System Identification and Robust Control

Lecture 4: Parametric Identification

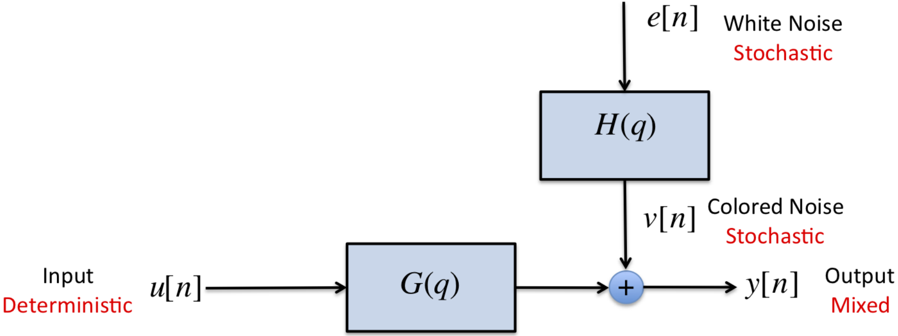
Eugenio Schuster

schuster@lehigh.edu
Mechanical Engineering and Mechanics
Lehigh University

Discrete-time LTI Systems

A liner time-invariant model is specified by

- Impulse response $\{g[k]\}_1^\infty$
- Spectrum $\Phi_{vv} = \lambda |H(e^{i\omega})|^2$ of the additive noise
- Probability density function (PDF) of the noise $e[n]$



Discrete-time LTI Systems

- A complete model is thus given by

$$\begin{aligned} y[n] &= G(q)u[n] + H(q)e[n] \\ f_e(\cdot) & \quad \text{the PDF of } e \end{aligned} \tag{1}$$

where

$$G(q) = \sum_{k=1}^{\infty} g[k]q^{-k}, \tag{2}$$

and

$$H(q) = \sum_{k=0}^{\infty} h[k]q^{-k} = 1 + \sum_{k=1}^{\infty} h[k]q^{-k}. \tag{3}$$

Discrete-time LTI Systems

- In practice, we work with structures that permit the specification of G and H in terms of a finite number of numerical coefficients.
- Quite often it is not possible to determine these coefficients a priori just from knowledge of the plant.
- Instead, the determination is left to estimation procedures. Therefore, these coefficients enter the model as to-be-identified parameters.
- In practice, the PDF of the noise is not specified as a function but described in terms of a few numerical characteristics. Typically, the first moment (mean) and second moment (variance).
- It is also common to assume that $e[n]$ is Gaussian, in which case the PDF is entirely specified by these two moments of the distribution.

Discrete-time LTI Systems

- We denote the parameter vector as θ.

- This vector ranges over a subset of R^d, where d is the dimension of θ:

$$\theta \in D_M \subset R^d.$$

- The model is then described as

$$\begin{aligned} y[n] &= G(q,\theta)u[n] + H(q,\theta)e[n] \qquad (4) \\ f_e(x,\theta) & \qquad \text{the PDF of } e[n] \text{ (white noise)} \end{aligned}$$

One-step Linear Predictor

- Given the system
$$y[n] = G(q)u[n] + H(q)e[n], \tag{5}$$
where $e[n]$ is white with zero mean and variance $E\{e^T e\} = \lambda^2 I$, we are interested in defining an estimator $\widehat{y}[n,\theta]$ that minimizes
$$E\{(y-\widehat{y})^T(y-\widehat{y})\}$$
.
- We propose
$$\widehat{y}[n,\theta] = L_y(q,\theta)y[n] + L_u(q,\theta)u[n]. \tag{6}$$
- We first note that
$$\begin{aligned} & y[n] = G(q)u[n] + H(q)e[n], \\ \iff \quad & H^{-1}(q)y[n] = H^{-1}(q)G(q)u[n] + e[n] \\ \iff \quad & y[n] = [I - H^{-1}(q)]y[n] + H^{-1}(q)G(q)u[n] + e[n] \end{aligned}$$

One-step Linear Predictor

- And now we write

$$\begin{aligned} y[n]-\widehat{y}[n,\theta] &= [I-H^{-1}(q)]y[n]+H^{-1}(q)G(q)u[n]+e[n] \\ &\quad -\{L_y(q,\theta)y[n]+L_u(q,\theta)u[n]\} \\ y[n]-\widehat{y}[n,\theta] &= [I-H^{-1}(q)-L_y(q,\theta)]y[n] \\ &\quad +[H^{-1}(q)G(q)-L_u(q,\theta)]u[n]+e[n] \\ y[n]-\widehat{y}[n,\theta] &= z[n]+e[n] \end{aligned}$$

- Then,

$$E\{(y-\widehat{y})^T(y-\widehat{y})\}=E\{z^Tz\}+E\{e^Te\}\geq \lambda^2 I \quad (7)$$

- The lower bound is achieved when $z=0$, i.e.,

$$L_y(q,\theta)=I-H^{-1}(q), \quad L_u(q,\theta)=H^{-1}(q)G(q). \quad (8)$$

One-step Linear Predictor

- We therefore define the predictor as

$$\widehat{y}[n,\theta] = [I - H^{-1}(q,\theta)]y[n] + H^{-1}(q,\theta)G(q,\theta)u[n], \quad (9)$$

and the prediction error as

$$\varepsilon[n,\theta] = y[n] - \widehat{y}[n,\theta] = H^{-1}(q,\theta)\{y[n] - G(q,\theta)u[n]\}. \quad (10)$$

- Note that both $I - H^{-1}(q,\theta)$ and $H^{-1}(q,\theta)G(q,\theta)$ have at least a one-step delay. Past inputs and outputs are used to provide a new predicted output.

- Note that if $G(q,\theta) = G(q)$ and $H(q,\theta) = H(q)$ then

$$\varepsilon[n,\theta] = e[n]$$

Family of Transfer-Function Models

ARX - Autoregressive with exogeneous input:

$$y[n] + a_1 y[n-1] + a_2 y[n-2] + \cdots + a_{n_a} y[n-n_a]$$
$$= b_1 u[n-1] + b_2 u[n-2] + \cdots + b_{n_b} u[n-n_b] + e[n] \qquad (11)$$

- Since the white-noise term $e[n]$ enter as direct error in the difference equation, this model is often called equation error model.
- The to-be-identified parameter θ is define in this case as

$$\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_1 & b_2 & \cdots & b_{n_b} \end{bmatrix}^T \qquad (12)$$

- If we introduce

$$A(q,\theta) = 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_a} q^{-n_a} \qquad (13)$$
$$B(q,\theta) = b_1 q^{-1} + b_2 q^{-2} + \cdots + a_{n_b} q^{-n_b} \qquad (14)$$

Family of Transfer-Function Models

- By comparing (11) with (4), we conclude that

$$G(q,\theta)=\frac{B(q,\theta)}{A(q,\theta)}, \qquad H(q,\theta)=\frac{1}{A(q,\theta)} \tag{15}$$

- Using (9) we write the one-step predictor for this family of models as

$$\widehat{y}[n,\theta]=[I-A(q,\theta)]y[n]+B(q,\theta)u[n]. \tag{16}$$

- Defining

$$\phi[n]=\begin{bmatrix} -y[n-1] & \cdots & -y[n-n_a] & u[n-1] & \cdots & u[n-n_b] \end{bmatrix}^T$$

we can write

$$\widehat{y}[n,\theta]=\theta^T\phi[n]=\phi[n]^T\theta, \qquad \varepsilon[n,\theta]=y[n]-\phi[n]^T\theta \tag{17}$$

- **Remark:** This defines a *linear regression* model where $\phi[n]$ is known as the *regression vector*. This is very important because powerful and simple estimation methods can be applied for the determination of θ.

Family of Transfer-Function Models

ARMAX - Autoregressive moving average with exogeneous input:

$$
\begin{aligned}
& y[n] + a_1 y[n-1] + a_2 y[n-2] + \cdots + a_{n_a} y[n-n_a] \\
& = b_1 u[n-1] + b_2 u[n-2] + \cdots + b_{n_b} u[n-n_b] \\
& + e[n] + c_1 e[n-1] + \cdots + c_{n_c} e[n-n_c] \qquad (18)
\end{aligned}
$$

- We add flexibility by describing the equation error as a moving average of white noise $e[n]$. The to-be-identified parameter θ is define in this case as

$$
\theta = [\ a_1 \quad a_2 \quad \cdots \quad a_{n_a} \quad b_1 \quad b_2 \quad \cdots \quad b_{n_b} \quad c_1 \quad c_2 \quad \cdots \quad c_{n_c}\ ]^T \qquad (19)
$$

- If we introduce

$$
\begin{aligned}
A(q,\theta) &= 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_a} q^{-n_a} \qquad (20) \\
B(q,\theta) &= b_1 q^{-1} + b_2 q^{-2} + \cdots + a_{n_b} q^{-n_b} \qquad (21) \\
C(q,\theta) &= 1 + c_1 q^{-1} + c_2 q^{-2} + \cdots + c_{n_c} q^{-n_c} \qquad (22)
\end{aligned}
$$

- By comparing (18) with (4), we conclude that

$$
G(q,\theta) = \frac{B(q,\theta)}{A(q,\theta)}, \qquad H(q,\theta) = \frac{C(q,\theta)}{A(q,\theta)} \qquad (23)
$$

Family of Transfer-Function Models

- Using (9) we write the one-step predictor for this family of models as

$$\begin{aligned}\widehat{y}[n,\theta] &= [I - \frac{A(q,\theta)}{C(q,\theta)}]y[n] + \frac{B(q,\theta)}{C(q,\theta)}u[n] \\ C(q,\theta)\widehat{y}[n,\theta] &= [C(q,\theta) - A(q,\theta)]y[n] + B(q,\theta)u[n] \\ \widehat{y}[n,\theta] &= [1 - A(q,\theta)]y[n] + B(q,\theta)u[n] \\ &\quad +[C(q,\theta)-1][y[n] - \widehat{y}[n,\theta]] \end{aligned} \quad (24)$$

- Defining

$$\begin{aligned}\phi[n,\theta] &= [-y[n-1] \cdots - y[n-n_a] \; u[n-1] \cdots u[n-n_b] \\ &\quad +\varepsilon[n-1,\theta] \cdots \varepsilon[n-n_c,\theta]]^T\end{aligned}$$

we can write

$$\widehat{y}[n,\theta] = \theta^T\phi[n,\theta] = \phi[n,\theta]^T\theta, \quad \varepsilon[n,\theta] = y[n] - \phi[n,\theta]^T\theta \quad (25)$$

- **Remark:** This defines a *pseudo-linear regression* model where $\phi[n,\theta]$ is known as the *regression vector*. Note that this is indeed a nonlinear regression because of the dependence of ϕ on θ.

Family of Transfer-Function Models

OE - Output error:

$$\begin{aligned} w[n] + f_1 w[n-1] + f_2 w[n-2] + \cdots + f_{n_f} w[n-n_f] & \\ = \quad & b_1 u[n-1] + b_2 u[n-2] + \cdots + b_{n_b} u[n-n_b] \\ y[n] \quad = \quad & w[n] + e[n] \end{aligned} \tag{26}$$

- Input and noise transfer functions are parametrized independently. The to-be-identified parameter θ is define in this case as

$$\theta = [\ b_1 \quad b_2 \quad \cdots \quad b_{n_b} \quad f_1 \quad f_2 \quad \cdots \quad f_{n_f}\ ]^T \tag{27}$$

- If we introduce

$$B(q,\theta) = b_1 q^{-1} + b_2 q^{-2} + \cdots + a_{n_b} q^{-n_b} \tag{28}$$

$$F(q,\theta) = 1 + f_1 q^{-1} + f_2 q^{-2} + \cdots + f_{n_f} q^{-n_f} \tag{29}$$

- By comparing (26) with (4), we conclude that

$$G(q,\theta) = \frac{B(q,\theta)}{F(q,\theta)}, \qquad H(q,\theta) = 1 \tag{30}$$

Family of Transfer-Function Models

- Using (9) we write the one-step predictor for this family of models as

$$\widehat{y}[n,\theta] \quad = \quad \frac{B(q,\theta)}{F(q,\theta)}u[n] = w[n,\theta]$$

- Defining

$$\phi[n,\theta] \quad = \quad [u[n-1] \cdots u[n-n_b] \ -w[n-1,\theta] \cdots -w[n-n_f,\theta]]^T$$

  we can write

$$\widehat{y}[n,\theta] = \theta^T\phi[n,\theta] = \phi[n,\theta]^T\theta, \quad \varepsilon[n,\theta] = y[n] - \phi[n,\theta]^T\theta \quad (31)$$

- **Remark:** This also defines a *pseudo-linear regression* model where $\phi[n,\theta]$ is known as the *regression vector*. Note that ϕ depends on θ once again.

Family of Transfer-Function Models

BJ - Box-Jenkins:

$$\begin{aligned} w[n] + f_1 w[n-1] + f_2 w[n-2] + \cdots + f_{n_f} w[n-n_f] \\ &= b_1 u[n-1] + b_2 u[n-2] + \cdots + b_{n_b} u[n-n_b] \\ v[n] + d_1 v[n-1] + d_2 v[n-2] + \cdots + d_{n_d} v[n-n_d] \\ &= e[n] + c_1 c[n-1] + c_2 e[n-2] + \cdots + c_{n_c} e[n-n_c] \\ y[n] &= w[n] + v[n] \end{aligned} \tag{32}$$

- Input and noise transfer functions are parametrized independently.
- If we introduce

$$F(q,\theta) = 1 + f_1 q^{-1} + f_2 q^{-2} + \cdots + f_{n_f} q^{-n_f} \tag{33}$$

$$B(q,\theta) = b_1 q^{-1} + b_2 q^{-2} + \cdots + b_{n_b} q^{-n_b} \tag{34}$$

$$C(q,\theta) = 1 + c_1 q^{-1} + c_2 q^{-2} + \cdots + c_{n_c} q^{-n_c} \tag{35}$$

$$D(q,\theta) = 1 + d_1 q^{-1} + d_2 q^{-2} + \cdots + d_{n_d} q^{-n_d} \tag{36}$$

Family of Transfer-Function Models

- By comparing (32) with (4), we conclude that

$$G(q,\theta) = \frac{B(q,\theta)}{F(q,\theta)}, \qquad H(q,\theta) = \frac{C(q,\theta)}{D(q,\theta)} \tag{37}$$

- Using (9) we write the one-step predictor for this family of models as

$$\widehat{y}[n,\theta] \quad = \quad [I - \frac{D(q,\theta)}{C(q,\theta)}]y[n] + \frac{B(q,\theta)}{F(q,\theta)}\frac{D(q,\theta)}{C(q,\theta)}u[n] \tag{38}$$

- **Remark:** This model can also be written as a pseudo-linear regression.

Family of Transfer-Function Models

$$A(q,\theta)y[n] = \frac{B(q,\theta)}{F(q,\theta)}u[n] + \frac{C(q,\theta)}{D(q,\theta)}e[n] \tag{39}$$

Polynomials Used	Name of Model Structure
B	FIR
AB	ARX
ABC	ARMAX
AC	ARMA
ABD	ARARX
$ABCD$	ARARMAX
BF	OE
$BFCD$	BJ

- **Note:** See derivation of pseudo-linear regression form of (39) in the book.

Family of Transfer-Function Models

State-Space:

$$\begin{aligned} x[n+1] &= A(\theta)x[n] + B(\theta)u[n] + w[n] \\ y[n] &= C(\theta)x[n] + v[n] \end{aligned} \quad (40)$$

- The transfer function of system (40) is given by

$$G(z,\theta) = C(\theta)(zI - A(\theta))^{-1}B(\theta) + D(\theta) \quad (41)$$

- The one-step predictor is given by the Kalman Predictor

$$\begin{aligned} \widehat{x}[n+1,\theta] &= (A(\theta) - M[n,\theta]C(\theta))\,\widehat{x}[n,\theta] + B(\theta)u[n] + M[n,\theta]y[n] \\ M[n,\theta] &= A(\theta)\Sigma[n,\theta]C(\theta)\left(C(\theta)\Sigma[n,\theta]C(\theta)^T + R\right)^{-1} \\ \Sigma[n+1,\theta] &= Q + A(\theta)\Sigma[n,\theta]A(\theta)^T \\ &\quad -A(\theta)\Sigma[n,\theta]C(\theta)^T\left(C(\theta)\Sigma[n,\theta]C(\theta)^T + R\right)^{-1}C(\theta)\Sigma[n,\theta]A(\theta)^T \end{aligned}$$

Family of Transfer-Function Models

- Assumptions:
  - $x[0]$ is Gaussian with mean $\bar{x}_0$ and covariance P_0.
  - $w[n]$ is Gaussian, zero-mean, white stochastic process, i.e., $E\{w[n]w^T[m]\} = Q\delta[n-m]$, independent of x_0 and $v[n]$.
  - $v[n]$ is Gaussian, zero-mean, white stochastic process, i.e., $E\{v[n]v^T[m]\} = R\delta[n-m]$, independent of x_0 and $w[n]$.
- The covariance

$$\Sigma[n,\theta] = E\{(x[n] - \widehat{x}[n,\theta])(x[n] - \widehat{x}[n,\theta])^T\}$$

satisfies a Riccati Difference Equation (RDE).

- The Kalman Predictor is an observer!!!

$$\begin{aligned} \widehat{x}[n+1,\theta] &= A(\theta)\widehat{x}[n,\theta] + B(\theta)u[n] + M[n,\theta]e[n,\theta] \\ e[n,\theta] &= y[n] - C(\theta)\widehat{x}[n,\theta] \\ E\{e[n,\theta]e[n,\theta]^T\} &= C(\theta)\Sigma[n,\theta]C^T(\theta) + R \end{aligned}$$

- We can predict the output as

$$\widehat{y}[n,\theta] = C(\theta)\widehat{x}[n,\theta]$$

Prediction Error Method

- The estimation of θ based on N measurements of both input $u[n]$ and output $y[n]$ is computed as

$$\widehat{\theta}_N = \arg\min V_N(\theta) \tag{42}$$

where

$$V_N(\theta) = \frac{1}{N} \sum_{n=1}^{N} \|\varepsilon[n, \theta]\|_2^2 \tag{43}$$

- In the case of a linear regression model

$$\widehat{y}[n, \theta] = \theta^T \phi[n] = \phi[n]^T \theta, \qquad \varepsilon[n, \theta] = y[n] - \phi[n]^T \theta. \tag{44}$$

- When we consider a SISO system ($\phi : d \times 1$ and $\theta : d \times 1$), we can write

$$V_N(\theta) = \frac{1}{N} \sum_{n=1}^{N} [y[n] - \phi[n]^T \theta]^2 \tag{45}$$

Least Square Estimate

- This quadratic function can be minimized analytically.

- We find that

$$\left[\frac{1}{N}\sum_{n=1}^{N}\phi[n]\phi^T[n]\right]\widehat{\theta}_N = \frac{1}{N}\sum_{n=1}^{N}\phi[n]y[n] \quad (46)$$

- This set of linear equations is known as the *normal equations*.

- If the matrix of the left is invertible, we have the Least Square Estimate (LSE)

$$\widehat{\theta}_N = \left[\frac{1}{N}\sum_{n=1}^{N}\phi[n]\phi^T[n]\right]^{-1}\frac{1}{N}\sum_{n=1}^{N}\phi[n]y[n] \quad (47)$$

Least Square Estimate

- Sometimes the estimate (47) can be written more conveniently in matrix form.
- We define the $N \times 1$ vector and $N \times d$ matrix

$$Y_N = \begin{bmatrix} y[1] \\ \vdots \\ y[N] \end{bmatrix}, \qquad \Phi_N = \begin{bmatrix} \phi^T[1] \\ \vdots \\ \phi^T[N] \end{bmatrix} \tag{48}$$

  respectively.
- We can write

$$V_N(\theta) = \frac{1}{N}|Y_N - \Phi_N\theta|^2 = \frac{1}{N}(Y_N - \Phi_N\theta)^T(Y_N - \Phi_N\theta) \tag{49}$$

- The normal equations take the form

$$[\Phi_N^T\Phi_N]\widehat{\theta}_N = \Phi_N^T Y_N \tag{50}$$

Least Square Estimate

- And the estimate can be written as

$$\widehat{\theta}_N = [\Phi_N^T \Phi_N]^{-1} \Phi_N^T Y_N \tag{51}$$

where $[\Phi_N^T \Phi_N]^{-1} \Phi_N^T$ is the Moore-Penrose pseudoinverse of Φ_N.

- Therefore, the estimate (51) is the solution to the overdetermined ($N > d$ system of linear equations

$$Y_N = \Phi_N \theta \tag{52}$$

- Let us write

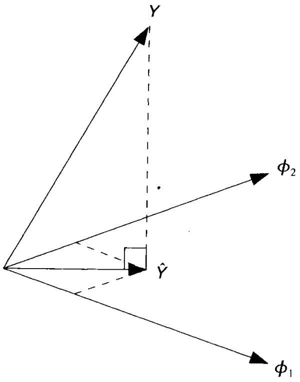
$$\Phi_N = [\bar{\phi}_1 \dots \bar{\phi}_d] \tag{53}$$

where $\bar{\phi}_i : N \times 1$ for $i = 1, \dots, d$.

- The problem in (52) is to find a linear combination of the vectors $\bar{\phi}_i : N \times 1$ for $i = 1, \dots, d$ that approximates Y_N as well as possible.

Least Square Estimate

- If Y_N belongs to the subspace generated by the columns of Φ_N, we can describe it as a unique linear combination of $\bar{\phi}_i : N \times 1$ for $i = 1, \ldots, d$.
- Otherwise, the best approximation is the vector in the subspace generated by the columns of Φ_N that has the smallest distance to Y_N, which is well know to be the orthogonal projection of Y_N on this subspace.



Least Square Estimate

- Let the projection be denoted as $\widehat{Y}_N$. Since it is the orthogonal projection, we have

$$(Y_N - \widehat{Y}_N) \perp \bar{\phi}_i \tag{54}$$

- That is,

$$(Y_N - \widehat{Y}_N)^T \bar{\phi}_i = 0, \qquad i = 1, \ldots, n \tag{55}$$

- We can write

$$\widehat{Y}_N = \sum_{j=1}^{d} \widehat{\theta}_j \bar{\phi}_j \tag{56}$$

- This gives

$$Y_N^T \bar{\phi}_i = \sum_{j=1}^{d} \widehat{\theta}_j \bar{\phi}_j^T \bar{\phi}_i, \qquad i = 1, \ldots, n \tag{57}$$

which are the normal equations (50).

Realization Algorithm

- Let us assume that we have identified the impulse response coefficients using a nonparametric method:

$$g[k] \qquad k = 0, \dots, 2N \tag{58}$$

How can we use this data to obtain a parametric state-space realization?

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] + Du[k] \end{aligned} \tag{59}$$

- The transfer function of system (59) is given by

$$G(z) = C(zI - A)^{-1}B + D \tag{60}$$

- We recall that

$$G(z) = \sum_{k=0}^{\infty} g[k]z^{-k} \tag{61}$$

Realization Algorithm

- Using the formal geometric series expansion we can write

$$(zI-A)^{-1} = \frac{1}{z}\left(I-\frac{A}{z}\right)^{-1} = \frac{1}{z}\left[I+\frac{A}{z}+\frac{A^2}{z^2}+\cdots\right] \tag{62}$$

- Therefore

$$G(z) = C(zI-A)^{-1}B + D = C\frac{1}{z}\left[I+\frac{A}{z}+\frac{A^2}{z^2}+\cdots\right]B + D \tag{63}$$

- Comparing (61) and (63) we can conclude that the impulse response coefficients $g[k]$ are given by the *Markov parameters*

$$g[k] = \begin{cases} D & k=0 \\ CA^{k-1}B & k\geq 1 \end{cases} \tag{64}$$

Realization Algorithm

- We define the *Hankel* matrix as

$$M[i,j] = \begin{bmatrix} g[i] & g[i+1] & \cdots & g[i+j] \\ g[i+1] & g[i+2] & \cdots & g[i+j+1] \\ \vdots & \vdots & & \vdots \\ g[i+j] & g[i+j+1] & \cdots & g[i+2j] \end{bmatrix} \quad (65)$$

- We define

$$H = M[1, N-1] \qquad \bar{H} = M[2, N-1] \quad (66)$$

and taking into account (64) we note that

$$H = H_1 H_2 \qquad \bar{H} = H_1 A H_2 \quad (67)$$

where

$$H_1 = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} = \bar{O}, \;\; H_2 = [B \;\; AB \; \cdots \; A^{N-1}B] = \bar{C} \quad (68)$$

Realization Algorithm

- We then can note that

$$D = g[0],\ C = H_1(1,:),\ B = H_2(:,1),\ A = H_1^+ \bar{H} H_2^+ \tag{69}$$

where

$$H_1^+ = (H_1^T H_1)^{-1} H_1^T \Rightarrow H_1^+ H_1 = I \tag{70}$$

$$H_2^+ = H_2^T (H_2 H_2^T)^{-1} \Rightarrow H_2 H_2^+ = I \tag{71}$$

Realization Algorithm

- How do we compute H_1 and H_2?
- We compute the Singular Value Decomposition (SVD) of H, i.e.,

$$H = H_1 H_2 = U\Sigma V^* = U\Sigma^{1/2}\Sigma^{1/2}V^* \tag{72}$$

where U and V are unitary matrices.

- We compute

$$H_1 = U\Sigma^{1/2} \Rightarrow H_1^* H_1 = \Sigma^{1/2}U^*U\Sigma^{1/2} = \Sigma \tag{73}$$

$$H_2 = \Sigma^{1/2}V^* \Rightarrow H_2 H_2^* = \Sigma^{1/2}V^*V\Sigma^{1/2} = \Sigma \tag{74}$$

and also note that

$$H_1 = \bar{O} \Rightarrow H_1^* H_1 = \sum_{k=0}^{N-1} (A^T)^k C^T C A^k \tag{75}$$

$$H_2 = \bar{C} \Rightarrow H_2 H_2^* = \sum_{k=0}^{N-1} (A)^k B B^T (A^T)^k \tag{76}$$

Realization Algorithm

- If A is stable,

$$H_1^* H_1 \to Q \qquad H_2 H_2^* \to P \tag{77}$$

when $N \to \infty$, where P and Q denote the observability and controllability gramians that satisfy

$$A^T Q A + C^T C = Q \tag{78}$$
$$A P A^T + B B^T = P \tag{79}$$

- We can finally note that

$$P = Q = \Sigma \Rightarrow \text{ BALANCED REALIZATION} \tag{80}$$

Can we obtain a BALANCED TRUNCATION?

Realization Algorithm

- Note that

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_N \end{bmatrix} \tag{81}$$

where σ_i, for $i = 1, \ldots, N$, are called Hankel singular values and where

$$H = U\Sigma V^* = [U_n \; U_s] \begin{bmatrix} \Sigma_n & 0 \\ 0 & \Sigma_s \end{bmatrix} \begin{bmatrix} V_n^* \\ V_s^* \end{bmatrix} \tag{82}$$

- It is usually the case that

$$\sigma_1 > \sigma 2 > \cdots > \sigma_n >> \sigma_{n+1} > \cdots \sigma_N$$

Realization Algorithm

- In this case we can adopt n as the order of the system and approximate

$$\begin{bmatrix} \Sigma_n & 0 \\ 0 & \Sigma_s \end{bmatrix} \approx \begin{bmatrix} \Sigma_n & 0 \\ 0 & 0 \end{bmatrix} \tag{83}$$

and we can write

$$H \approx H_n = U_n \Sigma_n V_n^* \tag{84}$$

and

$$\tilde{H}_1 = U_n \Sigma_n^{1/2}, \qquad \tilde{H}_2 = \Sigma_n^{1/2} V_n^* \tag{85}$$

to finally conclude that

$$D = g[0],\ C = \tilde{H}_1(1,:),\ B = \tilde{H}_2(:,1),\ A = \tilde{H}_1^+ \bar{H} \tilde{H}_2^+ \tag{86}$$

Curve Fitting

- Let us assume that we have identified the frequency response of a system using a nonparametric method:

$$G(e^{j\omega_k}) \qquad \omega_k = \frac{2\pi k}{N},\; k = 0, \ldots, N-1 \tag{87}$$

How can we use this data to obtain a parametric transfer function?

$$\widehat{G}(q,\theta) = \frac{B(q,\theta)}{A(q,\theta)} \tag{88}$$

where

$$A(q,\theta) = 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_a} q^{-n_a} \tag{89}$$

$$B(q,\theta) = b_0 + b_1 q^{-1} + b_2 q^{-2} + \cdots + a_{n_b} q^{-n_b} \tag{90}$$

Curve Fitting

- We define

$$E_N(\omega_k,\theta) = \left[G(e^{j\omega_k}) - \widehat{G}(e^{j\omega_k},\theta)\right] W(e^{j\omega_k}) \tag{91}$$

where $W(e^{j\omega_k})$ is a frequency weighting function.

- We also define the cost function

$$J(\theta) = \sum_{k=0}^{N-1} E_N(\omega_k,\theta) E_N^*(\omega_k,\theta) = \left\|\bar{E}_N(\omega,\theta)\right\|_2^2 \tag{92}$$

with $\bar{E}_N(\omega,\theta) = [E_N(\omega_0,\theta)\, E_N(\omega_1,\theta)\ \ldots\ E_N(\omega_{N-1},\theta)]$.

- We seek

$$\theta = \arg\min_{\theta \in R} J(\theta) \tag{93}$$

Curve Fitting

- **Output Error:**

$$E(\omega_k, \theta) = G(e^{j\omega_k}) - \frac{B(e^{j\omega_k}, \theta)}{A(e^{j\omega_k}, \theta)} \tag{94}$$

  This defines a **nonlinear** problem.

- **Equation Error:**

$$\begin{aligned} \tilde{E}(\omega_k, \theta) &= E(\omega_k, \theta) A(e^{j\omega_k}, \theta) \\ &= G(e^{j\omega_k}) A(e^{j\omega_k}, \theta) - B(e^{j\omega_k}, \theta) \end{aligned} \tag{95}$$

  This defines a **linear** problem.

- Note that $\tilde{E}(\omega_k, \theta)$ is a weighted function of $E(\omega_k, \theta)$.

Curve Fitting

- We can write

$$\tilde{E}(\omega_k,\theta) = E(\omega_k,\theta)W(e^{j\omega_k}) = E_N(\omega_k,\theta) \tag{96}$$

with

$$W(e^{j\omega_k}) = A(e^{j\omega_k},\theta). \tag{97}$$

- Recalling

$$\begin{aligned} A(e^{j\omega_k},\theta) &= 1 + a_1\left(e^{j\omega_k}\right)^{-1} + a_2\left(e^{j\omega_k}\right)^{-2} + \cdots + a_{n_a}\left(e^{j\omega_k}\right)^{-n_a} \\ B(e^{j\omega_k},\theta) &= b_0 + b_1\left(e^{j\omega_k}\right)^{-1} + b_2\left(e^{j\omega_k}\right)^{-2} + \cdots + a_{n_b}\left(e^{j\omega_k}\right)^{-n_b} \end{aligned}$$

we can write

$$\tilde{E}(\omega_k,\theta) = G(e^{j\omega_k}) - \theta\phi(e^{j\omega_k}) \tag{98}$$

where

$$\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_0 & b_1 & b_2 & \cdots & b_{n_b} \end{bmatrix} \tag{99}$$

and

$$\begin{aligned} \phi(e^{j\omega_k}) = \Big[& -G(e^{j\omega_k})\left(e^{j\omega_k}\right)^{-1} \;\; -G(e^{j\omega_k})\left(e^{j\omega_k}\right)^{-2} \;\; \cdots \;\; -G(e^{j\omega_k})\left(e^{j\omega_k}\right)^{-n_a} \\ & 1 \;\; \left(e^{j\omega_k}\right)^{-1} \;\; \left(e^{j\omega_k}\right)^{-2} \;\; \cdots \;\; \left(e^{j\omega_k}\right)^{-n_b} \Big]^T \end{aligned} \tag{100}$$

Curve Fitting

- We can write the to-be-minimized error as

$$\bar{E}(\omega,\theta) = \bar{G}(\omega) - \theta\bar{\phi}(\omega) \tag{101}$$

where

$$\begin{aligned} \bar{E}(\omega,\theta) &= [\tilde{E}(\omega_0,\theta)\ \tilde{E}(\omega_1,\theta)\ \ldots\ \tilde{E}(\omega_{N-1},\theta)] && (102) \\ \bar{G}(\omega) &= [G(\omega_0)\ G(\omega_1)\ \ldots\ G(\omega_{N-1})] && (103) \\ \bar{\phi}(\omega) &= [\phi(\omega_0)\ \phi(\omega_1)\ \ldots\ \phi(\omega_{N-1})] && (104) \end{aligned}$$

- And the cost function as

$$\tilde{J}(\theta) = \sum_{k=0}^{N-1} \tilde{E}(\omega_k,\theta)\tilde{E}^*(\omega_k,\theta) = \left\|\bar{E}(\omega,\theta)\right\|_2^2 \tag{105}$$

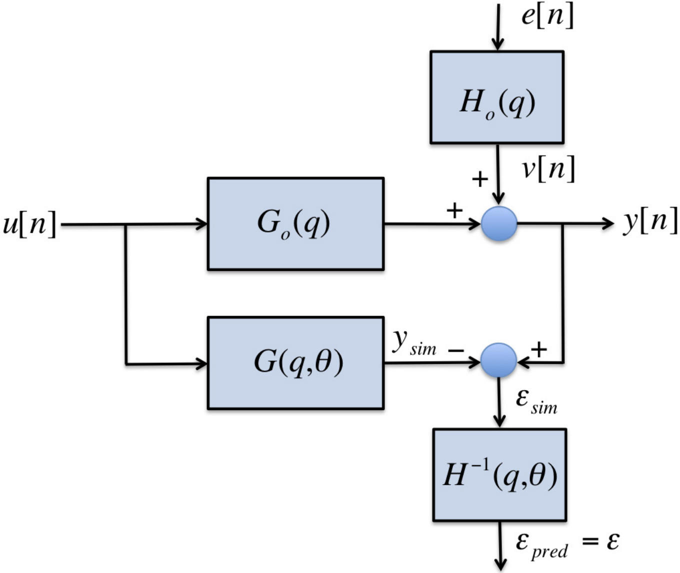
- The minimizing solution is obtained as

$$\left.\begin{array}{c} \min_\theta \bar{E}(\omega,\theta)\bar{E}(\omega,\theta)^* \\ \bar{E}(\omega,\theta) = \bar{G}(\omega) - \theta\bar{\phi}(\omega) \end{array}\right\} \Rightarrow \bar{E}\bar{\phi}^* = \bar{G}\bar{\phi}^* - \theta\bar{\phi}\bar{\phi}^* \tag{106}$$

which results in

$$\theta = \bar{G}\bar{\phi}^* \left[\bar{\phi}\bar{\phi}^*\right]^{-1} \tag{107}$$

Prediction vs Simulation



Prediction vs Simulation

- The simulated output is computed as

$$y_{sim}[n,\theta] = G(q,\theta)u[n], \tag{108}$$

and the simulation error as

$$\varepsilon_{sim}[n,\theta] = y[n] - y_{sim}[n,\theta] = y[n] - G(q,\theta)u[n]. \tag{109}$$

- The predicted output is computed as

$$\widehat{y}[n,\theta] = [I - H^{-1}(q,\theta)]y[n] + H^{-1}(q,\theta)G(q,\theta)u[n], \tag{110}$$

and the prediction error as

$$\varepsilon[n,\theta] = y[n] - \widehat{y}[n,\theta] = H^{-1}(q,\theta)\{y[n] - G(q,\theta)u[n]\}. \tag{111}$$

- Note that $\varepsilon[n,\theta] = H^{-1}(q,\theta)\varepsilon_{sim}[n,\theta]$.

Prediction vs Simulation

- Note that if $G(q,\theta) = G(q)$, then

$$\varepsilon_{sim}[n,\theta] = v[n] \Rightarrow R_{u\varepsilon_{sim}}[\tau] = 0$$

since $u[n]$ and $v[n]$ are uncorrelated by assumption (not true for feedback systems).

- Note that if $G(q,\theta) = G(q)$ and $H(q,\theta) = H(q)$ then

$$\varepsilon[n,\theta]{=}H^{-1}(q,\theta)\varepsilon_{sim}[n,\theta]{=}H^{-1}(q,\theta)v[n]{=}e[n] \Rightarrow R_{\varepsilon\varepsilon}[\tau]{=}\delta[\tau]$$

since $e[n]$ is white noise by assumption.

Consistency and Convergence

- The estimation of θ based on N measurements of both input $u[n]$ and output $y[n]$ is computed as

$$\widehat{\theta}_N = \arg\min V_N(\theta) \tag{112}$$

where

$$V_N(\theta) = \frac{1}{N}\sum_{n=1}^{N} \|\varepsilon[n,\theta]\|_2^2 = \frac{1}{N}\sum_{n=1}^{N} \varepsilon[n,\theta]\varepsilon^T[n,\theta] \tag{113}$$

and

$$\begin{aligned} \varepsilon[n,\theta] = y[n] - \widehat{y}[n,\theta] &= H^{-1}(q,\theta)\{y[n] - G(q,\theta)u[n]\} \\ &= H^{-1}(q,\theta)\varepsilon_{sim}[n,\theta]. \end{aligned} \tag{114}$$

- **Consistency of Estimate:** Is $\widehat{\theta}_N \to \theta_o$ when $N \to \infty$?

Consistency and Convergence

- **System** - $\mathcal{S}$:

$$\begin{aligned} y[n] &= G(q,\theta_o)u[n] + H(q,\theta_o)e[n] \\ &= \frac{B(q,\theta_o)}{A(q,\theta_o)}u[n] + \frac{1}{A(q,\theta_o)}e[n] \end{aligned} \tag{115}$$

- **Model** - $\mathcal{M}$

$$\begin{aligned} y[n] &= G(q,\theta)u[n] + H(q,\theta)e[n] \\ &= \frac{B(q,\theta)}{A(q,\theta)}u[n] + \frac{1}{A(q,\theta)}e[n] \end{aligned} \tag{116}$$

- We assume that $\mathcal{S} \in \mathcal{M}$. The model $\mathcal{M}$ has an ARX structure.

Consistency and Convergence

- We write the system $\mathcal{S}$ in linear regression form

$$y[n] = \phi^T[n]\theta_o + e[n] \tag{117}$$

where

$$\phi[n] = \begin{bmatrix} -y[n-1] & \cdots & -y[n-n_a] & u[n] & \cdots & u[n-n_b] \end{bmatrix}^T$$

and

$$\theta_o = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_0 & b_1 & \cdots & b_{n_b} \end{bmatrix}^T$$

- The prediction error can be written as

$$\begin{aligned} \varepsilon[n,\theta] = y[n] - \phi^T[n]\theta &= H^{-1}(q,\theta)\{y[n] - G(q,\theta)u[n]\} \\ &= A(q,\theta)y[n] - B(q,\theta)u[n]. \end{aligned} \tag{118}$$

Consistency and Convergence

- Where we have written

$$\begin{aligned}\widehat{y}[n,\theta] &= [I - H^{-1}(q,\theta)]y[n] + H^{-1}(q,\theta)G(q,\theta)u[n] \\ &= [I - A(q,\theta)]y[n] + B(q,\theta)u[n] \\ &= \phi^T[n]\theta\end{aligned} \quad (119)$$

- By defining

$$Y_N = \begin{bmatrix} y[1] \\ \vdots \\ y[N] \end{bmatrix}, \ \Phi_N = \begin{bmatrix} \phi^T[1] \\ \vdots \\ \phi^T[N] \end{bmatrix}, \ \varepsilon_N = \begin{bmatrix} \varepsilon[1] \\ \vdots \\ \varepsilon[N] \end{bmatrix} \quad (120)$$

we can write

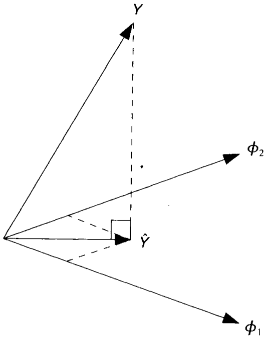
$$\varepsilon_N = Y_N - \Phi_N\theta \quad (121)$$

Consistency and Convergence

- The to-be-minimized cost function can be written as

$$V_N(\theta) = \frac{1}{N}\sum_{n=1}^{N} \varepsilon[n,\theta]\varepsilon^T[n,\theta] = \frac{1}{N}\varepsilon_N\varepsilon_N^T \qquad (122)$$

- ε_N is minimized when $\varepsilon_N \perp \widehat{Y}_N = \Phi_N\theta$.



Consistency and Convergence

- Then,

$$0 = \Phi_N^T \varepsilon_N = \Phi_N^T Y_N - \Phi_N^T \Phi_N \theta \tag{123}$$

and the estimate can be written as

$$\widehat{\theta}_N = [\Phi_N^T \Phi_N]^{-1} \Phi_N^T Y_N = R(N)^{-1} f(N) \tag{124}$$

where

$$R(N) = \frac{1}{N} \sum_{n=1}^{N} \phi[n] \phi^T[n] \qquad f(N) = \frac{1}{N} \sum_{n=1}^{N} \phi[n] y[n] \tag{125}$$

Consistency and Convergence

- Since

$$y[n] = \phi^T[n]\theta_o + e[n] \tag{126}$$

we can write

$$\begin{aligned}
\widehat{\theta}_N &= R(N)^{-1}\left[\frac{1}{N}\sum_{n=1}^{N}\phi[n]\left(\phi^T[n]\theta_o + e[n]\right)\right] \qquad (127)\\
&= R(N)^{-1}\left[\frac{1}{N}\sum_{n=1}^{N}\phi[n]\phi^T[n]\theta_o + \frac{1}{N}\sum_{n=1}^{N}\phi[n]e[n]\right]\\
&= \theta_o + R(N)^{-1}\frac{1}{N}\sum_{n=1}^{N}\phi[n]e[n]
\end{aligned}$$

Then,

$$\widehat{\theta}_N \to \theta_o \iff R(N)^{-1}\frac{1}{N}\sum_{n=1}^{N}\phi[n]e[n] = 0 \tag{128}$$

Consistency and Convergence

- We can conclude that

$$\frac{1}{N}\sum_{n=1}^{N}\phi[n]e[n] = \begin{bmatrix} -\widehat{R}_{ye}^{N}[-1] \\ -\widehat{R}_{ye}^{N}[-2] \\ \vdots \\ -\widehat{R}_{ye}^{N}[-n_a] \\ \widehat{R}_{ue}^{N}[0] \\ \widehat{R}_{ue}^{N}[1] \\ \vdots \\ \widehat{R}_{ue}^{N}[n_b] \end{bmatrix} \tag{129}$$

is zero if $e[n]$ and $u[n]$ are uncorrelated.

- We can note that

$$R(N) = \frac{1}{N}\sum_{n=1}^{N}\phi[n]\phi^T[n] = \left[\begin{array}{c|c} R_{yy} & R_{yu} \\ \hline R_{uy} & R_{uu} \end{array}\right] \tag{130}$$

is non-singular if persistent excitation for $u[n]$ is guaranteed. R_{yy} is always non-singular due to $e[n]$. We need $u[n]$ sufficiently exciting for R_{uu} to be non-singular.

Consistency and Convergence

- This result can be generalized to any kind of model structure

$$\widehat{\theta}_N \rightarrow \theta_o \iff \begin{cases} 1) & \mathcal{S} \in \mathcal{M} \\ 2) & \text{input uncorrelated with noise} \\ 3) & \text{persistent excitation of the input} \end{cases} \tag{131}$$

- When the parameters of $G(q,\theta)$ and $H(q,\theta)$ are independent (FIR, OE, BJ) we can replace

$$1) \qquad \mathcal{G} \in \mathcal{M}$$

- In closed-loop systems ($u[n]$ becomes correlated with $e[n]$), we can replace

$$2) \qquad \text{reference uncorrelated with noise} \rightarrow \text{IV Method}$$

Consistency and Convergence

- We can also conclude that

$$(\widehat{\theta}_N - \theta_o) \sim \mathcal{N}(0, \lambda) \tag{132}$$

with

$$\lambda = cov(\widehat{\theta}_N) = \frac{\sigma_e^2}{N} R(N)^{-1} \tag{133}$$

- Provided 1), 2) and 3) hold:
  - $\widehat{\theta}_N$ is an unbiased estimate
  - $cov(\widehat{\theta}_N) \to 0$ when $N \to \infty$

Instrumental Variables

- There is no assumption about the structure of the model.

$$
\begin{aligned}
y[n] &= G(q,\theta)u[n] + H(q,\theta)e[n] \\
&= \frac{B(q,\theta)}{A(q,\theta)}u[n] + \frac{C(q,\theta)}{D(q,\theta)}e[n]
\end{aligned} \qquad (134)
$$

- By multiplying both sides of the equation by $A(q,\theta)$ we write

$$
\begin{aligned}
A(q,\theta)y[n] &= B(q,\theta)u[n] + \frac{A(q,\theta)C(q,\theta)}{D(q,\theta)}e[n] \\
A(q,\theta)y[n] &= B(q,\theta)u[n] + v[n]
\end{aligned}
$$

and finally

$$
y[n] = \phi^T[n]\theta + v[n] \qquad (135)
$$

where

$$
\phi[n] = \begin{bmatrix} -y[n-1] & \cdots & -y[n-n_a] & u[n] & \cdots & u[n-n_b] \end{bmatrix}^T
$$

and

$$
\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_0 & b_1 & \cdots & b_{n_b} \end{bmatrix}^T
$$

Instrumental Variables

- The Instrumental Variable method connects parametric and correlation methods.
- We correlate $y[n] = \phi^T[n]\theta + v[n]$ with an instrumental variable $\xi[n]$ of dimension $n_a + n_b + 1$ that is uncorrelated from the noise $v[n]$, i.e.,

$$\begin{aligned} \frac{1}{N}\sum_{n=1}^{N}\xi[n]y[n] &= \frac{1}{N}\sum_{n=1}^{N}\xi[n]\phi^T[n]\theta + \frac{1}{N}\sum_{n=1}^{N}\xi[n]v[n] \quad (136) \\ &= \frac{1}{N}\sum_{n=1}^{N}\xi[n]\phi^T[n]\theta \end{aligned}$$

- The estimate can be written as

$$\widehat{\theta}_N^{IV} = R(N)^{-1}f(N) \quad (137)$$

where

$$R(N) = \frac{1}{N}\sum_{n=1}^{N}\xi[n]\phi^T[n] \qquad f(N) = \frac{1}{N}\sum_{n=1}^{N}\xi[n]y[n] \quad (138)$$

Instrumental Variables

- Since

$$y[n] = \phi^T[n]\theta_o + v[n] \tag{139}$$

we can write

$$\begin{aligned} \widehat{\theta}_N^{IV} &= R(N)^{-1}\left[\frac{1}{N}\sum_{n=1}^{N}\xi[n]\left(\phi^T[n]\theta_o + v[n]\right)\right] \qquad (140) \\ &= R(N)^{-1}\left[\frac{1}{N}\sum_{n=1}^{N}\xi[n]\phi^T[n]\theta_o + \frac{1}{N}\sum_{n=1}^{N}\xi[n]v[n]\right] \\ &= \theta_o + R(N)^{-1}\frac{1}{N}\sum_{n=1}^{N}\xi[n]v[n] \end{aligned}$$

- Then,

$$\widehat{\theta}_N^{IV} \to \theta_o \iff R(N)^{-1}\frac{1}{N}\sum_{n=1}^{N}\xi[n]v[n] = 0 \tag{141}$$

Instrumental Variables

- We need
  - $\widehat{R}_{\xi v}[\tau] = 0 \Rightarrow \xi[n]$, $v[n]$ uncorrelated
  - $R(N)$ no singular $\Rightarrow \xi[n]$, $\phi[n]$ correlated enough
- In summary,

$$\widehat{\theta}_N^{IV} \to \theta_o \iff \begin{cases} 1) & \mathcal{G} \in \mathcal{M} \\ 2) & \xi[n] \text{ uncorrelated with noise } v[n] \\ 3) & \xi[n] \text{ correlated enough with } \phi[n] \end{cases} \tag{142}$$

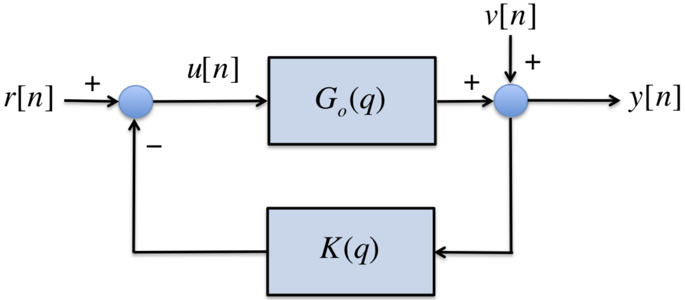
- In open loop, the i.v. is constructed from input $u[n]$

$$\xi[n] = \begin{bmatrix} u[n-n_b-1] & \cdots & u[n-n_b-n_a] & u[n] & \cdots & u[n-n_b] \end{bmatrix}^T$$

- In closed loop, the i.v. is constructed from reference $r[n]$

$$\xi[n] = \begin{bmatrix} r[n] & r[n-1] & \cdots & r[n-n_a-n_b-1] \end{bmatrix}^T$$

Instrumental Variables



- Alternatively,
  - Estimate $\frac{U}{R} = \frac{1}{1+G_o K} = S(\theta)$. Note that we can also estimate $\frac{Y}{R} = \frac{1}{1+G_o K} = T(\theta)$. Knowing $K(q)$ we can obtain an estimate for $G_o(q)$ either from $S(\theta)$ or $T(\theta)$.
  - Filter the noise by computing $\widehat{u}[n] = S(q,\theta) * r[n]$. Therefore, $\widehat{u}[n]$ is uncorrelated with $v[n] \Rightarrow \xi[n] = \widehat{u}[n]$.

Approximate Identification

- Let us assume now the case where $\mathcal{S} \notin \mathcal{M}$:
  - $\mathcal{G} \in \mathcal{M}$, $\mathcal{H} \notin \mathcal{M}$: Still consistent estimation of G_o
    - IV Method
    - LS Method for FIR, OE, BJ
      (G and H independently parameterized)
  - $\mathcal{G} \notin \mathcal{M}$, $\mathcal{H} \in \mathcal{M}$: What can we expect?
  - $\mathcal{G} \notin \mathcal{M}$, $\mathcal{H} \notin \mathcal{M}$: What can we expect?
- **System** - $\mathcal{S}$:

$$y[n] = G(q,\theta_o)u[n] + H(q,\theta_o)e[n] \triangleq G_o(q)u[n] + H_o(q)e[n] \tag{143}$$

- **Model** - $\mathcal{M}$

$$y[n] = G(q,\theta)u[n] + H(q,\theta)e[n] \triangleq G_\theta(q)u[n] + H_\theta(q)e[n] \tag{144}$$

Approximate Identification

- The prediction error is given by

$$
\begin{aligned}
\varepsilon[n,\theta] &= H_\theta^{-1}(q)\{y[n] - G_\theta(q)u[n]\} \\
&= H_\theta^{-1}(q)\{G_o(q)u[n] + H_o(q)e[n] - G_\theta(q)u[n]\} \\
&= H_\theta^{-1}(q)\{(G_o(q) - G_\theta(q))u[n] + (H_o(q) - H_\theta(q))e[n]\} + e[n]
\end{aligned}
$$

  which can be written as

$$
\varepsilon_\theta[n] = H_\theta^{-1}(q) \left[\begin{array}{cc} (G_o(q) - G_\theta(q)) & (H_o(q) - H_\theta(q)) \end{array} \right] \left[\begin{array}{c} u[n] \\ e[n] \end{array} \right] + e[n]
$$

  where we have defined $\varepsilon[n,\theta] \triangleq \varepsilon_\theta[n]$.

- Assumptions:
  - There is a delay both in the system and in the model (G_o and G_θ both contain a delay) or in the controller. That is, $u[n]$ depends only on $y[n-1]$ and earlier values in the case of feedback control.
  - H_o and H_θ are monic. That is, $(H_o - H_\theta)e[n]$ is independent of $e[n]$.

Approximate Identification

- Therefore, $e[n]$ will be uncorrelated with the first term of $\varepsilon_\theta[n]$!

$$\Phi_{\varepsilon\varepsilon}(\omega,\theta) = \frac{1}{|H_\theta|^2}\left[(G_o - G_\theta)\ (H_o - H_\theta)\right]\begin{bmatrix} \Phi_{uu} & \Phi_{ue} \\ \Phi_{eu} & \lambda \end{bmatrix}\begin{bmatrix} \bar{G}_o - \bar{G}_\theta \\ \bar{H}_o - \bar{H}_\theta \end{bmatrix} + \lambda$$

where

$$\begin{bmatrix} \Phi_{uu} & \Phi_{ue} \\ \Phi_{eu} & \lambda \end{bmatrix} = \begin{bmatrix} I & 0 \\ \frac{\Phi_{eu}}{\Phi_{uu}} & I \end{bmatrix}\begin{bmatrix} \Phi_{uu} & 0 \\ 0 & \lambda - \frac{|\Phi_{eu}|^2}{\Phi_{uu}} \end{bmatrix}\begin{bmatrix} I & \frac{\Phi_{eu}}{\Phi_{uu}} \\ 0 & I \end{bmatrix}$$

- Let us introduce

$$B_\theta(e^{j\omega}) = \frac{H_o(e^{j\omega}) - H_\theta(e^{j\omega})}{\Phi_{uu}(\omega)}\Phi_{ue}(\omega)$$

- Then, we can write

$$\Phi_{\varepsilon\varepsilon}(\omega,\theta) = \frac{|G_o - G_\theta + B_\theta|^2}{|H_\theta|^2}\Phi_{uu} + \frac{|H_o - H_\theta|^2\left(\lambda - \frac{|\Phi_{eu}|^2}{\Phi_{uu}}\right)}{|H_\theta|^2} + \lambda$$

Approximate Identification

- The estimation of θ based on N measurements of both input $u[n]$ and output $y[n]$ is computed as

$$\widehat{\theta}_N = \arg\min_{\theta} V_N(\theta) \tag{145}$$

where

$$V_N(\theta) = \frac{1}{N}\sum_{n=1}^{N} \varepsilon^2[n,\theta] = \widehat{\sigma}_\varepsilon(\theta) \tag{146}$$

- By Parseval's theorem

$$\sigma_\varepsilon = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{\varepsilon\varepsilon}(\omega) d\omega \tag{147}$$

- Then,

$$\widehat{\theta}_N = \arg\min_{\theta} \int_{-\pi}^{\pi} \Phi_{\varepsilon\varepsilon}(\omega,\theta) d\omega \tag{148}$$

Approximate Identification

- Open-loop case: $u[n] \perp e[n] \Rightarrow B_\theta(e^{j\omega}) \equiv 0$.

$$\begin{aligned}\Phi_{\varepsilon\varepsilon}(\omega,\theta) &= \frac{|G_o - G_\theta|^2}{|H_\theta|^2}\Phi_{uu} + \frac{|H_o - H_\theta|^2}{|H_\theta|^2}\lambda + \lambda \\ &= \frac{|G_o - G_\theta|^2}{|H_\theta|^2}\Phi_{uu} + \left(\left|\frac{H_o}{H_\theta} - 1\right|^2 + 1\right)\lambda\end{aligned}$$

- But as $\min \int_{-\pi}^{\pi} |R|^2 d\omega = \min \int_{-\pi}^{\pi} (|R-1|^2+1) d\omega$, we can write the expression to minimize as

$$\Phi^m_{\varepsilon\varepsilon}(\omega,\theta) = |G_o - G_\theta|^2 \frac{\Phi_{uu}}{|H_\theta|^2} + \frac{|H_o|^2\lambda}{|H_\theta|^2} = |G_o - G_\theta|^2 \frac{\Phi_{uu}}{|H_\theta|^2} + \frac{\Phi_{vv}}{|H_\theta|^2}$$

Approximate Identification

- **ARX Model** - $\mathcal{M}$

$$y[n] = G_\theta(q)u[n] + H_\theta(q)e[n] = \frac{B_\theta(q)}{A_\theta(q)}u[n] + \frac{1}{A_\theta(q)}e[n] \qquad (149)$$

- The to-be-minimized expression is given by

$$\Phi^m_{\varepsilon\varepsilon}(\omega,\theta) = |G_o - G_\theta|^2\Phi_{uu}|A_\theta|^2 + \frac{\Phi_{vv}}{\left|\frac{1}{A_\theta}\right|^2}$$

- If $\Phi_{uu} \equiv 1$, the limit model is a compromise between fitting $1/|A_\theta|^2$ to the noise spectrum and minimizing

$$\int_{-\pi}^{\pi} |G_o - G_\theta|^2 |A_\theta|^2 d\omega$$

- Note that this problem is a (linear) curve fitting problem, where we minimize the equation error

$$\tilde{E}(\omega,\theta) = E(\omega,\theta)A(\omega,\theta) = \left(G_o(e^{j\omega}) - G_\theta(e^{j\omega})\right) A_\theta(e^{j\omega})$$

Approximate Identification

- **OE Model** - $\mathcal{M}$

$$y[n] = G_\theta(q)u[n] + H_\theta(q)e[n] = \frac{B_\theta(q)}{A_\theta(q)}u[n] + e[n] \tag{150}$$

- The to-be-minimized expression is given by

$$\Phi^m_{\varepsilon\varepsilon}(\omega,\theta) = |G_o - G_\theta|^2\Phi_{uu} + \Phi_{vv}$$

- If $\Phi_{uu} \equiv 1$, the goal is the minimization of

$$\int_{-\pi}^{\pi} |G_o - G_\theta|^2 d\omega$$

- Note that this problem is a (nonlinear) curve fitting problem, where we minimize the output error

$$E(\omega,\theta) = G_o(e^{j\omega}) - G_\theta(e^{j\omega})$$

Practical Identification

- Given

$$Z^N = \{y[n], u[n]; n \leq N\}$$

- We want:
  - A model for the plant
  - A model for the noise
  - An estimate of the accuracy
- We know how to identify a "model" inside an a-priori given "model structure"

– We need to choose a model structure
  - Input design
  - Pre-treatment of data
  - Model set selection
  - Model validation

Input Design

- The input must be **sufficiently exciting**

- White Noise: It is persistently exciting of order ∞. Advantage: It excites all the frequencies. Drawback: It is hard to generate in practice because it has infinity energy. Solution: Filtered white noise

$$u_f[n] = L(q)u[n] \Rightarrow \Phi_{u_f u_f} = |L(e^{j\omega})|^2$$

  The white noise $u[n]$ is generated in the computer. The filter $L(q)$ can be a low/band/high-pass filter.

- Pseudo-random Binary Signal (PRBS):

$$x[n] = 1, \quad x[n+1] = \begin{cases} x[n] & \text{with probability} \quad p \\ -x[n] & \text{with probability} \quad 1-p \end{cases}$$

- Sum of Sinusoids: It excites specific frequencies.

Input Design

- We must remember that

$$\widehat{\theta}_N = \arg\min_{\theta} \int_{-\pi}^{\pi} \Phi_{\varepsilon\varepsilon}(\omega, \theta) d\omega \tag{151}$$

$$\Phi_{\varepsilon\varepsilon}(\omega,\theta) = \frac{|G_o - G_\theta + B_\theta|^2}{|H_\theta|^2} \Phi_{uu} + \frac{|H_o - H_\theta|^2 \left(\lambda - \frac{|\Phi_{eu}|^2}{\Phi_{uu}}\right)}{|H_\theta|^2} + \lambda$$

where

$$B_\theta(e^{j\omega}) = \frac{H_o(e^{j\omega}) - H_\theta(e^{j\omega})}{\Phi_{uu}(\omega)} \Phi_{ue}(\omega)$$

- Open-loop case: $u[n] \perp e[n] \Rightarrow B_\theta(e^{j\omega}) \equiv 0.$

$$\Phi_{\varepsilon\varepsilon}(\omega, \theta) = \frac{|G_o - G_\theta|^2}{|H_\theta|^2} \Phi_{uu} + \frac{|H_o - H_\theta|^2}{|H_\theta|^2} \lambda + \lambda$$

Φ_{uu} works as a weight function in the fitting process.

Data Pre-treatment

- **Bias Removal:** Given

$$A(q)y[n] = B(q)u[n] + v[n]$$

- If $E\{v[n]\} = 0$, then the relation between the static input $\bar{u}$ and static output $\bar{y}$ is given by

$$A(1)\bar{y} = B(1)\bar{u}$$

- The static component of $y[n]$, $\bar{y}$, may not be entirely due to $\bar{u}$, i.e., the noise might be biased ($E\{v[n]\} \neq 0$).
- Method 1: Subtract the means. Define

$$\bar{y} = \frac{1}{N}\sum_{n=1}^{N} y^m[n]; \qquad \bar{u} = \frac{1}{N}\sum_{n=1}^{N} u^m[n]$$

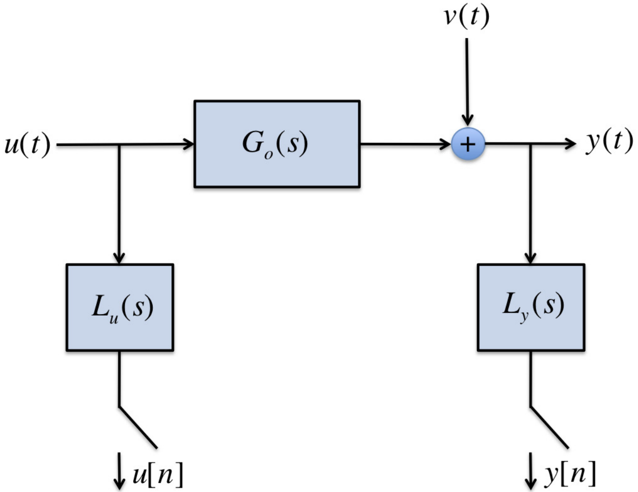
where $y^m[n]$ and $u^m[n]$ represent the measured data. Generate new data:

$$y[n] = y^m[n] - \bar{y}; \qquad u[n] = u^m[n] - \bar{u}$$

- Method 2: Model the offset by an unknown constant β and estimate it

$$A(q)y[n] = B(q)u[n] + \beta + v[n]$$

- **Sampling:** Without an anti-aliasing filter, high frequency content is folded to low frequency



Data Pre-treatment

- The sampling interval $T_s = \frac{1}{f_s}$

1- Defines the maximum frequency $f_{max} = \frac{1}{2} f_s = \frac{1}{2} \frac{1}{T_s}$ that we will see in the sampled signal. Do not sample too slowly.

2- Determines the observation time assuming the number of samples N fixed, i.e., $T = NT_s$. Do not sample too fast.

3- Defines pole location: $z = e^{sT_s}$, where s denotes the pole in continuous time. If $T_s \sim 0$, all the poles of the sampled system are driven to 1 (bad conditioned system near to instability). Do not sample too fast.

Data Pre-treatment

- **Outliers:** These data points can be either erroneous or highly-disturbed. They can have a very bad effect on the estimate since the PEM will try to fit them. They must be removed.

Data Pre-treatment

- **High-frequency Content Filtering:** "High" means "above the frequency range of interest." We filter both input and output with a LTI low-pass (LP) filter $L(q)$, i.e.,

$$y_F[n] = L(q)y[n], \qquad u_F[n] = L(q)u[n].$$

- The model can now be written as

$$A(q)y_F[n] = B(q)u_F[n] + v[n]$$

  with $v[n] = H(q)e[n]$.

- Equivalently,

$$A(q)y[n] = B(q)u[n] + \frac{1}{L(q)}v[n].$$

- Therefore, we multiply the noise by $1/L(q)$ (high-pass filter $\rightarrow$ low-frequency attenuation).

Data Pre-treatment

- We must remember that

$$\widehat{\theta}_N = \arg\min_{\theta} \int_{-\pi}^{\pi} \Phi_{\varepsilon\varepsilon}^{m}(\omega,\theta)d\omega \tag{152}$$

where now

$$\begin{aligned}\Phi_{\varepsilon\varepsilon}^{m}(\omega,\theta) &= |G_o - G_\theta|^2 \frac{|L|^2 \Phi_{uu}}{|H_\theta|^2} + \frac{|H_o|^2\lambda}{|H_\theta|^2} \\ &= |G_o - G_\theta|^2 \frac{|L|^2 \Phi_{uu}}{|H_\theta|^2} + \frac{\Phi_{vv}}{|H_\theta|^2}\end{aligned}$$

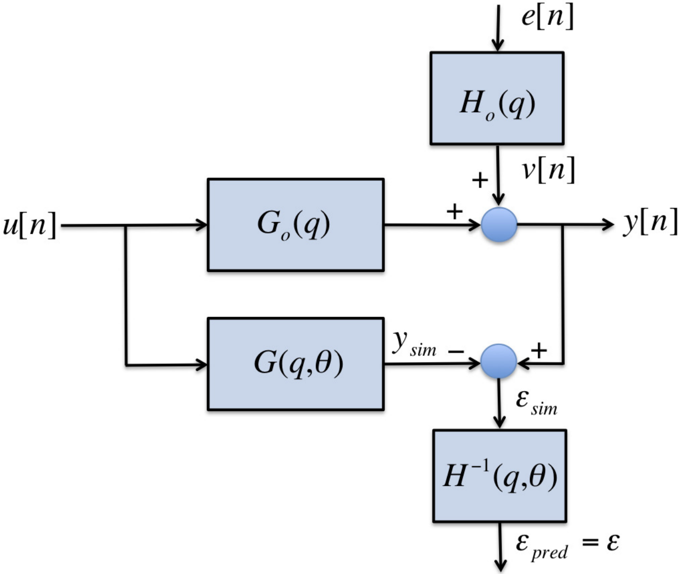
- Note that if G_θ and H_θ are independently parameterized, the fitting method will use H_θ to fit the noise spectrum Φvv and G_θ to fit G_o.

- Note that L can be used as a frequency weighting function to emphasize those frequencies where the fitting is more important.

Model Set Selection

- The goal is to fit the data with the least complex model structure and in this way to avoid over-fitting, which amounts to fitting noise.
- **Order Selection:** Use the singular values of the Hankel matrix to determine the order n of the system. Avoid over-fitting the data with n too high.
- **Delay Selection:** Estimate time delay using
  - Correlation Method
  - Parametric Identification using FIR model structure
- **Model Selection:** Lots of trial and error!!!

Model Validation



Model Validation

- The simulated output is computed as

$$y_{sim}[n,\theta] = G(q,\theta)u[n], \tag{153}$$

and the simulation error as

$$\varepsilon_{sim}[n,\theta] = y[n] - y_{sim}[n,\theta] = y[n] - G(q,\theta)u[n]. \tag{154}$$

- The predicted output is computed as

$$\widehat{y}[n,\theta] = [I - H^{-1}(q,\theta)]y[n] + H^{-1}(q,\theta)G(q,\theta)u[n], \tag{155}$$

and the prediction error as

$$\varepsilon[n,\theta] = y[n] - \widehat{y}[n,\theta] = H^{-1}(q,\theta)\{y[n] - G(q,\theta)u[n]\}. \tag{156}$$

- Note that $\varepsilon[n,\theta] = H^{-1}(q,\theta)\varepsilon_{sim}[n,\theta]$.

Model Validation

- Note that if $G(q,\theta) = G(q)$, then

$$\varepsilon_{sim}[n,\theta] = v[n] \Rightarrow R_{u\varepsilon_{sim}}[\tau] = 0$$

since $u[n]$ and $v[n]$ are uncorrelated by assumption (not true for feedback systems).

- Note that if $G(q,\theta) = G(q)$ and $H(q,\theta) = H(q)$ then

$$\varepsilon[n,\theta]{=}H^{-1}(q,\theta)\varepsilon_{sim}[n,\theta]{=}H^{-1}(q,\theta)v[n]{=}e[n] \Rightarrow R_{\varepsilon\varepsilon}[\tau]{=}\delta[\tau]$$

since $e[n]$ is white noise by assumption.

- We can always compare the Bode plot of the identified parametric model with the Bode plot obtained used non-parametric methods (ETFE/SPA).

Model Validation

- **Loss Function:** The estimation of θ based on N measurements of both input $u[n]$ and output $y[n]$ is

$$\widehat{\theta}_N = \arg\min_{\theta} V_N(\theta) \tag{157}$$

where

$$V_N(\theta) = \frac{1}{N}\sum_{n=1}^{N} \|\varepsilon[n,\theta]\|_2^2 \tag{158}$$

- Then, for an assumed model structure family we can plot the loss-function

$$V_N(\widehat{\theta}) = \frac{1}{N}\sum_{n=1}^{N} \|\varepsilon[n,\widehat{\theta}]\|_2^2 \tag{159}$$

as a function of the number of parameters, i.e., the size of the vector θ.

- Choose the number of parameters that minimize $V_N(\widehat{\theta})$.

Model Validation

Akaike's Information Theoretic Criterion (AIC):

$$\log V_N(\theta) + \frac{n}{N} \tag{160}$$

Akaike's Final Prediction Error Criterion (FPE):

$$\frac{1 + n/N}{1 - n/N} V_N(\theta) \tag{161}$$

These criteria penalize the number of parameters n compared with the number of data points N.

NOTE: Use different sets of data for identification and validation.