

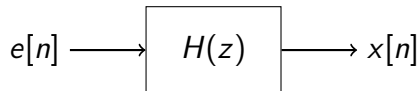
Summary of lecture 12

Tomas McKelvey

Department of Electrical Engineering
Chalmers University of Technology

Definition of ARMA

- Suppose $e[n]$ is a white noise process (zero mean, WSS).
- We model $x[n]$ as the output from a linear system



- $x[n]$ is an ARMA process if

$$H(z) = \frac{B(z)}{A(z)} = \frac{1 + b_1 z^{-1} + \dots + b_q z^{-q}}{1 + a_1 z^{-1} + \dots + a_p z^{-p}}$$

or, equivalently, if

$$x[n] + \dots + a_p x[n-p] = e[n] + \dots + b_q e[n-q] \quad (1)$$

- $x[n]$ is 1) AR(p) if $B(z) = 1$ 2) MA(q) if $A(z) = 1$

PSD of ARMA

- $e[n]$ is white noise $\Rightarrow r_e[n] = \delta[n]\sigma_e^2 \leftrightarrow P_e(e^{j\omega}) = \sigma_e^2$.
- We get

$$P_x(e^{j\omega}) = P_e(e^{j\omega}) |H(e^{j\omega})|^2 = \sigma_e^2 |H(e^{j\omega})|^2$$

\rightsquigarrow the filter $H(z)$ shapes the PSD of $x[n]$

- Examples indicate that
 - AR good for peaks
 - MA useful for notches

whereas ARMA can combine the strengths.

ACF of ARMA

- Techniques to compute the ACF:

- Find the PSD as above and compute

$$r_x[n] = IDTFT \{P_x(e^{j\omega})\}$$

↪ general solution, but often rather complicated.

- Derive linear equations from the difference equation (1)

↪ simple solution, but only works for AR and MA processes.

ACF of AR-processes

- Consider an AR(p)-process

$$x[n] + a_1x[n-1] + \cdots + a_px[n-p] = e[n] \quad (2)$$

- Multiply by $x[n-k]$, $k \geq 0$, and take expectations:

Yule-Walker (YW) equations

$$r_x[k] + a_1r_x[k-1] + \cdots + a_pr_x[k-p] = \sigma^2 \delta[k]$$

- Note:** 1) YW are linear in the autocorrelation function, $r_x[n]$
2) by changing k , we can collect any number of equations.
 \Rightarrow easy to find $r_x[n]$ using YW

ACF of AR-processes: using the YW eq's

- Suppose we know a_1, \dots, a_p and σ_e^2 .
- If we seek $r_x[k]$ for $k = 0, 1, \dots, p$: use YW for $k = 0, 1, \dots, p$
- Let us illustrate these equations when $p = 3$,

$$\begin{bmatrix} 1 & a_1 & a_2 & a_3 \\ a_1 & 1 + a_2 & a_3 & 0 \\ a_2 & a_1 + a_3 & 1 & 0 \\ a_3 & a_2 & a_1 & 1 \end{bmatrix} \begin{bmatrix} r_x[0] \\ r_x[1] \\ r_x[2] \\ r_x[3] \end{bmatrix} = \begin{bmatrix} \sigma_e^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

- **Remarks:**
 - ① $p + 1$ linear equations with $p + 1$ unknowns \Rightarrow easy to solve!
 - ② $r_x[k]$, $k > p$ can be found by using more YW-equations
 - ③ No need to memorize the matrix equation \rightsquigarrow better to know how to derive them!

ACF of MA-processes

- Consider an MA(q)-process

$$x[n] = e[n] + b_1 e[n-1] + \cdots + b_q e[n] \quad (3)$$

- Multiply by $x[n-k]$, $k \geq 0$, and take expectations:

Solution for $r_x[n]$ of an MA(q)-processes

(where $b_0 = 1$ and $b_i = 0$ for $i > q$ or $i < 0$),

$$r_x[k] = \begin{cases} \sum_{i=0}^q \sigma_e^2 b_i b_{i-k} & \text{if } |k| \leq q \\ 0 & \text{otherwise.} \end{cases}$$

- Note:** 1) we do not have to solve an equation system
2) q is often small and then it is easier to rederive the expression than to memorize it!