



Course material

Course:

Digital Signal Processing II

Video:

2.3.2.d Echo Cancellation

Concepts (extracted from automatically generated subtitles):

Return path of the signal. Adaptive echo cancellation. Adaptive filter. Error signal. Hands-free telephone conversation. Delay of capital m samples. Impulse response of the room. Low-pass filter. Input signal. Classic application of lms. Echo-free signal. Transfer function. N. s. Negative sign. Phase shift.



[to video sequence search](#)
(within Digital Signal Processing II.)

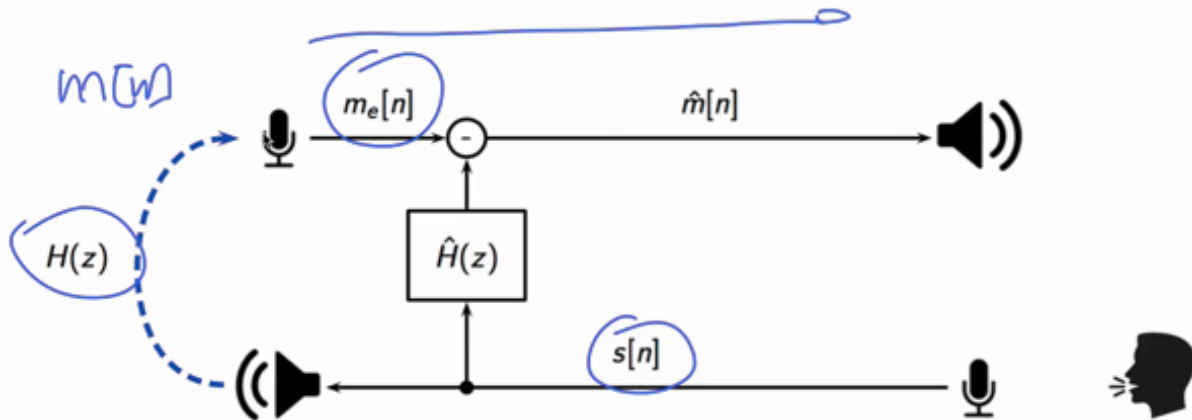


[to video](#)

Center for Digital Education. More educational support material here:

<https://exoset.epfl.ch/booc/>

Example: adaptive echo cancellation



113

A classic application of LMS is adaptive echo cancellation. This is something you use probably every day when you have a hands-free telephone conversation with a friend. Suppose your person A and your friend B here is talking over the microphone, their voice is picked up by a microphone and played over a loudspeaker. Now since you have a microphone too at your end, your friend's voice is picked up by your own microphone and sent back to your friend's loudspeaker. In the best of circumstances this only creates an annoying echo. In the worst case it gets picked up again by the microphone and if the gain is high enough this creates what is called a Larson effect, a high-pitched noise that obliterates all conversation. So in order to be able to talk hands-free over the phone you need a way to eliminate this feedback path. How is this done? Well if you could estimate the transfer function from the loudspeaker to the microphone at your end and simulate it with an adaptive filter then you could subtract the echo from the return path of the signal and transmit to your friend an echo-free signal. So the signal captured by the microphone here, $m_e[n]$, is a combination of what you are saying, let's call the signal $m[n]$, plus the echo which is the convolution of $s[n]$ with this transfer function which is the path between your loudspeaker and the microphone. Mathematically we have that $m_e[n]$ is

notes

summary

0m 0s



Signal captured by the microphone:

$$m_e[n] = \underbrace{m[n]} + \underbrace{h[n] * s[n]}$$

114

equal to M of N plus the convolution between H of N and S of N . S of N being the speech signal from your friend, H of N the impulse response of the room or the path from loudspeaker to microphone at your end. We need to estimate this

notes

summary

1m 33s



Signal captured by the microphone:

$$m_e[n] = m[n] + h[n] * s[n]$$

we need to estimate $h[n]$ in order to *subtract* the unwanted echo

114

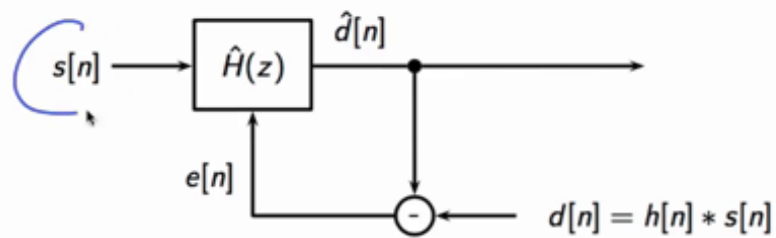
impulse response in order to subtract the unwanted echo from the return path. This estimation can be cast in the standard adaptive signal processing

notes

summary

1m 49s





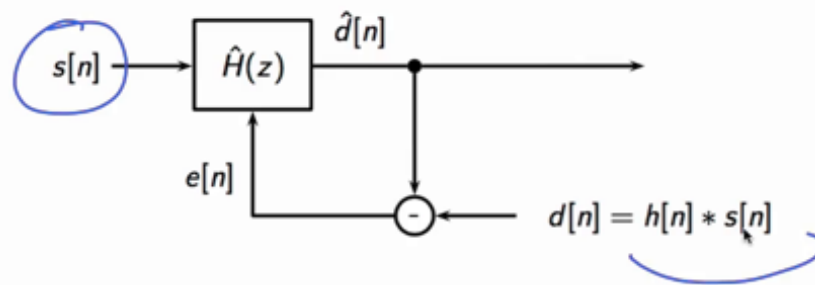
framework. Our input signal is your friend's voice S of N and our desired

notes

summary

1m 58s





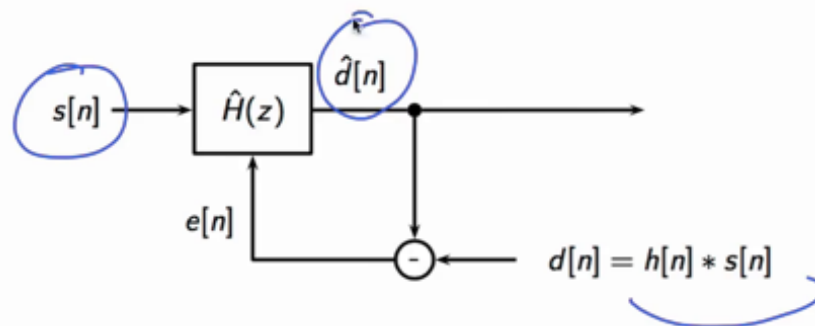
signal is the echo namely S of N convolved with the impulse response of the room at your end. So we want to estimate this echo so that we will be

notes

summary

2m 6s





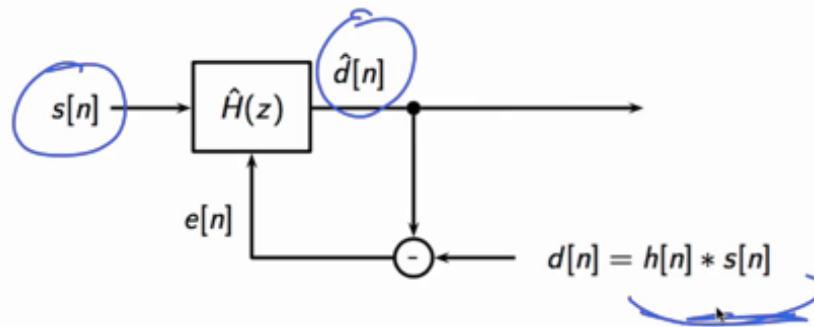
able to subtract it from the return path. So the error signal would be the difference between our estimation of the echo and the actual echo that we can

notes

summary

2m 13s





measure on the line. A potential difficulty is that what we're measuring

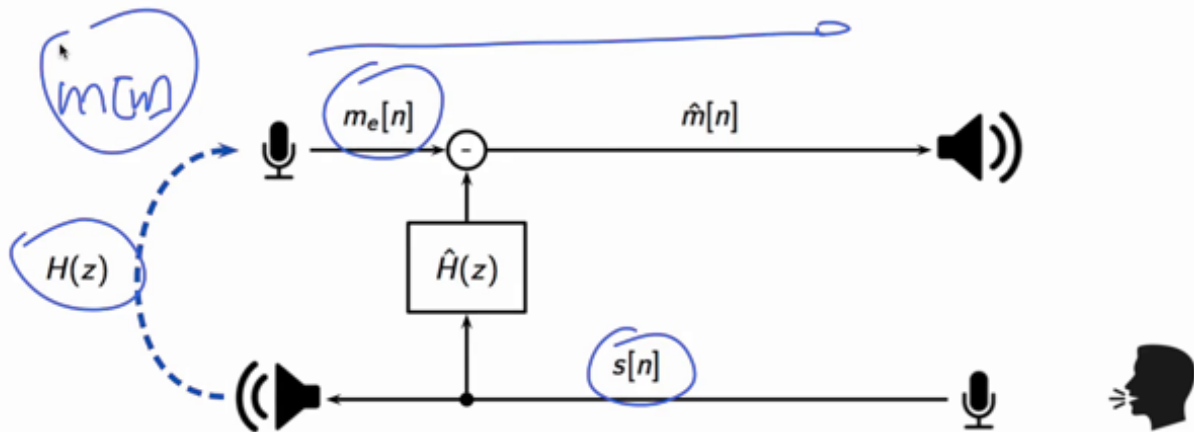
notes

summary

2m 25s



Example: adaptive echo cancellation



113

here at the line is not just the echo but the echo plus what we're saying at a certain point in time. But normally in a telephone conversation only one person

notes

summary

2m 29s



- ▶ "desired" signal is the echo (so we can subtract it)
- ▶ normally, only one person talks at a time: when B is speaking, $m_e[n] = h[n] * s[n]$
- ▶ people move, volume changes: $H(z)$ is time varying!
- ▶ use the LMS filter

talks at a time so at least for certain segments of the timeline we will have a signal that is just the echo as captured by the microphone. A second difficulty is that as people move in the room and the volume of the conversation changes the transfer function from loudspeaker to microphone will change as well. So H of Z is time varying and that's why we need to use the LMS filter to be able to track these changes in the statistics of the signal. As an example here is a

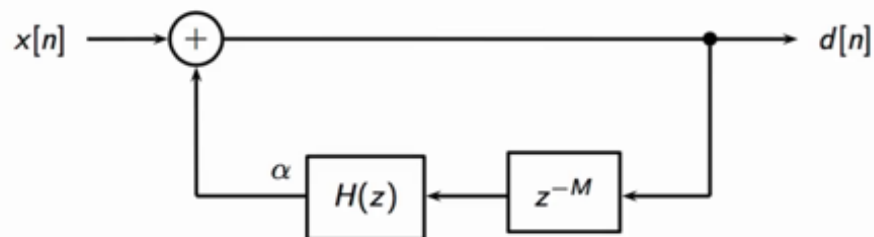
notes

summary

2m 38s



Example: simple echo model



$$H(z) = (1 - \lambda)/(1 - \lambda z^{-1})$$

117

simple block diagram that encodes a simple type of echo. You have a feedback

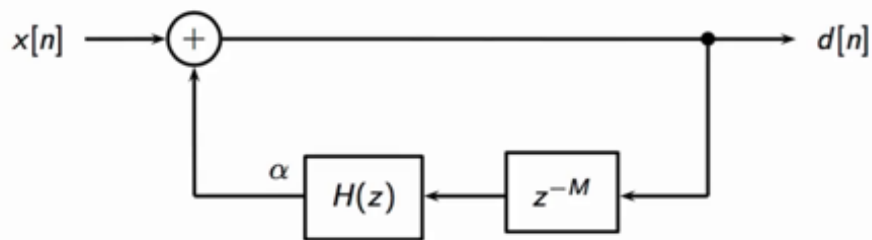
notes

summary

3m 10s



Example: simple echo model



$$H(z) = (1 - \lambda)/(1 - \lambda z^{-1})$$

117

path with a delay of capital M samples and every time the signal goes through this feedback path it gets filtered by a simple low-pass filter, in this case a

notes

summary

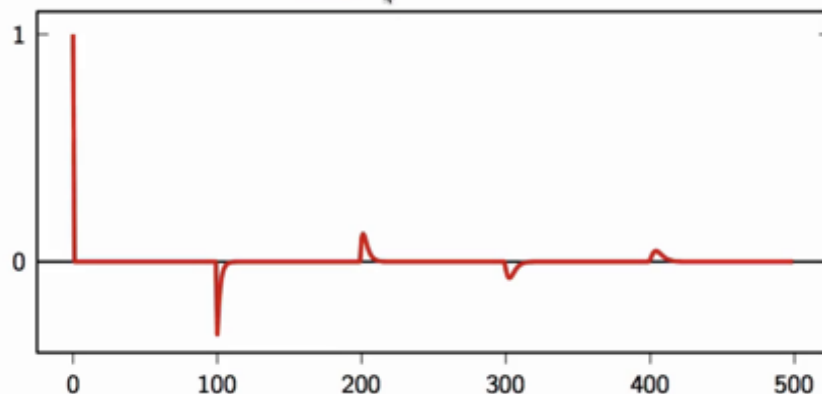
3m 13s



Echo impulse response



$$M = 100, \alpha = -0.8, \lambda = 0.6$$



118

leaky integrator, and attenuated by a constant alpha. So if we choose say the delay to be a hundred samples, alpha to be minus 0.8, the negative sign imposes a

notes

summary

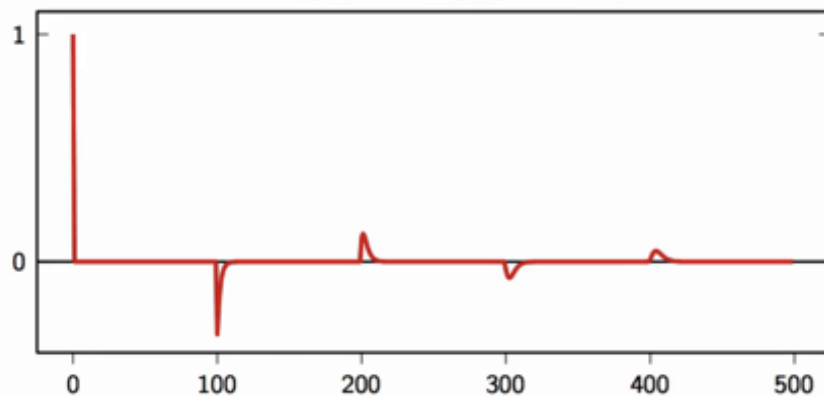
3m 25s



Echo impulse response



$$M = 100, \alpha = -0.8, \lambda = 0.6$$



118

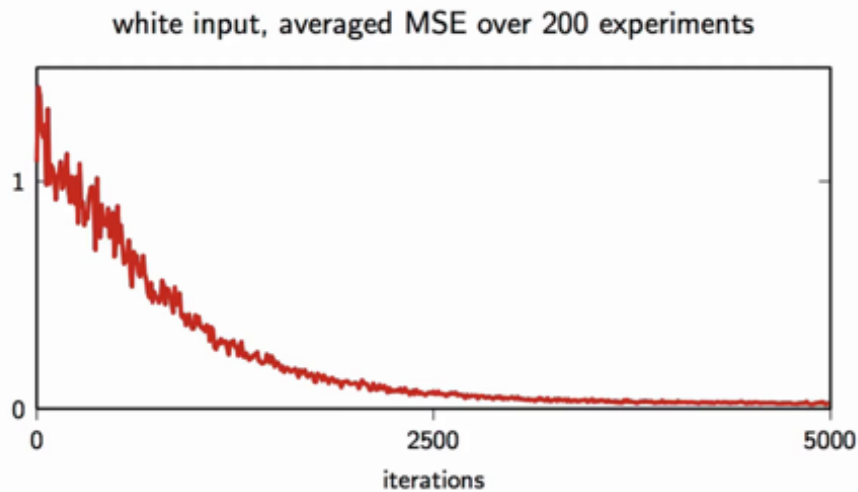
phase shift with every reflection and lambda equal to 0.6, we have an input response that looks like so. We use a white input to drive the LMS adaptation

notes

summary

3m 37s





and this is the mean square error averaged over 200 experiments. You see that the convergence is quite fast so in about 2,500 iterations we've already converged to a very low error. That means that the taps of the LMS filter have converged to the exact shape of the impulse response. Now the trick is if at one point we change something in the echo function and here we for instance

notes

summary

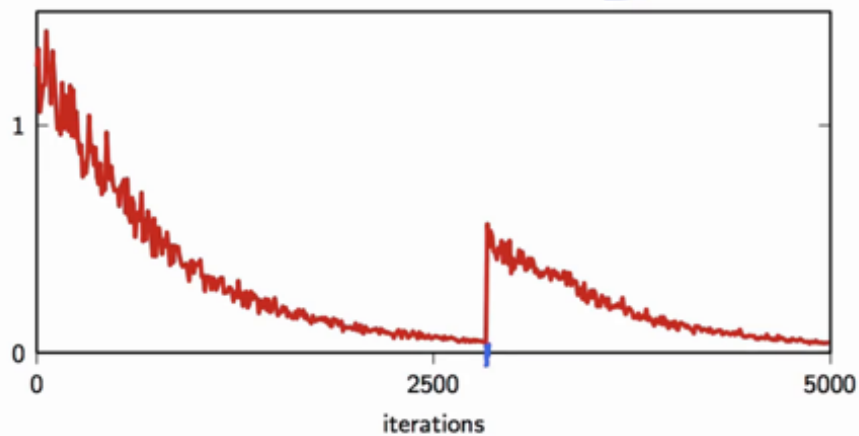
3m 49s



LMS can catch up with changes



echo delay changes from $M = 100$ to $M = 90$ at $n = 3000$



120

change the delay from a hundred samples to 90 samples, the adaptation can recover rather quickly. So at this point we change the delay of the echo. This could

notes

summary

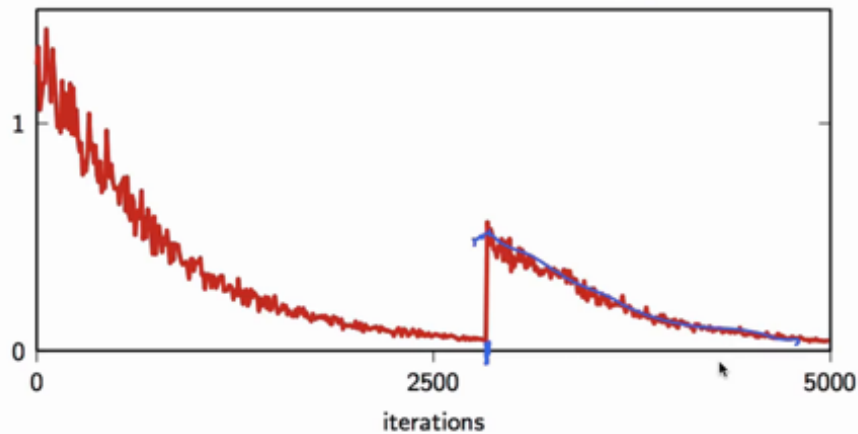
4m 19s



LMS can catch up with changes



echo delay changes from $M = 100$ to $M = 90$ at $n = 3000$



120

correspond to someone moving in the room and therefore changing the shortest reflection path from microphone to loudspeaker. At first the error jumps up of course because there's a mismatch between the taps of the filter and the actual impulse response of the room but the LMS filter is constantly adapting the filter taps and quickly the error goes down to zero again.

notes

summary

4m 30s

