Dynamic and Time Series Modeling for Process Control

Sachin C. Patwardhan
Dept. of Chemical Engineering
IIT Bombay

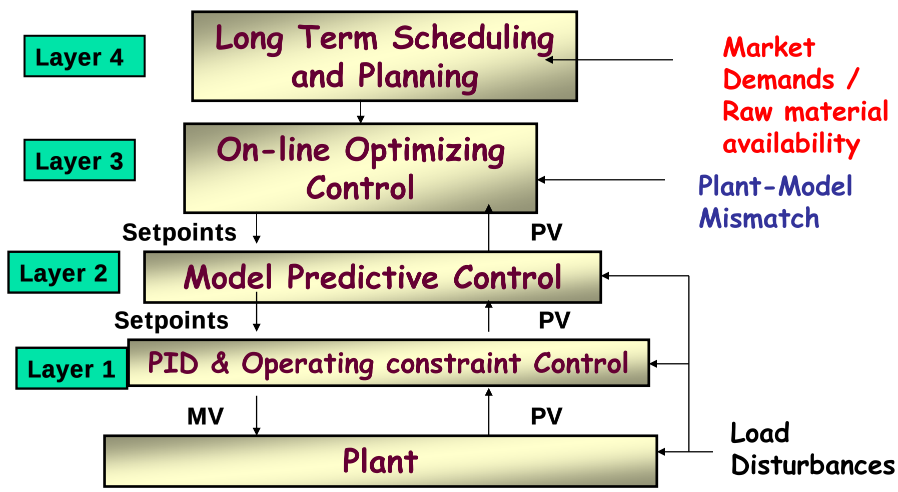
Why Mathematical Modeling?

Key Component of All Advanced Monitoring, Control and Optimization Schemes

- Process Synthesis and Design (offline)
- Operation scheduling and planning
- Process Control
  - Soft sensing / Inferential measurement
  - Optimal control (batch operation)
  - On-line optimization (continuous operation)
  - On-line control (Single loop / multivariable)
- Online performance monitoring Fault diagnosis / fault prognosis

Automation Lab
IIT Bombay

Plant Wide Control Framework



Automation Lab
IIT Bombay

Models for Plant-wide Control

- Layer 4: Aggregate Production Rate Models
- Layer 3: Steady State / Dynamic First Principles Models
- Layer 2: Dynamic Multivariable Time Series Models
- Layer 1: SISO Time Series Models, ANN/PLS/Kalman Filters (Soft Sensing)

Mathematical Models

Qualitative
- Qualitative Differential Equation
- Qualitative signed and directed graphs
- Expert Systems

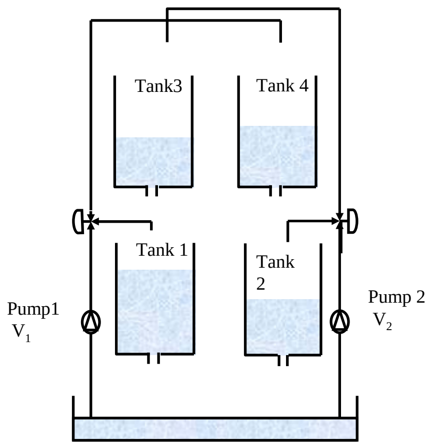
Quantitative
- Differential Algebraic systems
- Mixed Logical and Dynamical Systems
- Linear and Nonlinear time series models
- Statistical correlation based (PCA/PLS)

Mixed
- Fuzzy Logic based models

White Box Models

First Principles / Phenomenological / Mechanistic
- Based on
  - energy and material balances
  - physical laws, constitutive relationships
  - Kinetic and thermodynamic models
  - heat and mass transfer models
- Valid over wide operating range
- Provide insight in the internal working of systems
- Development and validation process:
  difficult and time consuming

Example: Quadruple Tank System



$$\frac{dh_1}{dt} = -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1 k_1}{A_1} v_1$$

$$\frac{dh_2}{dt} = -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2 k_2}{A_2} v_2$$

$$\frac{dh_3}{dt} = -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_1)k_1}{A_3} v_1$$

$$\frac{dh_4}{dt} = -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_2)k_2}{A_4} v_2$$

Manipulated Inputs : v_1 and v_2

Measured Outputs : h_1 and h_2

Example: Non-isothermal CSTR

Reaction : $A \longrightarrow B$

Material Balance

$$V\frac{dC_A}{dt} = F(C_{A0} - C_A) - Vk_0 \exp(-E/RT)C_A$$

Enery Balance

$$V\rho c_p \frac{dT}{dt} = \rho C_p F(T_0 - T) - Q$$
$$+ (-\Delta H_{rxn})Vk_0 \exp(-E/RT)C_A$$

Heat Tranfer to Cooling Jacket

$$Q = \frac{aF_c^{b+1}}{F_c + (aF^b / 2\rho_c C_{pc})}(T - T_{cin})$$

Example: Fed-Batch Fermenter

$$\frac{d(XV)}{dt} = \mu(S_1,S_2)XV \qquad (X : \text{Biomass Conc.})$$

$$\frac{d(S_2V)}{dt} = F_2S_{2F} - \sigma_2(S_1,S_2)XV \qquad (S_2 : \text{Substrate - 2 Conc.})$$

$$\frac{d(PV)}{dt} = \pi(S_1,S_2)XV - kPV \qquad (P : \text{Product Conc.})$$

$$\frac{dV}{dt} = F_2 \qquad (V : \text{Reactor Volume})$$

$$\mu(S_1,S_2) = \frac{0.086S_1S_2}{2.0 + S_1 + 0.0303S_1^2}$$

$$\sigma_2(S_1,S_2) = \mu(S_1,S_2)/1.05 \quad ; \quad \pi(S_1,S_2) = 117.7e^{-0.311S_2}\mu(S_1,S_2)$$

Fixed Bed Reactor

- **Material Balances** **(Distributed Parameter System)**

$$\frac{\partial C_A}{\partial t} = -v_l\frac{\partial C_A}{\partial z} - k_{10}e^{-E_1/RT_r}C_A \qquad \text{........Reactant A}$$

$$\frac{\partial C_B}{\partial t} = -v_l\frac{\partial C_B}{\partial z} + k_{10}e^{-E_1/RT_r}C_A - k_{20}e^{-E_2/RT_r}C_B \qquad \text{........Product B}$$

- **Energy Balances**

$$\frac{\partial T_r}{\partial t} = -v_l\frac{\partial T_r}{\partial z} + \frac{(-\Delta H_{r1})}{\rho_m C_{pm}}k_{10}e^{-E_1/RT_r}C_A + \frac{(-\Delta H_{r2})}{\rho_m C_{pm}}k_{20}e^{-E_2/RT_r}C_B + \frac{U_w}{\rho_m C_{pm}V_r}(T_j - T_r) \qquad \text{........Reactor Temp.}$$

$$\frac{\partial T_j}{\partial t} = u\frac{\partial T_j}{\partial z} + \frac{U_{wj}}{\rho_{mj}C_{pmj}V_j}(T_r - T_j) \qquad \text{........Jacket Temp.}$$

Grey Box Models

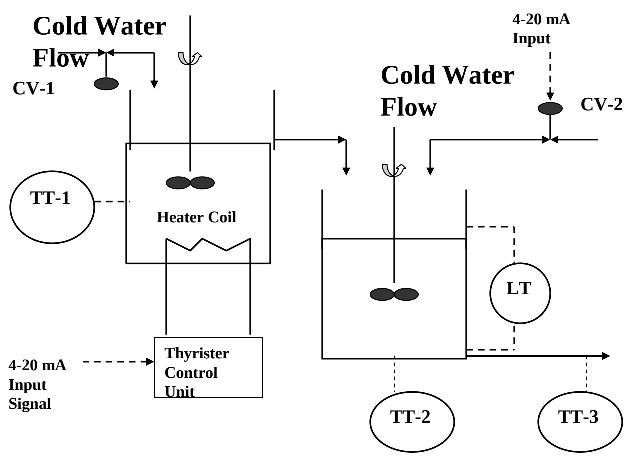
Semi-Phenomenological

Part of model developed from the first principles and part developed from data

Example: dynamic model for reactor model using energy and material balance and Reaction kinetics modeled using neural network

Better choice than complete black box models

Example: Stirred Tank Heater-Mixer



Experimental Setup: Schematic Diagram

Example: Stirred Tank Heater-Mixer

$$\frac{dT_1}{dt} = \frac{F_1}{V_1}(T_{i1} - T_1) + \frac{Q(I_1)}{V_1 \rho C_p}$$

$$\frac{dh_2}{dt} = \frac{1}{A_2}[F_1 + F_2(I_2) - F]$$

$$\frac{dT_2}{dt} = \frac{1}{h_2 A_2}\left[F_1(T_1 - T_2) + F_2(T_{i2} - T_2) - \frac{UA(T_2 - T_{atm})}{\rho C_p}\right]$$

$$Q(I_1) = 7.979 I_1 + 0.989 I_1^2 - 0.0073 I_1^3$$

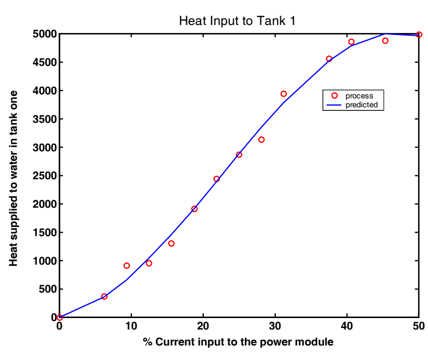
$$F_2(I_2) = 3.9 + 27 I_2 - 0.71 I_2^2 + 0.0093 I_2^3$$

$$U = 139.5\ J / m^{2\ 0}Ks \quad ; \quad F(h) = k\sqrt{h_2 - h}$$

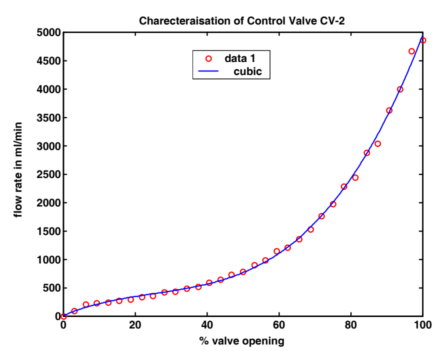
I_1 : % current input to thyrister power controller

I_2 : % current input to control valve

Example: Stirred Tank Heater-Mixer

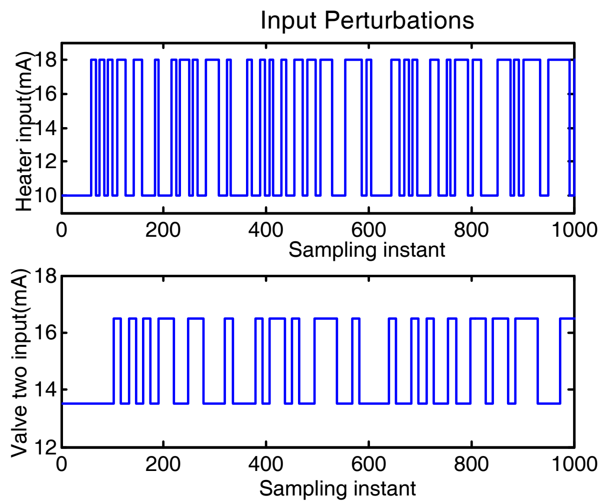


Thyrister Power Controller Characterization

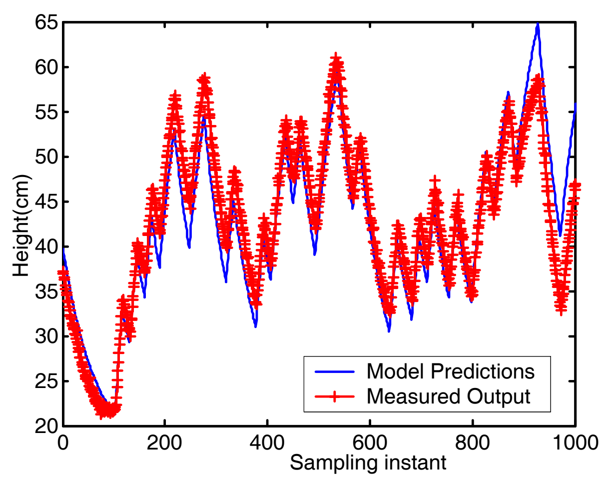


Control Valve Characterization

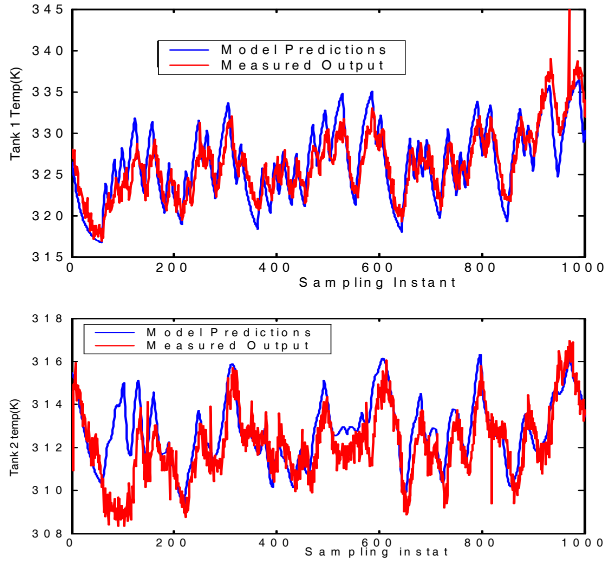
Model Validation: Input Excitations



Model Validation: Level Variations



Model Validation: Temperature Profiles



Dynamic Models for Control

- **Linear perturbation models:** Regulatory operation around fixed operating point of mildly nonlinear processes operated continuously. Developed using
  - Local linearization of white/gray box models
  - Identification from input output data

  Why use approximate Linear Models?
  - Linear control theory for controller synthesis and closed loop analysis is very well Developed
  - For **small** perturbations near operating point, processes exhibit linear dynamics
- **Nonlinear dynamic models:** strongly nonlinear systems, operation over wide operating range, batch / semi-batch processes

Local Linearization

Given a lumped parameter model

$$dX/dt = F(X,U,D) \quad ; \; Y = G(X)$$

and steady state operating point $(\overline{X},\overline{U},\overline{D})$,

we apply Taylor series expansion around $(\overline{X},\overline{U},\overline{D})$

to develop linear perturbation model

$$dx/dt = Ax + Bu + Hd \quad ; \; y = Cx$$

Perturbation variables

$$x(t) = X(t) - \overline{X} \;; y(t) = Y(t) - \overline{Y} \;;$$

$$u(t) = U(t) - \overline{U} \;; \; d(t) = D(t) - \overline{D} \;;$$

Local Linearization

where

$$A = [\partial F / \partial X] \, ; \, B = [\partial F / \partial U] \, ;$$

$$H = [\partial F / \partial D] \, ; \; C = [\partial G / \partial X]$$

computed at $(\overline{X},\overline{U},\overline{D})$

Transfer Function Matrix:
Can be obtained by taking Laplace transform together with assumption $x(0) = \overline{0}$
(i.e. initial state of the process corresponds to operating steady state)

$$y(s) = G_p(s)u(s) + G_d(s)d(s)$$

$$G_u(s) = C[sI - A]^{-1}B \quad ; \; G_d(s) = C[sI - A]^{-1}H$$

Perturbation Model for CSTR

Consider non-isothermal CSTR dynamics

$$\frac{dC_A}{dt} = f_1(C_A, T, F, F_c, C_{A0}, T_{cin})$$

$$\frac{dT}{dt} = f_2(C_A, T, F, F_c, C_{A0}, T_{cin})$$

feed flow rate

coolant flow rate

States $(X) \equiv [C_A \;\; T]^T$ Measured Output $(Y) \equiv [T]$

Manipulated Inputs $(U) \equiv [F \;\; F_c]^T$

Unmeasured Disturbances $(D_u) \equiv [C_{A0}]$

Measured Disturbances $(D_m) \equiv [T_{cin}]$

Feed conc.

Cooling water Temp.

CSTR: Model Parameters and Steady state Operating Point

V (Reactor volume) = 1 m^3 ; F (Inlet flow) = 1 m^3/min ;

C_{A0}(Inlet concentration of A) = 2.0 kmol/m^3;

T_0 (Inlet temperature) = 50 ^0C ; F (Coolant flow) = 15 m^3/min ;

C_p(Specific heat of reacting mixture) = 1 cal/(g K) ;

Tcin (Coolent Inlet Temperature) = 92 ^0C;

C_{pc} (spacific heat of coolent) = 1 cal /(g K) ;

ρ (Reacting liquid density) = 10^6 g/m^3; ρ_c(Coolent density) = 10^6 g/m^3;

- ΔH_{rnx} (Heat of reaction) = 130 x 10^6 cal/kmol ;

a = 1.678 x 10^6 cal / min ; $b = 0.5$; E/ R = 8330.1 K

C_A(Concentration of A) = 0.265kmol/m^3
T(Reactor Temperature) $= 121^0 C$

Operating Steady State

Discrete Dynamic Models

Computer control relevant discrete models

$$x(k+1) = \Phi x(k) + \Gamma u(k)$$
$$y(k) = Cx(k)$$
$$\Phi = \exp(AT) \quad ; \quad \Gamma = \int_0^T \exp(A\tau)B\,d\tau$$

Definition

$$\Phi = \exp(AT) = I + T\Phi + \frac{T^2}{2!}\Phi^2 + ...$$

Note: Assumption of piece-wise constant inputs holds only for manipulated inputs and NOT for the disturbances or any other input

Transfer Function Matrix

q-Transfer Function Matrix: Can be obtained by taking q-transform together with assumption $x(0) = \overline{0}$

$$y(k) = G_p(q)u(k)$$
$$G_p(q) = C[qI - \Phi]^{-1}\Gamma$$

q : Shift Operator

$$q\{f(k)\} = f(k+1) \quad ; \quad q^{-1}\{f(k)\} = f(k-1)$$

Alternatively, taking z - transform on bothe sides of difference equation

$$zx(z) - x(0) = \Phi x(z) + \Gamma u(z)$$

When $x(0) = \overline{0}$

$$x(z) = [zI - \Phi]^{-1}\Gamma u(z)$$
$$y(z) = Cx(z) = [zI - \Phi]^{-1}\Gamma u(z)$$
$$G_p(z) = C[zI - \Phi]^{-1}\Gamma \quad \text{: Pulse Transfer Function}$$

Automation Lab
IIT Bombay

Computation of System Matrices

Compouation Method 1 :

Let $A = \Psi \Lambda \Psi^{-1}$ where Λ is diagonal matrix
with eigenvalues appearing on maion diagonal
Ψ : matrix with eigenvectors of A as columns

$$\Phi = \Psi \exp(\Lambda T)\Psi^{-1}$$

$$\Gamma = \Psi \left[\int_0^T \exp(\Lambda \tau) d\tau \right] \Psi^{-1} B$$

Compouation Method 2 :

$\Phi(t) = \exp(At)$ is solution of ODE - IVP

$$\frac{d\Phi}{dt} = A\Phi(t) \;\; ; \;\; \Phi(0) = I$$

Taking Laplace Transform

$$s\Phi(s) - \Phi(0) = A\Phi(s) \Rightarrow \Phi(s) = [sI - \Phi]^{-1} \Rightarrow \Phi(T) = L^{-1}\left[(sI - \Phi)^{-1}\right]_{t=T}$$

$$\Gamma = \left[\int_0^T \exp(A\tau) d\tau \right] B$$

When A is invertible matrix $\Gamma = [\exp(AT) - I]A^{-1}B = [\Phi - I]A^{-1}B$

Automation Lab
IIT Bombay

CSTR: Continuous Perturbation Model

Continuous linear state space model

$$x(t) = \begin{bmatrix} C_A(t) - \overline{C}_A \\ T(t) - \overline{T} \end{bmatrix} \; ; \; u(t) = \begin{bmatrix} F(t) - \overline{F} \\ F_c(t) - \overline{F}_c \end{bmatrix} \; ;$$

$$dx/dt = \begin{bmatrix} -7.56 & -0.09 \\ 852.72 & 5.77 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 1.735 \\ -6.07 & -70.95 \end{bmatrix} u(t)$$

$$y(t) = [0 \quad 1] x(t)$$

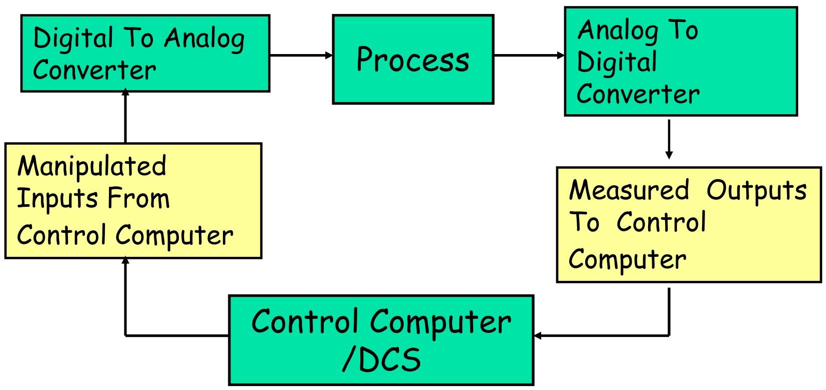
Laplace Transfer Function

$$G_p(s) = \begin{bmatrix} \dfrac{-6.07\,s - 45.9}{s^2 + 1.79\,s + 35.8} & \dfrac{-70.95\,s + 943.5}{s^2 + 1.79\,s + 35.83} \end{bmatrix}$$

Automation Lab
IIT Bombay

Models for Computer Control

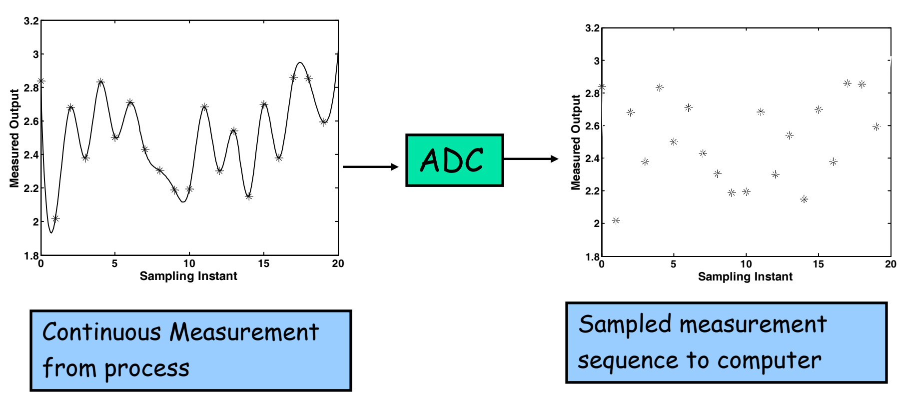
Computer controlled system / Distributed Digital Control system



Automation Lab
IIT Bombay

Digital Control: Measured Outputs

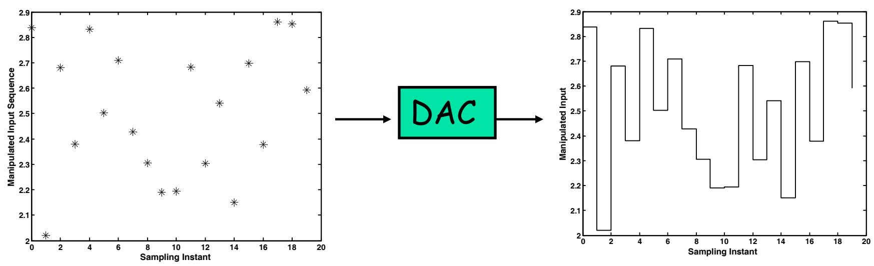
Output measurements are available only at discrete sampling instant $\{t_k = kT : k = 0,1,2,....\}$
Where T represents sampling interval



Digital Control: Manipulated Inputs

In computer controlled (digital) systems Manipulated inputs implemented through DAC are piecewise constant

$$u(t) = u(t_k) \equiv u(k) \quad for \;\; t_k \leq t \leq t_{k+1}$$



Input Sequence Generated by computer

Continuous input profile generated by DAC

CSTR: Discrete Perturbation Model

Discrete linear state space model

Sampling Time (T) = 0.1 min

$$x(k+1) = \begin{bmatrix} 0.185 & -0.008 \\ 73.492 & 1.333 \end{bmatrix} x(k) + \begin{bmatrix} 0.0026 & 0.134 \\ -0.7335 & -1.797 \end{bmatrix} u(k)$$

$$y(k) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(k)$$

Discrete q-transfer function model

$$G_p(q) = \begin{bmatrix} \dfrac{-6.07\,q - 45.9}{q^2 + 1.79\,q + 35.83} & \dfrac{-70.95\,q + 943.5}{q^2 + 1.79\,q + 35.83} \end{bmatrix}$$

$$G_p(q^{-1}) = \begin{bmatrix} \dfrac{-6.07\,q^{-1} - 45.9q^{-2}}{1 + 1.79\,q^{-1} + 35.83q^{-2}} & \dfrac{-70.95\,q^{-1} + 943.5q^{-2}}{1 + 1.79\,q^{-1} + 35.83q^{-2}} \end{bmatrix}$$

Black Box Models

Data Driven / Black Box Models

Static maps (correlations)/ dynamic models (difference equations) developed directly from *historical input-output data*

Valid over limited operating range

Provide no insight into internal working of systems

Development process: much less time consuming and comparatively easy

Black Box Models

Dynamic Models: Given observed data

$$\text{Set of past Inputs}: U^{(k)} = [u(1) \quad u(2) \quad \ldots \quad u(k)]$$

$$\text{Measured Outputs}: Y^{(k)} = [y(1) \quad y(2) \quad \ldots \quad y(k)]$$

we are looking for relationship

$$\mathbf{y}(k) = \Omega\left(U^{(k-1)}, Y^{(k-1)}, \theta\right) + e(k)$$

such that ***noise*** (residuals) e(k) are as small as possible

$\theta \in R^d$ represents parameter vector

Automation Lab
IIT Bombay

Tools for Black Box Modeling

Linear Difference equation (time series) models

Principle component analysis (PCA) / Projection Of latent structures (PLS) /
Statistical models based on linear correlation analysis of historical data

Artificial Neural Networks/Wavelet Networks
Excellent for capturing arbitrary nonlinear maps

Fuzzy Rule Based Models
Quantification of qualitative process knowledge

Automation Lab
IIT Bombay

Steps in Model Development

- Selection of model structure
- Planning of experiments for estimation of unknown model parameters
  - Design of input perturbation sequences
  - Open loop / closed loop experimentation
- Estimation of model parameters from experimental data using optimization techniques
- Model validation
  - Prediction capabilities
  - Steady state behavior

Model Structure Selection

Issues in Model Selection

- Process application (batch / continuous)
- Time scale of operation
- Type of application (scheduling/optimization/MPC/Fault Diagnosis)
- Availability of physical knowledge / historical data
- Development time and efforts

Model granularity decides how well we can make control / planning moves or diagnose / analyze process behavior

Data Driven Models

Development of linear state space/transfer models starting from first principles/gray box models is impractical proposition.

Practical Approach

- Conduct experiments by perturbing process around operating point
- Collect input-output data
- Fit a differential equation or difference equation model

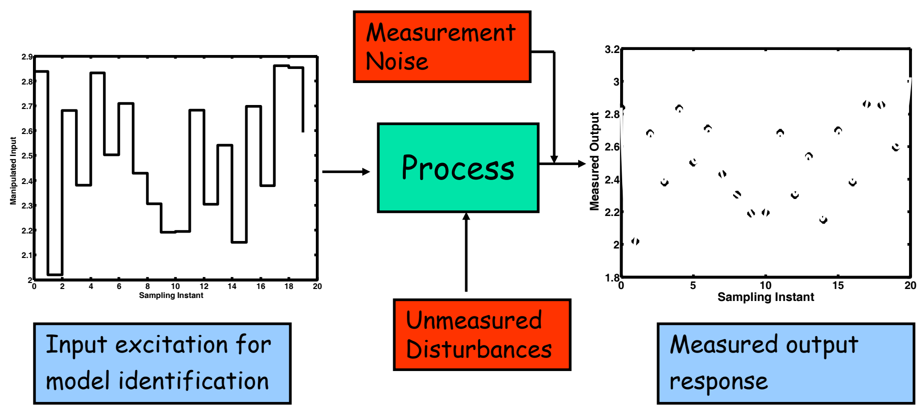
Difficulties

- Measurements are inaccurate
- Process is influenced by unknown disturbances
- Models are approximate

Automation Lab
IIT Bombay

Discrete Model Development

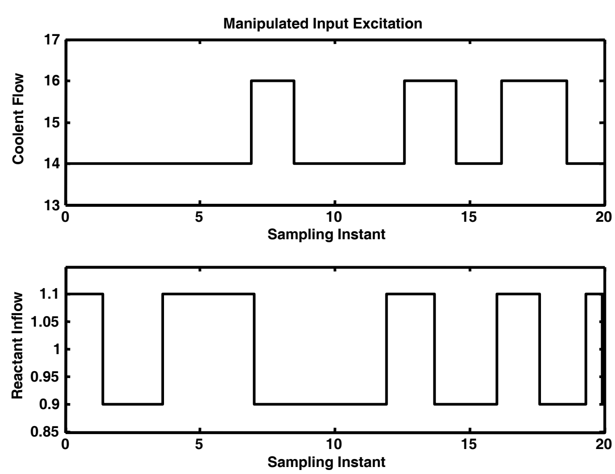
Excite plant around the desired operating point by injecting input perturbations



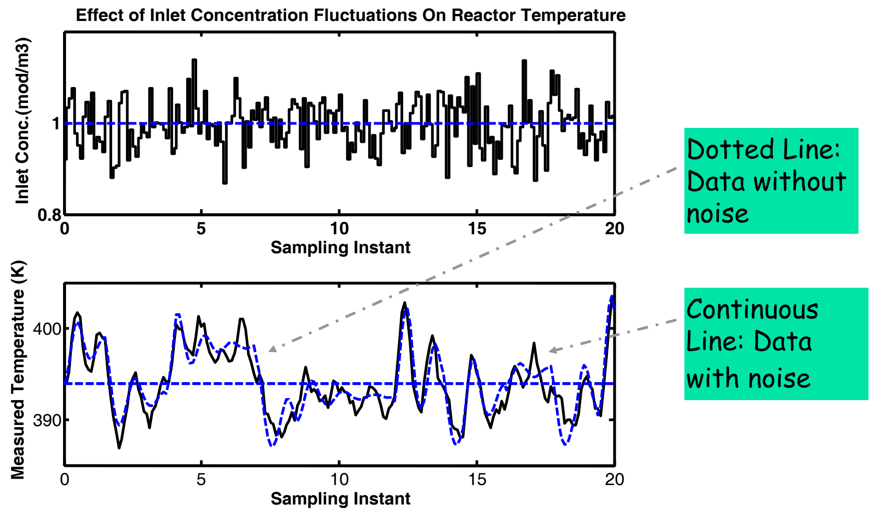
Automation Lab
IIT Bombay

CSTR: Input Excitation

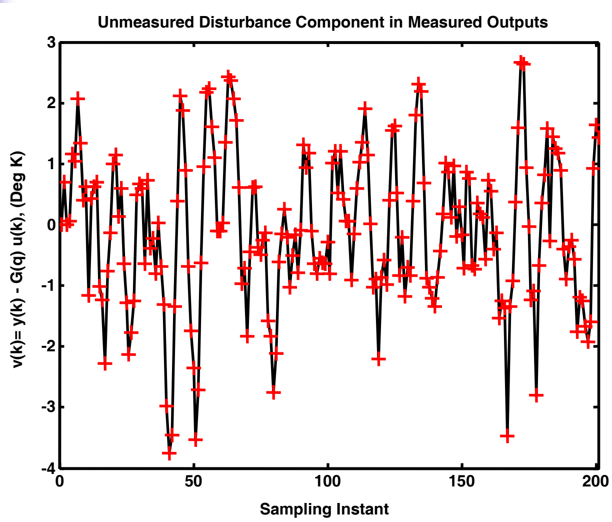
PRBS: Pseudo Random Binary Signal



CSTR: Identification Experiments

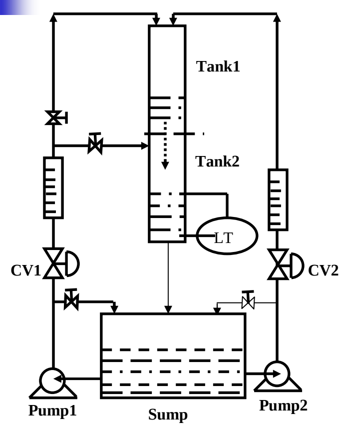


CSTR: Noise Component



Automation Lab
IIT Bombay

Two Non-Interacting Tanks Setup

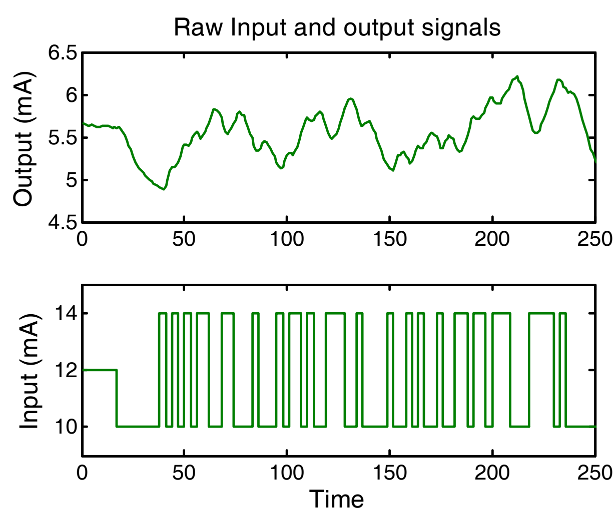


Non Interacting Tank Level Control setup

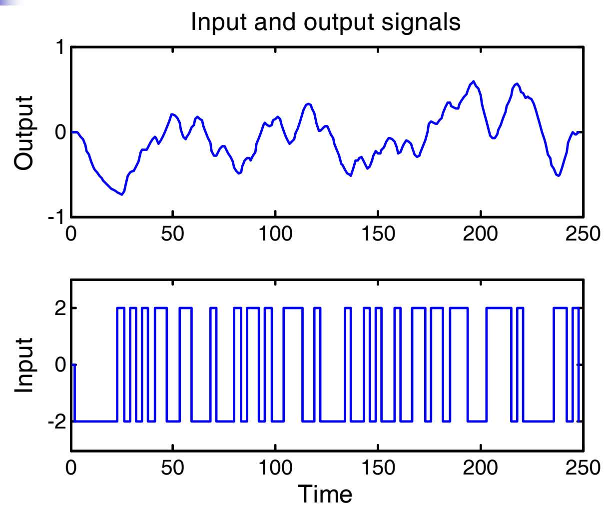
SISO System
Output: Level in tank 2
Manipulated Input :
Valve Position CV-2
Disturbance:
Valve Position CV-1

Automation Lab
IIT Bombay

Input Output Data



Perturbation Data for Identification



Mean values removed from Input and Output data

Impulse Response Model

Consider T.F. $\frac{y(s)}{u(s)} = g(s)$ with impulse input

$$y_{impulse}(t) = g(t) = L^{-1}[g(s)]$$

Convolution Integral : $y(t) = \int_0^\infty g(\tau)u(u-\tau)d\tau$

For piece - wise constant inputs

$$y(kT) = \left[\int_0^T g(\tau)d\tau\right]u[(k-1)T] + \left[\int_T^{2T} g(\tau)d\tau\right]u[(k-2)T] + \ldots$$

$$y(k) = \sum_{j=1}^{\infty}\left[\int_0^T g(\tau)d\tau\right]u(k-j) = \sum_{j=1}^{\infty} g_T(j)u(k-j)$$

Impulse Response Coefficients : $g_j = \left[\int_0^T g(\tau)d\tau\right]$

Impulse Response Model

Impulse Response Model

$$y(k) = \sum_{j=1}^{\infty} g_j u(k-j) = \sum_{j=1}^{\infty} g_j q^{-j} u(k)$$

Defining transfer operator $G(q) = \sum_{j=1}^{\infty} g_j q^{-j}$

$$y(k) = G(q)u(k)$$

- ✓Current output y(k) is viewed as weighted sum of all past inputs moves.
- ✓Impulse response coefficients determine weighting of each past move
- ✓G(q) is open loop BIBO stable if

$$\sum_{j=1}^{\infty} |g_j| < \infty$$

Discrete Model Forms

Finite Impulse Response (FIR) Model

For open loop stable systems

$$|g_k| \to 0 \text{ as } k \to \infty$$

$$y(k) \cong \sum_{j=1}^{N} g_j u(k-j)$$

Discrete Transfer Function Model

$$\frac{y(q^{-1})}{u(q^{-1})} = \frac{b_1 q^{-1} + b_2 q^{-2} + \ldots b_n q^{-2}}{1 + a_1 q^{-1} + .. a_n q^{-n}}$$

Example

$$\frac{y(q^{-1})}{u(q^{-1})} = \frac{b_1 q^{-1} + b_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}}$$

which is equivalent to

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-1)$$

Output Error (OE) Model

Data collected through experiments

Set of N output Measurements

$$Y^N \equiv \{y(k) : y(0), y(1), y(2), \ldots\ldots, y(N)\}$$

Set of Input Sequence

$$U^N \equiv \{u(k) : u(0), u(1), u(2), \ldots\ldots, u(N)\}$$

Output / Measurement Error Model

$$y(k) = G(q)u(k) + v(k)$$

Measured Value of Output

Deterministic component

Residue: unmeasured disturbances + measurement noise

Estimation of FIR Model

Consider FIR model with n coeffieints

$$y(k) = g_1 u(k-1) + \ldots\ldots + g_n u(k-n) + v(k)$$

Using experimental data we can write

$$y(n) = g_1 u(n-1) + \ldots. + g_n u(0) + v(n)$$

$$y(n+1) = g_1 u(n) + \ldots. + g_n u(1) + v(n+1)$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$y(N) = g_1 u(N-1) + \ldots\ldots + g_n u(N-n) + v(N)$$

Arranging in matrix form

$$\begin{bmatrix} y(n) \\ y(n+1) \\ \ldots \\ y(N) \end{bmatrix} = \begin{bmatrix} u(n-1) & u(n-2) & .. & u(0) \\ u(n) & u(n-1) & .. & u(1) \\ .. & .. & .. & .. \\ u(N-1) & .. & .. & u(N-n) \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ \ldots \\ g_n \end{bmatrix} + \begin{bmatrix} v(n) \\ v(n+1) \\ \ldots \\ v(N) \end{bmatrix}$$

Least square estimation

Resulting model is linear in parameters

$$\mathbf{Y} = \mathbf{A}\boldsymbol{\theta} + \mathbf{V}$$

Least square parameter estimation

$$\hat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \mathbf{V}^T\mathbf{V} = \min_{\boldsymbol{\theta}} [\mathbf{Y} - \mathbf{A}\boldsymbol{\theta}]^T [\mathbf{Y} - \mathbf{A}\boldsymbol{\theta}]$$

$$\hat{\boldsymbol{\theta}} = \left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}\mathbf{Y}$$

Let the noise sequence $\{v(k)\}$ have zero mean and let $\boldsymbol{\theta}_T$ represent true value of the parameter vector, i.e.

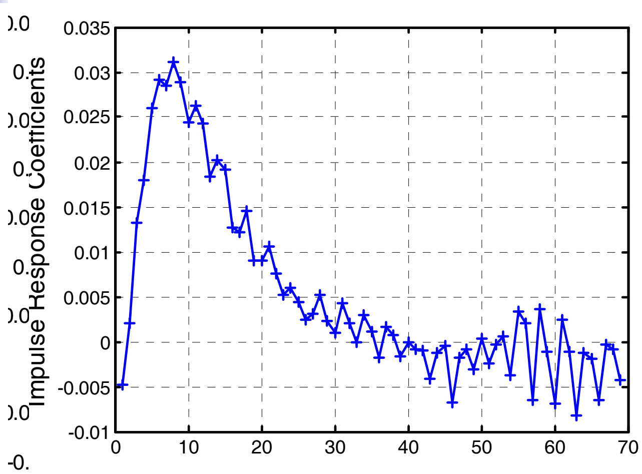
$$\mathbf{Y} = \mathbf{A}\boldsymbol{\theta}_T + \mathbf{V}$$

$$\hat{\boldsymbol{\theta}} = \left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}\mathbf{Y} = \left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T[\mathbf{A}\boldsymbol{\theta}_T + \mathbf{V}]$$

$$= \boldsymbol{\theta}_T + \left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T\mathbf{V}$$

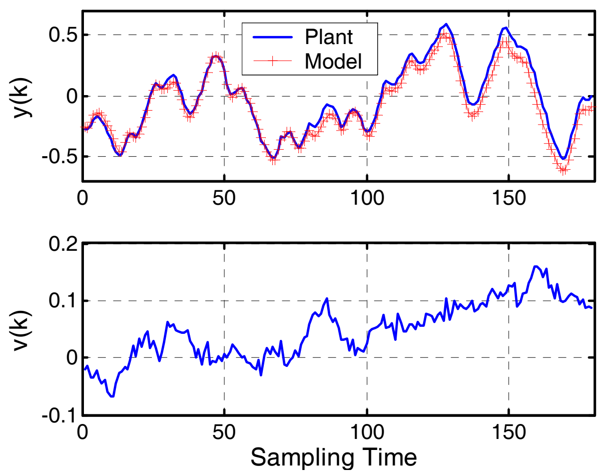
$$E\left[\hat{\boldsymbol{\theta}}\right] = \boldsymbol{\theta}_T + \left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T E[\mathbf{V}] = \boldsymbol{\theta}_T$$

Estimated FIR Model



Automation Lab
IIT Bombay

FIR Model Fit

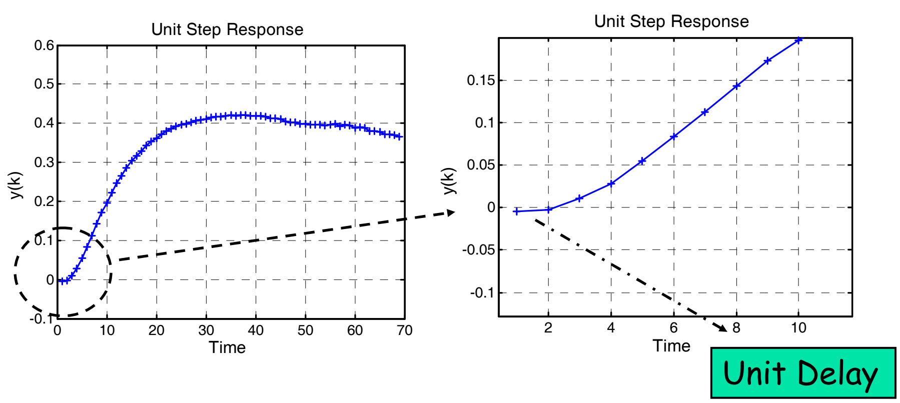


Automation Lab
IIT Bombay

Estimated step Response

Step response can be estimated from impulse response coefficients

$$\text{Unit Step Response Coefficient}: a_i = \sum_{j=1}^{i} g_j$$



Features of estimation

Thus, the least square estimation generates an unbiased estimate of model parameters when $E[\mathbf{V}] = \bar{0}$

If $\{v(k)\}$ is white noise sequence with variance σ^2, then

$$Cov[\mathbf{V}] = E[\mathbf{V}\mathbf{V}^T] = \sigma^2 I$$

$$Cov\{\hat{\boldsymbol{\theta}}\} = E\left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_T)(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_T)^T\right]$$

$$= [\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}^T E[\mathbf{V}\mathbf{V}^T]\mathbf{A}[\mathbf{A}^T\mathbf{A}]^{-1} = \sigma^2[\mathbf{A}^T\mathbf{A}]^{-1}$$

σ^2 can be estimated as

$$\hat{\sigma}^2 = \frac{1}{N}\hat{\mathbf{V}}^T\hat{\mathbf{V}} = \frac{1}{N}(\mathbf{Y} - \mathbf{A}\hat{\boldsymbol{\theta}})^T(\mathbf{Y} - \mathbf{A}\hat{\boldsymbol{\theta}})$$

Thus, estimated parameter covariance matrix is

$$\hat{C}ov\{\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}}\} = \frac{1}{N-n}(\hat{\mathbf{V}}^T\hat{\mathbf{V}})[\mathbf{A}^T\mathbf{A}]^{-1}$$

Difficulties with FIR Model

Advantages: Method can be easily extended to multiple input case

Difficulty:

Variance Errors in FIR Model Parameters

$$var(g_i)\ \alpha\ [1/(N-n)]$$

Variances of parameter estimates can be reduced by increasing data length (N)

Disadvantages:

✓Large number of parameters for MIMO case

✓Large data set required to get good parameter estimates, which implies long time for experimentation.

Alternate Model Form

$$y(k) = \frac{b_1 q^{-1} + ... + b_n q^{-2}}{1 + a_1 q^{-1} + ... + a_n q^{-2}} q^{-d} u(k) + v(k)$$

Output Error

Parameterized OE model

Two tank system under consideration is expected to have second order dynamics

$$x(s) = \frac{k_p}{(\tau_1 s + 1)(\tau_2 s + 1)} u(s)$$

which is equivalent to 2'nd order discrete time model

$$x(k) = \frac{b_1 q^{-1} + b_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}} u(k)$$

Since time delay (dead time) was found to be d = 1

$$x(k) = \frac{b_1 q^{-1} + b_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}} q^{-1} u(k)$$

which is equivalent to

$$x(k) = -a_1 x(k-1) - a_2 x(k-2) + b_1 u(k-2) + b_2 u(k-3)$$

$$y(k) = x(k) + v(k)$$

Parameter Estimation

x(k) : true value Y(k) : measured value
v(k) : measurement noise / disturbance

Difficulty: Only {y(k)} sequence is known. Sequence {x(k)} is unknown
Consequence: Linear least square method can't be used for parameter estimation

Given $(a_1, a_2, b_1, b_2, x(0), x(1), x(2))$ and d = 1

we can recursively estimate x(k) as

$$x(3) = -a_1 x(2) - a_2 x(1) + b_1 u(1) + b_2 u(0)$$

$$x(4) = -a_1 x(3) - a_2 x(2) + b_1 u(2) + b_2 u(1)$$

....

$$x(N) = -a_1 x(N-1) - a_2 x(N-2) + b_1 u(N-2) + b_2 u(N-3)$$

$$v(k) = y(k) - x(k) \quad for \quad k = 3,4,....N$$

Parameter Estimation

Nonlinear Optimization Problem

Estimate $(a_1, a_2, b_1, b_2, x(0), x(1))$ such that

$$\Psi[e(0),...e(k)] = \sum_{k=2}^{N}[v(k)]^2$$

is minimized with respect to $(a_1, a_2, b_1, b_2, x(0), x(1))$

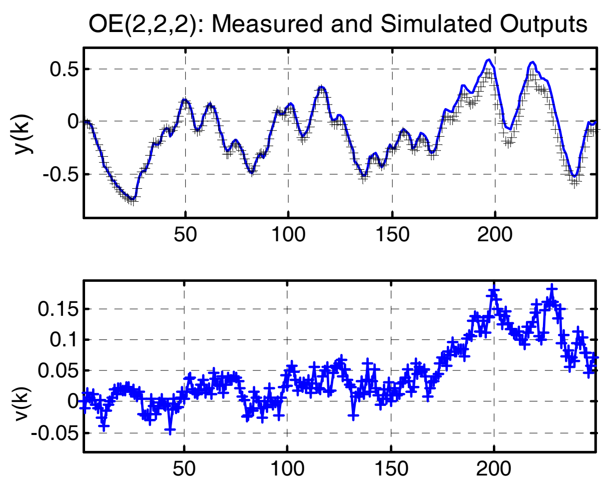
$$v(k) = y(k) - x(k)$$

$$x(k) = -a_1 x(k-1) - a_2 x(k-2) + b_1 u(k-2) + b_2 u(k-3)$$

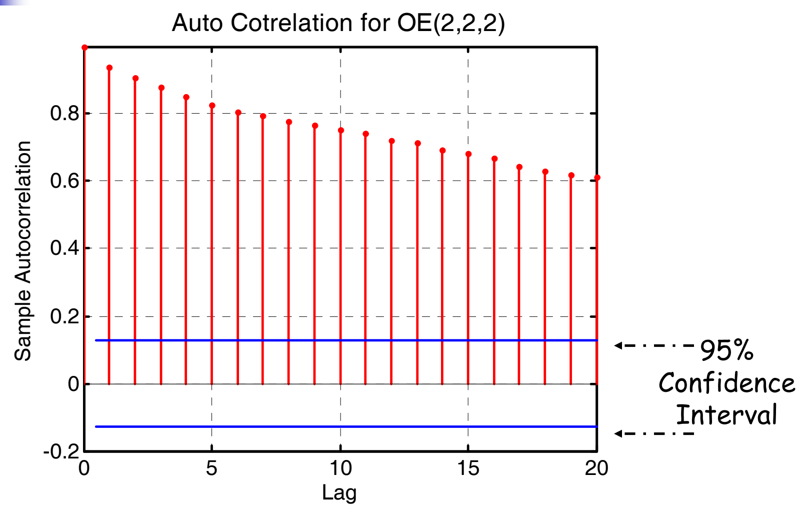
Simplification : Choose x(0) = x(1) = 0

Identified Model Parameters
y(k) = [B(q)/A(q)]u(k) + v(k)
B(q) = 4.567e-006 q^-2 + 0.01269 q^-3
A(q) = 1 - 1.653 q^-1 + 0.6841 q^-2

OE Model



OE Model : Autocorrelation



Unmeasured Disturbance Modeling

- The measured output y(k) contains contributions due to
  - Measurement errors (noise)
  - Unmeasured disturbances
- In additions modeling (equation) errors arise while developing approximate linear perturbation models

Thus, in order to extract true model parameters from the data, we need to carry out modeling of **unmeasured disturbances (or noise)**

Noise is modeled as a **stochastic process** (sequence of random variables, which are correlated in time)

Noise Modeling

$$y(k) = G(q)u(k) + v(k)$$

Deterministic component

Residue: unmeasured disturbances + measurement noise

Note: Information about unmeasured disturbances in the past is contained in the output measurement record. Thus, an obvious choice of model structure is

$$y(k) = f\left[\mathrm{u(k-1),....,u(k-m), y(k-1), ..., y(k-p)}\right] + e(k)$$

Equation Error Model

A discrete linear model, which captures the effect of past unmeasured disturbances, can be proposed as

$$y(k) = b_1 u(k-d-1) + ... + b_m u(k-d-m)$$
$$- a_1 y(k-1) - ... - a_n y(k-n) + e(k)$$

d : Time delay / dead time

How many past outputs do we include in the model? We can choose n such that

- error e(k) becomes uncorrelated with y(k) and contains no information about past disturbances
- Error e(k) is like a random variable uncorrelated with e(k-1), e(k-2),...

How do we mathematically state above requirement?

White Noise

Let us define **auto-correlation** in a random process {e(k): k= 1, 2, ...} as

$$R_v(\tau) = \text{cov}[e(k), e(k-\tau)]$$

$$= \lim_{N \to \infty} \frac{1}{(N-\tau)} \sum_{k=\tau}^{N} e(k)e(k-\tau)$$

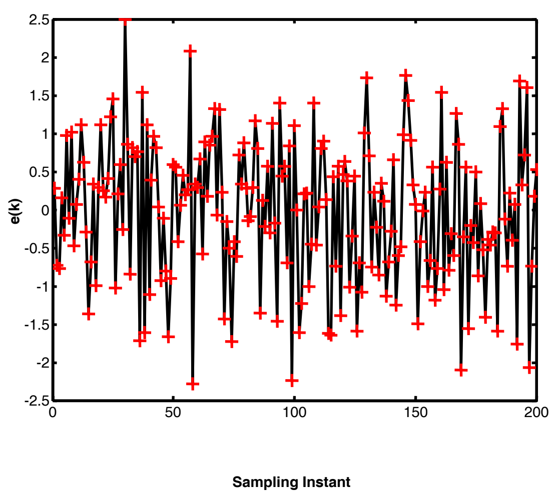
Equation error sequence e(k) in ARX model should be independent and equally distributed random variable sequence, i.e.

$$r_{ee}(\tau) = \begin{cases} \sigma^2 & for \quad \tau = 0 \\ 0 & for \quad \tau = \pm 1, \pm 2, ... \end{cases}$$

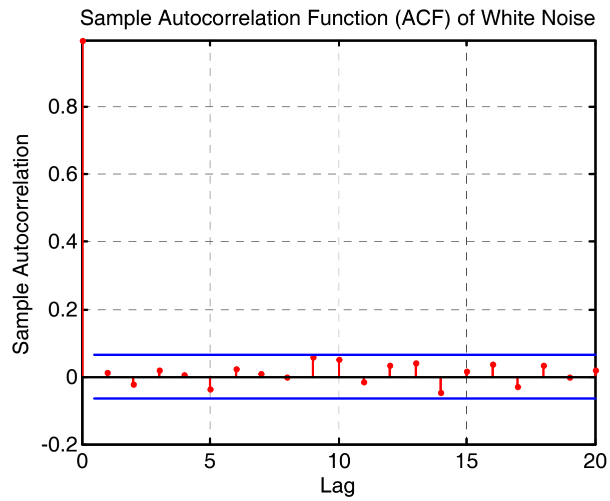
Such sequence is called **discrete time white noise**

Example: White Noise

Mean = 0 Variance = 1



White Noise: Autocorrelation



ARX Model Development

Consider 2'nd order ARX model with d = 1

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-2) + b_2 u(k-3) + e(k)$$

Advantages:

- ✓ Sequences {y(k)} and (u(k)} are known
- ✓ Linear in parameter model – optimum can be computed analytically

We can recursively estimate $\hat{y}(k)$ as

$$\hat{y}(3) = -a_1 y(2) - a_2 y(1) + b_1 u(1) + b_2 u(0)$$

$$\hat{y}(4) = -a_1 y(3) - a_2 y(2) + b_1 u(2) + b_2 u(1)$$

....

$$\hat{y}(N) = -a_1 y(N-1) - a_2 y(N-2) + b_1 u(N-2) + b_2 u(N-3)$$

$$e(k) = y(k) - \hat{y}(k) \quad for \;\; k = 3,4,......N$$

ARX : Parameter Identification

Arranging in matrix form

$$\begin{bmatrix} y(n) \\ y(n+1) \\ \dots \\ y(N) \end{bmatrix} = \begin{bmatrix} -y(1) & -y(0) & u(1) & u(0) \\ -y(2) & -y(1) & u(2) & u(1) \\ .. & .. & .. & .. \\ -y(N-1) & -y(N-2) & u(N-1) & u(N-2) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix} + \begin{bmatrix} e(2) \\ e(3) \\ \dots \\ e(N) \end{bmatrix}$$

Resulting model is linear in parameters

$$\mathbf{Y} = \mathbf{A}\boldsymbol{\theta} + \mathbf{e}$$

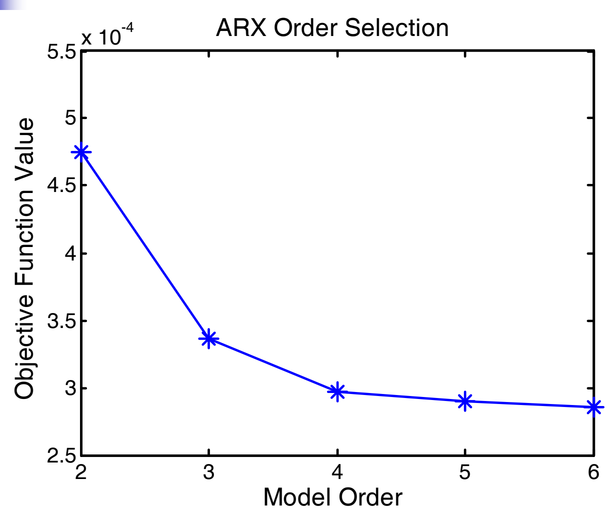
Least square parameter estimation

$$\hat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \mathbf{e}^T\mathbf{e} = \min_{\boldsymbol{\theta}} [\mathbf{Y} - \mathbf{A}\boldsymbol{\theta}]^T [\mathbf{Y} - \mathbf{A}\boldsymbol{\theta}]$$

$$\hat{\boldsymbol{\theta}} = [\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}\mathbf{Y}$$

Choose model order n such that sequence {e(k)} becomes white noise

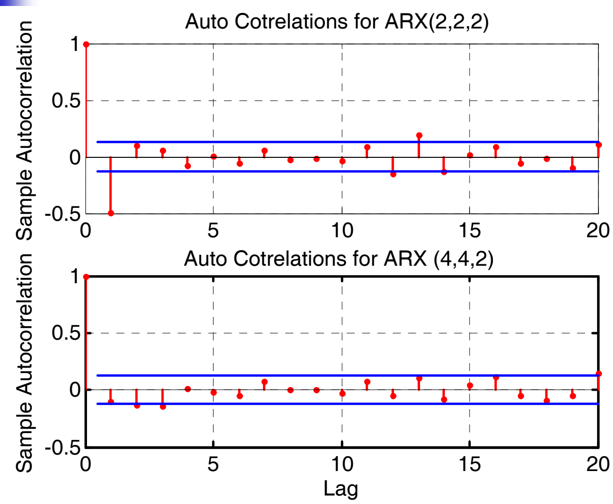
ARX: Order Selection



Time Delay
(d) = 1

Automation Lab
IIT Bombay

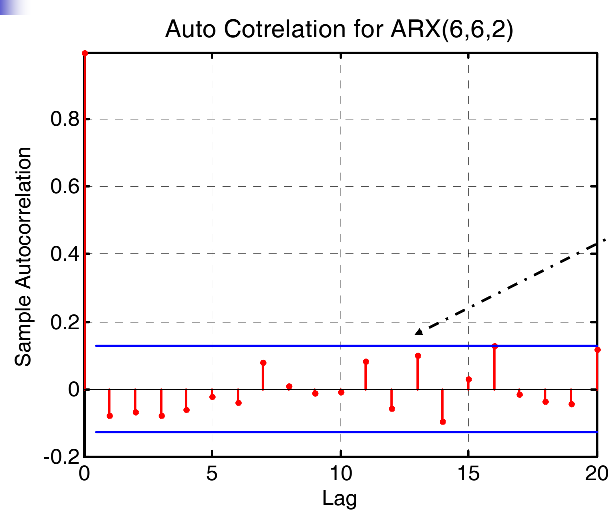
ARX: Order Selection



Model Residuals are not white

Automation Lab
IIT Bombay

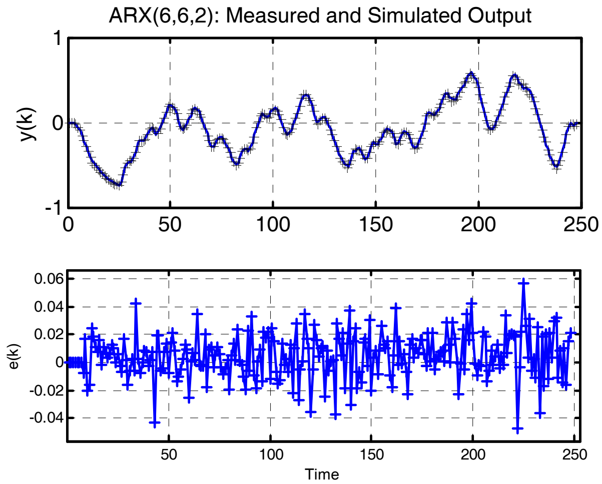
ARX: Order Selection



Model Residuals white

Automation Lab
IIT Bombay

ARX: Identification Results



Automation Lab
IIT Bombay

6'th Order ARX Model

Identified ARX Model Parameters

$A(q)y(k) = B(q)u(k) + e(t)$

$A(q) = 1 - 0.8135\, q^{-1} - 0.1949\, q^{-2} - 0.07831\, q^{-3} + 0.1107\, q^{-4}$
$+ 0.03542\, q^{-5} + 0.01755\, q^{-6}$

$B(q) = 0.00104\, q^{-2} + 0.013\, q^{-3} + 0.01176\, q^{-4} + 0.004681\, q^{-5}$
$+ 0.002472\, q^{-6} + 0.002197\, q^{-7}$

Error statistics

$$\text{Estimated Mean} : E\{e(k)\} = 4.8813 \times 10^{-3}$$

$$\text{Estimated Variance} : \hat{\lambda}^2 = 2.5496 \times 10^{-4}$$

{e(k)} is practically a zero mean white noise sequence

ARX: Estimated Parameter Variances

	Value	$\hat{\sigma}$		Value	$\hat{\sigma}$
a_1	-0.8135	0.0674	b_1	0.001	0.0009
a_2	-0.1949	0.0868	b_2	0.013	0.0011
a_3	-0.0783	0.0863	b_3	0.0118	0.0014
a_4	0.1107	0.0863	b_4	0.0047	0.0015
a_5	0.0354	0.0871	b_5	0.0025	0.0015
a_6	0.0175	0.0484	b_6	0.0022	0.0013

ARX Model

Auto Regressive with Exogenous input (ARX)

$$y(k) = b_1 u(k-1) + \ldots + b_m u(k-m)$$
$$- a_1 y(k-1) - \ldots - a_n y(k-n) + e(k)$$

Using shift operator (q), ARX model can be expressed as

$$y(k) = \frac{B(q^{-1})}{A(q^{-1})} q^{-d} u(k) + \frac{1}{A(q^{-1})} e(k)$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots. + a_n q^{-n}$$
$$B(q^{-1}) = b_1 q^{-1} + \ldots. + b_m q^{-m}$$

Noise Model

where e(k) is white noise sequence

Disadvantage

Large model order required to get white residuals

Noise Models

$$v(k) = \frac{1}{A(q^{-1})} e(k)$$

$e(k)$: Zero mean white noise process with variance λ^2

Auto Regressive (AR) Model

$$v(k) = -a_1 v(k-1) - ... - a_n v(k-n) + e(k)$$

Alternatively, if poles of A(q) are inside unit circle, then, by long division

$$\frac{1}{A(q^{-1})} = 1 + h_1 q^{-1} + h_2 q^{-2} + ... = H(q^{-1})$$

$$v(k) = H(q)e(k) = \sum_{i=0}^{\infty} h_i e(k-1)$$

Moving Average (MA) Process

$$\text{v(k)} = e(k) + h_1 e(k-1) + ...h_n e(k-n)$$

ARMA Model

AR and MA models can be combined to formulate a more general ARMA model

$$v(k) = -a_1 v(k-1) - ... - a_n v(k-m) + e(k) + c_1 e(k-1) + ... + c_m e(k-m)$$

$$\text{or} \quad v(k) = \frac{C(q^{-1})}{A(q^{-1})} e(k)$$

$e(k)$: Zero mean white noise process with variance λ^2

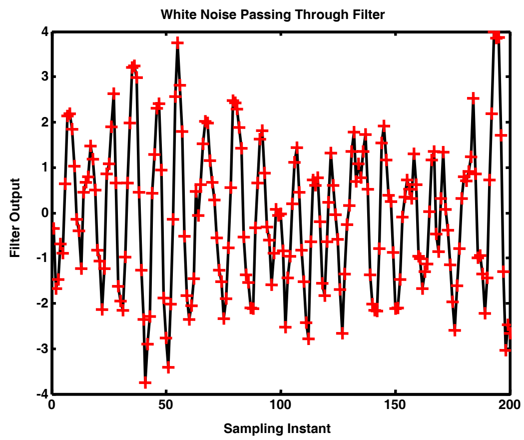
If poles of A(q) are inside unit circle, then, by long division

$$\frac{C(q^{-1})}{A(q^{-1})} = 1 + h_1 q^{-1} + h_2 q^{-2} + ... = H(q^{-1})$$

Advantage: Parsimonious in parameters
(significantly less number of model parameters required than AR or MA models for capturing noise characteristics)

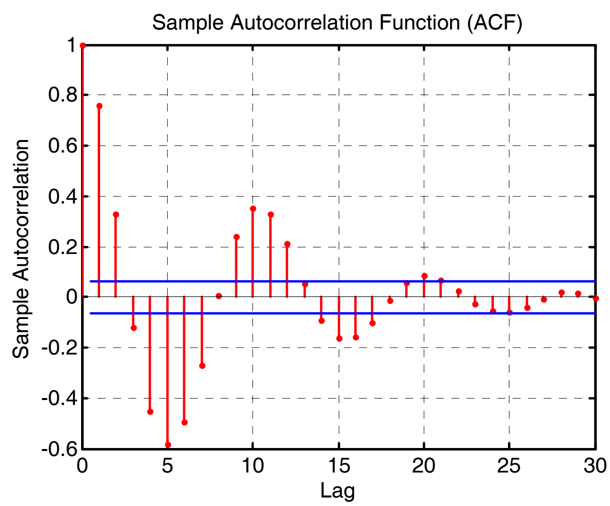
Example: Colored Noise

$$v(k) = \frac{0.8q^{-1} - 0.4q^{-2}}{1 - 1.5q^{-1} + 0.8q^{-2}} e(k)$$



Properties
Of e(k)
Mean = 0
Variance = 1

Colored Noise: Autocorrelation



Parameterized Models

ARMAX: Auto Regressive Moving Average with exogenous input (ARMAX)

$$y(k) = b_1 u(k-d-1) + ... + b_m u(k-d-m)$$
$$- a_1 y(k-1) - ... - a_n y(k-n)$$
$$+ e(k) + c_1 e(k-1) + ... + c_r e(k-r)$$

Or $$y(k) = \frac{B(q^{-1})}{A(q^{-1})} q^{-d} u(k) + \frac{C(q^{-1})}{A(q^{-1})} e(k)$$

Box-Jenkins (BJ) model: most general representation of time series models

$$y(k) = \frac{B(q^{-1})}{A(q^{-1})} q^{-d} u(k) + \frac{C(q^{-1})}{D(q^{-1})} e(k)$$

{e(k)} is white noise sequence in both the cases

Parameter Identification Problem

Given input output data collected from plant

$$Y^N \equiv \{y(k) : y(0), y(1), y(2), ..., y(N)\}$$
$$U^N \equiv \{u(k) : u(0), u(1), u(2), ..., u(N)\}$$

Choose a suitable model structure for the time series model and estimate the parameters of the model (coefficients of A(q), B(q), C(q) polynomials) such that some objective function of the residual sequence e(k)

$$\Psi[e(0), e(1), ..., e(N)]$$

is minimized.

The resulting residual sequence {e(k)} should be a white noise sequence

ARMAX: One Step Prediction

Consider 2'nd order ARMAX model with d = 1

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-2) + b_2 u(k-3)$$
$$+ e(k) + c_1 e(k-1) + c_2 e(k-2)$$

$$y(k) = \frac{b_1 q^{-2} + b_2 q^{-3}}{1 + a_1 q^{-1} + a_2 q^{-2}} u(k) + \frac{1 + c_1 q^{-1} + c_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}} e(k)$$

Difficulties:

✓Sequences {y(k)} and {u(k)} are known but {e(k)} is unknown

✓Non-Linear in parameter model – optimum can't be computed analytically

Solution Strategy

Problem solved numerically using nonlinear optimization procedures

Inevitability of Noise Model

Crucial Property of Noise Model :

Noise model and its inverse are stable and all its poles and zeros are inside unit circle

$$v(k) = H(q)e(k) = \sum_{i=0}^{\infty} h_i e(k-i)$$

$$\text{H(q) is stable i.e. } \sum_{i=0}^{\infty} |h_i| < \infty$$

$$e(k) = H^{-1}(q)v(k) = \sum_{i=0}^{\infty} \tilde{h}_i e(k-i)$$

$$\text{H}^{-1}\text{(q) is stable i.e. } \sum_{i=0}^{\infty} |\tilde{h}_i| < \infty$$

Key problem in identification is to find such H(q) and a white noise sequence {e(k)}

Note : H(q) is always 'monic' polynomial i.e. $h_0 = 1$

Example: A Moving Average Process

Consider a first order MA process

$$v(k) = e(k) + ce(k-1)$$

where $\{e(k)\}$ is a white noise sequence

i.e. $H(q) = 1 + cq^{-1} = \dfrac{q+c}{q}$ has a pole at $q = 0$ and zero at $q = -c$

Then, $H^{-1}(q) = \dfrac{1}{1+cq^{-1}} = \sum_{i=0}^{\infty}(-c)^i q^{-i}$ if $|c| < 1$

and e(k) can be recovered from measurements of v(k)

$$e(k) = \sum_{i=0}^{\infty}(-c)^i v(k-i)$$

Inversion of Noise Model plays a crucial role in the procedure for model identification

One Step Prediction

Suppose we have observed v(t) upto $t \le (k-1)$ and we want to predict v(k) based on measurements upto time (k - 1)

$$v(k) = \sum_{i=0}^{\infty} h_i e(k-i) = e(k) + \sum_{i=1}^{\infty} h_i e(k-i) = e(k)$$

$$\hat{v}(k \mid k-1) = \sum_{i=1}^{\infty} h_i e(k-i) \quad \text{as e(k) has zero mean}$$

$\hat{v}(k \mid k-1)$: Conditional expectation of v(k) based on information upto (k - 1)

$$\hat{v}(k \mid k-1) = v(k) - e(k) = [H(q) - 1]e(k)$$

$$= \frac{H(q)-1}{H(q)} v(k)$$

$$\hat{v}(k \mid k-1) = [H^{-1}(q) - 1]v(k) = \sum_{i=1}^{\infty} -\tilde{h}_i v(k-i)$$

One Step Output Prediction

Suppose we have observed y(t) and u(t) upto $t \le (k-1)$ and

We have $y(k) = G(q)u(k) + v(k)$

and we want to predict y(k) based on information upto time (k - 1)

$$\hat{y}(k \mid k-1) = G(q)u(k) + \hat{v}(k \mid k-1)$$
$$= G(q)u(k) + \left[1 - H^{-1}(q)\right]v(k)$$

However, $v(k) = y(k) - G(q)u(k)$

$$\hat{y}(k \mid k-1) = G(q)u(k) + \left[1 - H^{-1}(q)\right]\left[y(k) - G(q)u(k)\right]$$

Rearranging we have

$$\hat{y}(k \mid k-1) = H^{-1}(q)G(q)u(k) + \left[1 - H^{-1}(q)\right]y(k)$$

or

$$H(q)\hat{y}(k \mid k-1) = G(q)u(k) + \left[H(q) - 1\right]y(k)$$

ARX: One Step Predictor

Consider 2'nd order ARX model with d = 1

$$y(k) = \left[\frac{b_1 q^{-2} + b_2 q^{-3}}{1 + a_1 q^{-1} + a_2 q^{-2}}\right]u(k) + \left[\frac{1}{1 + a_1 q^{-1} + a_2 q^{-2}}\right]e(k)$$

One step ahead predictor for this model is

$$\hat{y}(k \mid k-1) = \left[\frac{b_1 q^{-2} + b_2 q^{-3}}{1}\right]u(k) + \left[\frac{-a_1 q^{-1} - a_2 q^{-2}}{1}\right]y(k)$$

which is equivalent to difference equation

$$\hat{y}(k \mid k-1) = -a_1 y(k-1) - a_2 y(k-2)$$
$$+ b_1 u(k-2) + b_2 u(k-3)$$

Advantage : All terms in RHS are known

Residual at k'th instant can be estimated as

$$e(k) = y(k) - \hat{y}(k \mid k-1)$$

ARMAX: One Step Predictor

Consider 2'nd order ARMAX model with d = 1

$$y(k) = \left[\frac{b_1 q^{-2} + b_2 q^{-3}}{1 + a_1 q^{-1} + a_2 q^{-2}}\right] u(k) + \left[\frac{1 + c_1 q^{-1} + c_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}}\right] e(k)$$

One step ahead predictor for this model is

$$\hat{y}(k \mid k-1) = \left[\frac{b_1 q^{-2} + b_2 q^{-3}}{1 + c_1 q^{-1} + c_2 q^{-2}}\right] u(k) + \left[\frac{(c_1 - a_1) q^{-1} + (c_2 - a_2) q^{-2}}{1 + c_1 q^{-1} + c_2 q^{-2}}\right] y(k)$$

which is equivalent to difference equation

$$\begin{aligned}\hat{y}(k \mid k-1) = & -c_1 \hat{y}(k-1 \mid k-2) - c_2 \hat{y}(k-2 \mid k-3) \\ & + b_1 u(k-2) + b_2 u(k-3) \\ & + (c_1 - a_1) y(k-1) + (c_2 - a_2) y(k-2)\end{aligned}$$

Residual at k'th instant can be estimated as

$$\varepsilon(k) = y(k) - \hat{y}(k \mid k-1)$$

ARMAX: One Step Predictor

Alternatively, using residuals at previous instances

$$\varepsilon(k-1) = [y(k-1) - \hat{y}(k-1 \mid k-2)]$$

$$\varepsilon(k-2) = [y(k-2) - \hat{y}(k-2 \mid k-3)]$$

we can rearrange one step predictor as

$$\begin{aligned}\hat{y}(k \mid k-1) = & -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-2) + b_2 u(k-3) \\ & + c_1 \varepsilon(k-1) + c_2 \varepsilon(k-2)\end{aligned}$$

$$\varepsilon(k) = y(k) - \hat{y}(k \mid k-1)$$

We can start prediction with initial guesses

$$\varepsilon(0) = \varepsilon(1) = 0$$

and, given model parameters $(a_1, a_2, b_1, b_2, c_1, c_2)$,

we can generate sequence $\{\varepsilon(k)\}$

using sequences {y(k)} and {u(k)}.

Automation Lab
IIT Bombay

2'nd Order ARMAX Model

Optimization formulation

Estimate $(a_1, a_2, b_1, b_2, , c_1, c_2)$ such that objective function

$$\Psi = \sum_{k=3}^{N} [e(k)]^2 = \sum_{k=3}^{N} [y(k) - \hat{y}(k \mid k-1)]^2$$

is minimized with respect to $(a_1, a_2, b_1, b_2, c_1, c_2)$

Identified Model Parameters

A(q) = 1 - 1.651 q^-1 + 0.68 q^-2

B(q) = 0.001748 q^-2 + 0.01154 q^-3

C(q) = 1 - 0.8367 q^-1 + 0.2501 q^-2

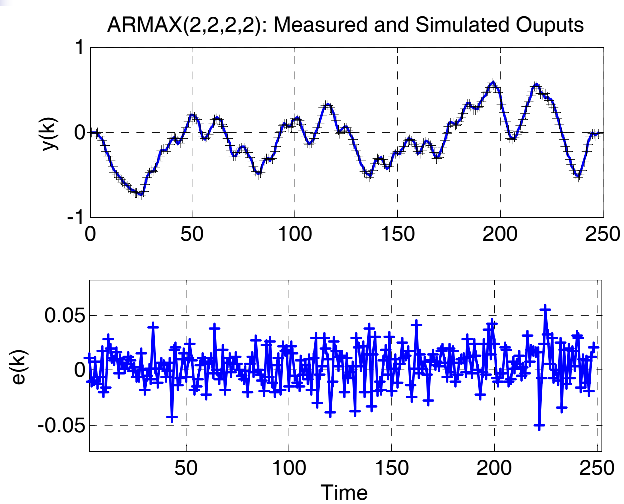
Residual {e(k)} Statistics

Estimated Mean : $E\{e(k)\} = 4.3601e-003$

Estimated Variance : $\hat{\lambda}^2 = 2.6813e-004$

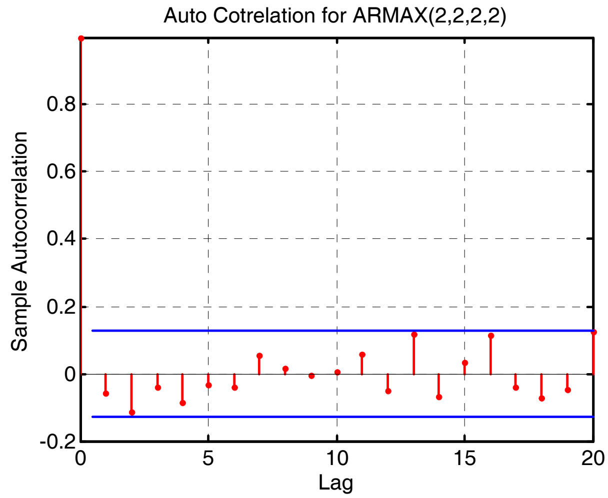
Automation Lab
IIT Bombay

2'nd Order ARMAX Model



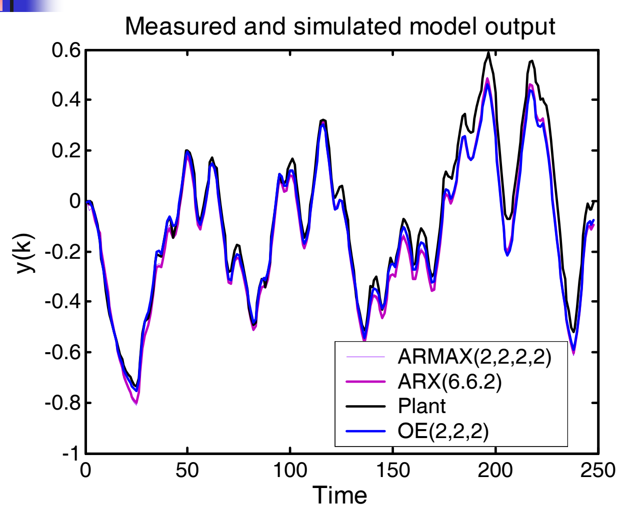
Automation Lab
IIT Bombay

ARMAX: Autocorrelation



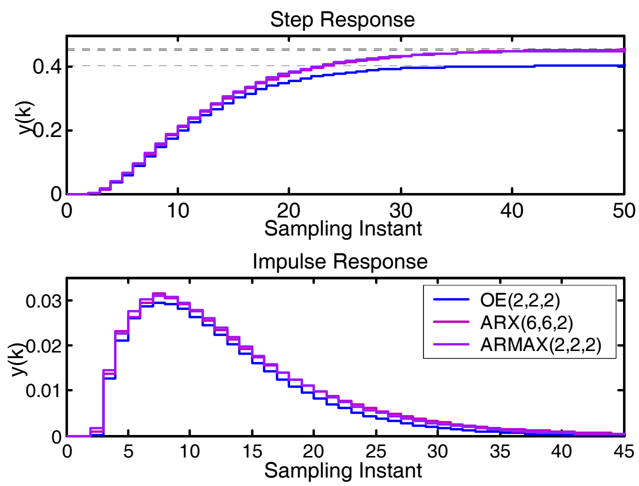
Automation Lab
IIT Bombay

Comparison of Model Predictions

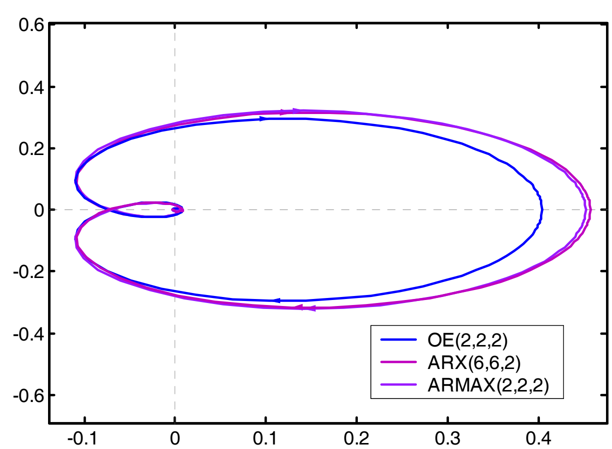


Best Fit (%)
ARMAX : 76.45
ARX : 76.37
OE : 77.38

Comparison of Models



Comparison of Models: Nyquist Plots



Automation Lab
IIT Bombay

Prediction Error Method

Given data set

$$Z_N = \{(y(K), u(k)) : k = 1,2....N\},$$

Model

$$y(k) = G(q,\theta)u(k) + H(q,\theta)e(k)$$

Optimal 1-step predictor

$$\hat{y}(k \mid k-1) = H^{-1}(q,\theta)G(q,\theta)u(k) + [1 - H(q,\theta)]y(k)$$

One step prediction error is defined as

$$\varepsilon(k,\theta) = y(k) - \hat{y}(k \mid k-1,\theta)$$

Parameter Estmimation by Prediction Error Method

Find θ that minimizes objective function

$$V(\theta, Z_N) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon(k,\theta)^2$$

Automation Lab
IIT Bombay

PEM: Parameter Estimation

$$\hat{\theta}_N = \underset{\theta}{\text{Min}}\, V(\theta, Z_N)$$

Typically, the resulting parameter estimation problem is solved numerically using
(a) Nonlinear optimization
(b) Gauss Newton Method
If it is desired to emphasize certain frequency of interest, then, we can minimize

$$V(\theta, Z_N) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon_F(k,\theta)^2$$

where $\varepsilon_F(k) = F(q^{-1})\varepsilon(k)$

$F(q^{-1})$ represents a filter

Alternate Choice of Objective Function

$$V(\theta, Z_N) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon(k,\theta)^2$$

Model Order Selection

Model order determined by minimizing Akaike Information Criterion (AIC)

$$AIC(\hat{\theta}_N) = N \ln\left[\frac{1}{N}\sum_{k=1}^{N} \varepsilon(k,\hat{\theta}_N)^2\right] + 2n$$

n : Number of model parameters

AIC = {Prediction Term} + { Model Order term}

✓Prediction Term: estimate of how ell the model fits data

✓Model Order Term: measure of model complexity required to obtain the fit

AIC strikes a balance between low residual variance and excessive number of model parameters, with smaller values indicating more desirable models

Basic Idea:

Penalize model complexity (measured by n) and obtain a model, which is reasonable w.r.t. variance errors and model complexity

ARMAX: State Realization

$$x(k+1) = \Phi x(k) + \Gamma u(k) + L_\infty e(k)$$

$$y(k) = Cx(k) + e(k)$$

$$\Phi = \begin{bmatrix} -a_1 & 1 & 0 & .... & 0 \\ -a_2 & 0 & 1 & .... & 0 \\ ... & .... & & .... & .... \\ -a_{n-1} & 0 & 0 & .... & 1 \\ -a_{n-2} & 0 & 0 & .... & 0 \end{bmatrix} ; \Gamma = \begin{bmatrix} b_1 \\ b_2 \\ ... \\ ... \\ b_n \end{bmatrix} ; L_\infty = \begin{bmatrix} c_1 - a_1 \\ c_2 - a_2 \\ ... \\ ... \\ c_n - a_n \end{bmatrix}$$

$$C = [1 \quad 0 \quad .... \quad 0]$$

$$G(q) = \frac{B(q)}{A(q)} = C[qI - \Phi]^{-1}\Gamma \ ; \ H(q) = \frac{C(q)}{A(q)} = C[qI - \Phi]^{-1}L_\infty + I$$

Interpretation as a State Observer

$$x(k+1 \mid k) = \Phi x(k \mid k-1) + \Gamma u(k) + L_\infty[y(k) - Cx(k \mid k-1)]$$

Connection with Steady State Kalman Estimator

Steady state form of Kalman prediction estimator (for large time) is given as

$$\hat{x}(k+1|k) = \Phi\hat{x}(k|k-1) + \Gamma u(k) + L_\infty e(k)$$
$$e(k) = y(k) - C\hat{x}(k|k-1)$$

$$L_\infty = \Phi P_\infty C^T (R_2 + C P_\infty C^T)^{-1}$$
$$P_\infty = \Phi P_\infty \Phi^T + R_1 - L_\infty C P_\infty \Phi^T$$

Thus, development of time series model can be viewed as identification of steady state Kalman estimator without requiring explicit knowledge of noise covariance matrices (R_1,R_2)

Steady state Kalman gain L_∞ is parameterized through H(q) and estimated directly from data.

MIMO System Identification

- ARX Model:

Method for ARX parameter identification can be extended to deal directly with multivariate data

- OE / ARMAX / BJ Models:

Typically, an n x m MIMO system is modeled as n MISO (Multi Input Single Output) systems

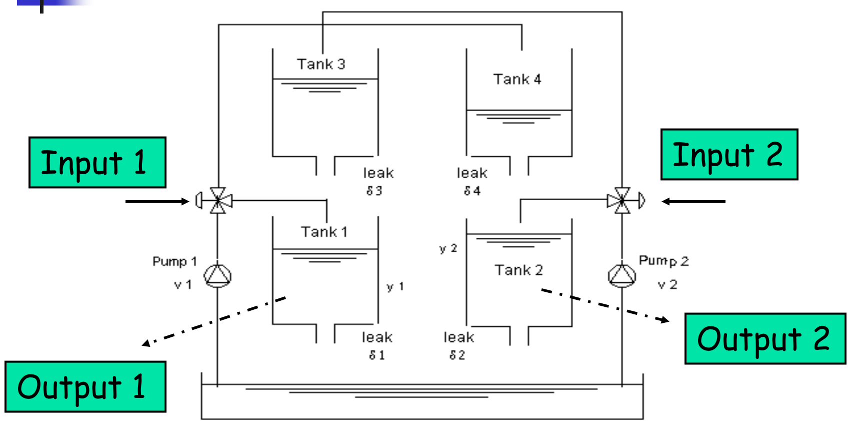
$$y_i(k) = G_{i1}(q)u_1(k) + ... + G_{im}(q)u_m(k) + H_i(q)e_i(k)$$
$$i = 1,2,.....n$$

MISO models are combined to form a one MIMO model

- Input excitation: Inputs can be perturbed sequentially or simultaneously

Automation Lab
IIT Bombay

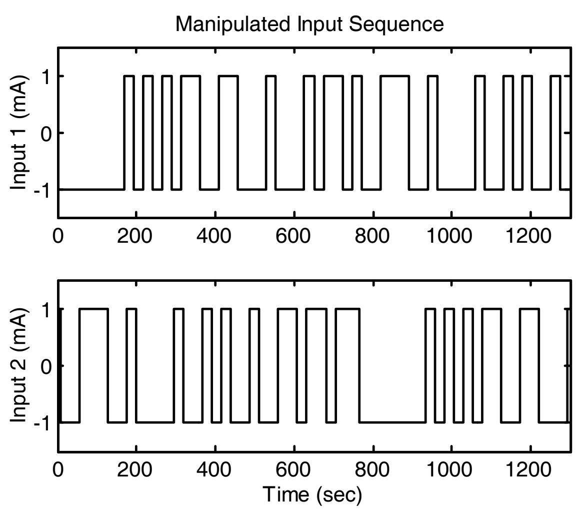
Identification Experiments on 4 Tank Setup



Schematic of Quadruple Tank Process

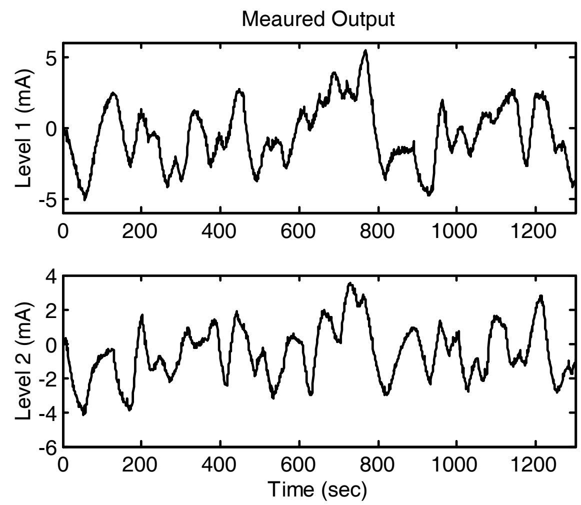
Automation Lab
IIT Bombay

4 Tank Setup: Input Excitations



Automation Lab
IIT Bombay

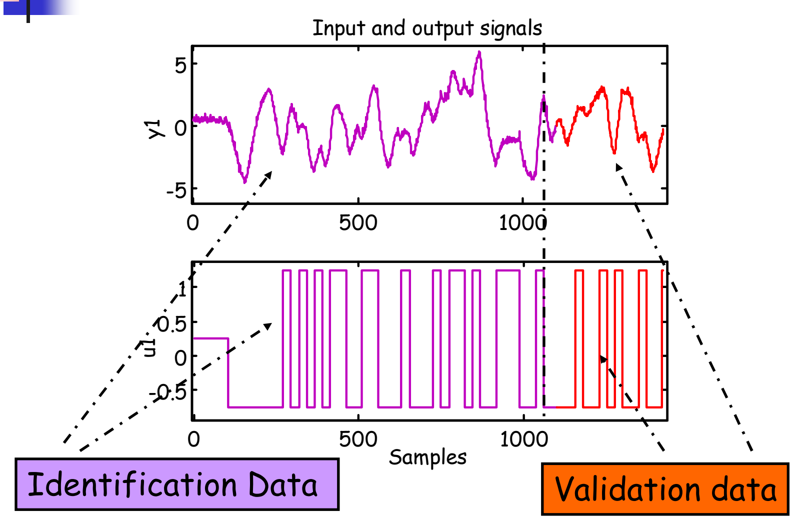
4 Tank Setup: Measured Output



1/18/2006 System Identification 103

Automation Lab
IIT Bombay

Splitting Data for Identification and Validation



1/18/2006 System Identification 104

MISO OE Model

MISO 2'nd Order Model

$$y_1(t) = [B_1(q)/A_1(q)]u_1(t) + [B_2(q)/A_2(q)]u_2(t) + e_1(t)$$

$$B1(q) = 0.1393\ q^{-1} + 0.04704\ q^{-2}$$

$$B2(q) = 0.002375\ q^{-1} + 0.01105\ q^{-2}$$

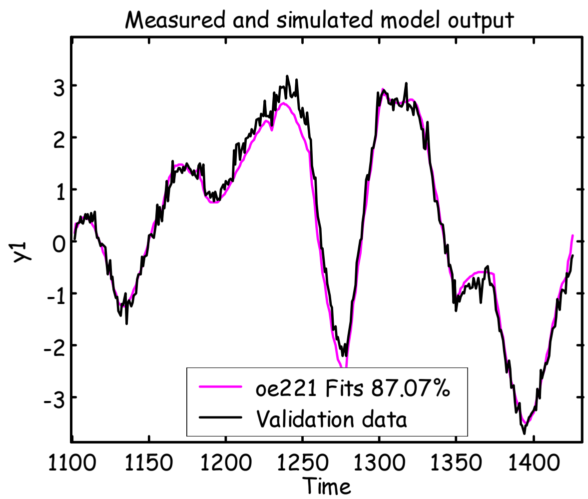
$$A1(q) = 1 - 0.2454\ q^{-1} - 0.6571\ q^{-2}$$

$$A2(q) = 1 - 1.887\ q^{-1} + 0.8903\ q^{-2}$$

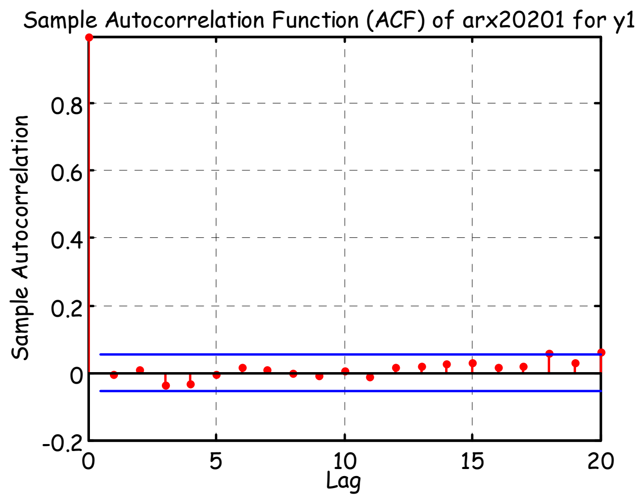
Estimated using Prediction Error Method

Loss function 0.114719 Sampling interval: 3

OE Model: Validation



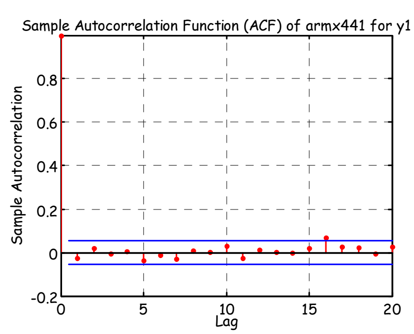
Residual Autocorrelation: ARX

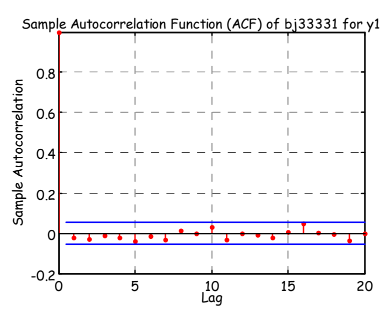


Residuals Autocorrelations

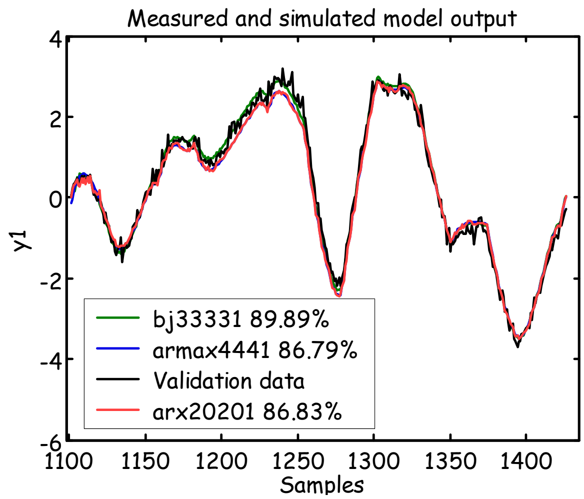
ARMAX
4'th Order

B-J
3'rd Order





Model Validation



ARMAX Model

ARMAX (4'th Order)

$A(q)y_1(t) = B_1(q)u_1(t) + B_2(q)u_2(t) + C(q)e_1(t)$

$A(q) = 1 - 0.6236\ q^{-1} - 0.8596\ q^{-2} - 0.0758\ q^{-3} + 0.568\ q^{-4}$

$B_1(q) = 0.08324\ q^{-1} + 0.02757\ q^{-2} + 0.02681\ q^{-3} - 0.1214\ q^{-4}$

$B_2(q) = 0.004045\ q^{-1} + 0.03261\ q^{-2} - 0.01841\ q^{-3} + 0.0201\ q^{-4}$

$C(q) = 1 - 0.4695\ q^{-1} - 0.8017\ q^{-2} - 0.1065\ q^{-3} + 0.4855\ q^{-4}$

Loss function 0.0243707

Box-Jenkins Model

$y(t) = [B(q)/F(q)]u(t) + [C(q)/D(q)]e(t)$

$B1(q) = 0.08196\, q^{-1} + 0.1035\, q^{-2} + 0.1323\, q^{-3}$

$B2(q) = 0.01197\, q^{-1} + 0.001306\, q^{-2} + 0.01304\, q^{-3}$

$C(q) = 1 - 1.976\, q^{-1} + 1.126\, q^{-2} - 0.1453\, q^{-3}$

$D(q) = 1 - 2.096\, q^{-1} + 1.209\, q^{-2} - 0.1128\, q^{-3}$

$F1(q) = 1 + 0.3058\, q^{-1} - 0.5066\, q^{-2} - 0.6204\, q^{-3}$

$F2(q) = 1 - 0.897\, q^{-1} - 0.9828\, q^{-2} + 0.8861\, q^{-3}$

Loss function 0.0239039

ARMAX:State Realization

$x(k+1) = \Phi x(k) + \Gamma x(k) + L_{\infty} e(k)$

$Y(k) = C\, x(k) + e(k)$

$$\Phi = \begin{bmatrix} 0.6236 & 1 & 0 & 0 \\ 0.8596 & 0 & 1 & 0 \\ 0.0758 & 0 & 0 & 1 \\ -0.5680 & 0 & 0 & 0 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0.0832 & 0.0040 \\ 0.0276 & 0.0326 \\ 0.0268 & -0.0184 \\ -0.1214 & 0.0201 \end{bmatrix} \qquad L_{\infty} = \begin{bmatrix} 0.1541 \\ 0.0579 \\ -0.0307 \\ -0.0826 \end{bmatrix};$$

$C = [\ 1 \quad 0 \quad 0 \quad 0\ ]$

Recursive Parameter Estimation

Consider 2'nd order ARX model with d = 1

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-2) + b_2 u(k-3) + e(k)$$

$$y(k) = \varphi^T(k)\theta + e(k)$$

$$\varphi^T(k) = [-y(k-1) \quad -y(k-2) \quad u(k-2) \quad u(k-3)]$$

Arranging in matrix form

$$\begin{bmatrix} y(2) \\ y(n+1) \\ \dots \\ y(N) \end{bmatrix} = \begin{bmatrix} \varphi^T(2) \\ \varphi^T(3) \\ \dots\dots \\ \varphi^T(N) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix} + \begin{bmatrix} e(2) \\ e(3) \\ \dots \\ e(N) \end{bmatrix}$$

N introduced as formal parameter to indicate that data up to instant N has been used

$$\mathbf{Y}(N) = \mathbf{A}(N)\boldsymbol{\theta} + \mathbf{e}(N)$$

Least square parameter estimation

$$\hat{\boldsymbol{\theta}}(N) = \left[\mathbf{A}(N)^T \mathbf{A}(N)\right]^{-1} \mathbf{A}(N)\mathbf{Y}$$

RLS: Problem Formulation

Nhen an additional measurement is obtained on-line ,matrix A becomes

$$A(N+1) = \begin{bmatrix} A(N) \\ \varphi^T(N+1) \end{bmatrix}, \quad Y(N+1) = \begin{bmatrix} Y(N) \\ y(N+1) \end{bmatrix}$$

New estimate $\hat{\theta}(N+1)$ *can be written as*

$$\hat{\theta}(N+1) = \left[A^T(N+1)A(N+1)\right]^{-1} A^T(N+1)Y(N+1)$$

Thus, A(N) keeps growing in size as new data arrives.
Instead of inverting $\left|A^T(N+1)A(N+1)\right|$ at every instat,
Can we rearrange calculations at (N+1) so that solution at
Instant N can be used to compute solution at instant (N+1)?

$$\hat{\theta}(N+1) = \left[A^T(N)A(N) + \varphi(N+1)\varphi^T(N+1)\right]^{-1} \times \left[A^T(N)Y(N) + \varphi(N+1)y(N+1)\right]$$

RLS: Solution

Using matrix inversion lemma

$$[A+BCD]^{-1} = A^{-1} - A^{-1}B\left[C^{-1} + DA^{-1}B\right]^{-1}DA^{-1}$$

and some rearrangement

Solution to above problem is given by following recursive set of equatiosns

$$\hat{\theta}(N+1) = \hat{\theta}(N) + L(N)\left[y(N+1) - \hat{y}(N+1 \mid N)\right]$$

where

$$\hat{y}(N+1 \mid N) = \varphi^T(N+1)\hat{\theta}(N)$$

Estimator gain matrix L(N) is computed by solving following Riccati equations

$$L(N) = P(N)\varphi(N+1)\left[1 + \varphi^T(N+1)P(N)\varphi(N+1)\right]^{-1}$$

$$P(N+1) = \left[I - L(N)\varphi^T(N+1)\right]P(N)$$

RLS: Initialization

In order to obtain an initial conditions to start RLS, it is necessary to choose $N = N_0$ such that $A^T(N_0)A(N_0)$ is nonsingular.

$$P(N_0) = \left[A^T(N_0)A(N_0)\right]^{-1}$$

$$\hat{\theta}(N_0) = P(N_0)A^T(N_0)Y(N_0)$$

Recursive canculations can be used for $N \geq N_0$

Alternatively, recursive equations are begun with the initial covariance matrix

$$P(0) = \alpha I \quad \text{and} \quad \hat{\theta}(0) = \overline{0}$$

where α is chosen large ($\approx 10^4$) indicating that we have not trust in the initial parameter estimate i.e. $\hat{\theta}(0) = \overline{0}$

This choice ensures $P(N) \rightarrow \left[A^T(N)A(N)\right]^{-1}$ as N increases

Time Varying Systems

For time varying systems, it is necessary to eliminate the influence of old data. This can be achieved using an exponential weighting in the loss function.

$$J(\theta) = \sum_{k=1}^{N} \lambda^{N-k} \left[y(k) - \varphi^T(k)\theta \right]^2$$

λ is called as forgetting factor

RLS for this system is given by,

$$\hat{\theta}(k+1) = \hat{\theta}(k) + K(k)\left[y(k+1) - \varphi^T(k+1)\hat{\theta}(k) \right]$$

$$K(k) = P(k)\varphi(k+1)\left[\lambda + \varphi^T(k+1)P(k)\varphi(k+1) \right]^{-1}$$

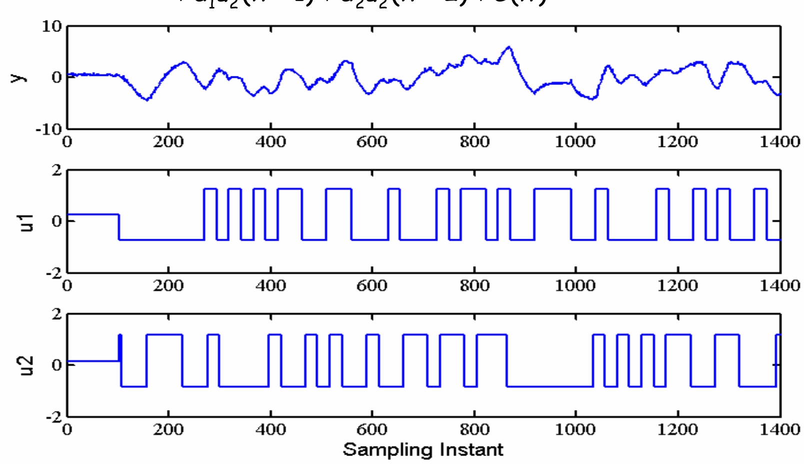
$$P(k+1) = \left[I - K(k)\varphi^T(k+1) \right] P(k)/\lambda$$

Asymptotic Data Length (ASL) = 1/(1 - λ)

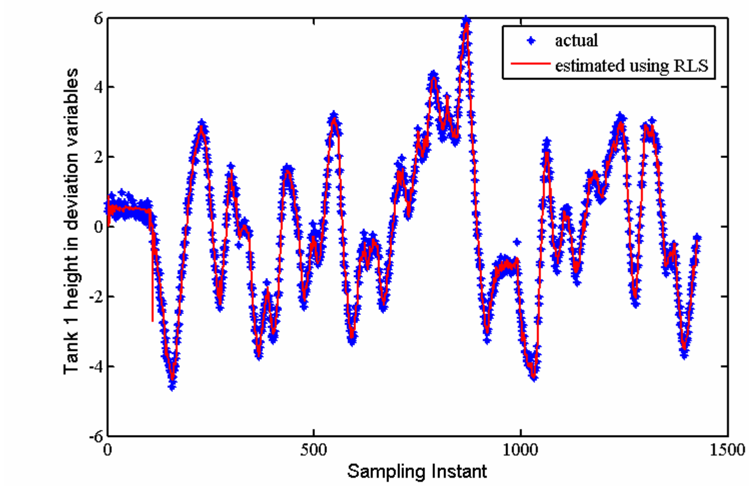
$\lambda = 0.999$: $ASL = 1000$; $\lambda = 0.95$: $ASL = 20$

RLS Application to 4 Tank Data

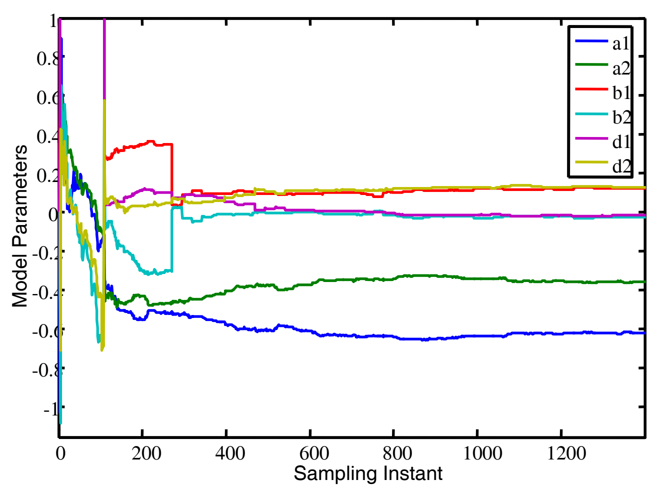
$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u_1(k-1) + b_2 u_1(k-2) + d_1 u_2(k-1) + d_2 u_2(k-2) + e(k)$$



Model Predictions



Parameter Variations



Extended Recursive Formulations (ELS)

OE Model

$$x(k) = a_1 x(k-1) + a_2 x(k-2) + b_1 u(k-2) + b_2 u(k-3)$$
$$y(k) = x(k) + v(k)$$

Recursive OE Formulation uses regressor vector

$$\varphi(k) = [\hat{x}(k-1) \quad \hat{x}(k-2) \quad u(k-1) \quad u(k-2)]$$

ARMAX Model

$$x(k) = a_1 x(k-1) + a_2 x(k-2) + b_1 u(k-2) + b_2 u(k-3)$$
$$+ e(k) + c_1 e(k-1) + c_2 e(k-2)$$

Extended Recursive Formulation uses regressor vector

$$\varphi(k) = [y(k-1) \quad y(k-2) \quad u(k-1) \quad u(k-2) \quad \varepsilon(k-1) \quad \varepsilon(k-2)]$$

where $\varepsilon(k-1)$ and $\varepsilon(k-2)$ represent model residuals at previous instants

Frequency Domain Analysis

- Time domain formulations of parameter estimation problem
  - Useful for carrying out parameter estimation
  - Does not provide any insight into internal working of optimization problem
- Frequency domain (power spectrum) analysis
  - Based on Fourier transform of auto-correlation and cross correlation function of signals
  - Powerful tool for analysis (analogous to use of Laplace transforms in linear control theory)
  - Provides insight into various aspects of optimization formulation
  - Can be used for perturbation signal design and estimation error analysis

Stationary Process

Consider noise model $v(k) = \sum_{i=0}^{\infty} h_i e(k-i)$

where $\{e(k)\}$ is a zero meanwhite noise process with variance λ^2

$$E\{v(k)\} = \sum_{i=0}^{\infty} h_i E\{e(k-i)\} = 0$$

and auto - covariance is

$$R_v(\tau) = E\{v(k), v(k-\tau)\} = \sum_{t=0}^{\infty}\sum_{j=0}^{\infty} h(t)h(j)E\{e(k-t), e(k-j-\tau)\}$$

$$= \lambda \sum_{t=0}^{\infty}\sum_{j=0}^{\infty} h(t)h(j)\delta(k-j-\tau) = \lambda \sum_{t=0}^{\infty} h(t)h(t-\tau)$$

Note : $h(r) = 0$ *if* $r < 0$. Covariance $R_v(\tau)$ is independent of k and is uniquely defined by $\{h(k)\}$ and λ^2.

Such stochastic process is called 'stationary' since it has zero mean and auto - covariance is independent of time (k).

Quasi-Stationary Process

$$y(k) = G(q)u(k) + H(q)e(k)$$

$$= \text{(deterministic)} + \text{(stochastic)}$$

Since $E\{e(k)\} = 0$, we have

$E\{y(k)\} = G(q)u(k)$ and $\{y(k)\}$ is not a stationary process

Quasi-stationary process

A signal $\{s(k)\}$ is said to be quasi stationary if it is subject to

(i) $E\{s(k)\} = m_s(k) \qquad |m_s(k)| \le C \qquad \forall k$

(ii) $E\{s(k)s(t)\} = R_s(k,t) \qquad |R_s(k,t)| \le C$

$$\lim_{N \to \infty} \frac{1}{N}\sum_{k=1}^{N} R_s(k, k-\tau) = R_s(\tau)$$

$R_s(\tau)$: auto - correlation function of signal $\{s(k)\}$

Signal Spectrum and Cross Spectrum

Defining $\bar{E}\{f(k)\} = \lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} f(k)$ we have

$$R_s(\tau) = \bar{E}\{s(k)s(k-\tau)\}$$

$$R_{sw}(\tau) = \bar{E}\{s(k)w(k-\tau)\}$$

We define power spectrum of signal $\{s(k)\}$

$$\Phi_s(\omega) = \sum_{\tau=-\infty}^{\infty} R_s(\tau) e^{-j\omega\tau}$$

and cross sprctrum between $\{s(k)\}$ and $\{w(k)\}$ as

$$\Phi_{sw}(\omega) = \sum_{\tau=-\infty}^{\infty} R_{sw}(\tau) e^{-j\omega\tau}$$

provided the infinite sum exists.

$\Phi_s(\omega)$ is always a real function of (ω)

$\Phi_{sw}(\omega)$ is, in general, complex valued function of ω

Power Spectrum (Contd.)

Note: Spectrum of signal s(t) represents Fourier transform of auto-covariance function

Inverse Transform:

By definition of inverse Fourier transform (Parseval's Theorem)

$$\bar{E}\left[s^2(k)\right] = R_s(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Phi_s(\omega) d\omega$$

Fundamental Modeling Problem

Given a disturbance with spectrum $\Phi_v(\omega)$
Can we find a transfer function H(q) such that the random process v(k) = H(q)e(k) has same spectrum with {e(k)} being a white noise?

Spectral Factorization

Main Result

Theorem : Suppose that $\Phi_v(\omega) > 0$ is a rational function of $\cos(\omega)$ (or $e^{i\omega}$). Then there exists a monic rational transfer function of z, R(z), with no poles and zeros outside unit circle such that

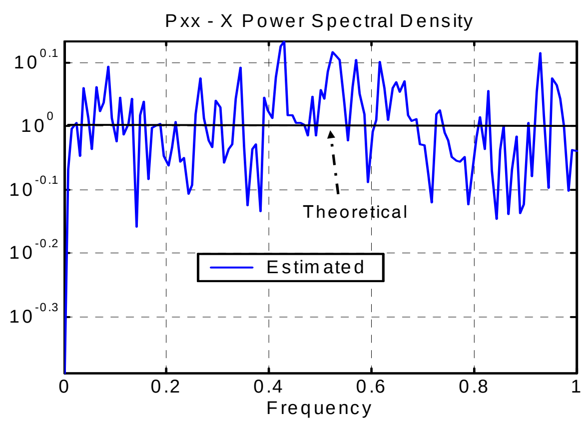
$$\Phi_v(\omega) = \lambda^2 \left|R(e^{i\omega})\right|^2$$

If the residue signal v(k) is **weakly stationary**, i.e. cov[v(k),v(s)] is function of only (k-s) **for any pair** (k,s), then spectral factorization theorem states that such a random sequence can be thought of being generated by a **stable linear transfer function driven by white noise**

Spectrum of White Noise

$$r_{ee}(\tau) = \begin{cases} \lambda^2 & for \quad \tau = 0 \\ 0 & for \quad \tau = \pm 1, \pm 2, ... \end{cases}$$

$$Case : \lambda^2 = 1$$



White Noise: Uniform power spectrum density at all frequencies

Example: Autocorrelation for AR

Numerical Estimate

$$\hat{R}_v(\tau) = \frac{1}{N-\tau}\left[\sum_{t=\tau+1}^{N} y(t)y(t-\tau)\right]$$

Theoretical (with knowledge of λ^2)

$$\text{Example: } y(t) = 0.5y(t-1) + e(t)$$

$$E[y(t)y(t-\tau)] = 0.5E[y(t-1)y(t-\tau)] + E[e(t)y(t-\tau)]$$

$$R_y(\tau) = 0.5R_y(\tau-1) + R_{ye}(\tau)$$

$$\text{But, } R_{ye}(\tau) = E[e(t)y(t-\tau)] = 0 \text{ if } \tau > 0$$

$$= \lambda^2 \text{ if } \tau = 0$$

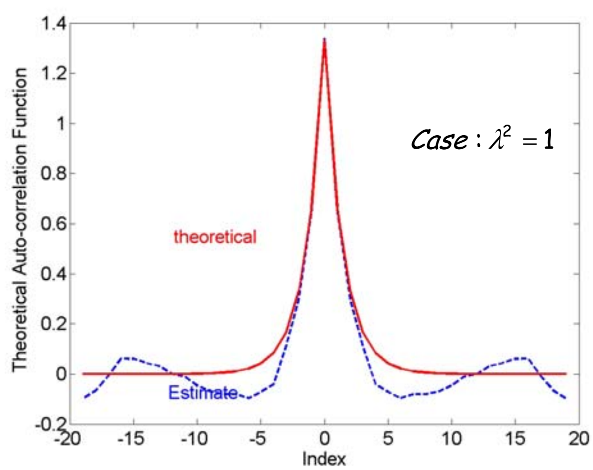
$$\text{For } \tau = 0: R_y(0) = 0.5R_y(1) + \lambda^2$$

$$\text{For } \tau = 1: R_y(1) = 0.5R_y(0)$$

$$\Rightarrow R_y(\tau) = \frac{4}{3}(0.5)^{\tau}\lambda^2$$

Example: Autocorrelation Function

$$y(t) = 0.5y(t-1) + e(t)$$



Spectrum of a Stochastic Process

Consider a stationary stochastic process

$$v(k) = H(q^{-1})e(k)$$

$\{e(k)\}$: zero mean white noise process with variance λ^2

$$\Phi_v(\omega) = \lambda^2 \left| H(e^{i\omega}) \right|^2$$

Example :

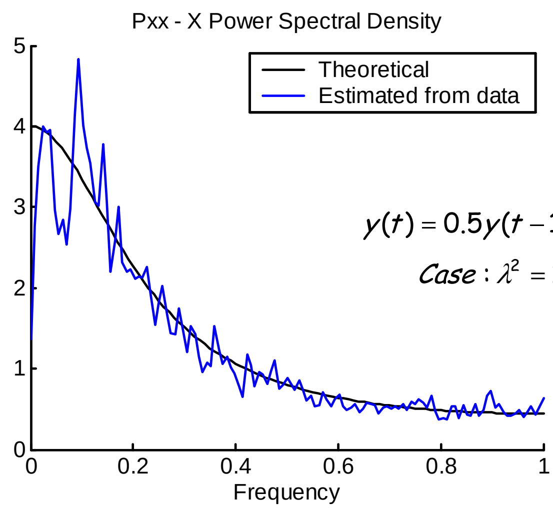
$$y(k) = \frac{1}{1-0.5q^{-1}} e(k)$$

$$\Phi_y(\omega) = \lambda^2 \left| \frac{1}{1-0.5\exp(-j\omega)} \right|^2$$

$$= \lambda^2 \frac{1}{[1+0.5\cos(\omega)]^2 + 0.25\sin^2(\omega)}$$

$\omega \in [0, \pi/T]$; π/T : Nyquist Frequency

Spectrum: Colored Noise



$$y(t) = 0.5y(t-1) + e(t)$$

$$Case : \lambda^2 = 1$$

Spectra for Linear Systems

Spectrum of mixed deterministic and stochastic signal

$$s(k) = u(k) + v(k)$$

$\{u(k)\}$: Quasi - stationary and deterministic signal with spectrum $\Phi_u(\omega)$

$\{v(k)\}$: Stationary stochastic process with spectrum $\Phi_v(\omega)$

$$\bar{E}[s(k)s(k-\tau)] = \bar{E}[u(k)u(k-\tau)] + \bar{E}[v(k)v(k-\tau)]$$
$$= R_u(\tau) + R_v(\tau)$$
$$as \quad \bar{E}[u(k)v(k-\tau)] = 0$$

$$y(k) = G(q)u(k) + H(q)e(k)$$

$\{u(k)\}$: Quasi - stationary and deterministic signal

$\{e(k)\}$: Zero mean white noise process with variance λ^2

$$\Phi_y(\omega) = |G(e^{i\omega})|^2 \Phi_u(\omega) + \lambda^2 |H(e^{i\omega})|^2$$
$$\Phi_{yu}(\omega) = G(e^{i\omega})\Phi_u(\omega)$$

Automation Lab
IIT Bombay

Errors Analysis

True Behavior

$$y(k) = G(q)u(k) + H(q)e(k)$$

Proposed / Identified Model

$$y(k) = \hat{G}(q)u(k) + \hat{H}(q)e(k)$$

Prediction error

$$\varepsilon(k) = \hat{H}^{-1}(q)\left[y(k) - \hat{G}(q)u(k)\right]$$
$$= \hat{H}^{-1}(q)\left[\left(G(q) - \hat{G}(q)\right)u(k) + H(q)e(k)\right]$$

Parameters estimated by minimizing

$$V(\theta, Z_N) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon^2(k,\theta)$$

$$\begin{Bmatrix} \text{Total Error} \\ \text{of Estimation} \end{Bmatrix} = \{\text{Bias Error}\} + \{\text{Variance Error}\}$$

Automation Lab
IIT Bombay

Variance Errors

Asymptotic variance of estimates using PEM are

$$Var\left[\hat{G}(e^{i\omega})\right] \cong \frac{n}{N}\frac{\Phi_v(e^{i\omega})}{\Phi_u(e^{i\omega})}$$

Noise to Signal Ratio

$$Var\left[\hat{H}(e^{i\omega})\right] \cong \frac{n}{N}\left|H(e^{i\omega})\right|^2$$

n : Model Order N : Data Length

Implications:

✓Variance errors can be reduced by

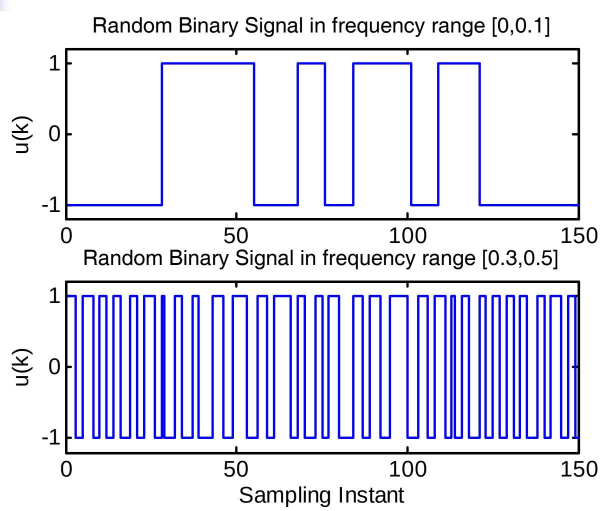
- increasing the data length (N)
- choosing high signal to noise ratio

$$\text{Signal to Noise Ratio (SNR)} = \frac{\Phi_u(e^{i\omega})}{\Phi_v(e^{i\omega})}$$

✓Models with large number of parameters require relatively larger data set for better parameter estimation.

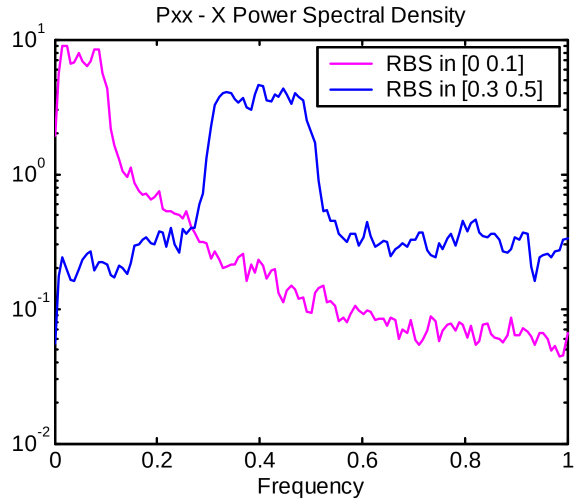
Automation Lab
IIT Bombay

Example: Input Selection



Generated By MATLAB Command 'idinput'

Input Spectrum



Bias Error: Concept

Real systems are of very high order and model is always chosen of lower order

Thus, bias errors are always present in any identification exercise

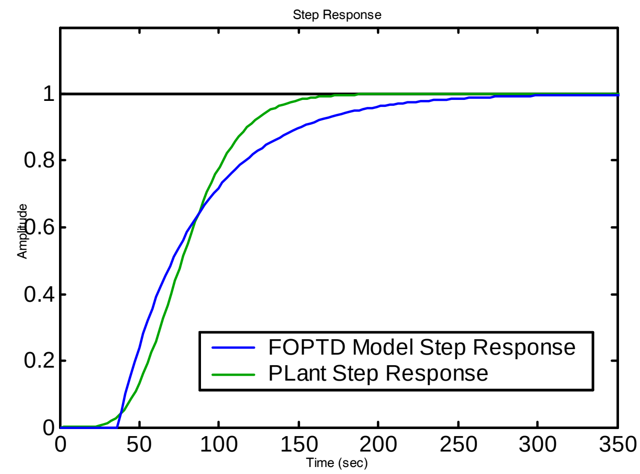
Classic Example in Process Control

Process Dynamics : $G(s) = \dfrac{1}{(10s+1)^8}$

Identified FOPTD model : $\hat{G}(s) = \dfrac{1}{(50s+1)} e^{-36s}$

Automation Lab
IIT Bombay

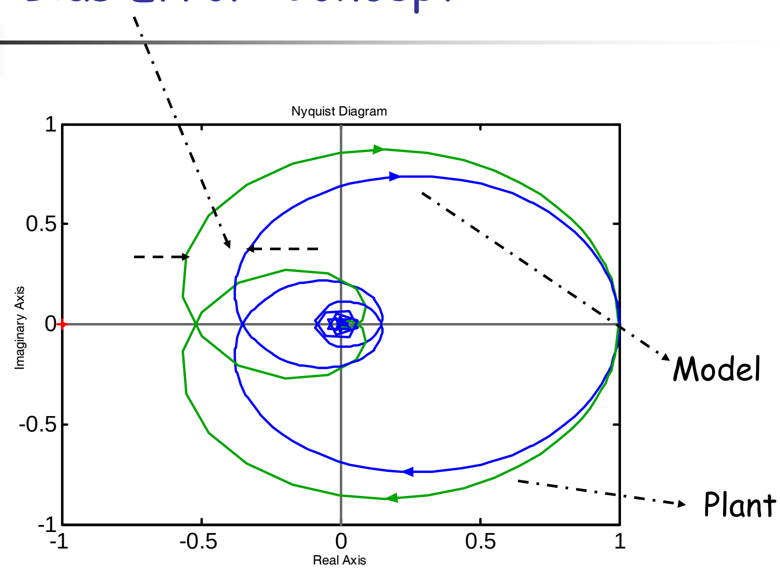
Bias Errors: Concepts



1/18/2006 System Identification 139

Automation Lab
IIT Bombay

Bias Error: Concept



1/18/2006 System Identification 140

Bias Errors

Prediction error

$$\varepsilon(k) = \hat{H}^{-1}(q)\left[\left(G(q) - \hat{G}(q)\right)u(k) + H(q)e(k)\right]$$

Parameters estimated by minimizing

$$V(\theta, Z_N) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k,\theta)$$

When data length N is large, we can write

$$R_\varepsilon(0) = \bar{E}\left[\varepsilon(k)\right] = \lim_{N\to\infty}\left[\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k,\theta)\right] = \lim_{N\to\infty}\left[V(\theta, Z_N)\right]$$

By Parseval's Theorem

$$\lim_{N\to\infty}\left[V(\theta, Z_N)\right] = \int_{-\pi}^{\pi}\Phi_\varepsilon(\omega)d\omega$$

$$\Phi_\varepsilon(\omega) = \left[\left|G(e^{i\omega}) - \hat{G}(e^{i\omega})\right|^2\Phi_u(\omega) + \Phi_v(\omega)\right]\frac{1}{\left|\hat{H}(e^{i\omega})\right|^2}$$

Bias Error: Interpretations

$$\lim_{N\to\infty}\left[\theta_N\right] = \int_{-\pi}^{\pi}\left[\left|G(e^{i\omega}) - \hat{G}(e^{i\omega})\right|^2\Phi_u(\omega) + \Phi_v(\omega)\right]\frac{1}{\left|\hat{H}(e^{i\omega})\right|^2}d\omega$$

- Bias distribution of $\left|G(e^{i\omega}) - \hat{G}(e^{i\omega})\right|^2$ in frequency domain is weighted by Signal To Noise Ratio
- Input spectrum can be chosen intelligently to reduce variance errors in certain frequency regions of interest
- For Output Error model (i.e. H(q)=1),

$$\lim_{N\to\infty}\left[\theta_N\right] = \int_{-\pi}^{\pi}\left[\left|G(e^{i\omega}) - \hat{G}(e^{i\omega})\right|^2\Phi_u(\omega)\right]d\omega$$

Thus, $\hat{G}(e^{i\omega}) \to G(e^{i\omega})$ if model is not under-parameterized

Summary

- Grey box models
  - Better choice for representing system dynamics.
  - Provide insight into internal working of the system
  - Development process time consuming and difficult
- Black Box Models
  - Relatively easy to develop
  - Provide no insight into internal working of systems
  - Limited extrapolation abilities.
- Black Box Model Development
  - Noise modeling is necessary to be able to extract the deterministic component of the model properly
  - Prediction error method used for parameter estimation

Summary

- Black Box Model Development
  - FIR and ARX models are relatively easy to develop but require large data set for reducing variance errors
  - OE, ARMAX or BJ models provide parsimonious description of model dynamics but require application nonlinear optimization for parameter estimation
  - Variance errors are directly proportional to number of model parameters and inversely proportional to data length
  - Frequency domain analysis provides insight into working of PEM. Variance errors can be reduced by appropriately selecting Signal to Noise Ratio
  - Bias errors in certain frequency region of interest can be reduced by appropriately choosing the spectrum of perturbation input sequences

References

- Ljung, L., System Identification: Theory for Users, Prentice Hall, 1987.
- Sodderstorm and Stoica, System Identification, 1989.
- Astrom, K. J., and B. Wittenmark, Computer Controlled Systems, Prentice Hall India (1994).