BIRZEIT UNIVERSITY
Electrical and Computer Engineering Department
ENCS4310, Digital Signal Processing | Project | Deadline 2/7/2023

Project submission deadline: 2/7/2023 23:55 PM on ITC
Project discussion: To be defined later
Project description:
Project title: **Line echo cancellation**
In communications over phone lines, a signal travelling from a far-end point to a near-end point is usually reflected at the near-end due to mismatches in circuitry (e.g., hybrid connections) The reflected signal travels back to the far-end point in the form of an echo. As a result, the speaker at the far-end receives, in addition to the desired signal from the near-end speaker, an attenuated replica of his own signal in the form of an echo. See Fig. 1

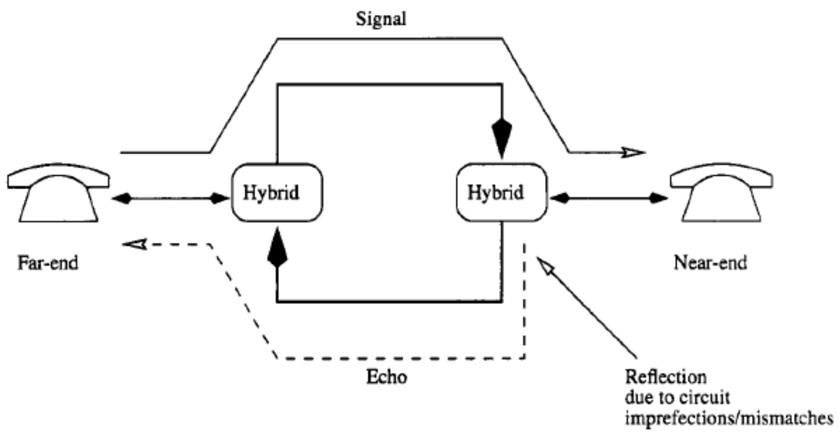

Fig. 1 The signal at the far-end is reflected at the near-end due to circuit mismatches and travels back to the far-end.

The echo interferes with the quality of the received signal. A common way to provide better voice quality at both ends is to employ adaptive line echo cancellers (LEC). At the near-end, for example, the signal feeding the LEC is the far-end signal while the reference signal is its reflected version - see Fig. 2

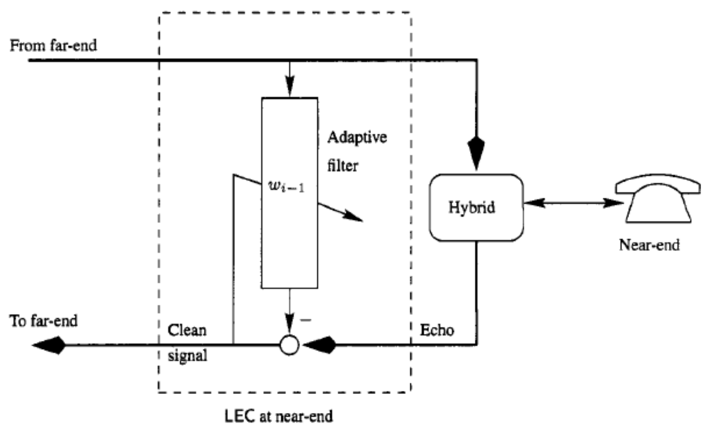

Fig. 2 An adaptive line echo canceller at the near-end.

In the figure, the output of the adaptive LEC generates a replica of the echo, and the error signal is therefore a "clean" signal that is transmitted to the far-end. The signals in this project are assumed to be sampled at 8 kHz.

A) Load the file path.mat, which contains the impulse response sequence of a typical echo path. Plot the impulse and frequency responses of the echo path.
B) Load the file css.mat, which contains 5600 samples of a composite source signal; it is a synthetic signal that emulates the properties of speech. Specifically, it contains segments of pause, segments of periodic excitation and segments with white-noise properties. Plot the samples of the CSS data, as well as their spectrum (Power Spectrum Density PSD).
C) Concatenate five such blocks and feed them into the echo path. Plot the resulting echo signal. Estimate the input and output powers in dB using

$$\hat{P} = 10\log_{10}\left(\frac{1}{N}\sum_{i=1}^{N}\left|signal(i)\right|^2\right)$$

where N denotes the length of the sequence. Evaluate the attenuation in dB that is introduced by the echo path as the signal travels through it; this attenuation is called the echo-return-loss (ERL).
D) Use 10 blocks of CSS data as far-end signal, and the corresponding output of the echo path as the echo signal. Choose an adaptive line echo canceller with 128 taps. Train the canceller by using as input data the far-end signal, i.e., x(i) = far-end(i), and as reference data the echo signal, i.e., d(i) = echo(i). Use ϵ-NLMS with $\epsilon = 10^{-6}$ and $\mu = 0.25$.
Plot the far-end signal, the echo, and the error signal provided by the adaptive filter. Plot also the echo path and its estimate by the adaptive filter at the end of the simulation.
E) Plot the amplitude and phase response for the estimated FIR channel at the end of the iterations. Compare it with the given FIR system (Path).
F) Propose a different appropriate Adaptive algorithm and compare it to the ϵ-NLMS

System Identification (Adaptive Algorithms)

The problem that arises is several applications is the identification of the system or, equivalently, finding its input-output response relationship. To succeed in finding the filter coefficients that represent a model of the unknown system, we set the system configuration as shown in Fig. 3.

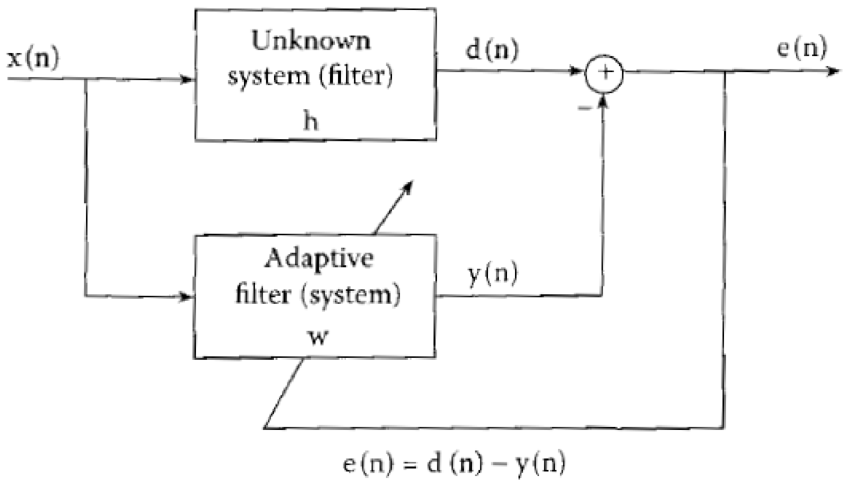

Fig. 3 System Identification

Assuming that the unknown system is time invariant, which indicate that the coefficients of its impulse response are constants and finite such that the desired response is given by

$$d[n] = \sum_{k=0}^{M-1} h[k]x[n-k]$$

The output of an adaptive FIR filter with the same number of coefficients, M, is given by

$$y[n] = \sum_{k=0}^{M-1} w[k]x[n-k]$$

For these two systems to be equal, the difference e[n]=d[n]-y[n] must be equal to zeros. It is the method of adaptive filtering that will enable us to learn the system coefficients and produce an error e[n], approximately equal zero.

Normalized Least mean squares ($\in$-NLMS) adaptive filter is an example of the well-known adaptive algorithms. The pseudo $\in$-NLMS adaptive algorithm is (Mth order FIR adaptive filter):

Inputs: M: filter length
μ:step-size factor
x(n): input data to the adaptive filter of length N (Vector)
w(0):initialization filter (vector) =zeros of length M

Outputs at each iteration (n) $y(n)=w^T(n)x(n)$
$e(n)=d(n)-y(n)$
$w(n+1)=w(n)+\dfrac{1}{\in+\|x\|^2}e(n)x(n)$, the updated filter coefficients , $\|x\|$ is the norm.

Project deliverables by each group:

1. Mini-report as described below in IEEE template. (you can use the Latex template Overleaf)
2. System demonstration of each part as described above.

You can use any programming language you prefer for implementing your project. However, we highly recommend MATLAB, Octave, or Python because they have many useful functions.

About the project: Teams of three students must do this project. The best arrangement is to choose a division of the project so that each of you can work on separate but interlocking parts. Teams of two or individual work will not be accepted.

Learning teamwork is also one of the more general goals of this course, so team projects will pick up points for demonstrating a successful ability to work with others.

The projects will be graded based on a project report (of around 3-4 pages) as well as in-class short presentations or discussion in the TA office.

Project submission must be via Ritaj only, but please use PDF format and not Word .DOC files if possible, since we often have formatting problems with Word files.

Your report must have the following structure, using these section headings and using IEEE paper format [will be attached]:

Introduction: A general description of the area of your project and why you are doing it.
Problem Specification: A clear technical description of the problem you are addressing.
Data: What are the real-world and/or synthetic signals you are going to use to develop and evaluate your work?
Evaluation Criteria: How are you going to measure how well your project performs? The best criteria are objective, quantitative, and discriminatory. You want to be able to demonstrate and measure improvements in your system.
Approach: A description of how you went about trying to solve the problem. Sometimes you can make a nice project by contrasting two or more different approaches.
Results and Analysis: What happened when you evaluated your system using the data and criteria introduced above? What were the principal shortfalls? (This may require you to choose or synthesize data that will reveal these shortcomings.) Your analysis of what happened is one of the most important opportunities to display your command of signal processing concepts.
Development: If possible, you will come up with ideas about how to improve the shortcomings identified in the previous section, and then implement and evaluate them. Did they, in fact, help? Were there unexpected side effects?
Conclusions: What did you learn from doing the project? What did you demonstrate about how to solve your problem?
References: Complete list of sources you used in completing your project, with explanations of what you got from each.

The reason for this somewhat arbitrary structure is simply to help you avoid some of the more problematic weaknesses we have seen in past years. If you are having trouble fitting your work into these sections, you should probably think more carefully about your project.