Adaptive Filter

2016.8(b)
2019.8(b)

Defination:- An Adaptive filter in DSP (Digital Signal processing) is a digital filter whose parameters are automatically adjusted to adapt to changing characteristics of the input signal or environment.

2016.8(b)

Implementation of the LMS Algorithm:

~~1. Initialize the filter coefficient $W_k(i) = 0$~~

~~2. At each sampling period:~~

~~a) Compute the filter output:~~

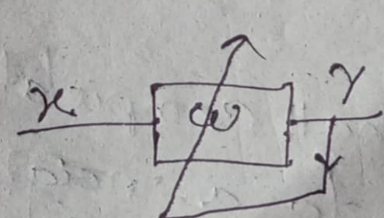
and steepest decent method:

$$\boxed{\xi(n) = E[|e(n)|^2] \rightarrow \text{Mean square error.}}$$

Step-1: Initialization

$w_0 \xrightarrow[\text{optimal value}]{} \omega$

Step-2:- Evaluate the gradient of $\xi(n)$ at current estimate, ω_n



Step-3:- $\omega_{n+1} = \omega_n - \mu \nabla \xi(n)$

$n = 0, 1, \ldots$

Step-4) Repeat step-2:

$$\nabla E(n) = \nabla E\{|e(n)|^2\}$$
$$= E\{\nabla |e(n)|^2\}$$
$$= E\{e(n) \nabla e^*(n)\}$$

$$\nabla e^*(n) = -x^*(n)$$
$$= -E\{e(n)\, x^*(n)\}$$

$$w_{n+1} = w_n - \mu \nabla E(n)$$
$$= w_n + \mu E\{e(n) x^*(n)\}$$

$$E\{e(n) x^*(n)\} = w_n + \mu E[\{d(n) - y(n)\} x^*(n)]$$
$$= w_n + \mu [E[d(n) x^*(n)] - E[y(n) x^*(n)]]$$
$$= w_n + \mu E[d(n) x^*(n)] - E[w_n^T x(n) x^*(n)]$$
$$= w_n + \mu (\underbrace{\hat{\gamma}_{dx}}_{\text{Cross correlation}} - \underbrace{\hat{\gamma}_{xx}}_{\text{auto correlation}} w_n)$$

This is stepest descent method.

-*-

Adaptive filter: (Iterative) Characteristics:-

- Don't need only knowledge of signal statistics before.
- Have small computational complexity
- Converge to a neighborhood of optional solution.

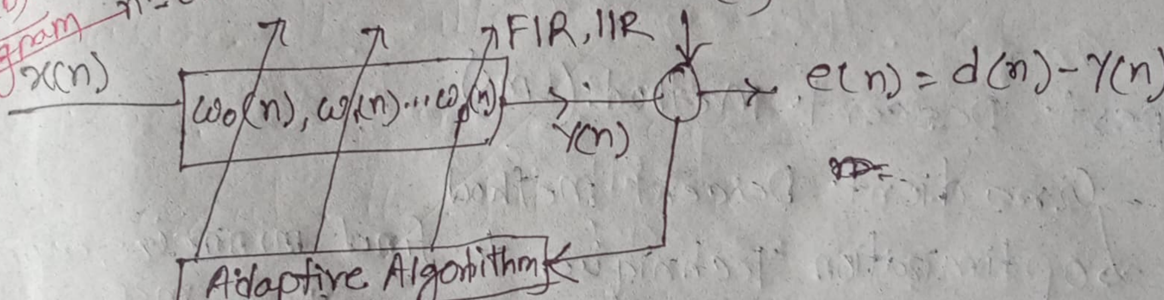
Adaptive filter is good for:

- Real time signal (nonstationary)

Application:

- System identification: (i) Determining the characteristics of an unknown system.
- Signal prediction: (ii) predicts future values of a signal
- noise cancellation: (iii) reducing unwanted noise in audio
- Acoustic Echo cancellation: (iv) Eliminating echoes in telecommunication systems.
- Speech & image coding: (v) Removing noise and artifacts from images

2019, 8(b) adaptive block diagram



$d(n)$ = designed signal

$y(n)$ = output

weight = $w_0(n), \ldots, w_p(n)$

$x(n)$ = input

Two parts:

(I) Filtering: Convolution of $x(n)$, $w(n) \to y(n)$

(II) Adaptive algorithm

error → $e(n)$ feedback

↓ minimize using LMS, RMS

* performance of adapting filter depend as -
  - stablish
  - equality
  - speed of adaptation
  - tracking capabilities

Mean square error: minimize MSE

MSE = $\varepsilon(n) = E\{|e(n)|^2\}$

$= E\{d(n) - y(n)\}^2$

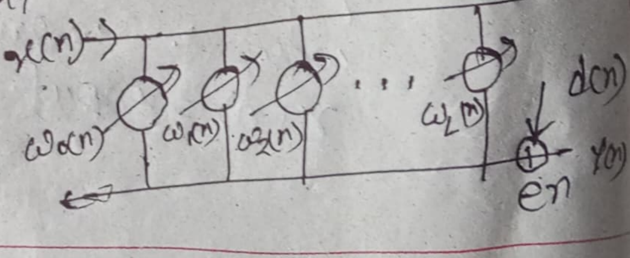
Steepest Descent Method: (LMS)

- Gradient Descent method
  - → optimization Technique to find minimize error MSE

$w(n) = [w_0(n), w_1(n), w_2(n), \ldots, w_L(n)]^T$

$w(n+1) = \Delta w(n) + w(n)$

$\Delta w(n)$ → change in weight



2022

8.a How adaptive filters can be used for noise cancellation.

⇒ Adaptive filters can be used for noise cancellation by dynamically adjusting their coefficients to minimize the difference between a desired signal and the filter output.

- The idea is to subtract the noise from the input signal using a reference noise signal.
- An Adaptive Algorithm (Like-LMS) adjusts the filter so that its output closely matches the noise.
  $\xi(n) = E\{|e(n)|^2\}$
- This output is then subtracted from the noisy signal to obtain the clean desired signal.

example:

In headphones, adaptive filters cancel ambient noise using a microphone that picks up external sounds and inverts them to cancel the noise.

—a—

8.b

Adaptive filter adjust by continuously modifying their coefficients based on the error between the filter output and the desired signal.

The adjustment is required in non-stationary environments such as when the noise or signal characteristics change over time.

Gradient descent method:

1. Compute the ~~error~~ output:

$$y(n) = w^T(n)\,x(n)$$

$x(n)$ → input vector

$w(n)$ → filter weight

2. Compute the error:

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

$d(n)$ → desired signal

3. updating the weights:

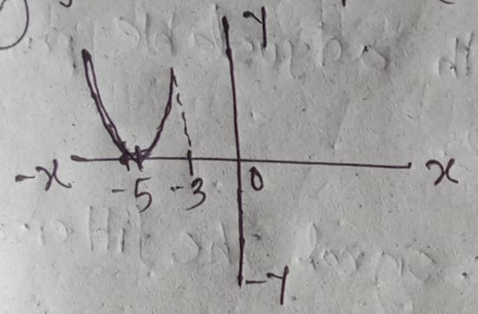
$$w(n+1) = w(n) + \mu\, e(n)\, x(n)$$

μ → the step-size parameter.

—*—

(A) optimization technique to find the minimum value of a function.

⇒ Sag function $y = (x+5)^2$



~~Step~~

(1) choose a starting point,

Let $x = -3$

$y = (x+5)^2$ → Gradient at ⊖ direction

(ii) $\frac{dy}{dx} = 2(x+5)$ Step size, $\mu = 0.01$

(3) $x_0 = -3$, $\frac{dy}{dx} = 2(x+5)$, $\mu = 0.01$

iteration 1:

$x_1 = x_0 - \mu \times \frac{dy}{dx}$

$= (-3) - (0.01) \times 2 \times (-3+5) = -3 - 0.01 \times 2(2)$

$= \cancel{-3.044}\ -3.04$

iterative 2:

$x_2 = x_1 - \mu \times \frac{dy}{dx}$

$= -3.04 - 0.01 \times 2(-3.04+5)$

$= \cancel{-3.044}\ -3.07$

—o—

II adaptive filter এর Block diagram ৭০ define:

(i) Input signal, $x(n)$ ⇒ The signal being processed by the filter.

(ii) Adaptive Filter: A filter with adjustable parameters using FIR or IIR.

(iii) Desired Signal, $d(n)$ ⇒ The signal the filter aims to estimate.

(iv) Error signal, $e(n)$ → difference between the filter's output $y(n)$ and the desired signal $d(n)$.

(v) Adaptive algorithm: based on error signal, usually to minimize it.

(vi) Filter output, $y(n)$ → The output of the adaptive filter.

— ❀ —