

[Next](#)
[Up](#)
[Previous](#)
[Contents](#)

Next: [6 Adaptive Interference Canceling](#)
Up: [5 Echo Cancellation](#)
Previous: [1 Network Echo Cancelers](#)
[Contents](#)

2 Acoustic Echo Cancelers

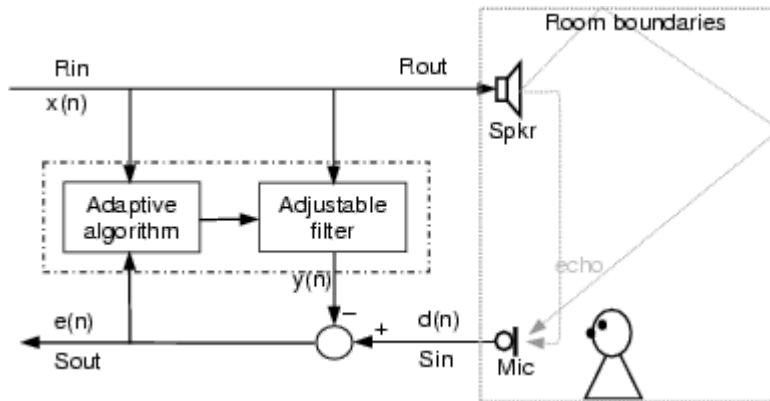


Figure 2.15: Block diagram of the acoustic echo canceler.

Acoustic echo cancelers are necessary in applications such as hands-free telephony, speakerphones, desktop communication, audio and video conferencing, voice command, desktop dictation, speech recognition, and many more. The problem addressed by the acoustic echo canceler in all those applications is illustrated in Fig. 2.15. In this figure, one side of a communication channel (of an audio conference for instance) is shown. The received speech or audio signal (R_{out}) is played through a loudspeaker so that all users in the local conference room can hear what the remote users say. The speech of the local users (S_{in}) is collected by one or more microphones and transmitted through the communication channel to the remote users. The problem in this setup is that the voice of the remote users played through the loudspeaker and its reflections off the room boundaries will also be collected by the microphones and transmitted back with the voice of the local user to the remote users who will hear their own voice delayed by the network and acoustic delay of the communication chain. The presence of the acoustic echo in the communication chain makes the users feel that they are being interrupted by their own echo, forcing them to stop speaking until the echo is faded away and the process is repeated over and over again. When the delay is large enough, which is usually the case in most communication applications especially mobile and voice over Internet protocol, this acoustic echo degrades the quality of the communication considerably. The acoustic echo problem mentioned above can be considerably reduced by an acoustic echo canceler as shown in Fig. 2.15. The main function of the AEC is to estimate the acoustic echo and subtract this estimate from the microphone signal. In the best case, when the estimate is accurate, this leads to eliminating the acoustic echo completely. This best case, however, is not easy to realize in practical echo cancelers. The AEC usually employs a transversal finite impulse response filter to estimate the acoustic echo. The FIR coefficients are adjusted using an adaptive algorithm to minimize the (S_{out}) signal, which is the adaptive filter error signal. The input (reference) signal for the adaptive filter is the far end speech (R_{in}) while its desired signal is the near end speech (S_{in}). After convergence, the adaptive filter coefficients will have an impulse response equals to the acoustic impulse response between the loudspeaker and microphone, including the reflections off the room boundaries. Since this acoustic impulse response can last several hundreds of milliseconds, acoustic echo cancelers usually employ adaptive FIR filters of several thousands coefficients. Adapting such a huge filter in real time is a real challenge, and therefore, advanced and efficient adaptive algorithms that reduce computational complexity and speed the convergence rate must be used. Examples of adaptive algorithms that have been successfully used in AEC are the Block Frequency Domain Adaptive Filter (see Section 4.2), and its partitioned version (see Section 4.12). The Normalized Least Mean Squares (see Section 4.11) and Leaky NLMS (4.7) have also been used in simple AEC applications when the acoustic impulse response is short. Several echo canceler application scripts are also included with ASPT such as `echo_nlms` (Section 10.16), `echo_leakynlms` (Section 10.15), `echo_bfdaf` (Section 10.14), and `echo_pbfdaf` (Section 10.17). Finally, it should be noted that a communication channel of a speaker phone for instance should employ both types of echo cancelers as shown in Fig. 2.16.

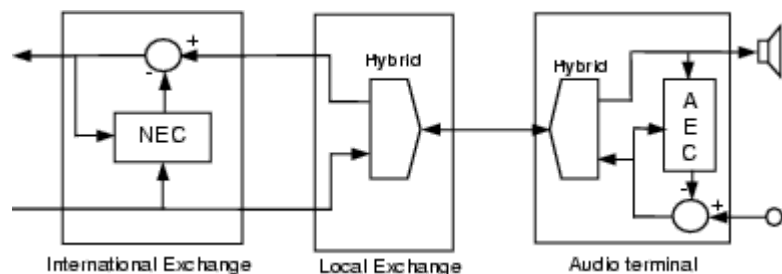


Figure 2.16: Block diagram of a communication channel employing both acoustic and network echo cancelers.

[Next](#)
[Up](#)
[Previous](#)
[Contents](#)

Next: [6 Adaptive Interference Canceling](#)
Up: [5 Echo Cancellation](#)
Previous: [1 Network Echo Cancelers](#)
[Contents](#)