Assignment 1

Subject: Introduction to SYSID
Deadline: March 2, 2010, 08:45h
Note: Check the *Assignment Guide* on Blackboard for the demands on your report: brief answers with Matlab-code in the appendix (do not hand in m-files).

Introduction

Most signal processing techniques have started in the continuous domain. Due to more and more common use of computers, these techniques were inevitably transformed to the discrete domain; in computers no continuous signals exist, only arrays of numbers. The goal of this assignment is to obtain insight in the correct interpretation of signals in the discrete time and learn about techniques to assess the relation between them.

Assignment 1.1: Covariance functions

Use the simulink model 'LinearWhiteNoise' to obtain an input signal *u(t)* and output signal *y(t)*.

a) Calculate the co-variance function $C_{uu}(\tau)$, $C_{uy}(\tau)$, $C_{yu}(\tau)$ and $C_{yy}(\tau)$ using the Matlab function 'xcov' (Use the 'biased' option: C_{uy}=xcov(y,u,'biased')).
b) Plot and analyze the results. Which variables should be at the axes? What is the effect of parameter MAXLAG in the function of xcov? What is the difference between $C_{uy}(\tau)$ and $C_{yu}(\tau)$?
c) What is the theoretical value of the auto co-variance function $C_{uu}(\tau)$ for all τ? Note that *u(t)* is white noise. What would be the theoretical value of the cross co-variance function $Cnu(\tau)$ between input *u(t)* and the noise? Note that *u(t)* and the noise are independent white noises.
d) Calculate the covariance $C_{uy}(\tau)$ with the 'unbiased' option. What is the effect of 'biased' and 'unbiased'?
e) Calculate from $C_{uu}(\tau)$ and $C_{uy}(\tau)$ the cross-correlation coefficients $r_{uu}(\tau)$ and $r_{uy}(\tau)$. For which τ is the maximal value obtained?
f) Write a Matlab program for the covariance function. What is the effect of the length of the vectors *u* and *y* for different τ?
g) Verify that the covariance figures obtained using your own Matlab program coincide with the previously calculated covariances using xcov. Create a 'biased' and an 'unbiased' version.

Assignment 1.2: Cross spectral densities obtained through the covariance function

System identification can be done in time and in frequency domain. In frequency domain, spectral analysis is used to estimate the transfer function of unknown systems. This part of the assignment will familiarize you with spectral analysis and the relation between the covariance function, the spectral densities and the systems transfer function. To allow verification of the applied techniques a known system is used as "unknown system" in this assignment:

$$G(\omega) = \frac{1}{0.0025s^2 + 0.015s + 1} \tag{1}$$

Build a Simulink model using the Simulink library to obtain input signal *u(t)* and output signal *y(t)*. Use a white noise signal (with variance one) as stochastic input *u(t)* to identify the system (Simulink => Sources => Random number block. Use Mean=0, Variance=1, choose a random Initial seed and Sample time=0.01.). Connect the stochastic input block to a transfer function block with transfer function *G(s)* (Simulink => Continuous => Transfer Fcn. Numerator=[1], Denominator=[0.0025 .015 1], Absolute tolerance=auto). Now add a second stochastic input (Use Mean=0, Variance=0.01, choose a different random Initial seed and Sample time=0.01.) to simulate noise and connect it to a summation point (Simulink => Math Operations => Sum) adding it to the output of the transfer function. Finally you need to include sinks. Sinks allow you to measure and export signals: use two To Workspace blocks to export input *u(t)* and output *y(t)* to Matlab's workspace. Run the model with a simulation time of 10 seconds (simulation stop time is set at the top of the screen).

h) Calculate the biased covariance functions $C_{uu}(\tau)$, $C_{yu}(\tau)$, $C_{uy}(\tau)$ and $C_{yy}(\tau)$.Use proper lag (τ) axes and plot all four covariance functions.
i) Use Matlab function `'impulse()'` to plot an impulse response of the system *G*. Plot a graph of $C_{uy}(\tau)$ in the same plot. Use the Simulink file to generate a signal *u(t)* and *y(t)* of 100 seconds long. Plot $C_{uy\_long}(\tau)$ in the same figure. Comment on the similarities and differences. Hint: zoom in on the time axis.
j) Use a discrete Fourier Transformation (matlab commands `FFT` and `FFTshift`) to transform the covariances $C_{uu}(\tau)$ and $C_{uy}(\tau)$ into the cross-spectral densities $S_{uu}(\omega)$ and $S_{uy}(\omega)$. Use both the 'biased' and 'unbiased' option. How can the results be interpreted?

Assignment 1.3: Cross-spectral density obtained through Fourier transformation of *u(t)* and *y(t)*

The cross-spectral densities $S_{uu}(\omega)$ and $S_{uy}(\omega)$ can also be directly obtained from Fourier transformed signals u(t) and y(t): U(ω) and Y(ω) respectively:

$$S_{uy}(\omega) = \frac{1}{N} U^*(\omega) \cdot Y(\omega) \qquad (2)$$

where *N* is the number of time samples.

Generate a signal *u(k)* that is the summation of a 5 Hz sine with amplitude 2 and a random signal with zero mean and variance one. The signal lasts for 8 seconds with a sample frequency f_s = 150 Hz. Develop a second order Butterworth filter with a cut-off frequency of 10 Hz. Apply a zero lag non-causal filter to *u(k)* in order to generate the filtered signal $u_{150}(k)$. Generate $u_{50}(k)$, $u_{30}(k)$ and $u_{10}(k)$ by taking every third, fifth and fifteenth sample of $u_{150}(k)$ respectively. This results in effective sample frequencies of respectively 50 Hz, 30 Hz and 10 Hz.
Continuous signals are denoted by *u(t)* and their Fourier transform as $U(\omega)$. Discrete time signals will be shown as *u(k)* and the Fourier transform as *U(n)*. The sample frequency is denoted by f_s, the total number of samples by *N*, observation time by *T* and the discrete time interval by Δt. Note that $T = N\Delta t$ and $\Delta t = \frac{1}{f_s} = \frac{T}{N}$.

k) Use a discrete Fourier Transformation (DFT) to transform $u_{150}(k)$ into $U_{150}(n)$. How should the coefficients be interpreted? Note that the results are complex vectors, so you can present the results as real and imaginary parts or as magnitudes and phases.

l) Determine a proper frequency axis and plot the result. Show the symmetric and anti-symmetric properties of the Fourier coefficients.

m) Apply the DFT to $u_{50}(k)$, $u_{30}(k)$ and $u_{10}(k)$. Generate a proper frequency axes for each signal and plot the results together with $u_{150}(k)$. Explain the effect of sample frequency f_s on the DFT, i.e. differences in magnitude and frequency characteristics.

n) Take the first 4 seconds of $u_{150}(k)$. Apply the DFT to this signal and plot the results together with $U_{150}(n)$. Explain the effect of observation time *T*.

Assignment 1.4: Spectral system estimation

In the frequency domain a transfer function can be estimated based on the input-output signals. The transfer function is estimated using spectral densities of the input and output signal of the system. Use the same system as in Part 1.2 to answer the following questions:

o) Given the input signal $u(t)$ is white noise what would the input spectrum $U(\omega)$ and the output spectrum $Y(\omega)$ theoretically look like?

p) Show that $S_{uy}(\omega) = G(\omega).S_{uu}(\omega)$.

q) A spectral estimator of $G(s)$ is given by Equation (3). Make a Bode diagram of your estimate of $G(s)$ using spectral densities as determined from covariance functions according to part 1.2. In the same figure, plot the Bode diagram of the “unknown system”. What is the effect of the `'biased'` and `'unbiased'` estimation?

r) Use the same spectral estimator of $G(s)$, only now use the spectral densities as determined from Fourier transformed signals according to part 1.3. Compare the results with q).

$$G(\omega) = \frac{S_{uy}(\omega)}{S_{uu}(\omega)} \tag{3}$$