ESE 531: Digital Signal Processing

Lec 18: March 30, 2017

IIR Filters and Adaptive Filters

Today

- IIR Filter Design
  - Impulse Invariance
  - Bilinear Transformation
- Transformation of DT Filters
- Adaptive Filters
- LMS Algorithm

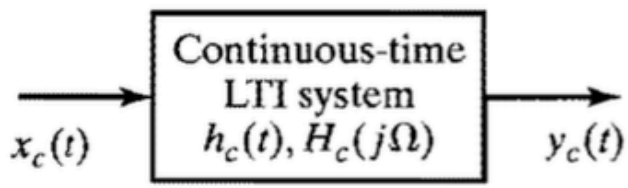
IIR Filter Design

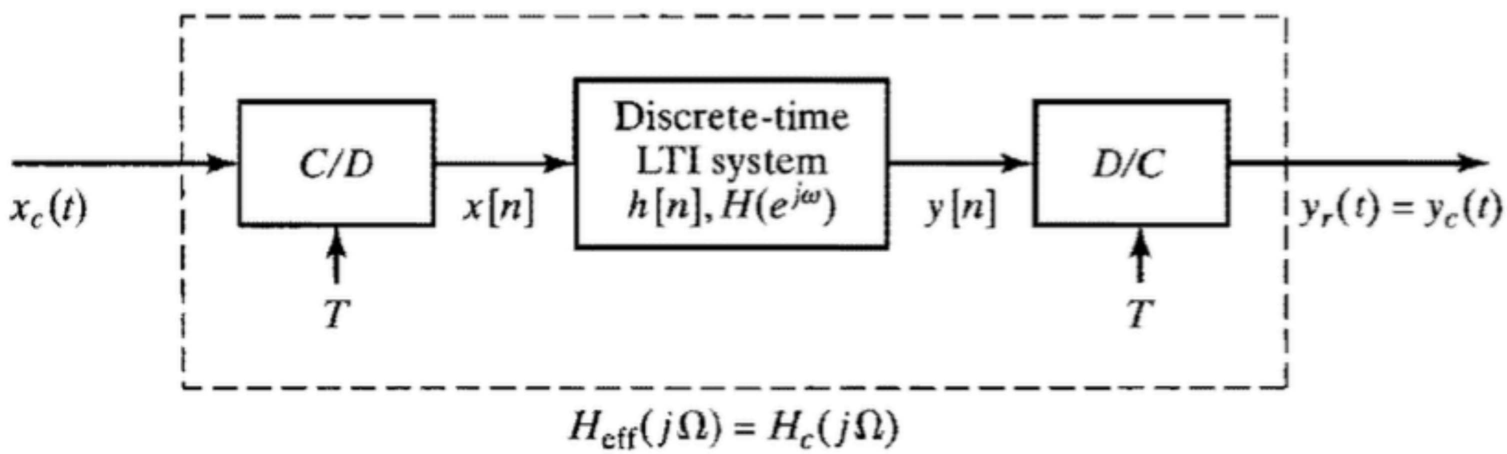
- Transform continuous-time filter into a discrete-time filter meeting specs
  - Pick suitable transformation from s (Laplace variable) to z (or t to n)
  - Pick suitable analog $H_c(s)$ allowing specs to be met, transform to $H(z)$

- We've seen this before… impulse invariance

Impulse Invariance

- Want to implement continuous-time system in discrete-time





Impulse Invariance

- With $H_c(j\Omega)$ bandlimited, choose

$$H(e^{j\omega}) = H_c(j\frac{\omega}{T}), \quad |\omega| < \pi$$

- With the further requirement that T be chosen such that

$$H_c(j\Omega) = 0, \quad |\Omega| \geq \pi / T$$

Impulse Invariance

- With $H_c(j\Omega)$ bandlimited, choose

$$H(e^{j\omega}) = H_c(j\frac{\omega}{T}), \quad |\omega| < \pi$$

- With the further requirement that T be chosen such that

$$H_c(j\Omega) = 0, \quad |\Omega| \geq \pi / T$$

$$\boxed{h[n] = Th_c(nT)}$$

IIR by Impulse Invariance

- If $H_c(j\omega)\approx 0$ for $|\omega_d| > \pi/T$, no aliasing and $H(e^{j\omega}) = H(j\omega/T)$, $\omega<\pi$
- To get a particular $H(e^{j\omega})$, find corresponding H_c and T_d for which above is true (within specs)
- Note: T_d is not for aliasing control, used for frequency scaling.

Example

Example: If $\boxed{H_c(s) = \frac{A_k}{s - p_k}}$ (e.g. one term in PF expansion)

Example

Example: If $\boxed{H_c(s) = \frac{A_k}{s - p_k}}$ (e.g. one term in PF expansion)

$h_c(t) = A_k e^{p_k t}, \quad t \geq 0;$

$$e^{at} \overset{L}{\longleftrightarrow} \frac{1}{s-a}$$

Example

Example: If $\boxed{H_c(s) = \dfrac{A_k}{s - p_k}}$ (e.g. one term in PF expansion)

$$h_c(t) = A_k e^{p_k t}, \quad t \geq 0; \qquad h[n] = T_d A_k e^{p_k T_d n} = T_d A_k \left(e^{p_k T_d}\right)^n$$

Example

Example: If $\boxed{H_c(s) = \frac{A_k}{s - p_k}}$ (e.g. one term in PF expansion)

$h_c(t) = A_k e^{p_k t}, \;\; t \geq 0; \quad h[n] = T_d A_k e^{p_k T_d n} = T_d A_k \left(e^{p_k T_d}\right)^n$

$\therefore \;\; \boxed{H(z) = T_d A_k \frac{1}{1 - e^{p_k T_d} z^{-1}}}$

Pole mapping is $z \leftarrow e^{sT_d}$

Zeros do *not* map the same way; *not* the general mapping of s to z

Example

Example: If $\boxed{H_c(s) = \frac{A_k}{s - p_k}}$ (e.g. one term in PF expansion)

$h_c(t) = A_k e^{p_k t}, \quad t \geq 0; \quad h[n] = T_d A_k e^{p_k T_d n} = T_d A_k \left(e^{p_k T_d}\right)^n$

$\therefore \quad \boxed{H(z) = T_d A_k \frac{1}{1 - e^{p_k T_d} z^{-1}}}$ Pole mapping is $z \leftarrow e^{sT_d}$

Zeros do *not* map the same way; *not* the general mapping of s to z

- Stability, causality, preserved.
- $j\Omega$ axis mapped linearly to unit-circle, with aliasing
- No control of zeros or of phase

Impulse Invariance

- Let,

$$H_c(j\Omega) = 0, \qquad |\Omega| \geq \pi / T$$

$$h[n] = T h_c(nT)$$

- If sampling at Nyquist Rate then

$$H(e^{j\omega}) = \frac{T}{T} \sum_{k=-\infty}^{\infty} H_c\left[j\left(\frac{\omega}{T} - \frac{2\pi k}{T} \right) \right]$$

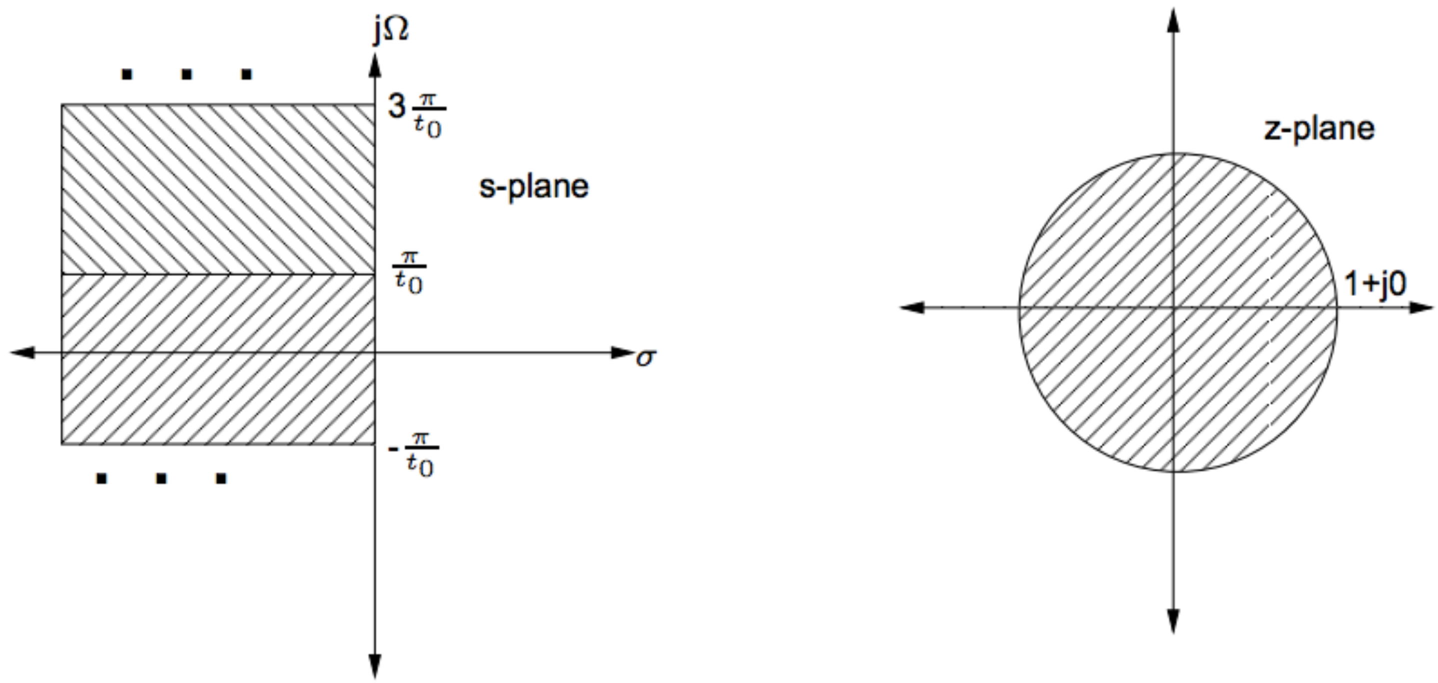
$$H(e^{j\omega}) = \frac{T}{T} H_c\left(j\frac{\omega}{T} \right), \qquad |\omega| < \pi$$

Impulse Invariance

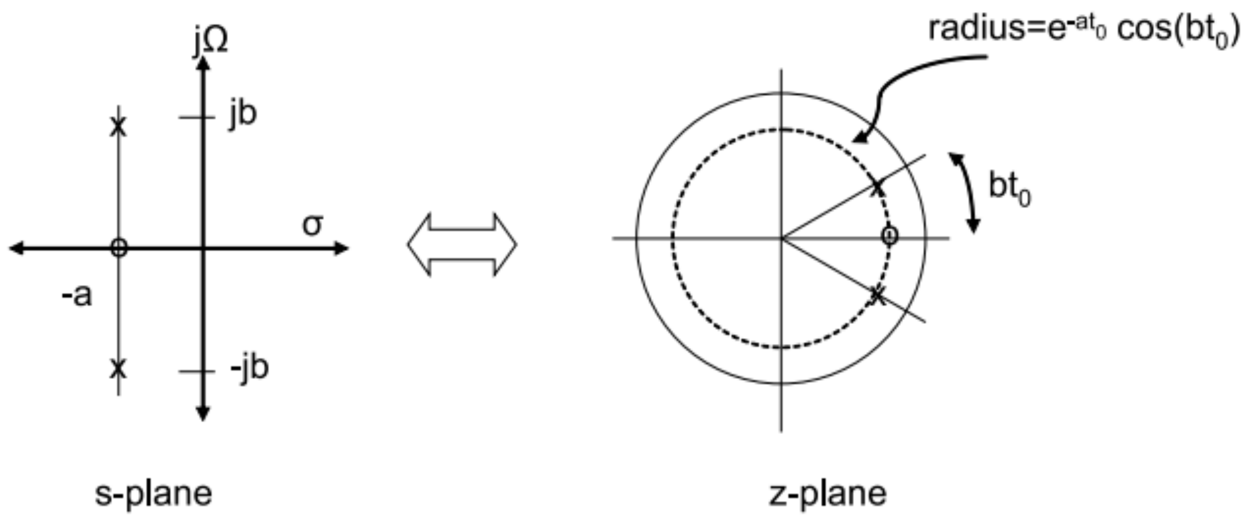
- Sampling the impulse response is equivalent to mapping the s-plane to the z-plane using:
  - $z = e^{st0} = r\ e^{j\omega}$
- The entire Ω axis of the s-plane wraps around the unit circle of the z-plane an infinite number of times
- The negative half s-plane maps to the interior of the unit circle and the RHP to the exterior
- This means stable analog filters (poles in LHP) will transform to stable digital filters (poles inside unit circle)
- This is a many-to-one mapping of strips of the s-plane to regions of the z-plane
  - Not a conformal mapping
  - The poles map according to $z = e^{st0}$, but the zeros do not always

Impulse Invariance Mapping

Mapping

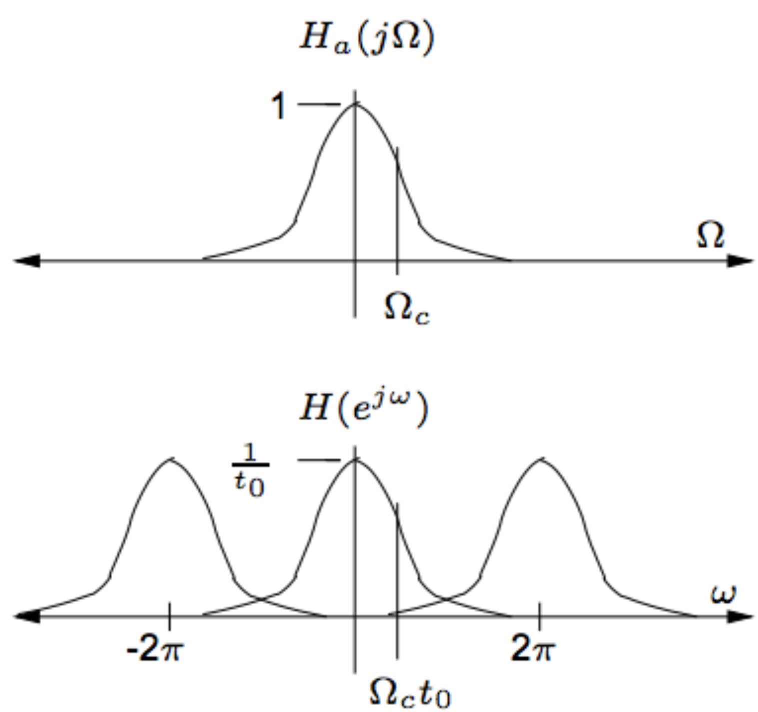


Impulse Invariance Mapping



Impulse Invariance

- Limitation of Impulse Invariance: overlap of images of the frequency response. This prevents it from being used for high-pass filter design



Bilinear Transformation

- The technique uses an algebraic transformation between the variables s and *z* that maps the entire jΩ-axis in the s-plane to one revolution of the unit circle in the *z*-plane.

$$s = \frac{2}{T_d}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right);$$

$$H(z) = H_c\left(\frac{2}{T_d}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)\right).$$

Bilinear Transformation

$$s = \frac{2}{T_d}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right);$$

- Substituting $s = \sigma + j\,\Omega$ and $z = e^{j\omega}$

$$s = \frac{2}{T_d}\left(\frac{1 - e^{-j\omega}}{1 + e^{-j\omega}}\right),$$

$$s = \sigma + j\Omega = \frac{2}{T_d}\left[\frac{2e^{-j\omega/2}(j\sin\omega/2)}{2e^{-j\omega/2}(\cos\omega/2)}\right] = \frac{2j}{T_d}\tan(\omega/2).$$

Bilinear Transformation

$$\Omega = \frac{2}{T_d} \tan(\omega/2).$$

$$\omega = 2 \arctan(\Omega T_d/2).$$

Bilinear Transformation

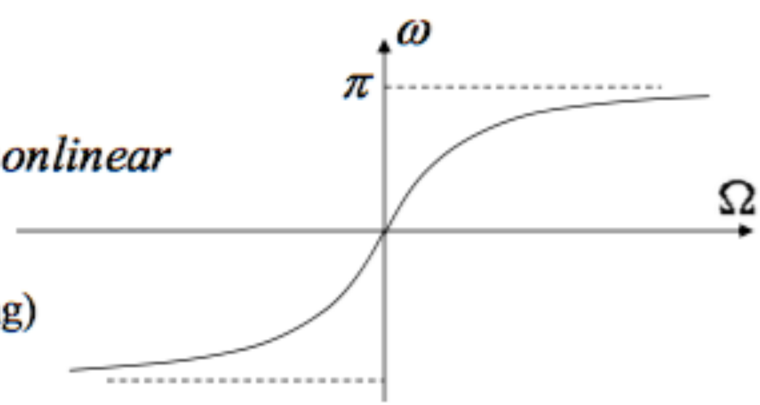
$$\Omega = \frac{2}{T_d}\tan(\omega/2),$$

$$\omega = 2\arctan(\Omega T_d/2).$$

No aliasing, but mapping nonlinear

(Impulse invariance:

linear mapping, but with aliasing)



Example: Notch Filter

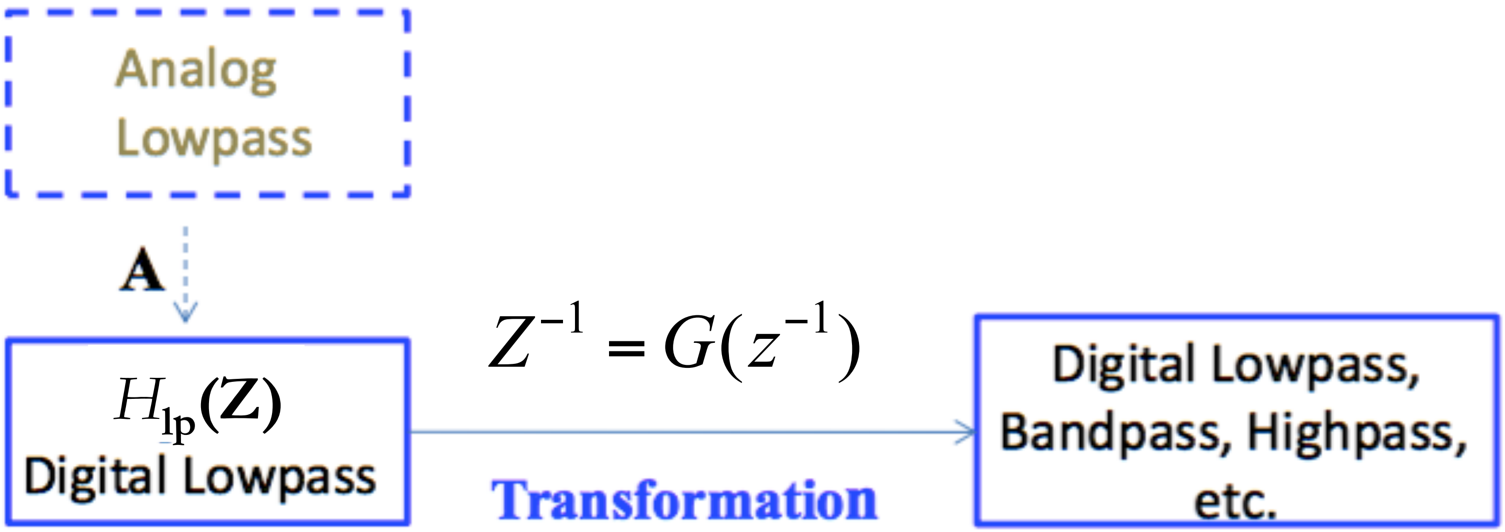
- The continuous time filter with:

$$H_a(s) = \frac{s^2 + \Omega_0^2}{s^2 + Bs + \Omega_0^2}$$

$$\Omega = \frac{2}{T_d} \tan(\omega/2),$$

$$\omega = 2 \arctan(\Omega T_d/2).$$

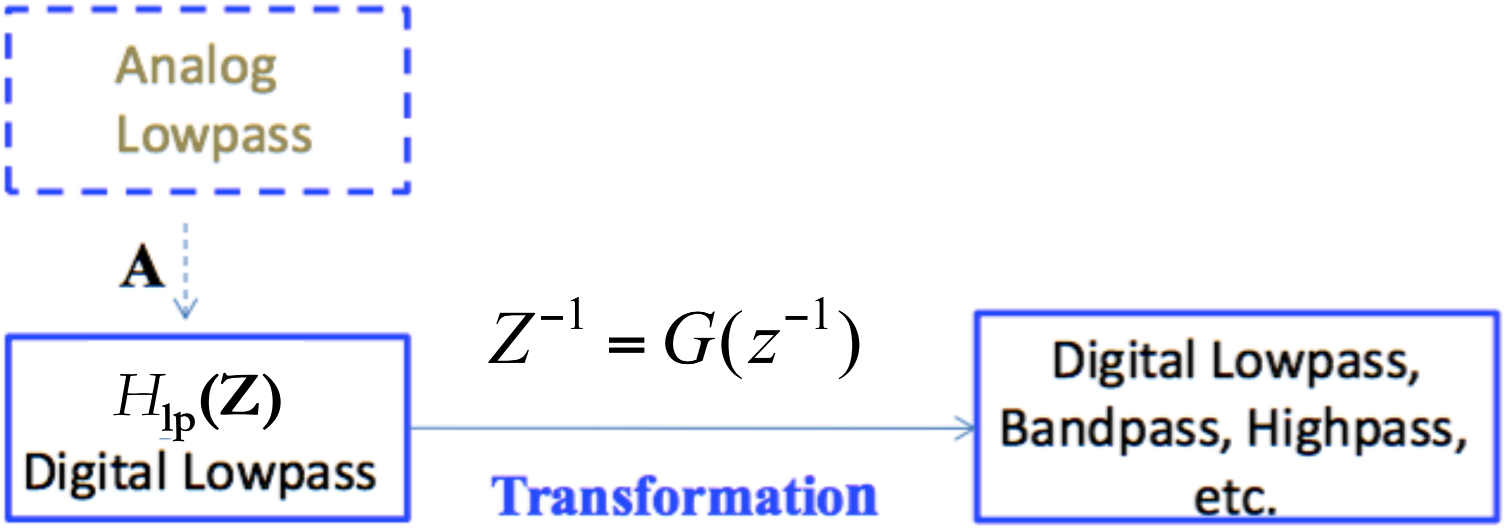
Transformation of DT Filters



- Z – complex variable for the LP filter
- z – complex variable for the transformed filter

- Map Z-plane→z-plane with transformation G

Transformation of DT Filters



- Map Z-plane→z-plane with transformation G

$$H(z) = H_{\mathrm{lp}}(Z)\big|_{Z^{-1}=G(z^{-1})}$$

Example 1:

- Lowpass→highpass
  - Shift frequency by π

so $\omega \to \omega - \pi$ (Lowpass to highpass)

$$Z^{-1} = -z^{-1} \quad \text{or} \quad e^{-j\omega} \to e^{-j(\omega-\pi)}$$

Example 1:

- Lowpass→highpass
  - Shift frequency by π

so $\omega \to \omega - \pi$ (Lowpass to highpass)

$$Z^{-1} = -z^{-1} \quad \text{or} \quad e^{-j\omega} \to e^{-j(\omega-\pi)}$$

$$\boxed{y[n] = 0.9y[n-1] + 0.1x[n]} \quad \text{lowpass;} \quad \textit{pole}: z = 0.9, \quad H(z) = \frac{0.1}{1-0.9z^{-1}}$$

$$H(-z) = \frac{0.1}{1+0.9z^{-1}} \quad \text{highpass;} \quad \textit{pole}: z = -0.9 \quad \boxed{y[n] = -0.9y[n-1] + 0.1x[n]}$$

Example 2:

- Lowpass→bandpass

$$Z^{-1} = -z^{-2}$$

$$H_{lp}(z) = \frac{1}{1 - az^{-1}} \longrightarrow H_{bp}(z) = \frac{1}{1 + az^{-2}}$$

Pole at z=a

Pole at z=±j√a

Example 2:

- Lowpass→bandpass

$$Z^{-1} = -z^{-2}$$

$$H_{lp}(z) = \frac{1}{1 - az^{-1}} \longrightarrow H_{bp}(z) = \frac{1}{1 + az^{-2}}$$

Pole at z=a

Pole at z=±j√a

- Lowpass→bandstop

$$Z^{-1} = z^{-2}$$

$$H_{lp}(z) = \frac{1}{1 - az^{-1}} \longrightarrow H_{bs}(z) = \frac{1}{1 - az^{-2}}$$

Pole at z=±√a

Transformation Constraints on $G(z^{-1})$

- If $H_{lp}(Z)$ is the rational system function of a causal and stable system, we naturally require that the transformed system function $H(z)$ be a rational function and that the system also be causal and stable.
  - $G(Z^{-1})$ must be a rational function of z^{-1}
  - The inside of the unit circle of the Z-plane must map to the inside of the unit circle of the z-plane
  - The unit circle of the Z-plane must map onto the unit circle of the z-plane.

Transformation Constraints on $G(z^{-1})$

- Respective unit circles in both planes

$$Z = e^{j\theta} \text{ and } z = e^{j\omega}$$

Transformation Constraints on $G(z^{-1})$

- Respective unit circles in both planes

$$Z = e^{j\theta} \text{ and } z = e^{j\omega}$$

$$Z^{-1} = G(z^{-1})$$

$$e^{-j\theta} = G(e^{-j\omega})$$

$$e^{-j\theta} = \left|G(e^{-j\omega})\right| e^{j\angle G(e^{-j\omega})}$$

Transformation Constraints on $G(z^{-1})$

- Respective unit circles in both planes

$$Z = e^{j\theta} \text{ and } z = e^{j\omega}$$

$$Z^{-1} = G(z^{-1})$$

$$e^{-j\theta} = G(e^{-j\omega})$$

$$e^{-j\theta} = \left|G(e^{-j\omega})\right| e^{j\angle G(e^{-j\omega})}$$

$$1 = \left|G(e^{-j\omega})\right| \qquad -\theta = \angle G(e^{-j\omega})$$

Transformation Constraints on $G(z^{-1})$

- General form that meets all constraints:
  - a_k real and $|a_k|<1$

$$G(z^{-1}) = \pm \prod_{k=1}^{N} \frac{z^{-1} - \alpha_k}{1 - \alpha_k z^{-1}}$$

General Transformation

- Lowpass→lowpass

$$G(z^{-1}) = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$$

- Changes passband/stopband edge frequencies

General Transformation

- Lowpass→lowpass

$$G(z^{-1}) = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$$

- Changes passband/stopband edge frequencies

$$\text{From } e^{-j\theta} = \frac{e^{-j\omega} - \alpha}{1 - \alpha e^{-j\omega}}, \text{ get}$$

$$\omega(\theta) = \tan^{-1}\left(\frac{(1-\alpha^2)\sin(\theta)}{2\alpha + (1+\alpha^2)\cos(\theta)} \right)$$

General Transformation

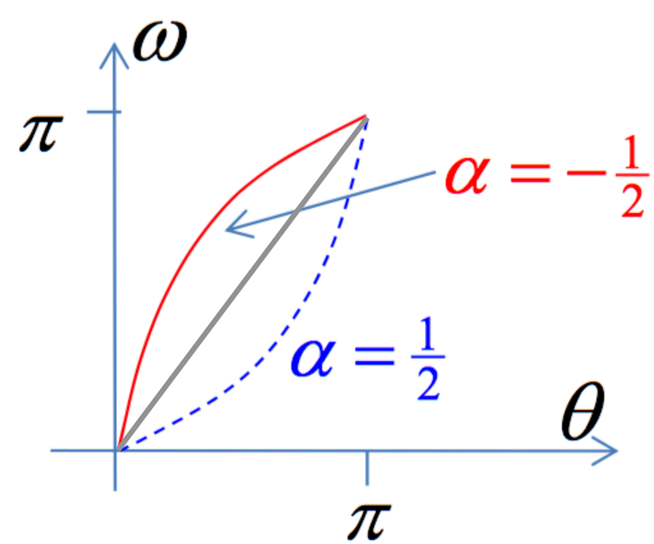
- Lowpass→lowpass

$$G(z^{-1}) = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$$

- Changes passband/stopband edge frequencies

$$\text{From } e^{-j\theta} = \frac{e^{-j\omega} - \alpha}{1 - \alpha e^{-j\omega}}, \text{ get}$$

$$\omega(\theta) = \tan^{-1}\left(\frac{(1-\alpha^2)\sin(\theta)}{2\alpha + (1+\alpha^2)\cos(\theta)}\right)$$



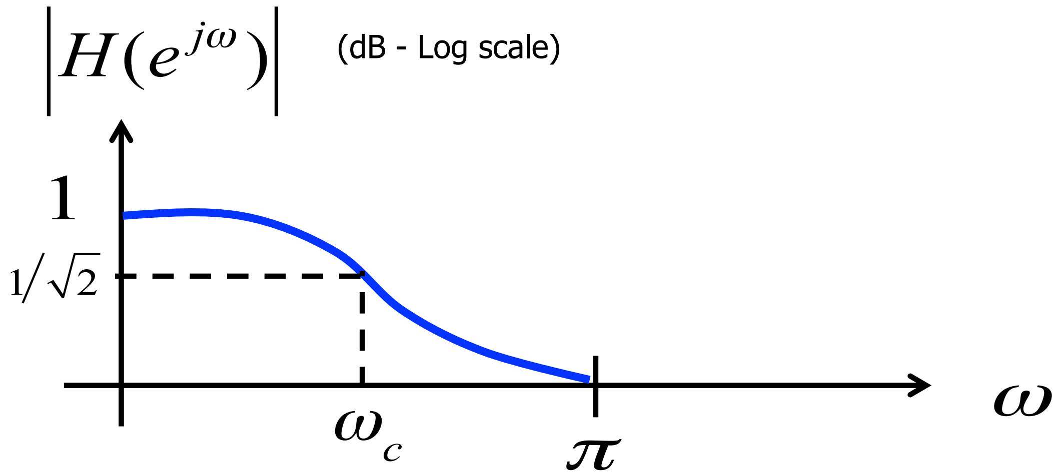
General Transformations

TABLE 7.1 TRANSFORMATIONS FROM A LOWPASS DIGITAL FILTER PROTOTYPE OF CUTOFF FREQUENCY θ_p TO HIGHPASS, BANDPASS, AND BANDSTOP FILTERS

Filter Type	Transformations	Associated Design Formulas
Lowpass	$Z^{-1} = \frac{z^{-1} - \alpha}{1 - az^{-1}}$	$\alpha = \frac{\sin\left(\frac{\theta_p - \omega_p}{2}\right)}{\sin\left(\frac{\theta_p + \omega_p}{2}\right)}$ ω_p = desired cutoff frequency
Highpass	$Z^{-1} = -\frac{z^{-1} + \alpha}{1 + \alpha z^{-1}}$	$\alpha = -\frac{\cos\left(\frac{\theta_p + \omega_p}{2}\right)}{\cos\left(\frac{\theta_p - \omega_p}{2}\right)}$ ω_p = desired cutoff frequency
Bandpass	$Z^{-1} = -\frac{z^{-2} - \frac{2\alpha k}{k+1} z^{-1} + \frac{k-1}{k+1}}{\frac{k-1}{k+1} z^{-2} - \frac{2\alpha k}{k+1} z^{-1} + 1}$	$\alpha = \frac{\cos\left(\frac{\omega_{p2} + \omega_{p1}}{2}\right)}{\cos\left(\frac{\omega_{p2} - \omega_{p1}}{2}\right)}$ $k = \cot\left(\frac{\omega_{p2} - \omega_{p1}}{2}\right) \tan\left(\frac{\theta_p}{2}\right)$ ω_{p1} = desired lower cutoff frequency ω_{p2} = desired upper cutoff frequency
Bandstop	$Z^{-1} = \frac{z^{-2} - \frac{2\alpha}{1+k} z^{-1} + \frac{1-k}{1+k}}{\frac{1-k}{1+k} z^{-2} - \frac{2\alpha}{1+k} z^{-1} + 1}$	$\alpha = \frac{\cos\left(\frac{\omega_{p2} + \omega_{p1}}{2}\right)}{\cos\left(\frac{\omega_{p2} - \omega_{p1}}{2}\right)}$ $k = \tan\left(\frac{\omega_{p2} - \omega_{p1}}{2}\right) \tan\left(\frac{\theta_p}{2}\right)$ ω_{p1} = desired lower cutoff frequency ω_{p2} = desired upper cutoff frequency

Reminder: Simple Low Pass Filter

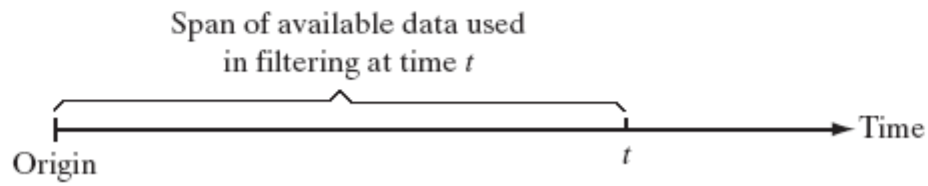
$$H_{LP}(z) = \frac{1-\alpha}{2}\frac{1+z^{-1}}{1-\alpha z^{-1}} \qquad |\alpha| < 1$$



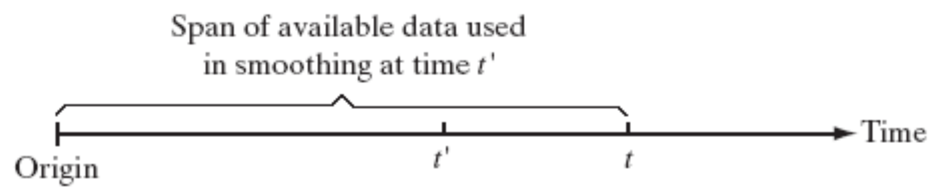
w_c is the 3dB cutoff frequency $\qquad \alpha = \frac{1-\sin(\omega_c)}{\cos(\omega_c)}$

The Filtering Problem

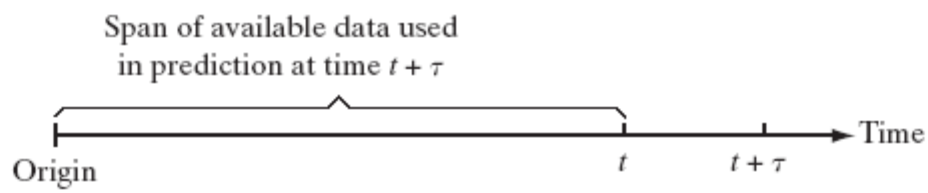
- Filters may be used for three information-processing tasks
  - Filtering



  - Smoothing



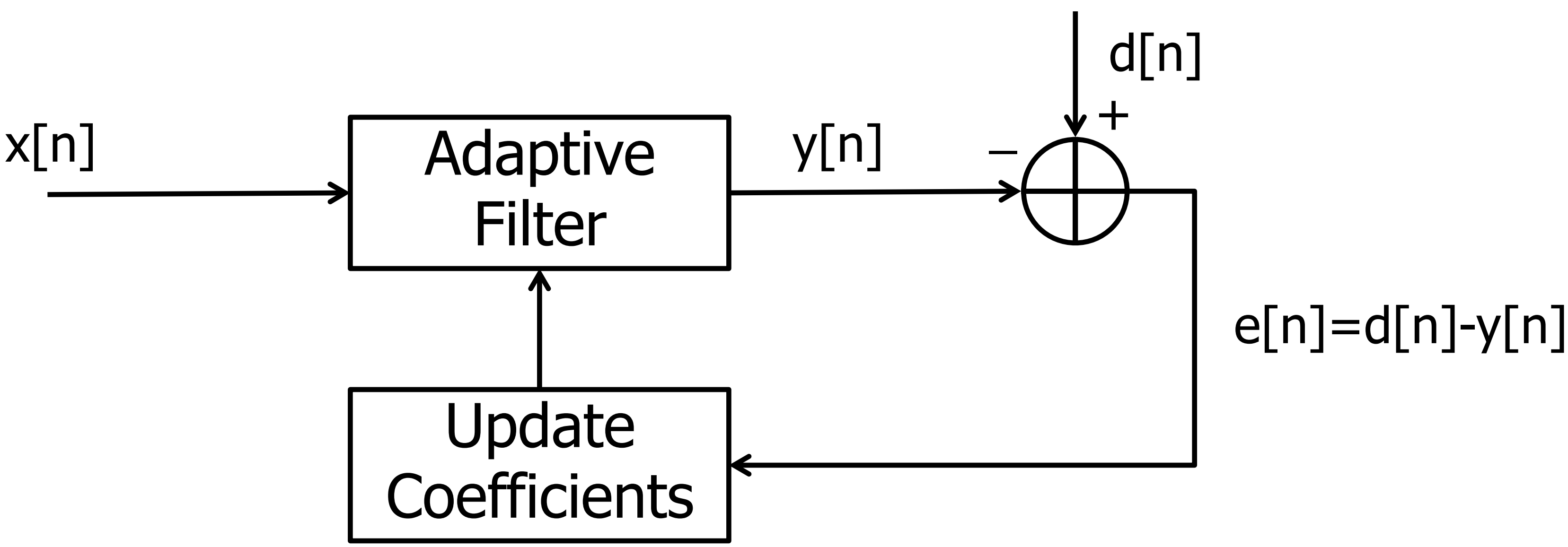
  - Prediction



- Given an optimality criteria we often can design optimal filters
  - Requires a priori information about the environment
- Adaptive filters are self-designing using a recursive algorithm
  - Useful if complete knowledge of environment is not available a priori

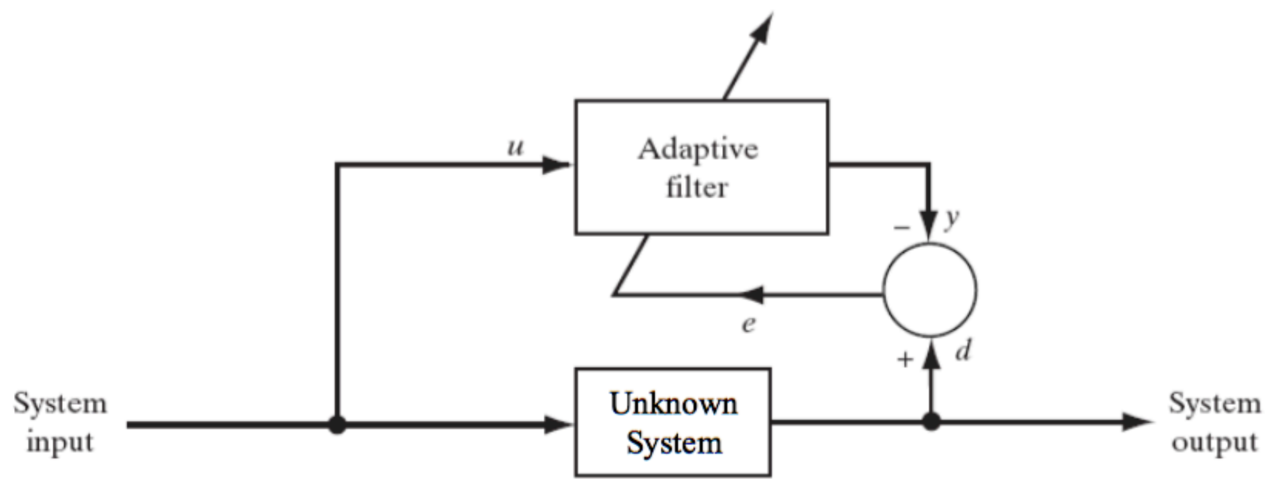
Adaptive Filters

- An adaptive filter is an adjustable filter that processes in time
  - It adapts…



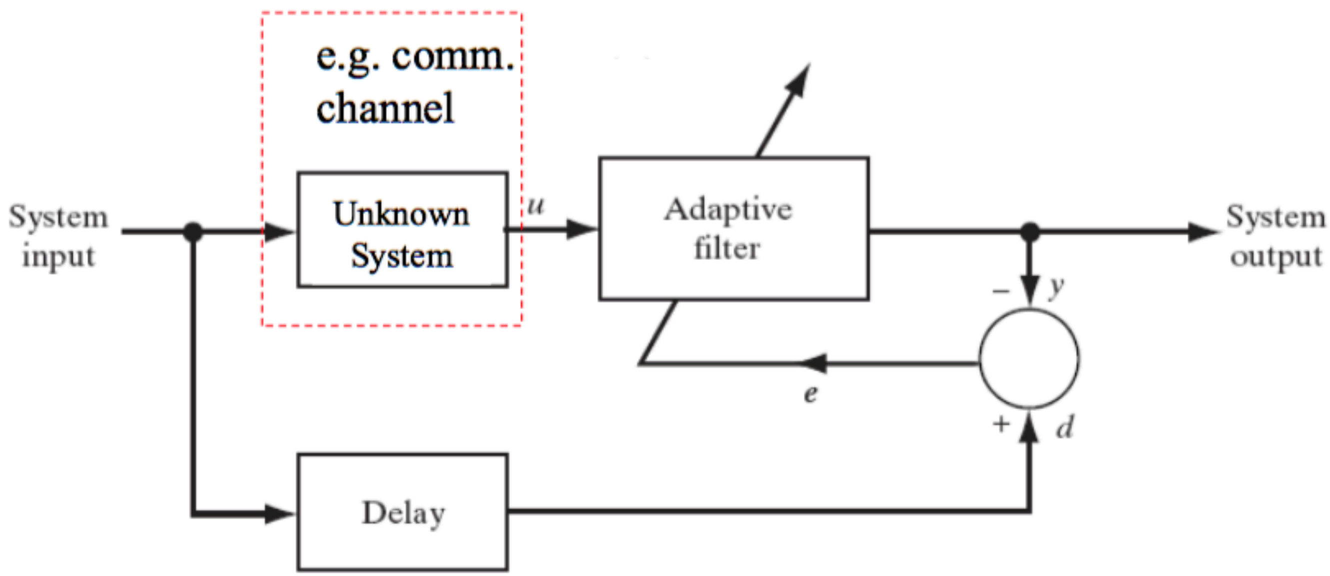
Adaptive Filter Applications

- System Identification



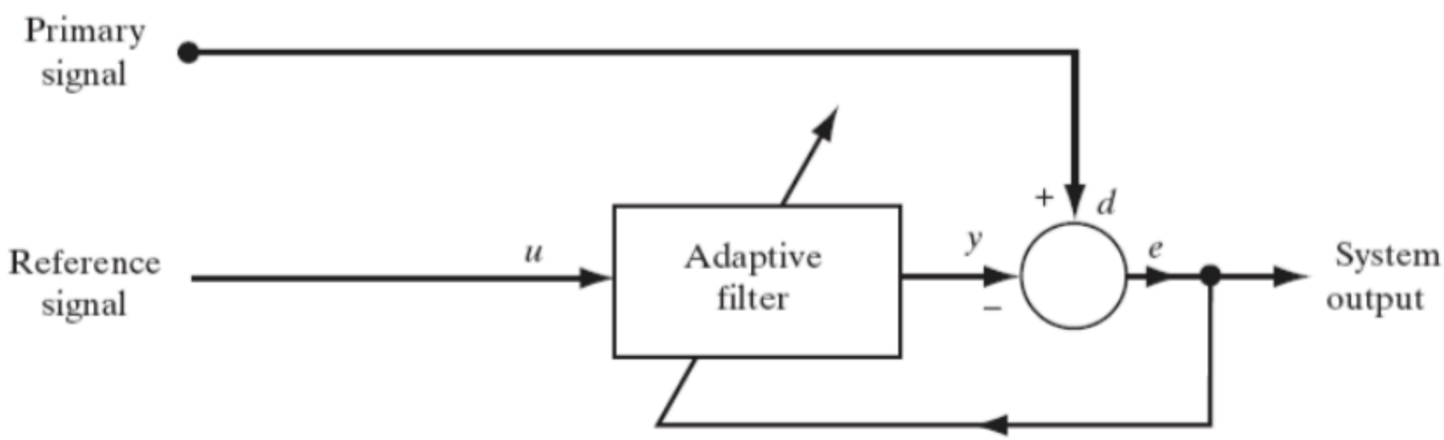
Adaptive Filter Applications

- Identification of inverse system



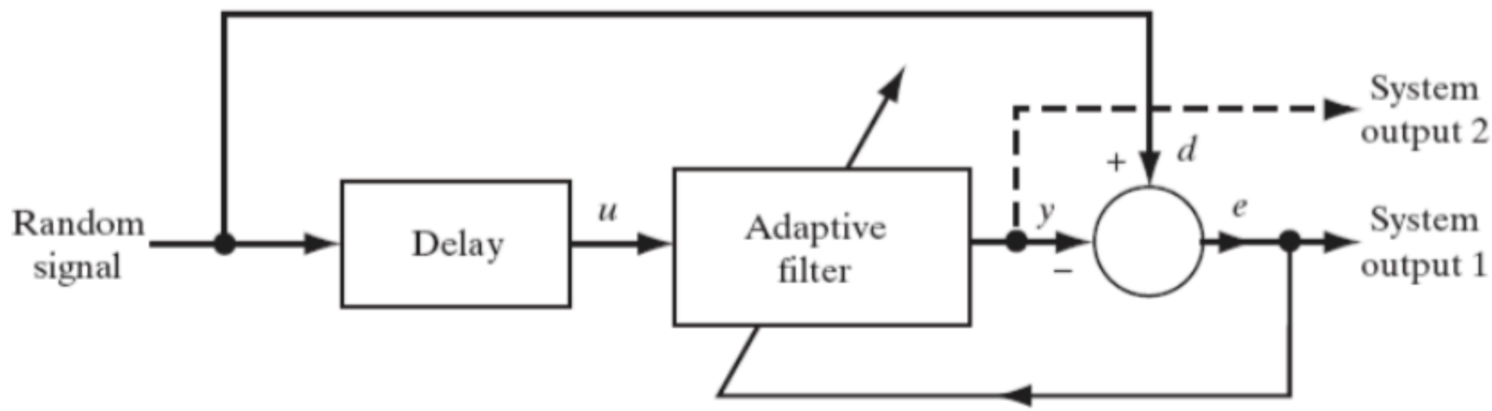
Adaptive Filter Applications

- Adaptive Interference Cancellation



Adaptive Filter Applications

- Adaptive Prediction

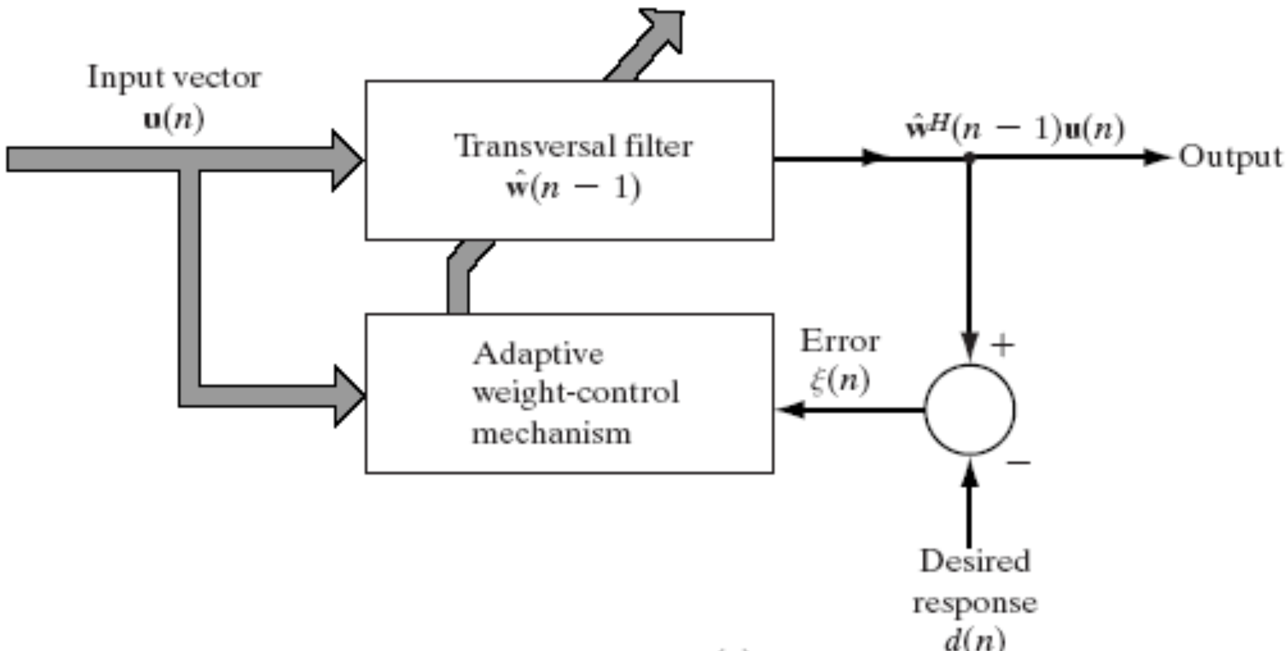


Stochastic Gradient Approach

- Most commonly used type of Adaptive Filters
- Define cost function as mean-squared error
    - Difference between filter output and desired response
- Based on the method of steepest descent
    - Move towards the minimum on the error surface to get to minimum
    - Requires the gradient of the error surface to be known

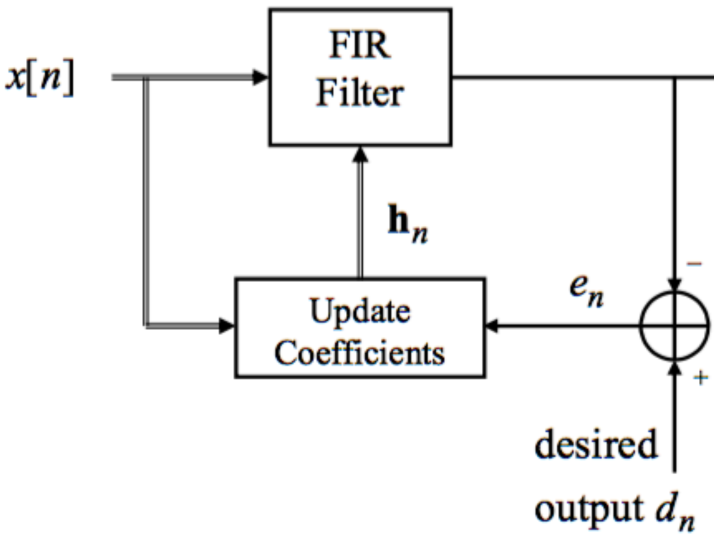
Least-Mean-Square (LMS) Algorithm

- The LMS Algorithm consists of two basic processes
  - Filtering process
    - Calculate the output of FIR filter by convolving input and taps
    - Calculate estimation error by comparing the output to desired signal
  -



(a)

Adaptive FIR Filter: LMS

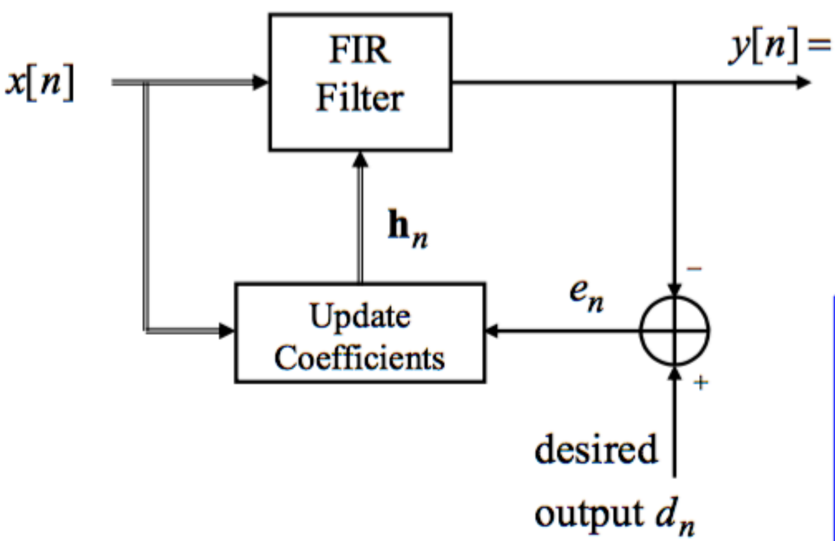


$$y[n] = \mathbf{h}_n^T \mathbf{x}_n$$

$$\mathbf{x}_n = (x[n], x[n-1]\ldots, x[n-M])^T$$

$$\mathbf{h}_n = (h_n[0], h_n[1], \ldots, h_n[M])^T$$

Adaptive FIR Filter: LMS



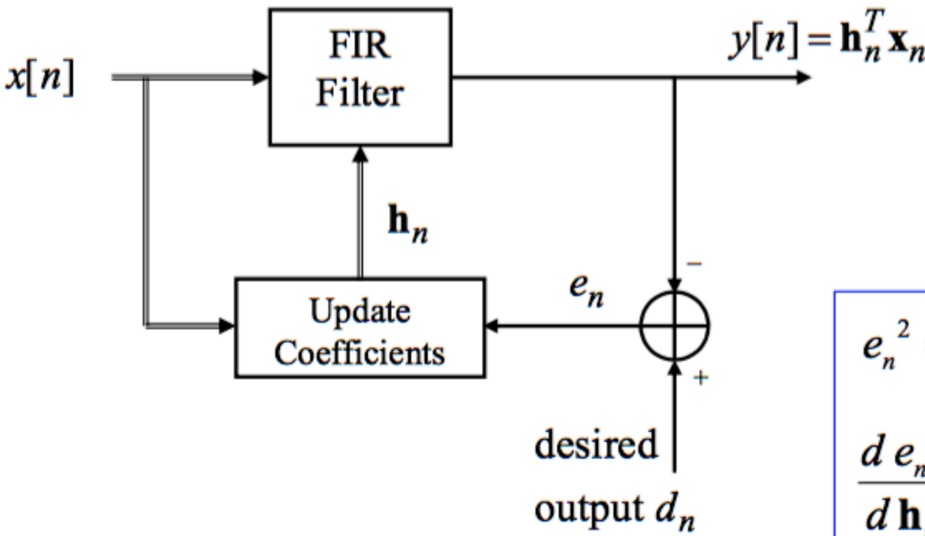
$$\mathbf{x}_n = (x[n], x[n-1] \ldots, x[n-M])^T$$

$$\mathbf{h}_n = (h_n[0], h_n[1], \ldots, h_n[M])^T$$

$$e_n^{\ 2} = (d[n] - y[n])^2 = (d[n] - \mathbf{h}_n^T \mathbf{x}_n)^2$$

$$\frac{d\, e_n^{\ 2}}{d\, \mathbf{h}_n} = -2(d[n] - \mathbf{h}_n^T \mathbf{x}_n)\mathbf{x}_n = -2 e_n \mathbf{x}_n$$

Adaptive FIR Filter: LMS



$$\mathbf{x}_n = (x[n], x[n-1] \ldots, x[n-M])^T$$

$$\mathbf{h}_n = (h_n[0], h_n[1], \ldots, h_n[M])^T$$

$$e_n^2 = (d[n] - y[n])^2 = (d[n] - \mathbf{h}_n^T \mathbf{x}_n)^2$$

$$\frac{d\, e_n^2}{d\, \mathbf{h}_n} = -2(d[n] - \mathbf{h}_n^T \mathbf{x}_n)\mathbf{x}_n = -2 e_n \mathbf{x}_n$$

Coefficient Update: Move in direction *opposite* to sign of gradient, proportional to magnitude of gradient

$$\mathbf{h}_{n+1} = \mathbf{h}_n + 2\mu e_n \mathbf{x}_n$$

Stochastic Gradient Algorithm

LMS Algorithm Steps

- Filter output

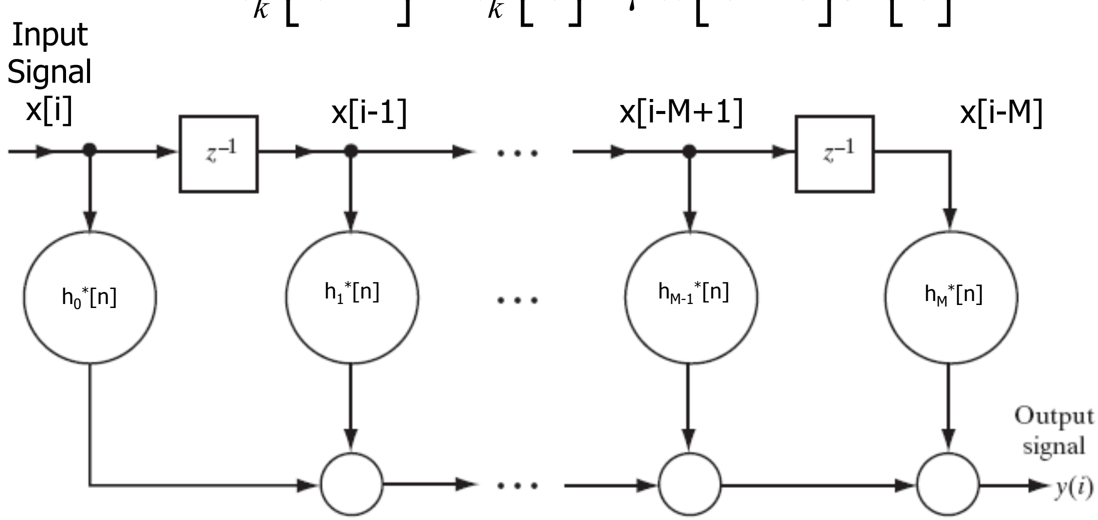
$$y[n]=\sum_{k=0}^{M-1} x[n-k]h_k^*[n]$$

- Estimation error

$$e[n]=d[n]-y[n]$$

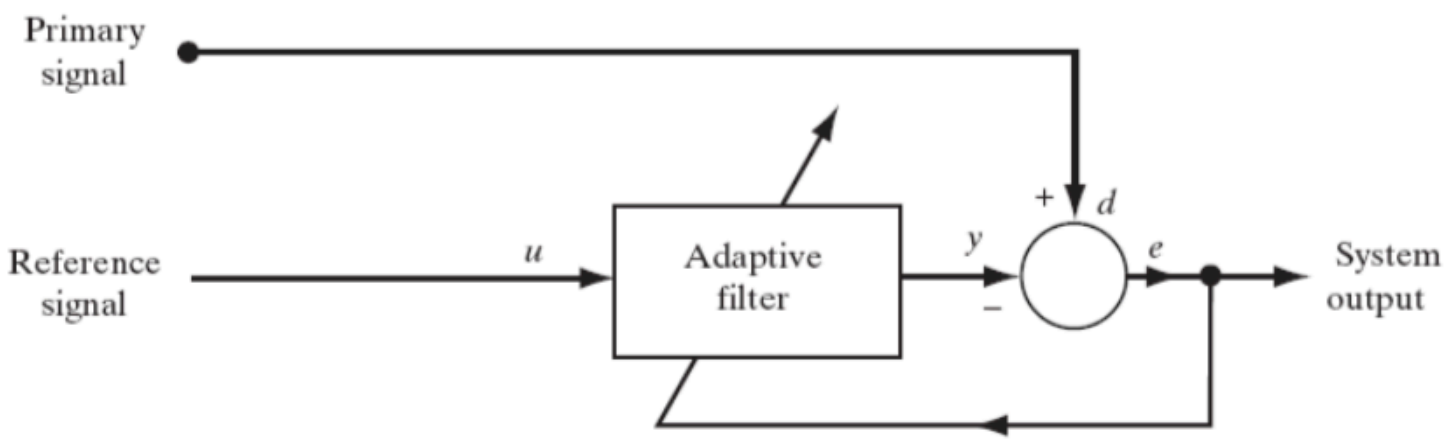
- Tap-weight adaptation

$$h_k[n+1]=h_k[n]+\mu x[n-k]e^*[n]$$

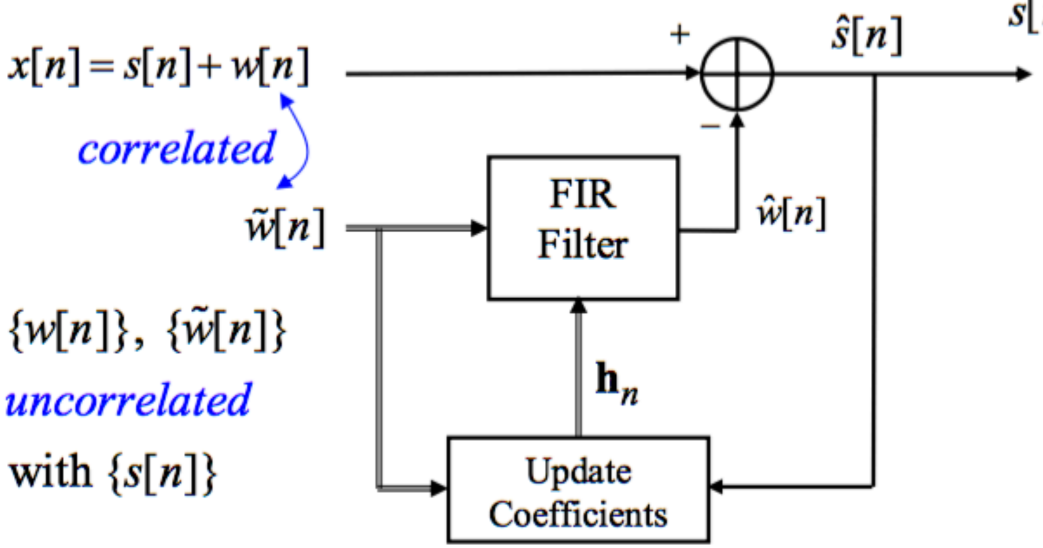


Adaptive Filter Applications

- Adaptive Interference Cancellation



Adaptive Interference Cancellation



$$\hat{s}[n] = x[n] - \mathbf{h}_n^T \tilde{\mathbf{w}}_n$$
$$= s[n] + \{w[n] - \mathbf{h}_n^T \tilde{\mathbf{w}}_n\}$$

uncorrelated

Minimize *instantaneous power* $\hat{s}^2[n]$ for best interference cancellation

$$\frac{d\left(\hat{s}[n]\right)^2}{d\,\mathbf{h}_n} = -2\hat{s}[n]\,\tilde{\mathbf{w}}_n$$

$$\boxed{\mathbf{h}_{n+1} = \mathbf{h}_n + 2\mu\,\hat{s}[n]\,\tilde{\mathbf{w}}_n}$$

Stability of LMS

- The LMS algorithm is convergent in the mean square if and only if the step-size parameter satisfy

$$0 < \mu < \frac{2}{\lambda_{max}}$$

- Here λ_{max} is the largest eigenvalue of the correlation matrix of the input data
- More practical test for stability is

$$0 < \mu < \frac{2}{\text{input signal power}}$$

- Larger values for step size
  - Increases adaptation rate (faster adaptation)
  - Increases residual mean-squared error

Admin

- HW 7 posted after class tonight
  - Due 4/6