 30%)

This problem is divided into three tasks which are described below:

1. Your first task is to identify the parameters of a linear system of the form

$$A(z)v(k) = B(z)w(k) + e(k), \tag{1}$$

where the polynomials $A(z), B(z)$ are given as

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} \quad B(z) = b_1 z^{-1} + b_2 z^{-2},$$

$w(k)$ is a zero-mean i.i.d. input and $e(k)$ is zero-mean noise such that

$$\mathbb{E}[e(k)e(\tau)] = \begin{cases} 0.3 & \text{if } k = \tau \\ 0.05 & \text{if } |k - \tau| \leq 3 \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

Use the data given in the vectors `p1_w`, `p1_v`, corresponding to signals w and v, respectively, to obtain an estimate of the parameters $\theta_1 = [a_1, a_2, b_1, b_2]^\top$ using an *asymptotically unbiased* and *minimum variance* linear estimation method.

Assume that all the signals have a value of 0 for $k \leq 0$.

(a) Explain the procedure used to obtain such an estimate in the command window. What is the **regression problem** that you are solving? How did you derive it? Clearly define all the symbols.

(b) Return the estimated parameters in the vector `p1_theta1` in the specified order.

[**Weight: 10%**]

2. Now, consider a second system of the form

$$y(k) = \frac{D(z)}{C(z)} u(k) + f(k), \tag{3}$$

where $C(z) = 1 + c_1 z^{-1} + c_2 z^{-2}$ and $D(z) = d_1 z^{-1} + d_2 z^{-2}$ and $f(k)$ is an exogenous signal. Consider the problem of identifying $C(z)$ and $D(z)$ using the Instrumental Variable (IV) method.

(a) Let $f(k) = \frac{1}{C(z)} v(k)$, where $v(k)$ is from eq. (1) and consider the following instruments

$$\zeta(k) = [y(k-3) \quad y(k-4) \quad u(k-1) \quad u(k-2)]. \tag{4}$$

State whether the instruments $\zeta(k)$ at time k in eq. (4) are uncorrelated with the noise $v(k)$ at time k, followed by the reasoning you used to determine it.

(b) Now take

$$f(k) = \varepsilon(k). \tag{5}$$

In this case, $\varepsilon(k)$ is a zero-mean i.i.d noise. Consider the three possible choices for the instruments:

- (Option 1) $\zeta(k) = [y(k-1) \quad y(k-2) \quad u(k-2) \quad u(k-3)]$,
- (Option 2) $\zeta(k) = [y(k-2) \quad y(k-3) \quad u(k-2) \quad u(k-3)]$,
- (Option 3) $\zeta(k) = [y(k-3) \quad y(k-4) \quad u(k-1) \quad u(k-2)]$.

Which one of these three instruments might allow one to obtain a consistent estimate of the parameters $\theta_2 = [c_1, c_2, d_1, d_2]^\top$ using the Instrumental Variable (IV) method? Justify your answer and return the option number of the selected instrument in the variable `p1_opt` (return i for Option i).

(c) Use the instruments vector you selected in part b) and the data provided in the vectors `p1_u`, `p1_y`, corresponding to the signals u and y, respectively, to obtain an estimate of the parameters $\theta_2 = [c_1, c_2, d_1, d_2]^\top$ using the Instrumental Variable (IV) approach. Return the parameters $\theta_2 = [c_1, c_2, d_1, d_2]^\top$ in the variable `p1_theta2`.

[**Weight: 15%**]

3. Generate **one** MATLAB figure (`figure(131)`) where you plot: 1) the measured output `p1_y` and 2) the output $\hat{y}(k)$ predicted by your estimates of $C(z)$, and $D(z)$, when the input signal is `p1_u`. Generate the signal $\hat{y}(k)$ using solely `p1_u` (i.e., $\hat{y}(k)$ is a multiple-steps ahead prediction). Assume that all the signals have a value of 0 for $k \leq 0$.

   (a) Return the predicted vector $\hat{y}(k)$ in the variable `p1_y_hat`.

[**Weight: 5%**]

Problem 2 (Weight: 35%)

Consider the problem of identifying a stable causal linear time-invariant system with the following pulse response model:

$$y(k) = \sum_{i=0}^{\infty} g(i)u(k-i) + v(k)$$

where $g(i)$ is the pulse response of the system, and $u(k), y(k), v(k)$ are the input, the output, and the measurement noise respectively. The measurement noise is independent and identically distributed with $\mathcal{N}(\mu, \sigma^2)$, $\mu = 0$, $\sigma = 0.25$.

Your task is to estimate the pulse response of the system. For this purpose, you are given the following experiment data:

- `p2_u`: The input data, $u(k)$, for identification.
- `p2_y`: The output data, $y(k)$, for identification.

You should assume that the system starts at rest, i.e., the input $u(k) = 0$ for $k < 0$.

Using the provided experiment data, you should do the following:

1. With finite experiment data, you can only estimate a finite truncation of the pulse response, denoted by $\mathbf{g} = [g(0)\ g(1)\ \ldots\ g(\tau_{\max})]^\top \in \mathbb{R}^{\tau_{\max}+1}$. This leads to a truncation error

   $$e(k) = \sum_{i=\tau_{\max}+1}^{\infty} g(i)u(k-i).$$

   You want to find the smallest $\tau_{\max}$ such that $\|e(k)\|_\infty \leq 0.01$. To do this, based on the experiment data, estimate an approximate model of the form

   $$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-1) + e(k),$$

   using the ARX approach. Then, estimate the truncation error using the pulse response of the estimated approximate transfer function model.

   Please return the following variables:

   - `p21_ab` (vector of dimension 3×1): values of the estimate $[a_1\ a_2\ b_1]^\top$.
   - `p21_tau_max` (positive scalar): the smallest $\tau_{\max}$ value such that $\|e(k)\|_\infty \leq 0.01$.

   Please explain the following in the command window:

   (a) The least squares problem to find `p21_ab` with symbols clearly defined.

   (b) How did you find `p21_tau_max` from the approximate transfer function model?

   [Weight: 12%]

2. Select $\tau_{\max}$ as `p21_tau_max` that you returned in Part 1. If you were unable to work out Part 1, use `p21_tau_max` = 30. Based on the experiment data, estimate the finite pulse response $\hat{\mathbf{g}}$ using the least-squares approach. Find the covariance matrix of the estimate.

   Please return the following variables:

   - `p22_ghat` (vector of dimension (`p21_tau_max` $+ 1) \times 1$): the pulse response estimate $\hat{\mathbf{g}}$.
   - `p22_covg` (matrix of dimension (`p21_tau_max`$+1) \times ($`p21_tau_max`$+1)$): the covariance matrix of the estimate `p22_ghat`.

   Please explain the following in the command window:

   (a) The least squares problem to find `p22_ghat` with symbols clearly defined.

   (b) The equation for computing the covariance matrix with symbols clearly defined.

   [**Weight: 15%**]

3. Upon closer inspection, it is found that the measurements with a value exactly equal to zero are due to a sensor malfunction and should not be used for identification. Exclude the effect of the corrupted measurements, and estimate the finite pulse response $\hat{\mathbf{g}}$ using the least-squares approach again.

   Please return the following variables:

   - `p23_ghat` (vector of dimension (`p21_tau_max` $+ 1) \times 1$): the pulse response estimate $\hat{\mathbf{g}}$ excluding the effect of the corrupted measurements.

   Please explain the following in the command window:

   (a) What data treatment(s) do you need to do to exclude the effect of the corrupted measurements? [**Weight: 8%**]

Problem 3 **(Weight 35%)**

Consider the following configuration where a plant $G(z)$ operates in a closed loop with a controller $C_0(z)$, where the error $e(k)$ is zero-mean and uncorrelated with the reference $r(k)$.

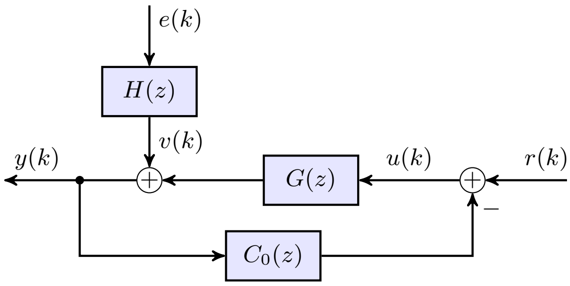

Figure 1: Structure of the control loop

The closed-loop transfer functions are given as

$$y = SGr + SHe$$
$$u = Sr - SC_0He,$$

where $S = \dfrac{1}{1 + C_0G}$ is the sensitivity function.

We are interested in the frequency response of the plant between $[\omega_l, \omega_u]$ rad/s, where the values of ω_l and ω_u are given as variables `p3_wl` and `p3_wu` respectively. The sampling time is $T_s = 1$ s.

You are given three sets of closed-loop experiment data with controllers in the form of $C_0(z) = \dfrac{a}{z-1}$:

- `p3_a` [dimension 3×1]: the a values used in the controllers. The i-th element corresponds to the i-th experiment.
- `p3_r` [dimension $N \times 3$]: the reference signal $r(k)$. The i-th column corresponds to the i-th experiment.
- `p3_y` [dimension $N \times 3$]: the output signal $y(k)$. The i-th column corresponds to the i-th experiment.

Assume that the system was at rest before data was collected, i.e., $y(k) = u(k) = r(k) = 0$ for $k < 0$.

Using the provided experiment data, you should do the following:

1. Using the ETFE method, estimate the frequency response from r to u for each of the three experiments, for the frequencies $\omega_k = 2k\pi/(NT_s)$, where $k = 1, 2, \ldots, N/2$. Which

experiment would obtain the smallest mean squared error in identifying the frequency response of the plant $G(z)$ at the frequency range $[\omega_l, \omega_u]$ rad/s?

Please return:

- p3_Tur (matrix of dimension $N/2 \times 3$): the estimated frequency response from r to u for frequencies ω_k in increasing order. The i-th column corresponds to the i-th experiment.
- p3_ID (scalar in $\{1, 2, 3\}$): index of the best experiment for the frequency range $[\omega_l, \omega_u]$ rad/s.

Please answer:

(a) Explain briefly the steps to estimate p3_Tur.

(b) Are your estimates asymptotically unbiased? Explain your reasoning.

(c) Plot the magnitudes of the three estimated transfer functions p3_Tur (in dB) with respect to ω_k (in rad/s and a logarithmic scale), in one figure (figure(31)).

(d) How did you choose the best experiment that results into the smallest mean squared error in identifying $G(z)$ in the frequency range $[w_l, w_u]$ rad/s?

[**Weight: 15%**]

2. In this task, you are required to perform closed-loop identification using the Dual-Youla method, using ONLY the best experiment you selected in part 1. If you didn't work out p3_ID, please use the first experiment.

   The factorization of the controller C_0 and the initial plant model P_0 are given by $C_0 = \frac{X_0}{Y_0}$, and $P_0 = \frac{N_0}{D_0}$, where X_0, Y_0 are the right normalized coprime factorization of C_0 generated by the rncf command in MATLAB, and $N_0 = \frac{1-0.2}{z-0.2}$, $D_0 = 1$.

   Please follow the following steps to find an estimate $\hat{G}(z)$ of the plant dynamics.

   (a) Formulate an equivalent open-loop identification problem in the form of $\beta = R\alpha + Fe$.

   (b) Estimate the finite pulse response of $\hat{R}$ using the least-squares approach with $\tau_{\max} = 20$.

   (c) Using the estimated Youla parameter $\hat{R}$, find the frequency response estimate $\hat{G}(e^{j\omega})$ of the plant, for the frequencies $\omega_k = 2k\pi/(NT_s)$, where $k = 1, 2, \ldots, N/2$.
   Hint: You may use $\hat{R}(z) = \sum_{k=0}^{\tau_{\max}} \hat{r}[k]z^{-k}$, where $\hat{r}[k]$ for $k \in \{0, 1, \ldots, \tau_{\max}\}$ are finite pulse response estimates of $\hat{R}$ computed above.

   Please return:

   - p3_beta (vector of length N): the filtered measurement data $\beta(k)$.
   - p3_alpha (vector of length N): the filtered excitation data $\alpha(k)$.
   - p3_Rhat (vector of length $\tau_{\max} + 1$): the estimated finite pulse response of $\hat{R}$.
   - p3_Ghat (vector of length $N/2$): the estimated frequency response $\hat{G}(e^{j\omega})$ for frequencies ω_k in increasing order.

   Please answer:

   (a) How did you construct the filtered data p3_alpha and p3_beta?

(b) Is the estimate `p3_Ghat` asymptotically unbiased? Explain your reasoning.

(c) Please determine if the following statements are correct. Explain your reasoning.

   i. $\hat{G}$ is not necessarily stabilized by the controller C_0 because the above Dual-Youla method does not enforce this condition.

   ii. The estimate $\hat{R}$ is unstable as it is a finite pulse response estimate of the Youla parameter R.

   iii. The above estimate $\hat{G}$ is stabilized by the controller C_0 because the estimate $\hat{R}$ of the Youla parameter R is stable.

[**Weight: 20%**]