Chapter 6 : Adaptive Signal Processing and its application

Outline

- Adaptive Signal Processing and
- Its Application

6.1 Adaptive Signal Processing

- Adaptive filters are best used in cases where signal conditions or system parameters are slowly changing and the filter is to be adjusted to compensate for this change.
- The least mean squares (LMS) criterion is a search algorithm that can be used to provide the strategy for adjusting the filter coefficients.
- Programming examples are included to give a basic intuitive understanding of adaptive filters.
- The coefficients of an adaptive filter are adjusted to compensate for changes in input signal, output signal, or system parameters.
- An adaptive filter can be very useful when there is uncertainty about the characteristics of a signal or when these characteristics change.

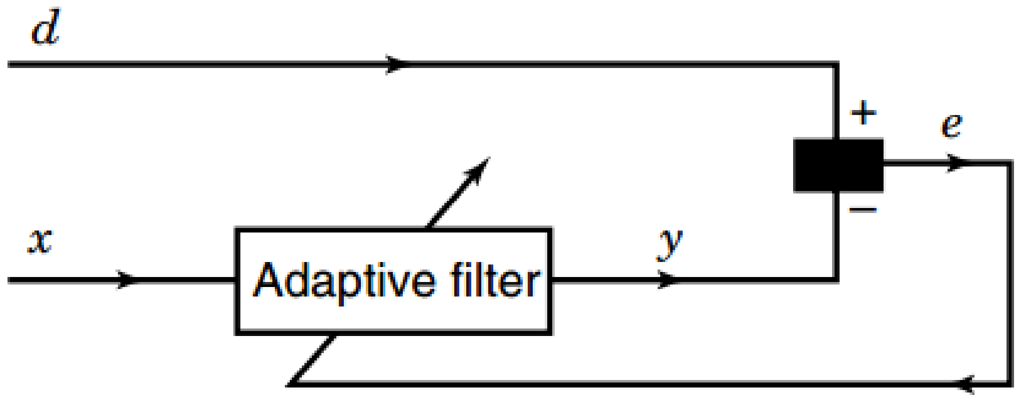

Figure 6.1.Basic adaptive filter structure.

- **The adaptive filter's output y is compared with a desired signal d to yield an error signal e, which is fed back to the adaptive filter.**
- **The coefficients of the adaptive filter are adjusted, or optimized, using a least mean squares (LMS) algorithm based on the error signal.**
- **We discuss here only the LMS searching algorithm with a linear combiner (FIR filter), although there are several strategies for performing adaptive filtering.**

$$y(n) = \sum_{k=0}^{N-1} w_k(n)x(n-k)$$

where $w_k(n)$ represent N weights or coefficients for a specific time n.

- **A performance measure is needed to determine how good the filter is. This measure is based on the error signal,**

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

which is the difference between the desired signal $d(n)$ and the adaptive filter's output $y(n)$. The weights or coefficients $w_k(n)$ are adjusted such that a mean squared error function is minimized. This mean squared error function is $E[e^2(n)]$, where E represents the expected value. Since there are k weights or coefficients, a gradient of the mean squared error function is required. An estimate can be found instead using the gradient of $e^2(n)$, yielding

$$w_k(n+1) = w_k(n) + 2\beta e(n)x(n-k) \qquad k = 0, 1, \ldots, N-1$$

- which represents the **LMS algorithm Equation** provides a simple but powerful and efficient means of updating the weights, or coefficients, without the need for averaging or differentiating, and will be used for implementing adaptive filters.
- The input to the adaptive filter is **x(n)**, and the rate of convergence and accuracy of the adaptation process (adaptive step size) is β
- For each specific time n, each coefficient, or weight ,$w_k(n)$ is updated or replaced by a new coefficient, unless the error signal **e(n)** is **zero**.
- After the filter's output y(n), the error signal e(n) and each of the coefficients $w_k(n)$ are updated for a specific time **n**, a new sample is acquired (from an ADC) and the adaptation process is repeated for a different time.

Adaptive Structures

- **A number of adaptive structures have been used for different applications in adaptive filtering.**
  - **For noise cancellation**
  - **For system identification**
  - **For Adaptive predictor**
  - **Additional structures have been implemented**

1. **For noise cancellation: The adaptive structure in Figure 6.1 modified for a noise cancellation application.**

- **The desired signal discorrupted by uncorrelated additive noise n. The input to the adaptive filter is a noise n' that is correlated with the noise n.**
- **The noise n' could come from the same source as n but modified by the environment.**

- The adaptive filter's output y is adapted to the noise n. When this happens, the error signal approaches the desired signal d.
- The overall output is this error signal and not the adaptive filter's output y.
- This structure will be further illustrated with programming examples using C code

2. For system identification:

- Figure 6.3 shows an adaptive filter structure that can be used for system identification or modeling.
- The same input is to an unknown system in parallel with an adaptive filter. The error signal e is the difference between the response of the unknown system d and the response of the adaptive filter y.
- This error signal is fed back to the adaptive filter and is used to update the adaptive filter's coefficients until the overall output y = d.
- When this happens, the adaptation process is finished, and e approaches zero.

- **In this scheme, the adaptive filter models the unknown system.**
- **This structure is illustrated later with three programming examples.**

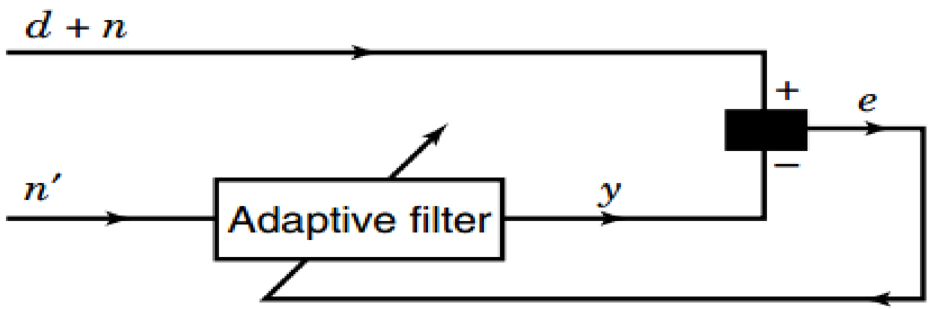

Figure 6.2.Adaptive filter structure for noise cancellation

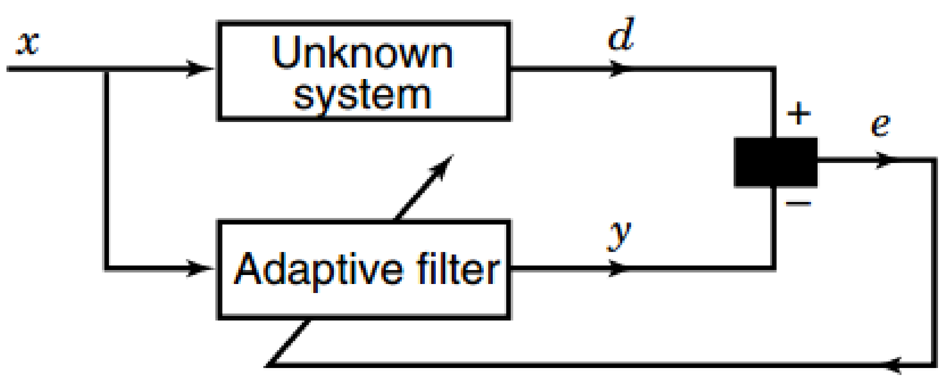

Figure 6.3.Adaptive filter structure for system identification

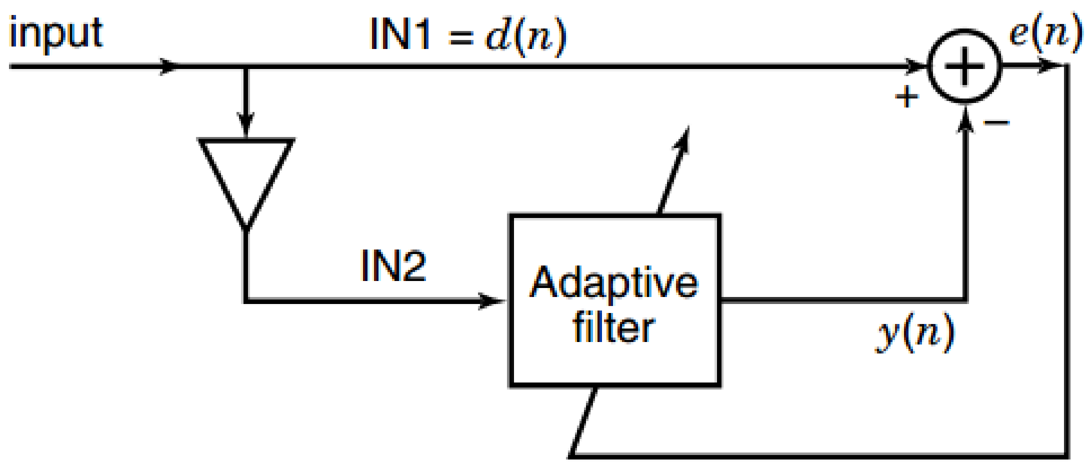

Figure 6.4.Adaptive predictor structure

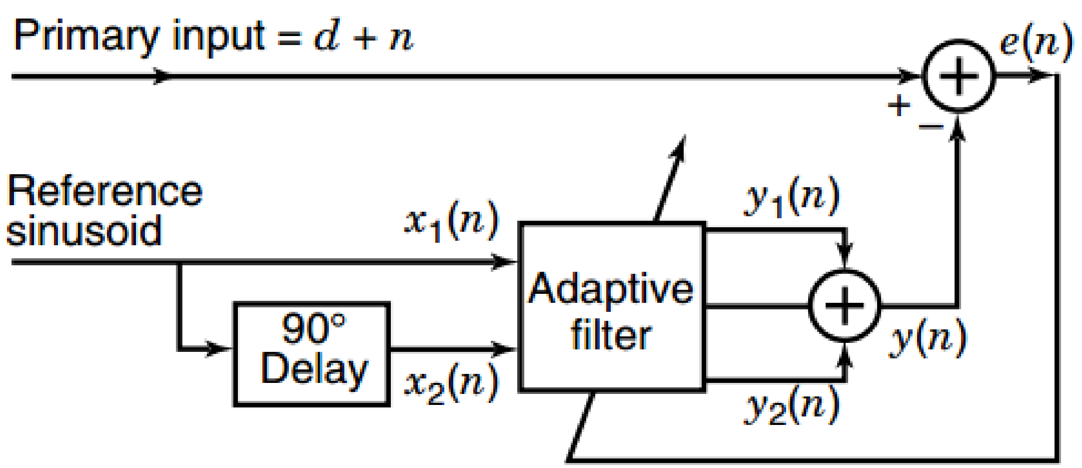

Figure 6.5.Adaptive notch structure with two weights.

3. For Adaptive predictor: Figure 6.4 shows an adaptive predictor structure which can provide an estimate of an input. This structure is illustrated later with a programming example.

4. Additional structures have been implemented, such as:

(a) **Notch with two weights**, which can be used to notch or cancel/reduce a sinusoidal noise signal. This structure has only two weights or coefficients. This structure is shown in Figure 6.5.

(b) **Adaptive channel equalization,** used in a modem to reduce channel distortion resulting from the high speed of data transmission over telephone channels.

- The LMS is well suited for a number of applications, including adaptive echo and noise cancellation, equalization, and prediction.
- Other variants of the LMS algorithm have been employed, such as the sign-error LMS, the sign-data LMS, and the sign-sign LMS.

1. For the sign-error LMS algorithm, becomes

$$w_k(n+1) = w_k(n) + \beta \, \text{sgn}[e(n)]x(n-k)$$

where sgn is the signum function,

$$\text{sgn}(u) = \begin{cases} 1 & \text{if } u \geqslant 0 \\ -1 & \text{if } u < 0 \end{cases}$$

2. For the sign-data LMS algorithm, becomes

$$w_k(n+1) = w_k(n) + \beta e(n) \, \text{sgn}[x(n-k)]$$

3. For the sign-sign LMS algorithm, becomes

$$w_k(n+1) = w_k(n) + \beta \, \text{sgn}[e(n)] \, \text{sgn}[x(n-k)]$$

which reduces to

$$w_k(n+1) = \begin{cases} w_k(n) + \beta & \text{if } \operatorname{sgn}[e(n)] = \operatorname{sgn}[x(n-k)] \\ w_k(n) - \beta & \text{otherwise} \end{cases}$$

which is more concise from a mathematical viewpoint because no multiplication operation is required for this algorithm.

Applications of adaptive signal processing

- **To introduce some practical aspects of signal processing, and in particular adaptive systems.**
- **Current applications for adaptive systems are in the fields of communications, radar, sonar, seismology, navigation systems and biomedical engineering.**

- The basic principles of adaptation, will cover various adaptive signal processing algorithms (e.g., the LMS algorithm).
- Many applications, such as adaptive noise cancellation, interference canceling, system identification, etc.

✓ Adaptive modeling and system identification

✓ Inverse adaptive modeling,

✓ Deconvolution and equalization

✓ Adaptive control systems

✓ Adaptive interference canceling

✓ Canceling noise, canceling periodic interference,

✓ Canceling interference in ECG signals, etc.