ADAPTIVE FILTERING APPLICATIONS EXPLAINED

DESPITE THE DIVERSITY AND COMPLEXITY OF ADAPTIVE FILTERING, A SIMPLE CLASSIFICATION DOES EMERGES AND PRACTICAL APPLICATIONS ARE DEMONSTRATED.

INTRODUCTION

Although well-known and widely used, adaptive filtering applications are not easily understood, and their principles are not easily simplified. Currently, adaptive filtering is applied in such diverse fields as communications, radar, sonar, seismology, and biomedical engineering. Although these various applications are very different in nature, one common feature can be noted: an input vector and a desired response are used to compute an estimation error, which is used, in turn, to control the values of a set of adjustable filter coefficients. The adjustable coefficients may take the form of tap weights, reflection coefficients, or rotation parameters, depending on the filter structure employed.

Despite the diversity and complexity, a simple classification of adaptive filtering does emerge and practical applications can be demonstrated. This app. note begins by describing four basic classes of adaptive filtering applications and follows with sections that detail various fundamentals, techniques, and algorithms of several selected adaptive applications (refer to Table 1).

Classifying Adaptive Filtering Applications

Various applications of adaptive filtering differ in the manner in which the desired response is extracted. In this context, we may distinguish four basic classes of adaptive filtering applications (depicted in Figures 1 to 4, which follow):

- Identification
- Inverse Modeling
- Prediction
- Interference Canceling

Table 1. Adaptive Filtering Applications

Adaptive Filtering Class	Application
Identification	System Identification Layered Earth Modeling
Inverse Modeling	Predictive Deconvolution Adaptive Equalization
Prediction	Linear Predictive Coding Adaptive Differential PCM Auto-Regressive Spectrum Analysis Signal Detection
Interference Canceling	Adaptive Noise Canceling Echo Cancellation Radar Polarimetry Adaptive Beamforming

The following notations are used in Figures 1–4:

u = input applied to the adaptive filter

y = output of the adaptive filter

d = desired response

e = d -y = estimation error

The functions of the four basic classes of adaptive filtering applications follow.

I. Identification (Figure 1). The notion of a mathematical model is fundamental to sciences and engineering. In the class of applications dealing with identification, an adaptive filter is used to provide a linear model that represents the best fit to an unknown plant.The plant and the adaptive filter are driven by the same input. The plant output supplies the desired response for the adaptive filter. If the plant is dynamic in nature, the model will be time varying.

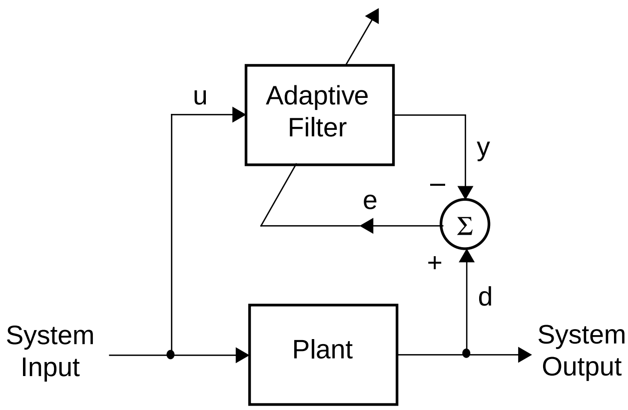


Figure 1. Identification

II. Inverse Modeling (Figure 2). In this second class of applications, the adaptive filter provides an inverse model representing the best fit to an unknown noisy plant. Ideally, the inverse model has a transfer function equal to the reciprocal of the plant's transfer function. A delayed version of the plant input constitutes the desired response for the adaptive filter. In some applications, the plant input is used without delay as the desired response.

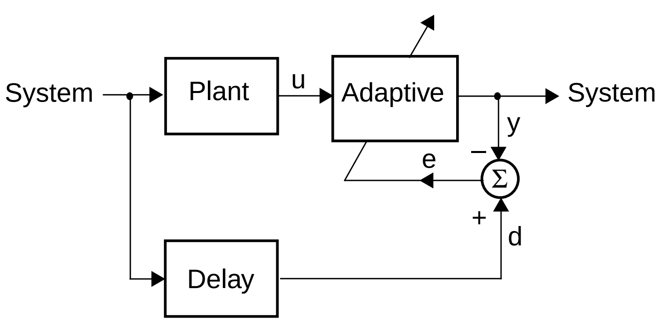


Figure 2. Inverse Modeling

III. Prediction (Figure 3). In this example, the adaptive filter provides the best prediction of the present value of a random signal. The present value of the signal serves the purpose of a desired response for the adaptive filter. Past values of the signal supply the input applied to the adaptive filter. Depending on the application of interest, the adaptive filter output or the estimation error may service as the system output. In the first case, the system operates as a predictor; in the latter case, it operates as a prediction error filter.

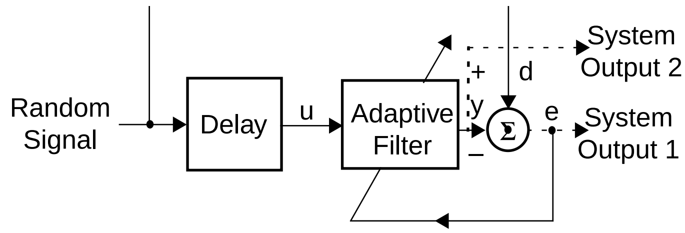


Figure 3. Prediction

IV. Interference Cancelling (Figure 4). In this final class of applications, the adaptive filter is used to cancel unknown interference contained in a primary signal, with the cancellation being optimized in some sense. The primary signal serves as the desired response for the adaptive filter. A reference signal is employed as the input to the adaptive filter. The reference signal is derived from a sensor or set of sensors located in relation to the sensor(s) supplying the primary signal in such a way that the information-bearing signal component is weak or essentially undetectable.

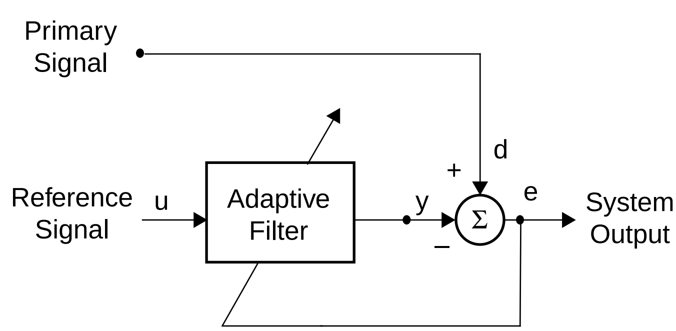


Figure 4. Interference Canceling

SELECTED ADAPTIVE FILTERING APPLICATIONS

Prediction

The coders used for the digital representation of speech signals fall into two broad classes: source coders and waveform coders. Source coders are model dependent, which means that they use *a priori* knowledge about how the speech signal is generated at the source. Source coders for speech are generally referred to as *vocoders*. Vocoders can operate at low coding rates; however, they provide a synthetic quality, with the speech signal having lost substantial naturalness.

Waveform coders, on the other hand, strive for facsimile reproduction of the speech waveform. In principle, these coders are signal independent. They may be designed to provide telephone-toll quality for speech at relatively high coding rates.

In the context of speech, linear predictive coding (LPC) strives to produce digitized voice data al low bit rates (2.4 to 4.8 Kbps) with two important motivations in mind:

1. The use of linear predictive coding permits the transmission of digitized voice over a narrow-band channel.
2. The realization of a low-bit rate makes the encryption of voice signals easier and more reliable than would otherwise be the case.

Figure 5 shows a simplified block diagram of the classical model for the speech production process. (In this particular example, the sound-generating mechanism is linearly separable from the intelligence-modulating, vocal-tract filter.) The precise form of the excitation depends on whether the speech sound is voiced or unvoiced.

Voiced speech sound is generated from quasi-periodic, vocal-cord sound. In the speech model, the impulse-train generator produces a sequence of impulses, which are spaced by a fundamental period equal to the pitch period. This signal, in turn, excites a linear filter whose impulse response equals the vocal-cord sound pulse.

An unvoiced speech sound is generated from random sound produced by turbulent air flow. In this case the excitation consists simply of a white noise source. The probability distribution of the noise samples does not appear to be critical.

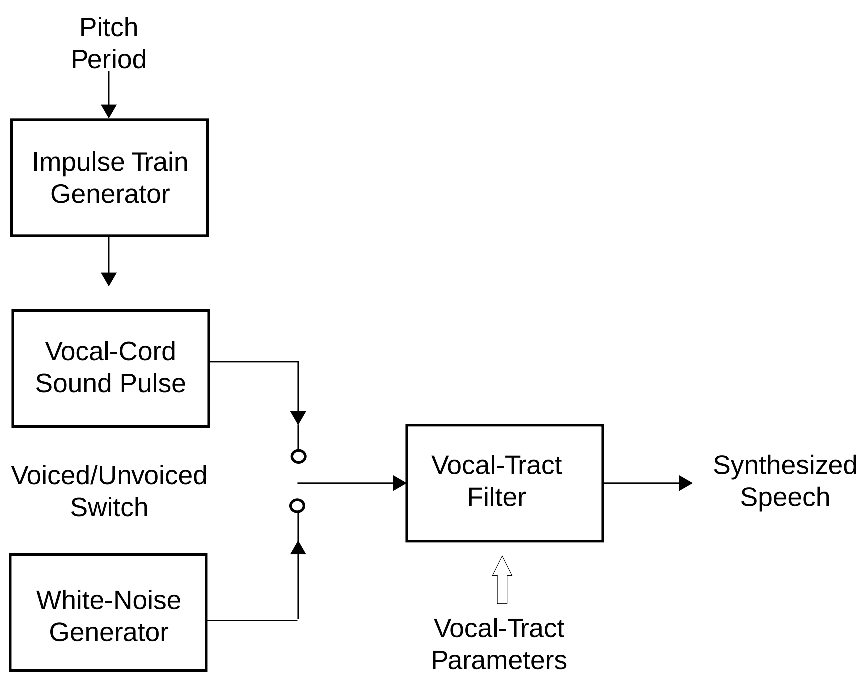


Figure 5. Block Diagram of a Simplified Model for the Speech Production Process

Figure 6 shows the block diagram of an LPC vocoder, consisting of a transmitter and a receiver. The transmitter first applies a window to the input speech signal, thereby identifying a block of speech samples for processing. This window is short enough for the vocal-tract shape to be nearly stationary, so the parameters of the speech-production model may be treated as essentially constant for the duration of the window. The transmitter then analyzes the input speech signal in an adaptive manner—block by block—by performing a linear prediction and pitch detection. Finally, it codes the parameters made up of the set of predictor coefficients, the pitch period, the gain parameter, and the voiced-unvoiced parameter, for transmission over the channel. The receiver performs the inverse operations by first decoding the incoming parameters. In particular, it computes the values of the predictor coefficients, the pitch period, and the gain parameter, and determines whether the segment of interest represents voiced or unvoiced sound. Finally, the receiver uses these parameters to synthesize the speech signal by utilizing the model of Figure 5.

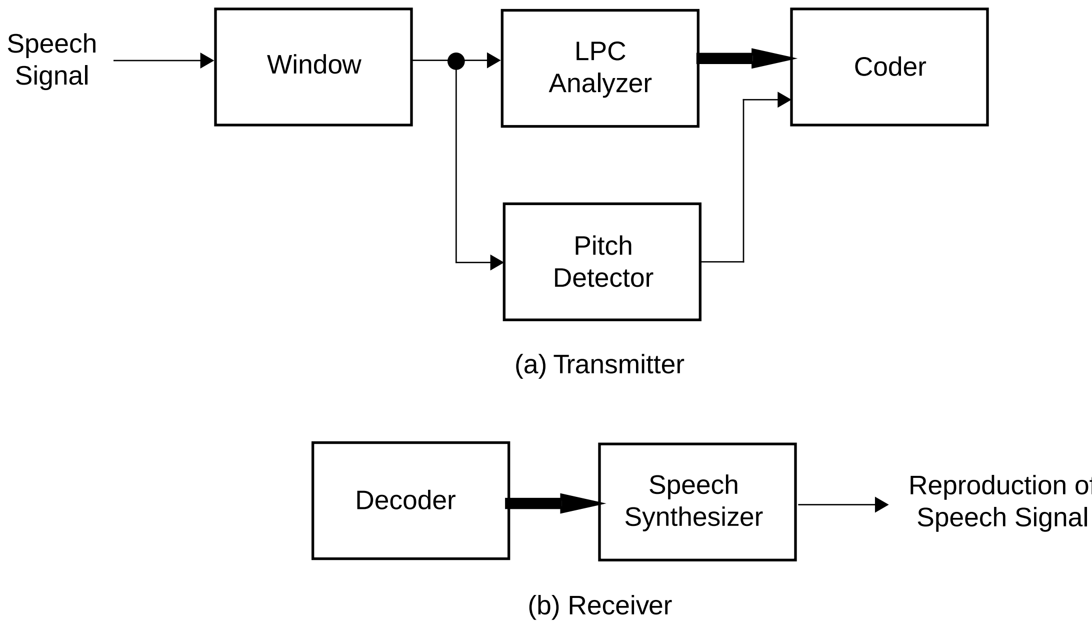


Figure 6. Block Diagram of LPC Vocoder: (a) Transmitter, (b) Receiver

Adaptive Equalization

During the past three decades, a considerable effort has been devoted to the study of data-transmission systems that utilize the available channel bandwidth efficiently. The objective here is to design the system to accommodate the highest possible rate of data transmission, subject to a specified reliability that is usually measured in terms of the error rate or average probability of symbol error.

The transmission of digital data through a linear communication channel is limited by two factors: 1) intersymbol interference, and 2) additive thermal noise.

Intersymbol Interference (ISI). Caused by dispersion in the transmit filter, the transmission medium, and the receive filter.

Additive Thermal Noise. Generated by the receiver at its front end. For bandwidth-limited channels, intersymbol interference seems to be the chief determining factor in the design of high-data-rate transmission systems. Figure 7 shows the equivalent baseband model of a binary pulse-amplitude (PAM) modulation system. The signal applied to the input of the transmitter part of the system consists of a binary data sequence, in which the binary symbol consists of 1 or 0. This sequence is applied to a pulse generator, the output of which is filtered first in the transmitter, then by the medium, and finally in the receiver. Let *u(k)* denote the sampled output of the receive filter in Figure 7; the sampling is performed in synchronism with the pulse generator in the transmitter. This output is compared to a threshold by means of a decision device. If the threshold is exceeded, the receiver makes a decision in favor of symbol 1. Otherwise, it decides in favor of symbol 0.

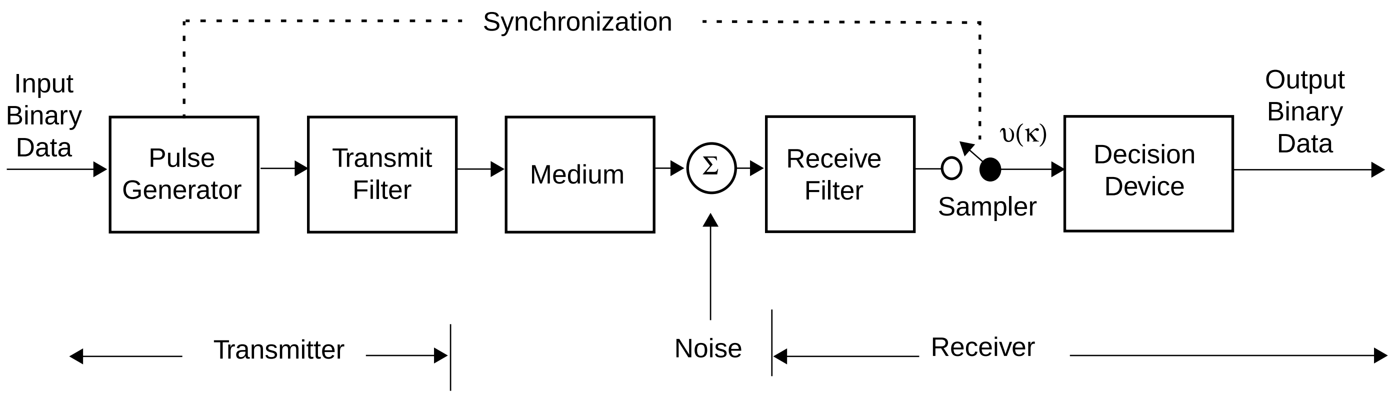


Figure 7. Block Diagram of a Baseband Data Transmission System (Without Equalization)

On a physical channel there is always intersymbol interference. To overcome intersymbol interference, control of the time-sampled function is required. In principle, if the characteristics of the transmission medium are known precisely, then it is virtually always possible to design a pair of transmit and receive filters that will make the effect of intersymbol interference arbitrarily small.

For adequately reducing the intersymbol interference, an adaptive equalizer will provide precise control over the time response of the channel.

An adaptive filtering algorithm requires knowledge of the desired response to form the error signal needed for the adaptive process to function. In theory, the transmitted sequence is the desired response for adaptive equalization.

In practice, however, with the adaptive equalizer located in the receiver, the equalizer is physically separated from the origin of its ideal desired response.

There are two methods in which a replica of the desired response may be generated locally in the receiver: 1) training method, and 2) decision-directed method.

Training method. In this method, a replica of the desired response is stored in the receiver. Naturally, the generator of this stored reference must be electronically synchronized with the known transmitted sequence.

Decision-directed method: Under normal operating conditions, a good replica of the transmitted sequence is being produced at the output of the decision device in the receiver. Accordingly, if this output was the correct transmitted sequence, it may be used as the desired response for the purpose of adaptive equalization. Such a method of learning is said to be “decision directed” because the receiver attempts to learn by employing its own decisions.

A final comment pertaining to performance evaluation: A popular experimental technique for assessing the performance of a data transmission system involves the use of an eye pattern. This pattern is obtained by applying the received wave to the vertical deflection plates of an oscilloscope, and a saw-tooth wave at the transmitted symbol rate to the horizontal deflection plates. The resulting display is called an eye pattern because of its resemblance to the human eye for binary data. Thus, in a system using adaptive equalization, the equalizer attempts to correct for intersymbol interference in the system and thereby open the eye pattern as far as possible.

Adaptive Differential Pulse-Code Modulation

In Pulse-Code Modulation (PCM), which is the standard technique for waveform coding, three basic operations are performed on the speech signal: 1) sampling, 2) quantization, and 3) coding. The operations of sampling and quantization are designed to preserve the shape of the speech signal. As for coding, it is merely a method of translating the discrete sequence of sample values into a more appropriate form of signal representation. The rationale for sampling follows from a basic property of all speech signals: They are band limited. This means that a speech signal can be sampled in time at a finite rate in accordance with the sampling theorem. For example, commercial telephone networks designed to transmit speech signals occupy a bandwidth from 200 to 3200 Hz. To satisfy the sampling theorem, a conservative sampling rate of 8 kHz is commonly used in practice.

In PCM, as used in telephony, the speech signal is sampled at the rate of 8 kHz, nonlinearly quantized, and the coded into 8-bit words, as shown in Figure 8(a). The result is a good signal-to-quantization noise ratio over a wide dynamic range of input signal levels. This method requires a bit rate of 64 Kbps.

Differential Pulse-Code Modulation (DPCM), another example of waveform coding, involves the use of a predictor as shown in Figure 5(b). The predictor is designed to exploit the correlation that exists between adjacent samples of the speech signal in order to realize a reduction in the number of bits required for the transmission of each sample of the speech signal and yet maintain a prescribed quality of performance. This is achieved by quantizing and then coding the prediction error that results from the subtraction of the predictor output from the input. If the prediction is optimized, the variance of the prediction error will be significantly smaller than that of the input signal, so a quantizer with a given number of levels can be adjusted to produce a quantizing error with a smaller variance than would be possible if the input signal were quantized directly as in a standard PCM system. Equivalently, for a quantizing error of prescribed variance, DPCM requires a smaller number of quantizing levels than PCM. Differential PCM uses a fixed quantizer and a fixed predictor. A further reduction in the transmission rate can be achieved by using an adaptive quantizer together with an adaptive predictor of sufficiently high order.

Adaptive Differential Pulse-Code Modulation (ADPCM) can digitize speech with toll (8-bit PCM) quality at 32 Kbps (see Figure 8[c]).

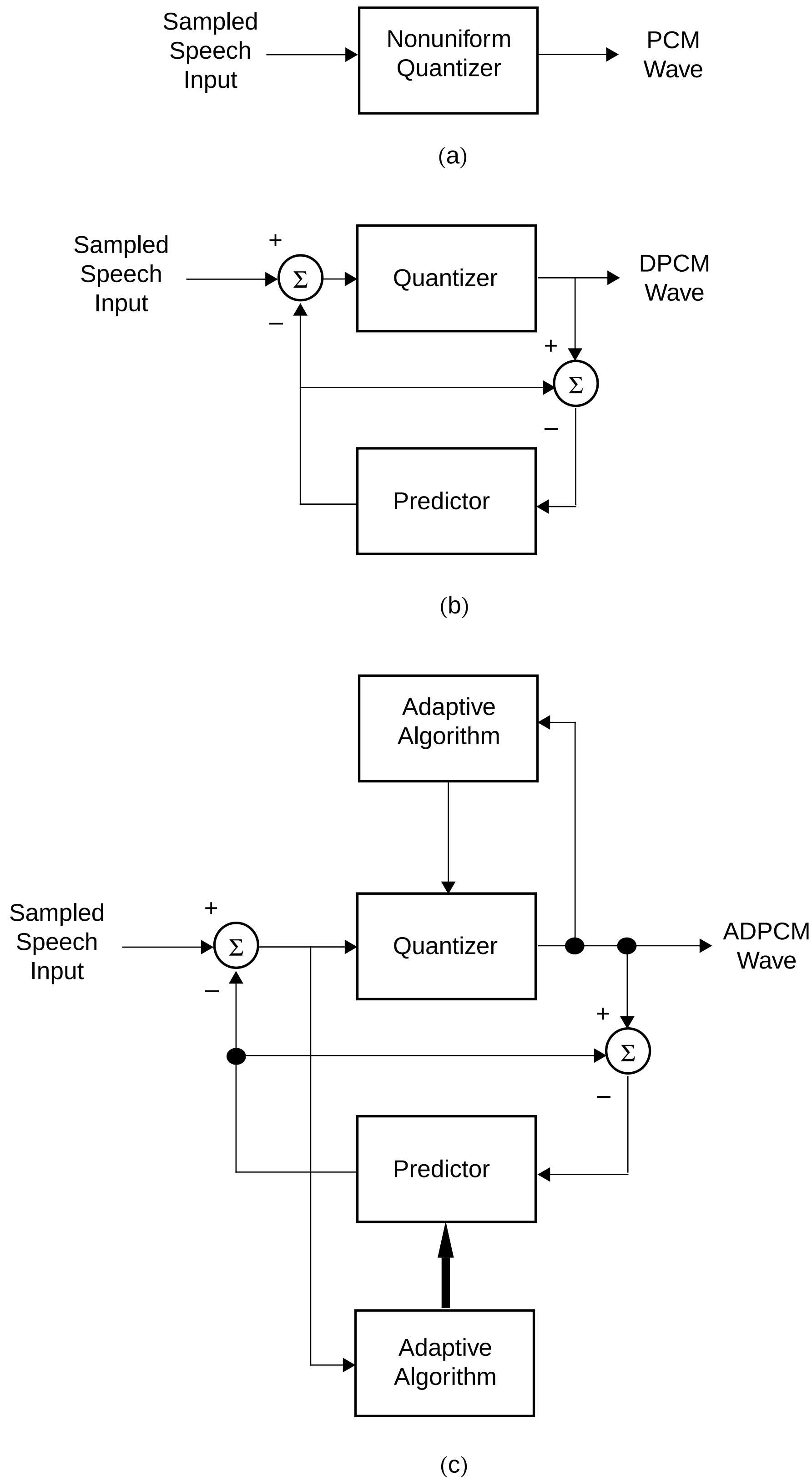


Figure 8. Waveform Coders: (a) PCM, (b) DPCM, (c) ADPCM

Adaptive Noise Canceling

As the name implies, adaptive noise cancelling relies on the use of noise cancelling by subtracting noise from a received signal, an operation controlled in an adaptive manner for the purpose of improved signal-to-noise ratio. Ordinarily, it is inadvisable to subtract noise from a received signal because such an operation could produce disastrous results by causing an increase in the average power of the output noise. However, when proper provisions are made, and filtering and subtraction are controlled by an adaptive process, it is possible to achieve a superior system performance compared to direct filtering of the received signal.

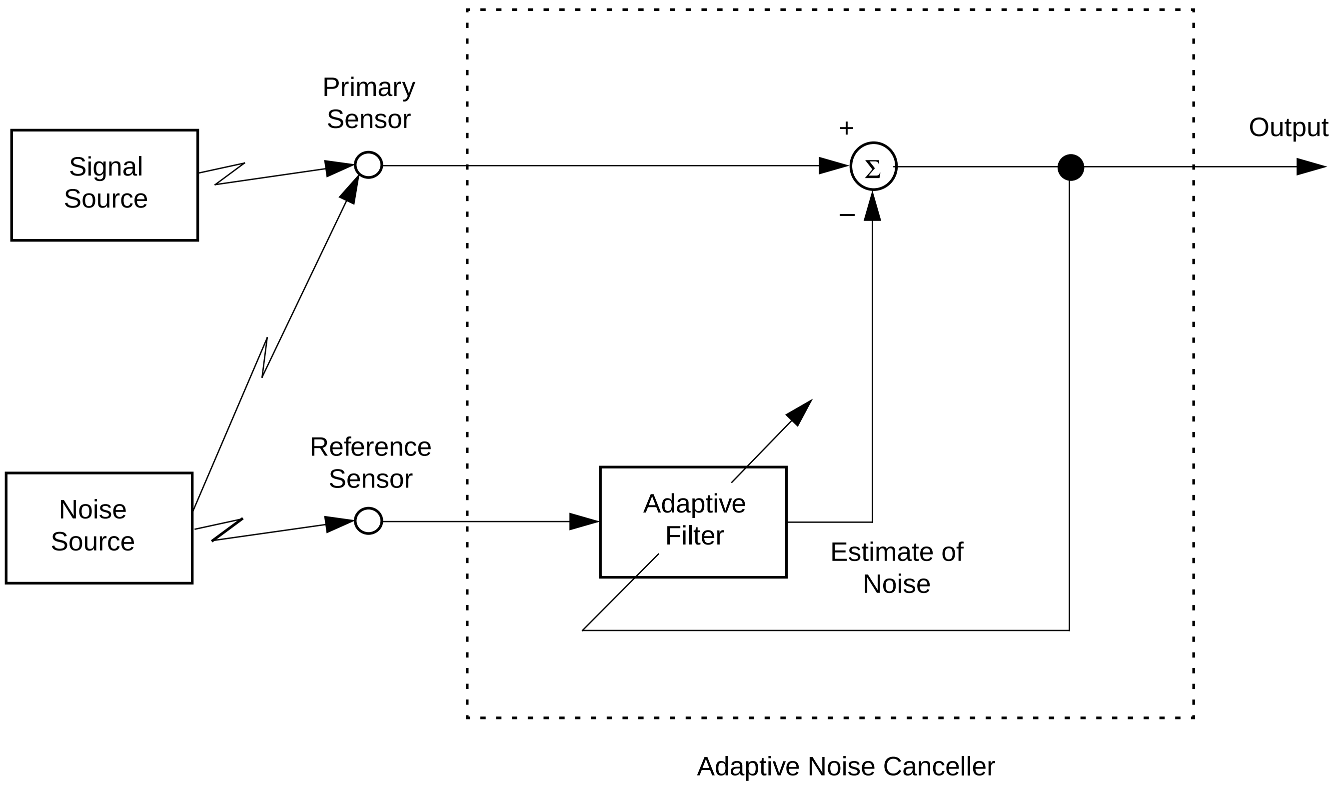


Figure 9. Adaptive Noise Cancellation

Basically, an adaptive noise canceller is a dual-input, closed-loop adaptive control system as illustrated in Figure 9 and Figure 1. The two inputs of the system are derived from a pair of sensors: a primary sensor and a reference sensor.

- The primary sensor receives an information-bearing signal *s(n)* corrupted by additive noise *v0(n)*.
- The signal and the noise are not correlated with each other. The reference sensor receives a noise *v1(n)* that is not correlated with the signal *s(n)* but correlated with the noise *v0(n)* in the primary sensor output in an unknown way:

E[s(n),v1(n-k)]=0, for all k

E[v0(n)v1(n-k)]=p(k)

where, as before, the signals are real valued and *p(k)* is an unknown cross correlation for lag *k*.

The reference signal *v1(n)* is processed by an adaptive filter to produce the output signal *y(n)*. The filter output is subtracted from the primary signal *d(n),* serving as the desired response for the adaptive filter. The error signal is defined by:

e(n) = d(n)-y(n)

The error signal is used, in tern, to adjust the tap weights of the adaptive filter, and the control loop around the operations of filtering and subtraction is thereby closed. Note that the information bearing signal *s(n)* is indeed part of the error signal *e(n)*. Now, the adaptive filter attempts to minimize the mean-square value (average power) of the error signal *e(n)*. The information bearing signal *s(n)* is essentially unaffected by the adaptive noise canceller. Hence, minimizing the mean-square value of the error signal *e(n)* is equivalent to minimizing the mean-square value of the output noise *v0(n)-y(n)*. With the signal *s(n)* remaining essentially constant, it follows that the minimization of the mean-square value of the error signal is indeed the same as the maximization of the output signal to noise ratio of the system.

The effective use of adaptive noise cancelling therefore requires that the reference sensor be placed in the noise field of the primary sensor with two specific objectives in mind:

1. The information-bearing signal component of the primary sensor output is undetectable in the reference sensor output.
2. The reference sensor output is highly correlated with the noise component of the primary sensor output. Moreover, the adaptation of the adjustable filter coefficients must be near optimum.

Noise Canceling Applications

Now, let us consider two useful applications of the adaptive noise cancelling operation.

Canceling 60-Hz Interference in Electrocardiography (ECG). In ECG, commonly used to monitor heart patients, an electrical discharge radiates energy through a human tissue, and the resulting output is received by an electrode. The electrode is usually positioned in such a way that the received energy is maximized. Typically, however, the electrical discharge involves very low potentials. Hence extra must be exercised in minimizing signal degradation due to external interference. By far, the strongest form of interference is that of a 60-Hz periodic waveform picked up by the receiving electrode from nearby electrical equipment. Figure 10 shows a block diagram of the adaptive noise canceller used to reduce the harmonics.

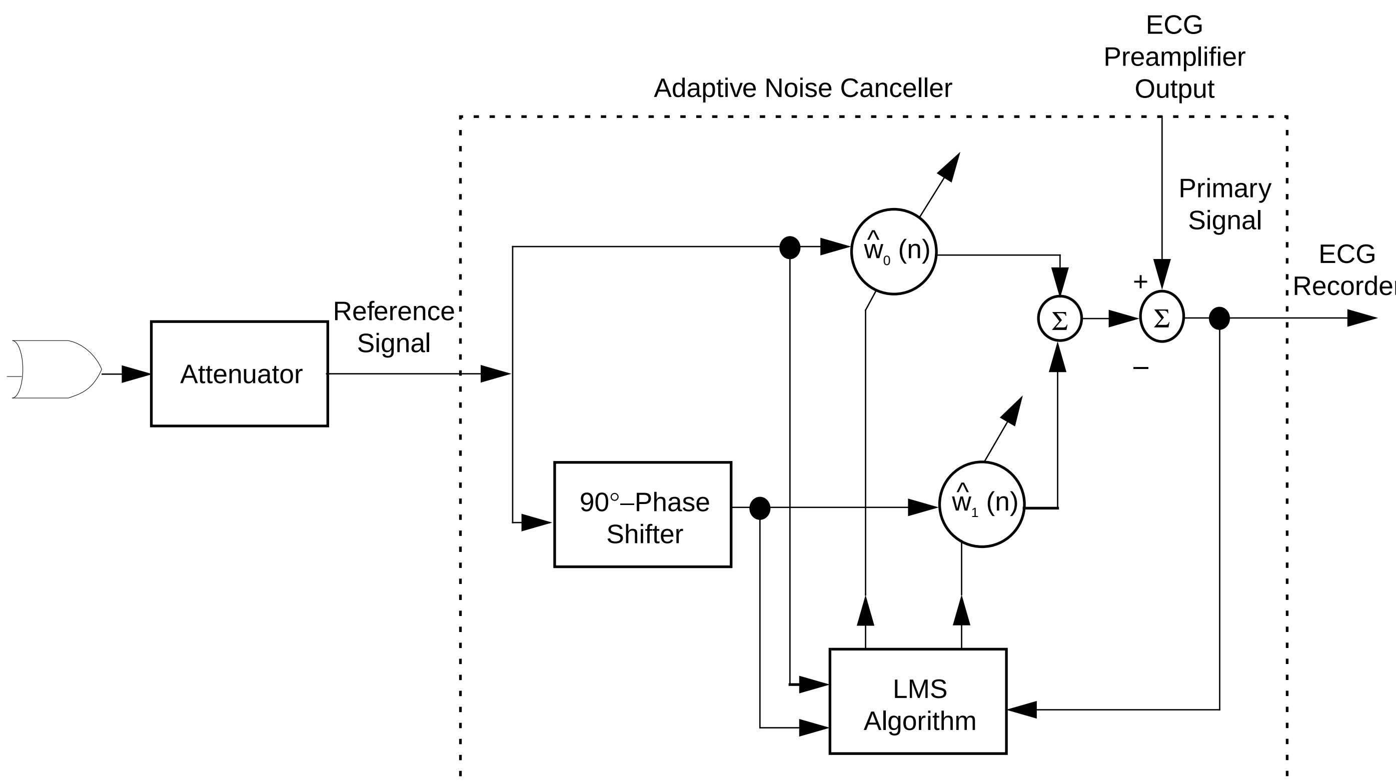


Figure 10. Adaptive Noise Canceller for Suppressing 60-Hz Interference in Electrocardiography. (After Widrow, and others [1975b].)

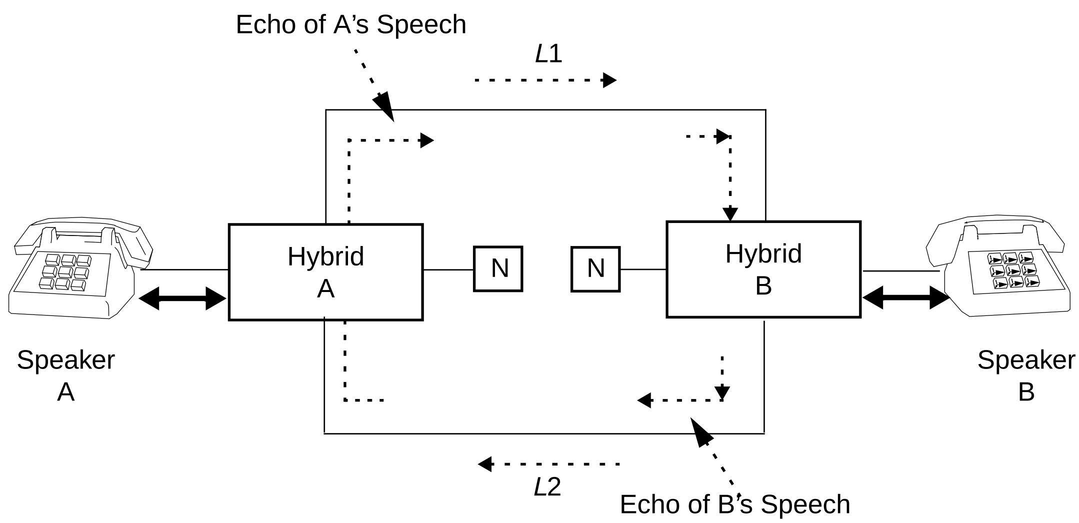
Reduction of Acoustic Noise in Speech. At a noisy site, for example, the cockpit of a military aircraft, voice communication is effected by the presence of acoustic noise. This is particularly serious when linear predictive coding (LPC) is used for the digital representation of voice signals at low-bit rates.

The noise corrupted speech is used as the primary signal. To provide the reference signal, a reference microphone is placed in a location where there is sufficient isolation from the source of speech.

Echo Cancellation

Almost all conversations are conducted in the presence of echoes. An echo may not be distinct, depending on the time delay involved. If the delay between the speech and the echo is short, the echo is not noticeable but perceived as a form of spectral distortion or reverberation. If, on the other hand, the delay exceeds a few tens of milliseconds, the echo is distinctly noticeable.

To see how echoes occur, consider a long-distance telephone circuit depicted in Figure 11. Every telephone is connected to a central office by a two-wire line called the “customer loop.” The two-wire line serves the need for communications in either direction. However, for circuits longer than 35 miles, a separate path is necessary for each direction of transmission.



The boxes marked *N* are balancing impedances.

Figure 11. Long-Distance Telephone Circuit

Accordingly, there must be provision for connecting the two-wire circuit to the four-wire circuit. This connection is accomplished by means of a hybrid transformer, commonly referred to as a hybrid.

Basically, a hybrid is a bridge circuit with three ports. If the bridge is not perfectly balanced, the “in” port becomes coupled to the “out” port, thereby giving rise to an echo (refer to Figure 12).

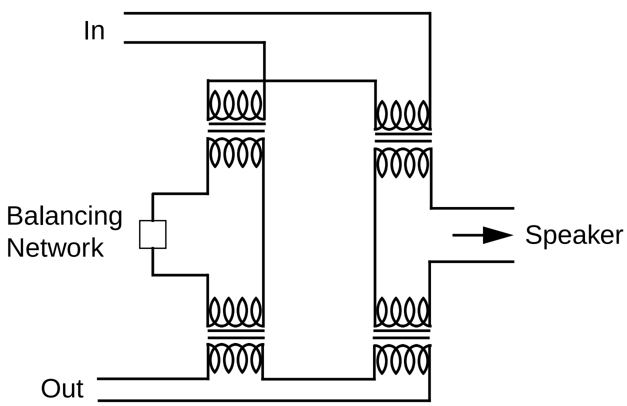


Figure 12. Hybrid Circuit

The basic principle of echo cancellation is to synthesize a replica of the echo and subtract it from the returned signal. This principle is illustrated in Figure 13 for only one direction of transmission. The adaptive canceller is placed in the four-wire path near the origin of the echo. The synthetic echo is generated by passing the speech signal from speaker A through an adaptive filter that ideally matches the transfer function of the echo path. The reference signal, passing through the hybrid, results in the echo signal. This echo, together with a near-end talker signal *x*, constitutes the desired response for the adaptive canceller. The synthetic echo is subtracted from the desired response to yield the canceller error signal. In any event, the error signal is used to control the adjustments made in the coefficiencies of the adaptive filter. For the adaptive echo cancellation circuit to operate satisfactorily, the impulse response of the adaptive filter should have a length greater than the longest echo path that needs to be accommodated.

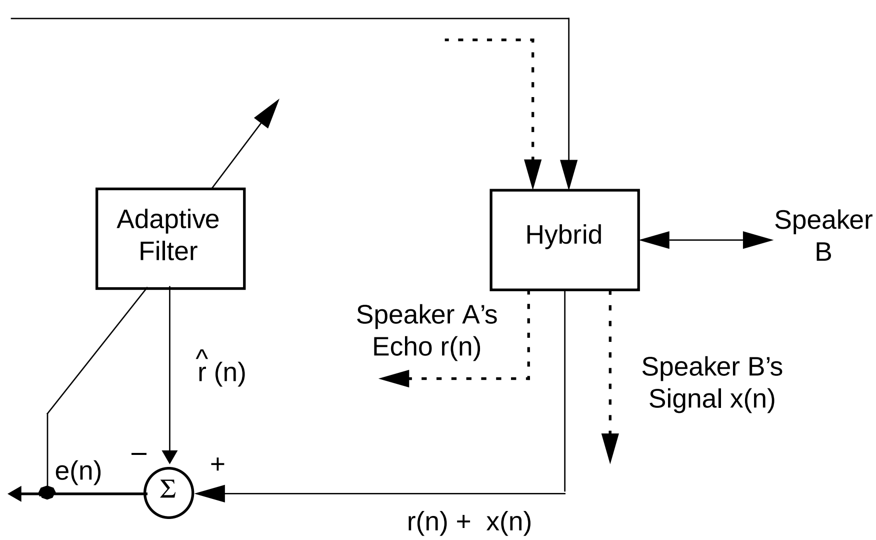


Figure 13. Signal Definitions for Echo Cancellation

LMS Algorithm

The well-known least-mean-square (LMS) algorithm is an important member of the family of stochastic gradient-based algorithms. A significant feature of the LMS algorithm is its simplicity. It does not require measurements of the pertinent correlation functions, nor does it require matrix inversion. Indeed, it is the simplicity of the LMS algorithm that has made it the standard against which other adaptive filtering algorithms are benchmarked.

The operation of the LMS algorithm is descriptive of a feedback control system. Basically, it consists of a combination of two basic processes:

1. An adaptive process, which involves the automatic adjustment of a set of tap weights.

2. A filtering process, which involves: (a) forming the inner product of a set of tap inputs and the corresponding set of tap weights emerging from the adaptive process to produce an estimate of a desired response, and (b) generating an estimation error by comparing this estimate with the actual value of the desired response. The estimation error is used, in turn, to actuate the adaptive process, thereby closing the feedback loop.

Correspondingly, we may identify two basic components in the structural constitution of the LMS algorithm, as illustrated in Figure 14.

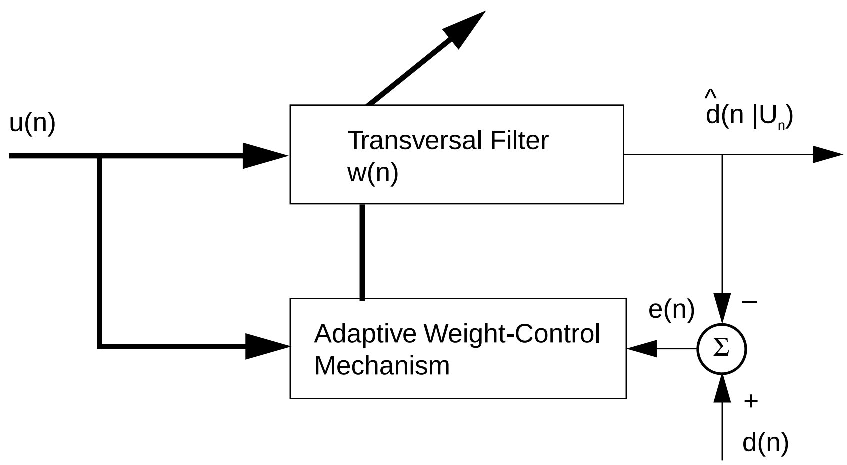


Figure 14. Block Diagram of Adaptive Transversal Filter

First we have a transversal filter, around which the LMS algorithm is built. This component is responsible for performing the filtering process. Second, we have a mechanism for performing the adaptive control process on the tap weights of the transversal filter.

Details of the transversal filter component are presented in Figure 15. The tap inputs from the elements of the M-by-1 tap input vector *u(n)*, where M-1 is the number of delay elements.

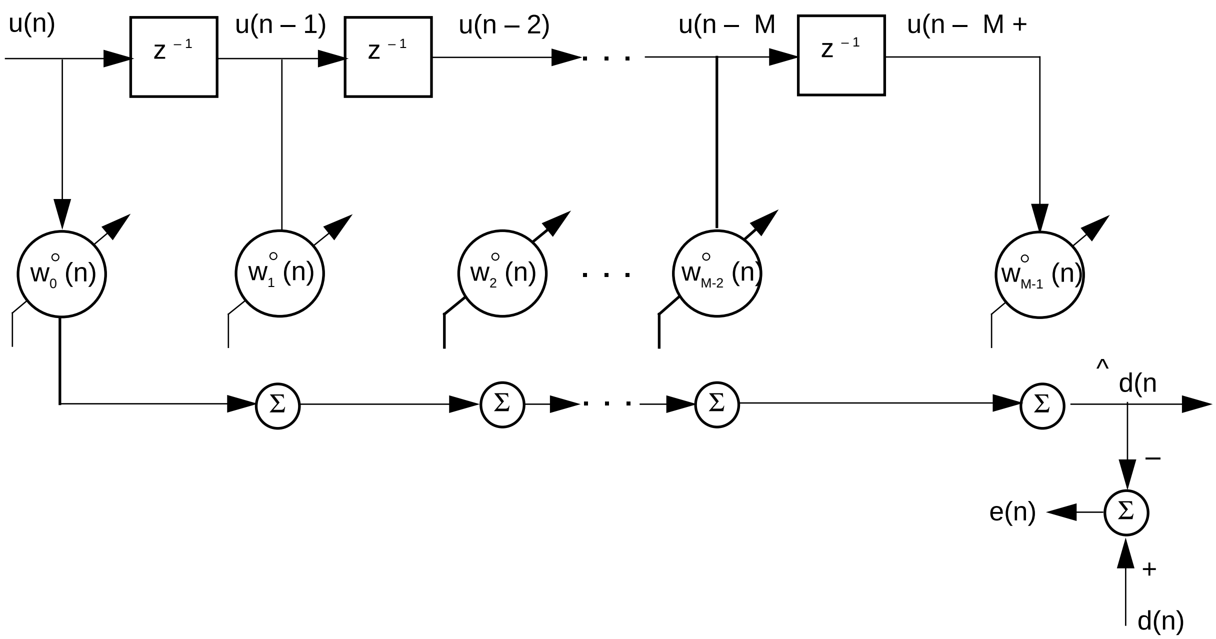


Figure 15. Detailed Structure of the Transversal Filter Component

Figure 16 presents details of the adaptive weight-control mechanism. Specifically, a scaled version of the inner product of the estimation error and the tap input is computed. The result obtained defines the correction applied to the tap weight. The scaling factor used in this computation is called the adaptation constant or step size parameter.

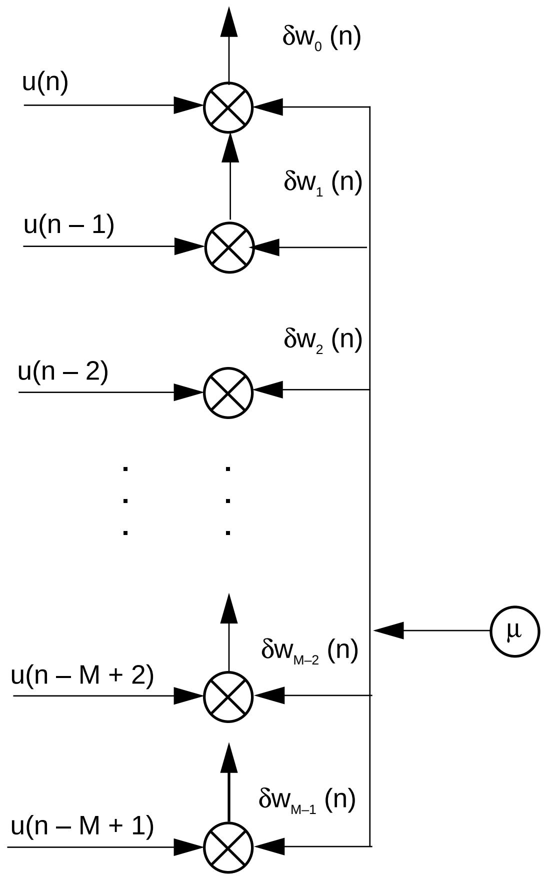


Figure 16. Detailed Structure of the Adaptive Weight-Control Mechanism

The tap weight vector computed by the LMS algorithm executes a random motion around the minimum point of the error performance surface. This random motion gives rise to two forms of convergence behavior for the LMS algorithm: 1) Convergence in the mean, and 2) Convergence in the mean square.

It is important to realize, however, that the "mis-adjustments" are under the designer's control. In particular, the feedback loop acting around the tap weights behaves like a low-pass filter, with an average time constant that is inversely proportional to the step size parameter *u*. Hence, by assigning a small value to *u*, the adaptive process is made to progress slowly, and the effects of gradient noise on the tap weights are largely filtered out. This, in turn, has the effect of reducing the mis-adjustments.

LMS Algorithm Summary

1. FIR Filter output: y(n) = w'(n)u(n)

2. Estimation error: e(n) = d(n) - y(n)

3. Tap weight adaptation: w(n+1) = w(n) + uu(n)e(n)
   u(n) .. tap-input vector: u(n), u(n-1), ... u(n-M+1)
   w(n) .. tap-weight vector: w0(n), w1(n) ... wM-1(n)

Equation 1 and 2 define the estimation error *e(n)*, the computation of which is based on the current estimate of the tap-weight vector, *w(n)*. Note also that the second term, *uu(n)e(n),* on the right side of Equation 3 represents the correction that is applied to the current estimate of the tap-weight vector, *w(n)*. The iterative procedure is started with the initial guess *w(0)=0*. In general, the LMS algorithm requires only 2M+1 complex multiplications and 2M complex additions per iteration, where *M* is the number of tap weights used in the adaptive transversal filter.

References

1. Haykin, S. Adaptive Filter Theory. Prentice-Hall International, Inc. 1991, pages 17–20, 31–40, 49–55, 303.

2. Honig, M. and Messerschmitt, D. Adaptive Filters: Structures, Algorithms, and Applications. Boston: Kluwer Academic Publishers. 1984.