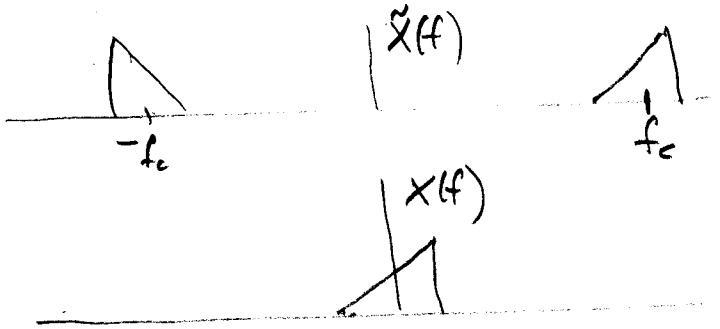
1.2 Random Processes – Time Domain Characterization

1.2.1 Justification for Complex Signals

We're using complex functions of time so that we can represent real bandpass waveforms

$$\tilde{x}(t) = Re\left[x(t)\, e^{j2\pi f_c t}\right] \qquad f_c \text{ nominal center}$$



this illustrates transforms, but similar picture applies to power spectra

1.2.2 Means, Variances, Etc

The samples are complex random variables, assumed to be circularly complex:

$$E[x] = \bar{x} = \mu$$

$$E\left[(x-\mu)^2\right] = 0 \qquad E\left[Re[x-\mu]^2\right] = E\left[Im[x-\mu]^2\right]$$

$$E\left[|x-\mu|^2\right] = \sigma^2 \quad \text{the variance}$$

$$= \overline{|x|^2} \quad \text{for zero mean.}$$

Note: Haykin convention differs from convention $\sigma^2 = \frac{1}{2} E\left[|x|^2\right]$

The factor $\frac{1}{2}$ is so σ^2 is power in bandpass signal

$$P = E\left[\tilde{x}(t)^2\right] = E\left[\tfrac{1}{2} Re\left[|x(t)|^2 + x^2(t)\, e^{j4\pi f_c t}\right]\right] = \tfrac{1}{2}\overline{|x(t)|^2}$$

So be careful

1.2.3 Random Vectors, pdfs, Characteristic Functions

Read Haykin 2.11

- Consider a vector of complex components (not necessarily samples from a single process), length N

$$\underline{u} \qquad \underline{\mu} = E[\underline{u}] \qquad R = E[\underline{u}\,\underline{u}^\dagger] \text{ or } E[\underline{u}\,\underline{u}^H]$$

mean — covariance matrix

- R is Hermitian and positive semidefinite (non-negative definite) whether Gaussian or not. Many consequences, such as

$$\underline{x}^\dagger R \underline{x} \geq 0 \quad \text{or all eivals } \lambda_i \geq 0 \text{ and real}$$

Usually pos def but occasionally semidef, such as
- samples from sum of fewer sine waves than samples
- interpolated sequence in multirate processing

Note Hermitian quadratic form

$$\underline{x}^\dagger R \underline{x} = \text{tr}[R\,\underline{x}\,\underline{x}^\dagger] = \text{tr}[R\,\tilde{R}_x]$$

($\tilde{R}_x$ ← sample cov matrix)

is a random variable itself, if $\underline{x}$ is random.

- If Gaussian, pdf is

$$p_{\underline{u}}(\underline{u}) = \frac{1}{(2\pi)^N \det(\frac{1}{2}R)} \exp\left(-(\underline{u}-\underline{\mu})^\dagger R^{-1}(\underline{u}-\underline{\mu})\right)$$

$$= \frac{1}{(2\pi)^N \det(\Lambda)} \exp\left(-\tfrac{1}{2}(\underline{u}-\underline{\mu})^\dagger \Lambda^{-1}(\underline{u}-\underline{\mu})\right) \qquad \Lambda = \tfrac{1}{2}R$$

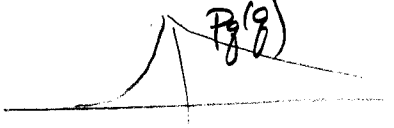
zero mean: $\exp(-\underline{u}^T R^{-1}\underline{u})$, $\exp(-\frac{1}{2}\underline{u}^\dagger \Lambda^{-1}\underline{u})$

again, be careful of the factor $\frac{1}{2}$

- The pdf of the scalar Hermitian q.f. in Gaussian rv's is often useful:

$q = \underline{u}^+ Q \underline{u}$ where Q is Hermitian, not necessarily definite

$P_q(q) = ?$



We obtain it through the characteristic function

$$M_q(s) = \int_{-\infty}^{\infty} P_q(q)\, e^{-sq}\, dq$$

$$= \frac{\exp\left(-s\underline{\mu}^+ (\Omega^{-1} + sR)^{-1} \underline{\mu}\right)}{\det(I + sRQ)}$$

$$= \frac{1}{|I + sRQ|} \quad \text{(zero mean)}$$

Ref:
M Schwartz, W Bennett and S Stein, Communication Systems and Techniques, McGraw Hill 1966
App B

what does this form imply?

Easy expansion for moments, even easier for 1st moment:

$$\bar{q} = E[\underline{u}^+ Q \underline{u}] = E[\mathrm{tr}[Q\, \underline{u}^+ \underline{u}]] = \mathrm{tr}[QR]$$

1.2.4 Random Processes

Haykin 2.1, 2.3, 2.4

sequence $u(n)$, equispaced samples

mean $\mu(n)$

auto corr'n f'n $r(n, n-k) = E[u(n)\, u^*(n-k)]$

auto cov f'n $c(n, n-k) = E[(u(n) - \mu(n))(u(n-k) - \mu(n-k))^*]$

$$\sigma_u^2(n) = c(n, n)$$

WSS: $\mu(n) = \mu$ $\quad r(n, n-k) = r(k)$, $\quad c(n, n-k) = c(k)$

$\overline{|u|^2} = r(0)$ $\qquad \sigma_u^2 = c(0)$

- Here's an example of conjugate symmetry:

$$r(k) = \overline{x(n)\,x^*(n-k)} = \overline{x(i+k)\,x^*(i)} = \overline{x^*(i)\,x(i+k)} = r^*(-k)$$

So we know that

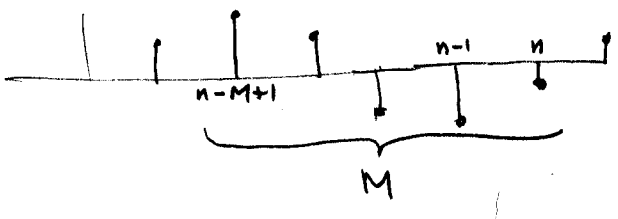
$S(z) = \mathcal{Z}[r(k)]$ satisfies $S(z) = S^*(1/z^*)$

and on the unit circle, $z = e^{j\omega}$, $1/z^* = z$

$S(\omega) = S^*(\omega)$ hence power spectrum is real on the circle.

- consider vector of M equispaced samples

$$\underline{u}(n) = \begin{bmatrix} u(n) \\ u(n-1) \\ \vdots \\ u(n-M+1) \end{bmatrix} \begin{matrix} \text{newest} \\ \\ \\ \text{oldest} \end{matrix}$$



correlation matrix $R = E[\underline{u}(n)\,\underline{u}^\dagger(n)]$

like any covariance matrix, it's Hermitian:

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r(-1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M+1) & r(-M+2) & \cdots & r(0) \end{bmatrix} = \begin{bmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r^*(1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r^*(M-1) & r^*(M-2) & \cdots & r(0) \end{bmatrix}$$

but it's also Toeplitz — only M different elements, a feature we can exploit

Two very important properties:

- reverse the order of samples (without conjugation):

$$\underline{u}^B(n) = \begin{bmatrix} u(n-M+1) \\ u(n-M+2) \\ \vdots \\ u(n-1) \\ u(n) \end{bmatrix}$$

Its correlation matrix

$$E[\mathbf{u}^B(n)\mathbf{u}^{BH}(n)] = \begin{bmatrix} r(0) & r^*(1) & \cdots & r^*(M-1) \\ r(1) & r(0) & \cdots & r^*(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(M-1) & r(M-2) & \cdots & r(0) \end{bmatrix} = R^T$$

Question – what is correlation matrix of conjugate reverse $\underline{u}^{*B}(n)$?

- partitioned correlation matrix. add another sample.

$$\underline{u}_{M+1}(n) = \begin{bmatrix} u(n) \\ u(n-1 \\ \vdots \\ u(n-M+1) \\ u(n-M) \end{bmatrix} = \begin{bmatrix} \underline{u}_M(n) \\ u(n-M) \end{bmatrix} = \begin{bmatrix} u(n) \\ \underline{u}_M(n-1) \end{bmatrix}$$

Its correlation matrix

$$\mathbf{R}_{M+1} = \left[\begin{array}{c|c} r(0) & \mathbf{r}^H \\ \hline \mathbf{r} & \mathbf{R}_M \end{array}\right] = \left[\begin{array}{c|ccc|c} r(0) & r(1) & r(2) & \cdots & r(M) \\ \hline r^*(1) & r(0) & r(1) & \cdots & r(M-1) \\ r^*(2) & r^*(1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r^*(M) & r^*(M-1) & r^*(M-2) & \cdots & r(0) \end{array}\right]$$

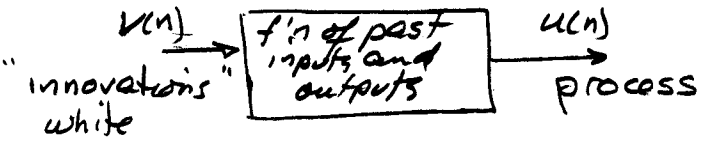
or

$$\mathbf{R}_{M+1} = \left[\begin{array}{c|c} \mathbf{R}_M & \mathbf{r}^{B*} \\ \hline \mathbf{r}^{BT} & r(0) \end{array}\right] = \left[\begin{array}{ccc|c} r(0) & r(1) & \cdots & r(M) \\ r^*(1) & r(0) & \cdots & r(M-1) \\ \vdots & \vdots & \ddots & \\ r^*(M-1) & r^*(M-2) & \cdots & r(1) \\ \hline r^*(M) & r^*(M-1) & \cdots & r(0) \end{array}\right]$$

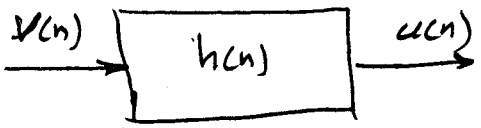
Lends itself to recursive approaches. Several algorithms in adaptive filtering exploit this property.

1.2.5 Models of Stochastic Processes

- We can model processes as

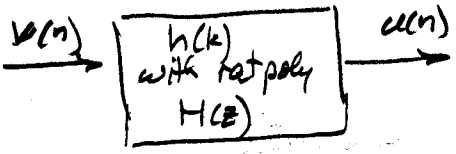


and a linear model is



$$= \sum_i v(n-i)h(i) = \sum_k v(k)h(n-k)$$

and a finite order linear model is



- we usually classify finite order linear processes as

  auto regressive (AR), $H(z)$ is all pole

  moving average (MA), $H(z)$ is all zero (FIR)

  auto reg, moving av (ARMA), $H(z)$ has both poles + zeroes

- for any linear process, finite order or not, the autocorrelation function is

$$r(k) = \overline{u(n)\,u^*(n-k)} = \sum_i \sum_m \overline{v(n-i)\,v^*(n-k-m)}\,h(i)\,h^*(m)$$

$$= \sigma_v^2 \sum_i \sum_m \delta(k+m-i)\,h(i)\,h^*(m)$$

$$= \sigma_v^2 \sum_i h(i)\,h^*(i-k)$$

proportional to impulse response autocorrelation

with z transform

$$R(z) = \sigma_v^2\, H(z)\, H^*(1/z^*)$$

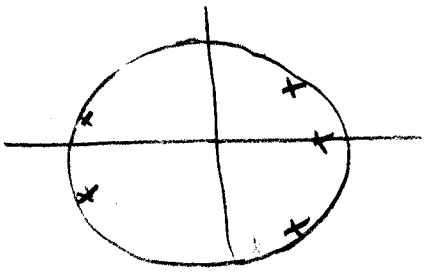
$$S(\omega) = \sigma_v^2 \left|H(\omega)\right|^2 \quad \text{on unit circle } z = e^{j\omega}$$

- AR processes satisfy ΔE

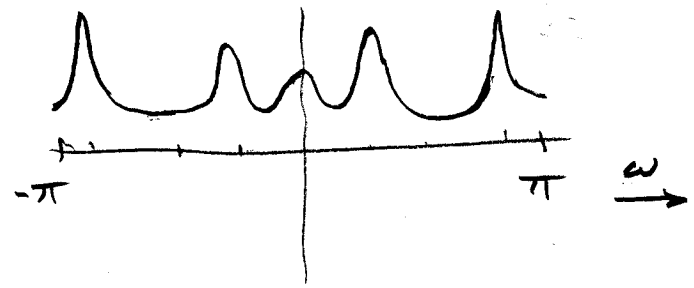
$$u(n) + a_1^* u(n-1) + \cdots + a_M^* u(n-M) = v(n) \qquad (a_0 = 1)$$

or

$$u(n) = \underbrace{w_1^* u(n-1) + \cdots + w_M^* u(n-M)}_{\text{an estimate } \hat{u}(n) \text{ based on } M \text{ past samples}} + v(n)$$



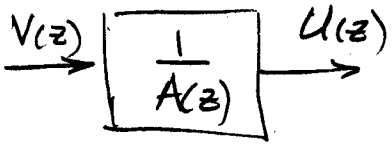
particularly good at representing spectra with peaks



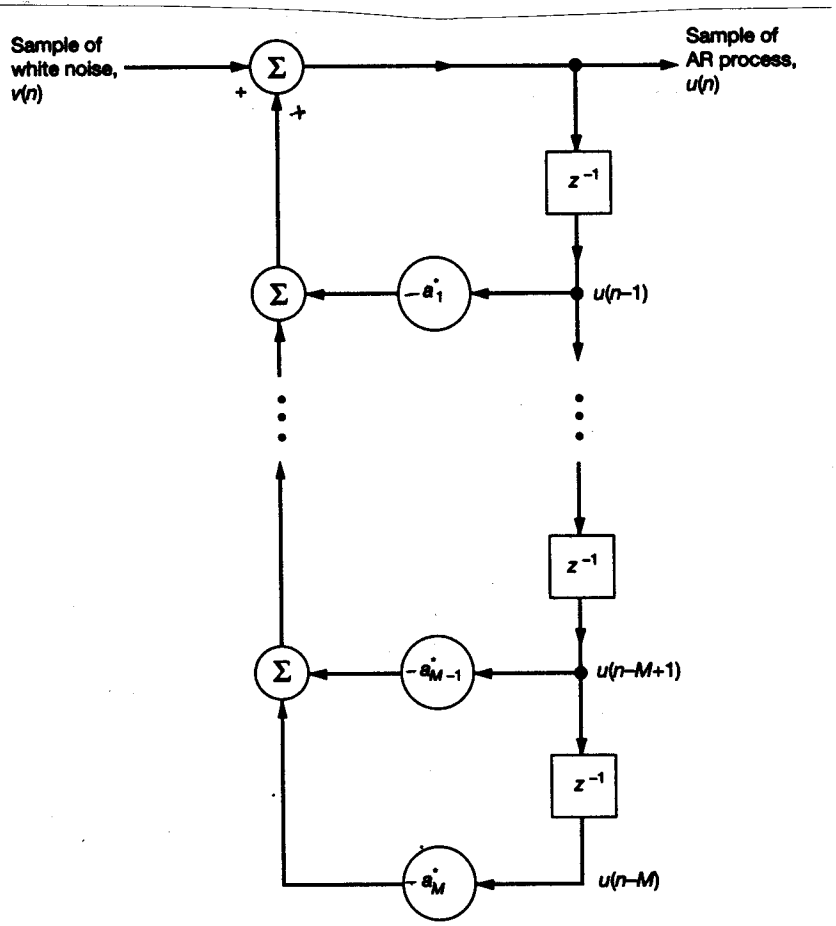
and the process is defined by $\{a_1, \cdots a_M, \sigma_v^2\}$

Using z-transforms:

$$U(z) = \frac{V(z)}{A(z)}$$



$$A(z) = 1 + a_1^* z^{-1} + \cdots + a_M^* z^{-M}$$



synthesis filter

(note sign convention differs from text)

Questions:

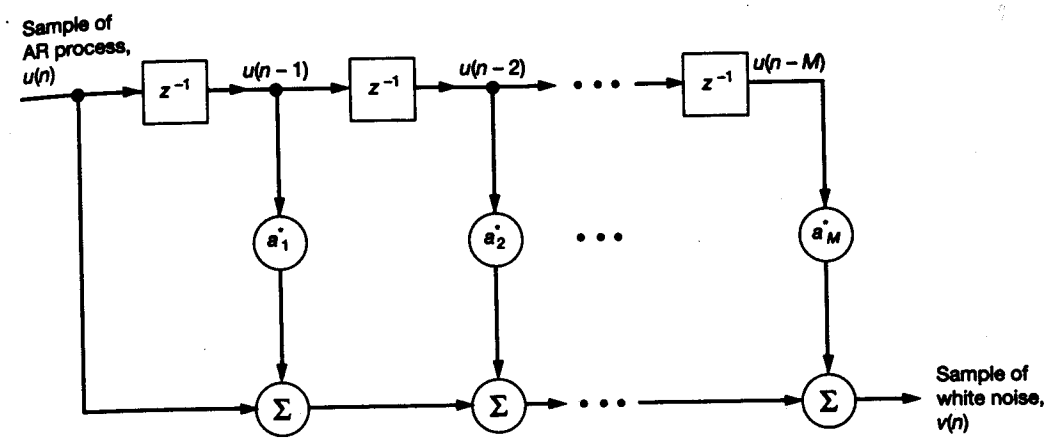
- Where are the poles?
- Where in the circuit is an estimate (prediction) of $u(n)$? $v(n) = u(n) - \hat{u}(n)$
- How many past samples of output are needed to make the prediction? What are the coefficients?
- What is the autocorrelation function $R(z)$?
- Can we go backwards from $R(z)$ and recover the filter coefficients?

- Turn it backwards to recover the innovations

$$V(z) = A(z)\,U(z)$$

$U(z) \rightarrow$ [$A(z)$] $\rightarrow V(z)$

FIR



$A(z)$ = "analysis filter" = "prediction error filter" = "whitening filter"

Can you see it in the synth filter?

- Prediction: $\hat{U}(z) = U(z) - V(z) = (1 - A(z))\,U(z) = W(z)\,U(z)$

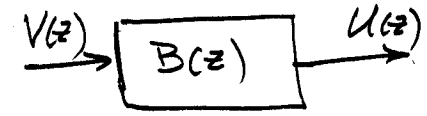
$W(z)$: prediction filter

- MA processes satisfy the ΔE

$$u(n) = v(n) + b_1^* v(n-1) + \cdots + b_K^* v(n-K) \qquad (b_0 = 1)$$

an FIR filter

$$U(z) = B(z)\,V(z)$$



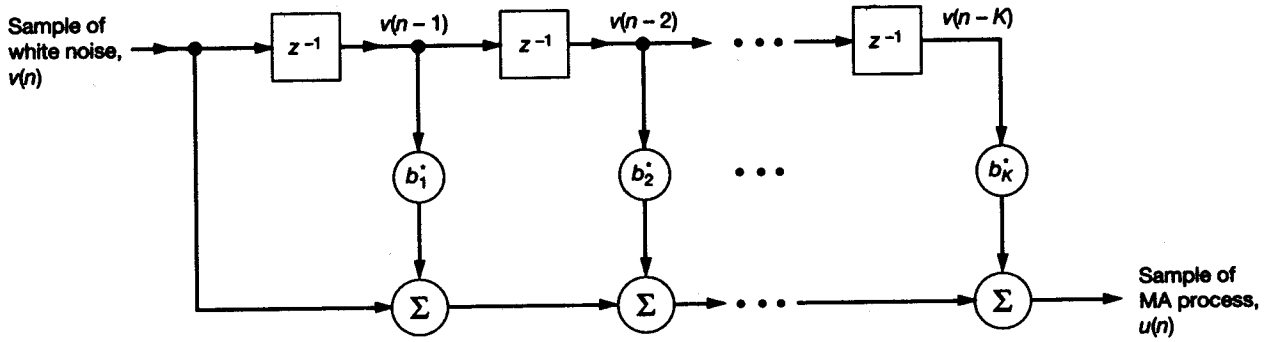
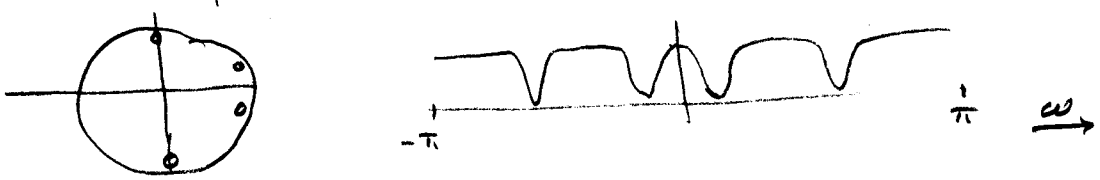
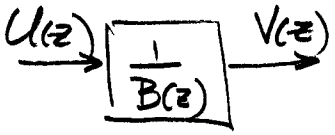


Figure 2.3 Moving average model (process generator).

Good for representing spectra with nulls



Its inverse filter, the one that recovers the innovations, is



an all pole filter "whitening filter" "prediction error filter" "analysis filter"

- Prediction filter

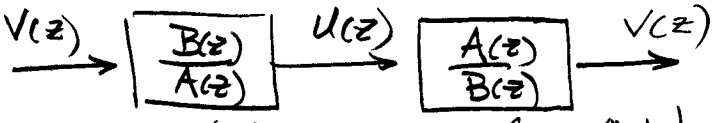
$$\hat{U}(z) = U(z) - V(z) = \left(1 - \frac{1}{B(z)}\right) U(z) = \frac{B(z)-1}{B(z)}\,U(z)$$

Not an all pole filter — poles and zeroes

- Can we go backwards from $R(z)$ to obtain $B(z)$?
- How many past samples are needed for prediction

- ARMA processes satisfy the AE

$$u(n) + a_1^* u(n-1) + \cdots + a_M^* u(n-M) = v(n) + b_1^* v(n-1) + \cdots + b_K^* v(n-K)$$



synthesis — analysis ("whitening", "prediction error")

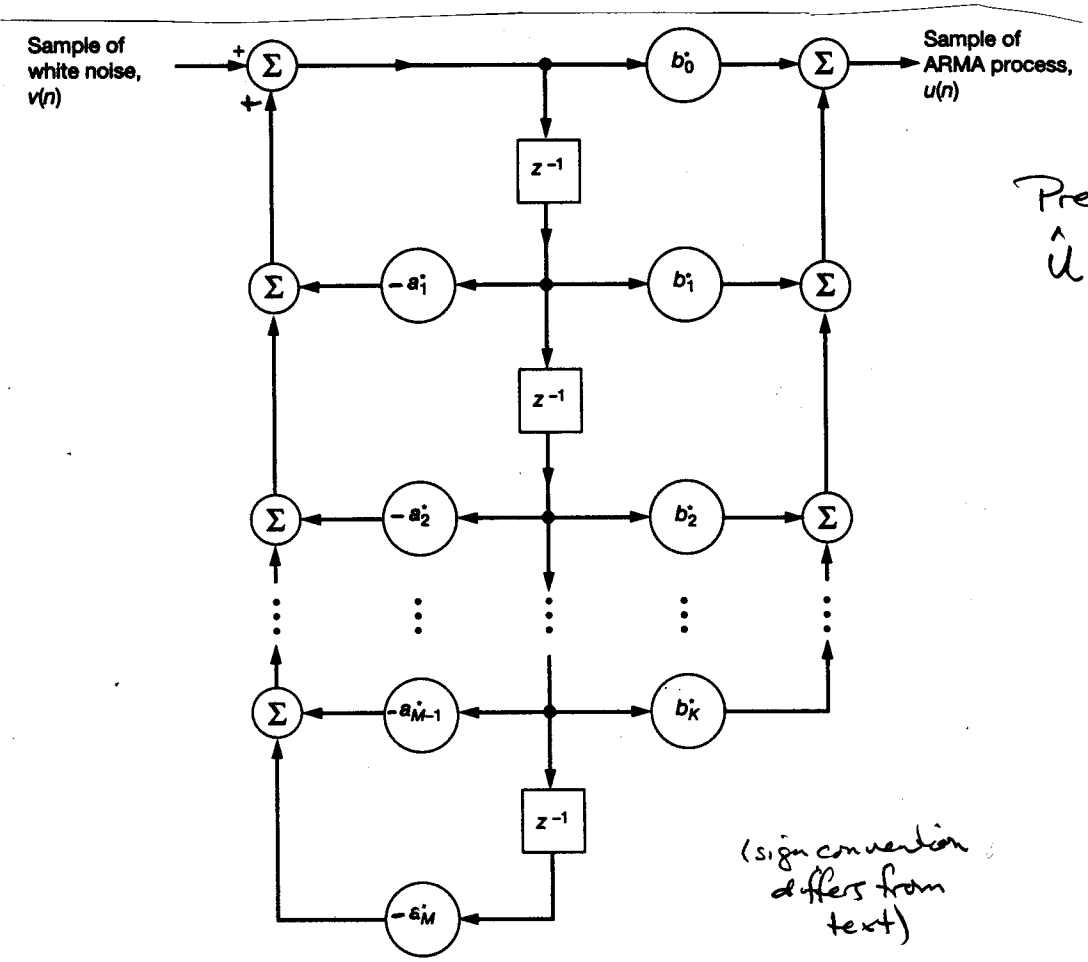


Figure 2.4 ARMA model (process generator) of order (M, K), assuming that $M > K$.

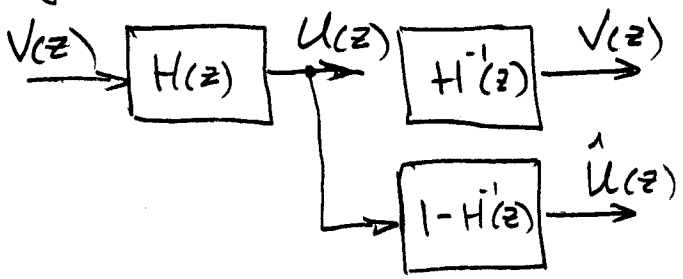
Prediction:

$$\hat{u}(z) = U(z) - V(z)$$

$$= \left(1 - \frac{A(z)}{B(z)}\right) U(z)$$

$$= \left(1 - H^{-1}(z)\right) U(z)$$

- In general, for linear filters



if the inverse filter exists

Therefore:

- If we have $H(z)$ and σ_v^2, then have $R(z) = \sigma_v^2 H(z) H^*(1/z^*)$ and therefore the power spectrum $R(e^{j\omega})$
- If we have $H(z)$, we have the
  - whitening, or prediction error, filter $H^{-1}(z)$
  - onestep predictor $1 - H^{-1}(z)$
- If we have the predictor, we have $H(z)$ and at least the normalized autocorrelation function
- A finite order FIR predictor is handy. It leads to an AR model.

- Obtaining AR model parameters from autocorrelation data: Assume you have $r(n)$ or $R(z)$ — either as ensemble averages or calculated from data as

$$r(n) = \frac{1}{N} \sum_{i=0}^{N-1} u(i)\, u^*(i-n)$$

zeropadded

$$0\,0\; u(0)\; u(1) \cdots u(N-1)\; 0\,0$$

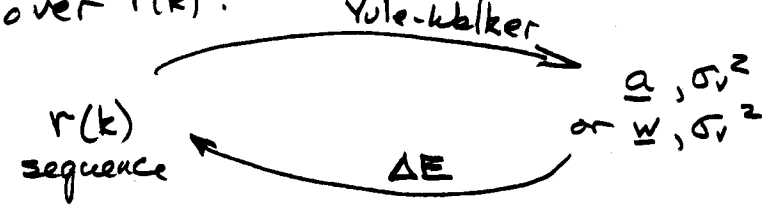
It's easy to calculate a FIR one step predictor from this, giving the model $A(z)$. The normal equations here are called the Yule-Walker equations:

$$u(n) + a_1^* u(n-1) + \cdots + a_M^* u(n-M) = v(n)$$

$$\overline{u(n)\, u^*(n-l)} + a_1^* \overline{u(n-1) u^*(n-l)} + \cdots + a_M^* \overline{u(n-M) u^*(n-l)} = \overline{v(n)\, u^*(n-l)}$$

$$r(l) + a_1^* r(l-1) + \cdots + a_M^* r(l-M) = \begin{cases} \sigma_v^2, & l = 0 \\ 0, & l \geq 1 \end{cases}$$

The Yule-Walker equations can be solved for $a_1, \cdots a_M, \sigma_v^2$. They can also be viewed as a difference equation (ΔE) to recover $r(k)$:



- It's easiest to solve the Yule Walker equations in matrix form. Each of the M+1 values of l generates its own row:

$$\begin{bmatrix} r(0) & r(-1) & \cdots & r(-M) \\ r(1) & r(0) & r(-1) & \cdots & r(1-M) \\ \vdots & & & & \vdots \\ & & & & r(-1) \\ r(M) & r(M-1) & & & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1^* \\ \vdots \\ a_M^* \end{bmatrix} = \begin{bmatrix} \sigma_v^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Here's how to solve for $a_1, \ldots, a_M, \sigma_v^2$:

- conjugate and partition, use $w_i = -a_i$, $i = 1 \ldots M$

$$\begin{bmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r^*(1) & r(0) & & \\ & & \ddots & \\ r^*(M-1) & & r^*(1) & r(0) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_M \end{bmatrix} = \begin{bmatrix} r^*(1) \\ \vdots \\ r^*(M) \end{bmatrix}$$

$$R\underline{w} = \underline{r}$$

$$\underline{w} = R^{-1}\underline{r}$$

$$\underline{a} = \begin{bmatrix} 1 \\ -\underline{w} \end{bmatrix}$$

and

$$\sigma_v^2 = \sum_{i=0}^{M} a_i r(i)$$

(easy with Levinson's recursion)

- How large do we make the order M?
  For any $r(n)$, we getter better and better fit as we increase M. When should we stop?
  - with experimental data, use Akaike Information Criterion (AIC) or Min Description Length (MDL) Haykin 2.10, optional. A problem is they require knowledge of underlying statistics
  - with ensemble averages, depends on computational load