Acoustic modem project

Session 2: Estimation and analysis of the acoustic channel

Alexander Bertrand, Paschalis Tsiaflakis and Hanne Deprez[1]

October 2017

Goal: Estimating the impulse response and frequency response of the acoustic channel.

Requirements: Matlab/Simulink in Windows, a sound card, a loudspeaker and a microphone.

Required files from DSP-CIS website: *recplay.mdl*

Required files from previous sessions: *initparams.m*, *analyze_rec.m*

Outcome: 2 m-files (+1 optional): *IR1.m*, *IR2.m*, *IR_bandstop.m* (optional) and 1 mat-file: *IRest.mat*

Deliverables: 1 m-file (+3 optional): *milestone1.m*, *IR_bandstop.m* (optional), *compute_shannon.m* (optional), *plot_shannon_vs_distance.m* (optional)

Important remarks: In all experiments in this session, use a sampling frequency $f_s = 16000$ Hz, if not specified otherwise. Make sure you set the Matlab path to your home-folder (and not on the local hard-drive, since this will be erased after you log off). If you use a stereo loudspeaker set, it is recommended to switch off one of the two loudspeakers. Try to optimize the dynamic range of loudspeakers and microphone, but **scale your signals to avoid clipping!**

1 Exercise 2-1: A first attempt to estimate the channel response

In this exercise a rough estimate is obtained of the impulse response of the acoustic channel used by the acoustic modem.

1. Create an m-file *IR1.m* that conducts a simple experiment to estimate the impulse response (IR) of the acoustic channel, by literally applying the definition of the IR (i.e., 'the response of the system when applying an impulse at the input'). The m-file should plot a figure with two subplots containing the estimated IR response (time-domain), and the (magnitude of the) frequency response. The time-domain scale must be in samples ('filter taps'), not seconds, and the frequency scale must be in Hz. The frequency response is plotted on a dB-scale (on the magnitude axis, not on the frequency axis). What do you observe?

[1]For questions and remarks: hanne.deprez@esat.kuleuven.be

2. How long is the estimated IR (approximately)?

3. If one would do the same experiment in a cathedral, with high-power audio equipment, and the distance between loudspeaker and microphone is larger than 20m, how would this IR change?

4. Is the acoustical environment the only factor that determines the IR? What else can have an influence?

5. *Optional*: First, use *IR1.m* to make a new estimate of the IR. Then, without moving the microphone, repeat the white-noise experiment from exercise 1-2. If you now convolve the transmitted white noise signal with the estimated IR (use the command `fftfilt`), this should yield an output signal with similar characteristics as the recorded signal (why?). Compare the spectrograms and PSDs of the recorded signal and the convolved signal. Do they look more or less the same?

2 Exercise 2-2: A robust channel response estimation

Our aim is now to obtain a better estimate of the IR of the channel.

1. How do the transmitted signal $u[k]$, the (unknown) channel IR-vector $\mathbf{h}$, and the recorded signal $y[k]$ relate to each other (assuming that noise can be neglected)? Give a matrix description of this relation (in the time domain). This yields an overdetermined system of linear equations, where the vector $\mathbf{h}$ contains the unknown variables. The data matrix in this system of equations will have a so-called Toeplitz structure to model a convolution.

2. Create an m-file *IR2.m* that estimates the IR $\mathbf{h}$ based on this matrix description, i.e., by solving the overdetermined system of equations in a least squares sense. What dimension should you choose for $\mathbf{h}$ (use the knowledge obtained in exercise 2-1)? Use white noise as input signal $u[k]$. Similar to *IR1.m*, the m-file *IR2.m* should make a plot of the estimated IR response (time-domain), and the (magnitude of the) frequency response. In addition, it should save the IR estimate $\mathbf{h}$ as a mat-file *IRest.mat*.

   Hint 1: Use the command `toeplitz`.

   Hint 2: Make sure that the samples of the input ($u[k]$) and output ($y[k]$) signal are (more or less) aligned in the matrix equation. Also make sure that the estimated IR $\mathbf{h}$ will be causal, i.e., the output samples cannot appear before the corresponding input samples! Therefore, since it is difficult to exactly align input and output samples, introduce a positive delay as a safety margin to avoid acausality. Note that a positive delay can be easily modeled in the vector $\mathbf{h}$, but a negative delay cannot be modeled.

3. Compare the time-domain IR and frequency response obtained with *IR2.m* with what you obtained with *IR1.m*. Do they resemble each other? Why (not)?

4. Repeat the white-noise experiment from exercise 1-2, and additionally estimate the IR using the same (simin/simout) signals (using your code in *IR2.m*). Can you observe a correspondence between the PSD of the recorded signal, and the estimated frequency response? Why (not)?

5. *Optional*: Predict what will happen to the IR if you use a stereo speaker setup. Verify experimentally.

6. *Optional*: Predict what will happen to the frequency response of the channel if you would put your hand against the loudspeaker. How will this change the shape of the frequency response of the channel, besides the obvious higher attenuation? Now do the experiment and check if you were right.

3 Exercise 2-3 *(Extra)*: Channel response estimation without full excitation

The following exercise will a) give you some experience with filter design in Matlab, and b) give more insight in *why* a white noise signal is used to estimate the IR in exercise 2-2.

1. Repeat the IR estimation procedure of exercise 2-2 a couple of times, without moving the microphone in between the experiments. Does the shape of the frequency response change a lot over the different experiments?

2. Create an m-file that generates a band-stop filtered white noise signal (a band-stop filter is the complement of a band-pass filter). The band-stop filter must attenuate the frequencies between 700 and 3000 Hz. The attenuation in the middle of the stop band must be *at least* 40 dB (use the `fir1` command).

3. Create an m-file *IR_bandstop.m* that does exactly the same as *IR2.m*, but it must use the band-stop filtered white noise signal instead of the original white noise signal.

4. Repeat the IR estimation procedure a couple of times (now using *IR_bandstop.m*), without moving the microphone inbetween the experiments. Is there a difference with the results from exercise 2-2? Does the shape of the frequency response now change a lot over the different experiments? Can you explain this?

4 Milestone demo

When you are ready with the above exercises, call the supervising TA, and show the following demo(s). E-mail the Matlab code of your demo(s) in one zip-package to

hanne.deprez@esat.kuleuven.be,

and add the group number and all the names of the group members. Do not send additional files (only files that are required to demonstrate the demo).

1. Demo 1: A single m-file named *milestone1.m* should conduct 3 experiments (automatically in a single run):

   - The white noise experiment of exercise 1-2 (week 1).
   - The estimation of the IR with *IR2.m* (use the simout from previous experiment without playing a new white noise sequence).
   - The estimation of the IR with *IR1.m*.

   This should be played and recorded live using the Simulink model *playrec.mdl* (i.e., do not use pre-recorded files or pre-estimated IR's). The m-file should end with showing four figures based on these experiments:

   - 2 subplots with spectrogram of transmitted white noise signal, and recorded signal (ex. 1-2).
   - 2 subplots with the PSD of transmitted white noise signal, and recorded signal (ex. 1-2).
   - Time and frequency response of the channel, based on *IR1.m* (ex. 2-1).
   - Time and frequency response of the channel, based on *IR2.m* (ex. 2-2).

   Use the correct scales on the axes, and give distinct titles to each figure. Make sure that the range of the Y-axis is the same in the different experiments to be able to compare the results.

2. Demo 2 (*Optional*): Demonstrate the results of exercise 1-3 by using *compute_shannon.m* to compute the channel capacity for a sampling frequency of 16000 Hz. If you have a plot of the distance-capacity dependency, show it using *plot_shannon_vs_distance.m*.

3. Demo 3 (*Optional*): Demonstrate the results of exercise 2-3, and try to explain the observations.