Assignment 2

Subject: Open loop, closed loop and multi-variable systems
Deadline: March 23, 2010, 08:45h

Part I: frequency domain identification

Introduction

An unknown system can be estimated in frequency domain (frequency response function, FRF). This part of the assignment focuses on the difference between open loop and closed loop situations. Estimated transfer functions of open loop and closed loop systems will be compared with the real transfer function in Bode diagrams.

Assignment 2.1: Open loop systems

a) Open and inspect the Simulink model `SISO_open_loop.mdl`. Give the formula of a spectral estimator of the frequency response function of the *unknown* system H and give the estimator of the coherence between input and output signals.
b) Inspect `si_openloop.m` and insert your estimators at the correct location. The m-file will now generate a Bode diagram, including coherence, of the estimated *unknown* system. Give the Bode diagram. What is presented in the Bode diagram?
c) Implement and calculate the coherence based on the raw (= not averaged) spectral densities. Plot the results and explain.
d) What is the result of averaging over multiple frequency bands? Change the number of frequency bands (variable `nbands`) over which the signal is averaged. What is the effect of averaging on the magnitude? (Hint: try a large number of frequency bands)
e) Again, calculate the coherence but now averaging over a number of frequency bands. What is the effect of averaging on the coherence? What are the differences with c)?
f) Change the variance of the noise signal (within the *unknown* system). What is the result on your estimates for the transfer function and coherence?
g) Assume the system in the model suffers from a lot of measurement noise. What would you advice to improve the estimates. You can think of adjusting the sampling frequency, observation time or by applying frequency averaging (see question d).

Assignment 2.2: Closed loop systems

Use the model `SISO_closed_loop.mdl` and the program `si_closedloop.m`.

h) Open and inspect the Simulink model `SISO_closed_loop.mdl`. Give the formula of a spectral estimator of *unknown* system H2.
i) Give the expression for the coherence between external input w(t) and output y(t). Is this a good measure for the linearity of the transfer function?
j) Implement your estimator and coherence function in `si_closedloop.m`. Identify H2 and give the Bode diagram and coherence of the estimates.
k) We will now check if alternative external inputs can also be used to estimate H2. Set the variance of input u(t) to 1 (and w(t) to 0). Give the expression for the spectral estimator of *unknown* system H2 using signal u(t) and determine the coherence between u(t) and y(t). Compare the estimate and coherence to the results of i).
l) Change transfer function H2 into a non-linear transfer function. Give your non linear function. Estimate the transfer function and the coherence and explain the differences (using input w(t)).

Part II: Input signal design

Introduction

Virtually any identification problem suffers from noise. This assignment covers some of the strategies on optimizing the input signal to minimize the effect of noise. This is particularly important when there are limitations to the maximum input (or output) amplitude that the system to be identified can handle. The system H1 in `MISO_open_loop.mdl` breaks when it receives an absolute input value greater than 1.5. Note you will design only one input for the MISO system effectively reducing it to SISO for Part I of this assignment. The goal is to design an input signal with the highest possible power, given the limitations on the amplitude. In this part of the exercise we will compare four signal designs. Make sure to scale all signals to the maximum amplitude of 1.5 before analyzing them.

Input signal design:

Signal A: Filtered Gaussian white noise
Generate a Gaussian white noise signal of 50 seconds at 200 Hz. Filter this signals with a second order low-pass Butterworth filter with a cut-off frequency of 40 Hz to create signal A.
Signal B: Multisine signal
Multi-sine inputs improve signal-to-noise ratio by enabling us to concentrate the power in the input signal on specific frequencies to prevent leakage (see lecture notes) Generate a multi-sine signal (signal B) with equal power on all frequencies in the range [1/T, 40] Hz with random phase using msinprep(). See the programming example in the Assignment Guide.
Signal C: Crested multisine signal
Cresting a signal is shifting the phase of the sines with the aim of obtaining maximum power with minimal amplitude. Functions msinprep() and msinclip() enable us to easily generate and crest multi-sine signals. Generate a signal C which is a crested version of signal B, use msinprep() and msinclip().
Signal D: Output crested multisine signal
The lower the amount of system output at a certain frequency, the higher the relative amount of noise (in case of white noise). This can be compensated for by shaping the input with the inverse gain of the system, such that the output of the system has now an approximately flat spectrum. Generate a crested multi-sine D. Scale the amplitude of the sines (before cresting) such that they are inversely proportional to the gain of the *unknown* system.

Assignment 2.3: Input signal design

m) Generate the four signals, simulate the system with each signal as input u1 to generate the output signals (set u2 to zeros). Calculate and plot the autospectral densities of the input and output signals. Include a table in your report stating for each signal (for both the input and the output): the maximum signal values, the standard deviation, the crest factor, average power between 0 and 40 Hz. Comment on the differences between the four signals.
n) How could you further improve the power in signal C? Explain the options and name the (dis-)advantages.
o) Use the four signals to estimate the FRF and the coherence (use averaging over 8 bands on all spectra). Show your estimates of the FRF and the true system in one Bode diagram, including the estimated coherence. Comment on the differences and similarities. Which fit is the best?
p) The course of the coherence using input signal D is different from the coherences using the other input signals. Explain the course of the coherence using signal D.

Part III: Multi-variable systems

Introduction

Until now, system identification techniques on single input/single output (SISO) systems have been discussed. This assignment covers open and closed loop multiple inputs/multiple outputs (MIMO) systems. In order to accurately estimate the separate contributions of multiple inputs in open loop MIMO systems, the inputs are generally designed to be uncorrelated. In this way input signals do not influence each other.
However, this technique alone is insufficient in closed loop MIMO systems since feedback pathways relate the inputs to each other. For this assignment the Simulink models `MISO_open_loop.mdl`, `MIMO_open_loop.mdl` and `MIMO_closed_loop.mdl` provide you with the input signals u(t) or z(t), and output signals y(t). Note the model requires inputsignals from Matlab's workspace. Generate 2 realizations of the crested multisine inputs (u1 and u2) as you generated in Part II, signal C.

Assignment 2.4: Coherence in MIMO systems

In multi-variable systems the ordinary, multiple and partial coherence can be determined as was explained in the lecture (see also De Vlugt et al. 2003a). `MISO_open_loop.mdl` presents a multiple inputs/single output (MISO) system: the output is the addition of the outputs of two initially linear identical systems with different inputs. Run the model for 50 seconds at 200 Hz and average the frequencies over 8 bands. Use the lecture sheets for reference to equations.

q) Calculate the ordinary coherence functions γ^2_{u1y} and γ^2_{u2y}. Explain the results.

r) Without the correction terms the transfer function $\hat{H}_{z1y}$ simplifies to $\hat{H}_{z1y} = \frac{S_{z1y}}{S_{z1z1}}$. Write the multiple coherence γ^2_{uy} as a summation of the ordinary coherence functions by substituting H_{z1y} and H_{z2y} by the simplified spectral estimators. Under what assumption is this allowed? What is compensated for by the correction term?

s) Calculate the multiple coherence both with the correction terms and without the correction terms (as in b). Explain the differences.

t) Nonlinearities and noise in the system decrease coherence. Now, implement a non-linearity by setting the lower limit of the dead zone of system H1 to -1.0 and the upper to 1.0, and the dead zone of system H2 to -0.1 and 0.1. Calculate the ordinary coherence functions and compare your results to a). How is this possible?

u) Calculate the multiple coherence (including the correction terms). Is the coherence lower? Can you determine from this coherence where the nonlinearities originated?

v) Calculate the partial coherences $\gamma_{u1y\cdot u2}$ and $\gamma_{u2y\cdot u1}$ (including the correction terms). Interpret their meaning. Can you determine where the nonlinearities originated?

Assignment 2.5: Open loop MIMO system

Use the Simulink model `MIMO_openloop.mdl` with the same inputs as before. Run the model for 50 seconds.

w) Identify the subsystems H11, H12, H21 and H22 (see lecture notes) and determine the multiple coherences. Plot the results.

x) Increase the noise in the system. Discuss the effect.

Assignment 2.6: Closed loop MIMO system

Use the Simulink model `MIMO_closedloop.mdl` with the same inputs as before. Run the model for 50 seconds.

y) Identify the subsystems H11, H12, H21 and H22 (see lecture notes) and determine the multiple coherences. Plot the results.

z) Increase the noise in the system. Discuss the effect.