Parametric Signal Modeling and Linear Prediction Theory
1. Discrete-time Stochastic Processes (2)

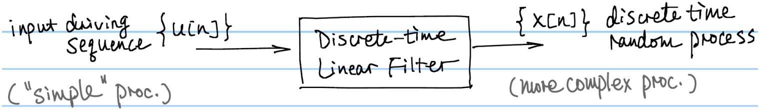
Electrical & Computer Engineering
University of Maryland, College Park

Acknowledgment: ENEE630 slides were based on class notes developed by Profs. K.J. Ray Liu and Min Wu. The LaTeX slides were made by Prof. Min Wu and Mr. Wei-Hong Chuang.

Contact: *minwu@umd.edu*. Updated: October 25, 2011.

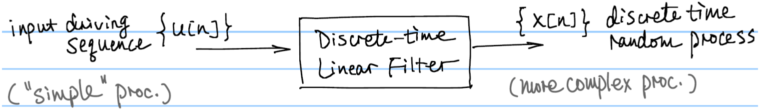
(1) The Rational Transfer Function Model

Many discrete-time random processes encountered in practice can be well approximated by a rational function model (Yule 1927).



Readings: Haykin 4th Ed. 1.5

The Rational Transfer Function Model



Typically $u[n]$ is a noise process, gives rise to randomness of $x[n]$.

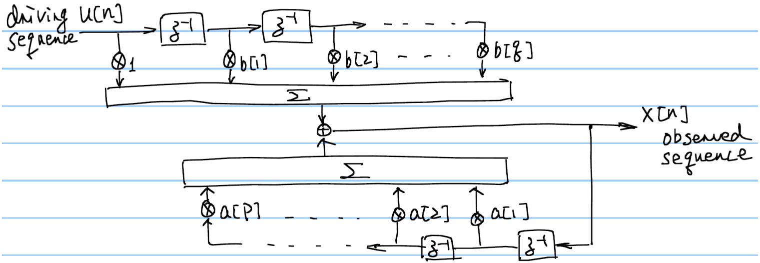
The input driving sequence $u[n]$ and the output sequence $x[n]$ are related by a linear constant-coefficient difference equation

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + \sum_{k=0}^{q} b[k]u[n-k]$$

This is called the autoregressive-moving average (ARMA) model:

- autoregressive on previous outputs
- moving average on current & previous inputs

The Rational Transfer Function Model



The system transfer function

$$H(z) \triangleq \frac{X(z)}{U(z)} = \frac{\sum_{k=0}^{q} b[k]z^{-k}}{\sum_{k=0}^{p} a[k]z^{-k}} \triangleq \frac{B(z)}{A(z)}$$

To ensure the system's stationarity, $a[k]$ must be chosen s.t. all poles are inside the unit circle.

(2) Power Spectral Density of ARMA Processes

Recall the relation in autocorrelation function and p.s.d. after filtering:

$$r_x[k] = h[k] * h^*[-k] * r_u[k]$$
$$P_x(z) = H(z)H^*(1/z^*)P_U(z)$$
$$\Rightarrow P_x(\omega) = |H(\omega)|^2 P_U(\omega)$$

$\{u[n]\}$ is often chosen as a white noise process with zero mean and variance σ^2, then $P_{\text{ARMA}}(\omega) \triangleq P_X(\omega) = \sigma^2 |\frac{B(\omega)}{A(\omega)}|^2$,
i.e., the p.s.d. of $x[n]$ is determined by $|H(\omega)|^2$.

We often pick a filter with $a[0] = b[0] = 1$ (normalized gain)

The random process produced as such is called an ARMA(p, q) process, also often referred to as a pole-zero model.

(3) MA and AR Processes

MA Process

If in the ARMA model $a[k] = 0 \ \forall k > 0$, then

$$x[n] = \sum_{k=0}^{q} b[k]u[n-k]$$

This is called an MA(q) process with $P_{\text{MA}}(\omega) = \sigma^2 |B(\omega)|^2$. It is also called an all-zero model.

AR Process

If $b[k] = 0 \ \forall k > 0$, then

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + u[k]$$

This is called an AR(p) process with $P_{\text{AR}}(\omega) = \frac{\sigma^2}{|A(\omega)|^2}$. It is also called an all-pole model.

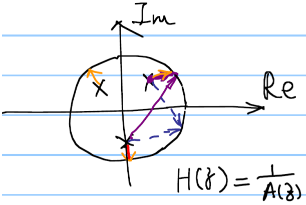
$$H(z) = \frac{1}{(1-c_1 z^{-1})(1-c_2 z^{-1})\cdots(1-c_p z^{-1})}$$

(4) Power Spectral Density: AR Model

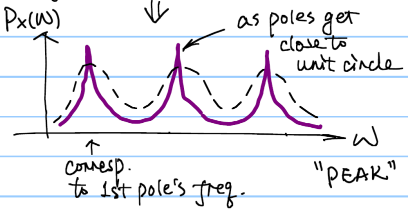
$$\text{ZT:} \quad P_X(z) = \sigma^2 H(z) H^*(1/z^*) = \sigma^2 \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}$$

$$\text{p.s.d.:} \quad P_X(\omega) = P_X(z)|_{z=e^{j\omega}} = \sigma^2 |H(\omega)|^2 = \sigma^2 |\frac{B(\omega)}{A(\omega)}|^2$$

- AR model: all poles $H(z) = \frac{1}{(1-c_1 z^{-1})(1-c_2 z^{-1})\cdots(1-c_p z^{-1})}$



The freq. response has large magnitude around the poles.

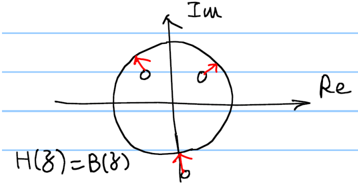


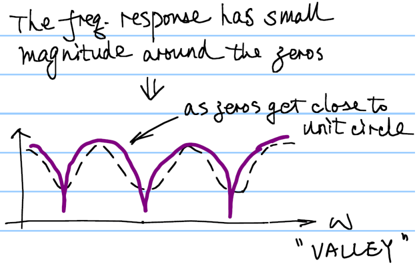
Power Spectral Density: MA Model

$$\text{ZT:} \quad P_X(z) = \sigma^2 H(z) H^*(1/z^*) = \sigma^2 \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}$$

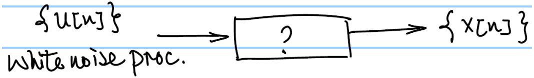
$$\text{p.s.d.:} \quad P_X(\omega) = P_X(z)|_{z=e^{j\omega}} = \sigma^2 |H(\omega)|^2 = \sigma^2 |\frac{B(\omega)}{A(\omega)}|^2$$

- MA model: all zeros





(5) Parameter Equations

Motivation: 

Want to determine the filter parameters that gives $\{x[n]\}$ with desired autocorrelation function?

Or observing $\{x[n]\}$ and thus the estimated $r(k)$, we want to figure out what filters generate such a process? (i.e., ARMA modeling)

Readings: Hayes §3.6

Parameter Equations: ARMA Model

Recall that the power spectrum for ARMA model

$$P_X(z) = H(z)H^*(1/z^*)\sigma^2$$

and $H(z)$ has the form of $H(z) = \frac{B(z)}{A(z)}$

$\Rightarrow P_X(z)A(z) = H^*(1/z^*)B(z)\sigma^2$

$\Rightarrow \sum_{\ell=0}^{p} a[\ell]r_x[k-\ell] = \sigma^2 \sum_{\ell=0}^{q} b[\ell]h^*[\ell-k], \forall k.$

(convolution sum)

Parameter Equations: ARMA Model

For the filter $H(z)$ (that generates the ARMA process) to be causal, $h[k] = 0$ for $k < 0$.
Thus the above equation array becomes

Yule-Walker Equations for ARMA process

$$\begin{cases} r_x[k] = -\sum_{\ell=1}^{p} a[\ell] r_x[k-\ell] + \sigma^2 \sum_{\ell=0}^{q-k} h^*[\ell] b[\ell+k], \ k = 0, \ldots, q \\ r_x[k] = -\sum_{\ell=1}^{p} a[\ell] r_x[k-\ell], k \geq q+1. \end{cases}$$

The above equations are a set of **nonlinear** equations (relate $r_x[k]$ to the parameters of the filter)

Parameter Equations: AR Model

For AR model, $b[\ell] = \delta[\ell]$. The parameter equations become

$$r_x[k] = -\sum_{\ell=1}^{p} a[\ell] r_x[k-\ell] + \sigma^2 h^*[-k]$$

Note:

1. $r_x[-k]$ can be determined by $r_x[-k] = r_x^*[k]$ ($\because$ w.s.s.)
2. $h^*[-k] = 0$ for $k > 0$ by causality, and
   $h^*[0] = [\lim_{z\to\infty} H(z)]^* = \left(\frac{b[0]}{a[0]}\right)^* = 1$

Yule-Walker Equations for AR Process

$$\Rightarrow r_x[k] = \begin{cases} -\sum_{\ell=1}^{p} a[\ell] r_x[-\ell] + \sigma^2 & \text{for } k = 0 \\ -\sum_{\ell=1}^{p} a[\ell] r_x[k-\ell] & \text{for } k \geq 1 \end{cases}$$

The parameter equations for AR are **linear** equations in $\{a[\ell]\}$

Parameter Equations: AR Model

Yule-Walker Equations in matrix-vector form

$$\Rightarrow \begin{bmatrix} r_x(0) & r_x(-1) & \cdots & r_x(-p+1) \\ r_x(1) & r_x(0) & \cdots & r_x(-p+2) \\ \vdots & & \ddots & \vdots \\ r(p-1) & \cdots & & r(0) \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \\ \vdots \\ a[p] \end{bmatrix} = - \begin{bmatrix} r_x(1) \\ r_x(2) \\ \vdots \\ r_x(p) \end{bmatrix}$$

put together $k=1,\ldots,p$

i.e., $\mathbf{R}^T \underline{a} = -\underline{r}$

- $\mathbf{R}$: correlation matrix
- $\underline{r}$: autocorrelation vector

If $\mathbf{R}$ is non-singular, we have $\underline{a} = -(\mathbf{R}^T)^{-1}\underline{r}$.

We'll see better algorithm computing $\underline{a}$ in §2.3.

Parameter Equations: MA Model

For MA model, $a[\ell] = \delta[\ell]$, and $h[\ell] = b[\ell]$. The parameter equations become

$$r_x[k] = \delta^2 \sum_{\ell=0}^{q} b[\ell] b^*[\underbrace{\ell - k}_{\triangleq \ell'}] = \sigma^2 \sum_{\ell'=-k}^{q-k} b[\ell' + k] b^*[\ell']$$

And by causality of $h[n]$ (and $b[n]$), we have

$$r_x[k] = \begin{cases} \sigma^2 \sum_{\ell=0}^{q-k} b^*[\ell] b[\ell + k] & \text{for } k = 0, 1, \ldots, q \\ 0 & \text{for } k \geq q + 1 \end{cases}$$

This is again a set of **non-linear** equations in $\{b[\ell]\}$.

(6) Wold Decomposition Theorem

Recall the earlier example: $y[n] = A\exp[j2\pi f_0 n + \phi)] + w[n]$

- ϕ: (initial) random phase
- $w[n]$ white noise

Theorem

Any stationary w.s.s. discrete time stochastic process $\{x[n]\}$ may be expressed in the form of $x[n] = u[n] + s[n]$, where

1. $\{u[n]\}$ and $\{s[n]\}$ are mutually uncorrelated processes, i.e., $\mathbb{E}\left[u[m]s^*[n]\right] = 0\ \forall m, n$
2. $\{u[n]\}$ is a general random process represented by MA model: $u[n] = \sum_{k=0}^{\infty} b[k]v[n-k]$, $\sum_{k=0}^{\infty} |b_k|^2 < \infty$, $b_0 = 1$
3. $\{s[n]\}$ is a predictable process (i.e., can be predicted from its own pass with zero prediction variance): $s[n] = -\sum_{k=1}^{\infty} a[k]s[n-k]$

Corollary of Wold Decomposition Theorem

ARMA(p,q) can be a good general model for stochastic processes: has a predictable part and a new random part ("innovation process").

Corollary (Kolmogorov 1941)

Any ARMA or MA process can be represented by an AR process (of infinite order).

Similarly, any ARMA or AR process can be represented by an MA process (of infinite order).

Example: Represent ARMA(1,1) by AR(∞) or MA(∞)

E.g., for an ARMA(1, 1), $H_{\text{ARMA}}(z) = \frac{1+b[1]z^{-1}}{1+a[1]z^{-1}}$

1. Use an AR(∞) to represent it:

2. Use an MA(∞) to represent it:

(7) Asymptotic Stationarity of AR Process

Example: we initialize the generation of an AR process with specific status of $x[0], x[-1], \ldots, x[-p+1]$ (e.g., set to zero) and then start the regression $x[1], x[2], \ldots,$

$$x[n] = -\sum_{\ell=1}^{p} a[\ell]x[n-\ell] + u[n]$$

The initial zero states are deterministic and the overall random process has changing statical behavior, i.e., non-stationary.

Asymptotic Stationarity of AR Process

If all poles of the filter in the AR model are inside the unit circle, the temporary nonstationarity of the output process (e.g., due to the initialization at a particular state) can be gradually forgotten and the output process becomes asymptotically stationary.

This is because $H(z) = \frac{1}{\sum_{k=0}^{p} a_k z^{-k}} = \sum_{k=1}^{p} \frac{A_k}{1-\rho_k z^{-1}}$

$$\Rightarrow h[n] = \sum_{k=1}^{p'} A_k \rho_k^n + \sum_{k=1}^{p''} c_k r_k^n \cos(\omega_k n + \phi_k)$$

p': # of real poles
p'': # of complex poles, $\rho_i = r_i e^{\pm j\omega_i}$
$\Rightarrow p = p' + 2p''$ for real-valued $\{a_k\}$.

If all $|\rho_k| < 1$, $h[n] \to 0$ as $n \to \infty$.

Asymptotic Stationarity of AR Process

The above analysis suggests the effect of the input and past outputs on future output is only **short-term**.

So even if the system's output is initially zero to initialize the process's feedback loop, the system can gradually forget these initial states and become **asymptotically stationary** as $n \to \infty$. (i.e., be more influenced by the "recent" w.s.s. samples of the driving sequence)

Detailed Derivations

Example: Represent ARMA(1,1) by AR(∞) or MA(∞)

E.g., for an ARMA(1, 1), $H_{\text{ARMA}}(z) = \frac{1+b[1]z^{-1}}{1+a[1]z^{-1}}$

1. Use an AR(∞) to represent it, i.e.,
   $H_{\text{AR}}(z) = \frac{1}{1+c[1]z^{-1}+c[2]z^{-2}+\ldots}$
   $\Rightarrow$ Let $\frac{1+a[1]z^{-1}}{1+b[1]z^{-1}} = \frac{1}{H_{\text{AR}}(z)} = 1 + c[1]z^{-1} + c[2]z^{-2} + \ldots$
   inverse ZT $\therefore c[k] = \mathbb{Z}^{-1}\left[H_{\text{ARMA}}^{-1}(z)\right]$
   $$\Rightarrow \begin{cases} c[0] = 1 \\ c[k] = (a[1] - b[1])(-b[1])^{k-1} \text{ for } k \geq 1. \end{cases}$$

2. Use an MA(∞) to represent it, i.e.,
   $H_{\text{MA}}(z) = 1 + d[1]z^{-1} + d[2]z^{-2} + \ldots$
   $\therefore d[k] = \mathbb{Z}^{-1}\left[H_{\text{ARMA}}(z)\right]$
   $$\Rightarrow \begin{cases} d[0] = 1 \\ d[k] = (b[1] - a[1])(-a[1])^{k-1} \text{ for } k \geq 1. \end{cases}$$