## Introduction to
# Soft Sensing and State Estimation

**Sachin C. Patawardhan**
**Department of Chemical Engineering**
**I.I.T. Bombay**
**Email: sachinp@iitb.ac.in**



## Outline

- Why state estimation?
- Observability
- Recursive Estimators and Luenberger Observer
- Optimal Recursive Estimation and Kalman Filtering
- Properties and Interpretations of Kalman Filtering
- Stationary Kalman Predictor and Time Series Models
- Extended Kalman Filtering
- Simulation examples and experimental case study

# Motivation

- Quality variables : product concentration, average molecular weight, melt viscosity etc.
  - Costly to measure on-line
  - Measured through lab assays: sampled at irregular intervals
- Measurements available from wireless sensors are at irregular intervals due to packet losses
- For satisfactory control of such processes: Quality variable / efficiency parameters should be estimated at a higher frequency
- Remedy: Soft Sensing and State Estimation





# State Feedback Controller Design

Discrete time State Space Model

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

State feedback multivariable control law

$$\mathbf{u}(k) = \mathbf{G}(\mathbf{x}_s(k) - \mathbf{x}(k))$$

- Step 1: Assume the states are measurable and design a stable control law / controller
- Step 2: Design a **state estimator** which constructs estimates of states by fusing measurements with model predictions
- Step 3: Implement the controller using the estimated states
- Separation principle ensures nominal closed loop stability with state estimator-controller pair

## Inferential Measurement: Basic Idea

Since fast sampled (primary) variables (temperatures, pressures, levels, pH) are correlated with the quality variable, can we infer values of quality variables from measurements of primary variables?

On line state estimation:
Feasible after availability of fast Computers





## Model Based Soft Sensing

Fast-rate Low-cost measurements from Plant (Temperature / Pressure / Speed)

Irregularly / Slowly sampled Quality variables from Lab assays

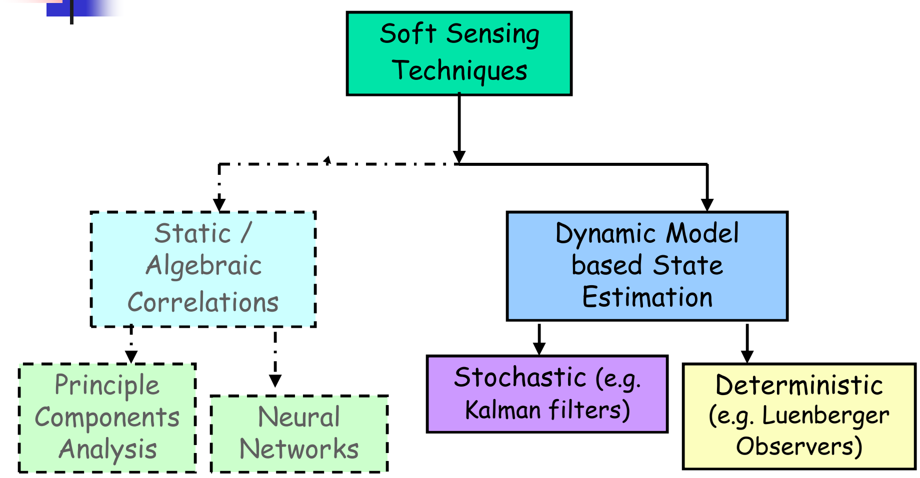
Dynamic Model
(ODEs/ PDEs)

On-line Fast Rate Estimates of Quality variables

Soft Sensing: Cost Effective Solution

Automation Lab
IIT Bombay

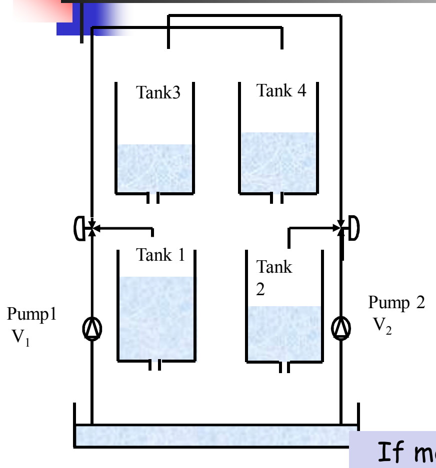
# Soft Sensing Approaches






Automation Lab
IIT Bombay

# Example: Quadruple Tank System




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

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

$$\frac{dh_3}{dt} = -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_1)k_1}{A_3}v_2$$

$$\frac{dh_4}{dt} = -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_2)k_2}{A_4}v_1$$

Manipulated Inputs : $v_1$ and $v_2$

Measured Outputs : $h_1$ and $h_2$

If model parameters are known accurately, can we estimate levels in Tanks 3 and 4 from measurements of levels in Tank 1 and 2?

## Example: Continuously Stirred Tank Reactor

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 $(\mathbf{X}) \equiv [C_A \ T]^T$   Measured Output $(\mathbf{Y}) \equiv [T]$

Manipulated Inputs $(\mathbf{U}) \equiv [F \ F_c]^T$

Unmeasured Disturbances $(\mathbf{D}_u) \equiv [C_{A0}]$   Feed conc.

Measured Disturbances $(\mathbf{D}_m) \equiv [T_{cin}]$   Cooling water Temp.

If model parameters are known accurately, can we estimate $C_A$ from measurements of T alone?



Automation Lab
IIT Bombay

## 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) ;

$\rho$ (Reacting liquid density) = $10^6$ g/$m^3$; $\rho_c$( Coolent density ) = $10^6$ g/$m^3$;

- $\Delta H_{rnx}$ (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) $= 121^0C$

Operating Steady State

## State Estimation Problem

It is desired to implement a state feedback control law. However, all the states are not measured.

Thus, given
Computer control relevant discrete model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \Psi\mathbf{d}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

and input-output data
$\{\mathbf{y}(0), \mathbf{y}(1).....\mathbf{y}(N)\}$ and $\{\mathbf{u}(0), \mathbf{u}(1).....\mathbf{u}(N)\}$

Can we estimate state sequence
$\{\hat{\mathbf{x}}(0), \hat{\mathbf{x}}(1).....\hat{\mathbf{x}}(k)\}$ ?





## Simplified Problem Statement

Consider ideal situation where
- disturbances and measurement errors are absent
- model is perfect

Problem: Given measurements y(0), y(1),...y(N) and inputs u(0), u(1), ....,u(N) together with model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

Estimate state sequence $\mathbf{x}(0), \mathbf{x}(1),....$

Since we have the model, it is sufficient to estimate only $\hat{\mathbf{x}}(0)$.
$\hat{\mathbf{x}}(1), \hat{\mathbf{x}}(2),....$can be estimated through recursive use of the model

# Initial State

Let $\mathbf{x}(0)$ denote initial state estimate
and given input sequence
$\{\mathbf{u}(0), \mathbf{u}(1), \mathbf{u}(2), .....\}$
we can use model to estimate

$$\mathbf{x}(1) = \Phi\mathbf{x}(0) + \Gamma\mathbf{u}(0)$$

$$\mathbf{x}(2) = \Phi\mathbf{x}(1) + \Gamma\mathbf{u}(1)$$
$$= \Phi^2\mathbf{x}(0) + \Phi\Gamma\mathbf{u}(0) + \Gamma\mathbf{u}(1)$$

$$\mathbf{x}(3) = \Phi^3\mathbf{x}(0) + \Phi^2\Gamma\mathbf{u}(0) + ....$$

How to find $\mathbf{x}(0)$?





# Estimation of Initial State

Given measurements $\mathbf{y}(0), \mathbf{y}(1), ...., \mathbf{y}(n-1)$
and inputs $\{\mathbf{u}(0), \mathbf{u}(1), \mathbf{u}(2), .....\}$
we can write

$$\mathbf{C}\mathbf{x}(0) = \mathbf{y}(0)$$

$$\mathbf{C}\mathbf{x}(1) = \mathbf{y}(1) = \mathbf{C}\Phi\mathbf{x}(0) + \mathbf{C}\Gamma\mathbf{u}(0)$$

$$\Rightarrow \mathbf{C}\Phi\mathbf{x}(0) = \mathbf{y}(1) - \mathbf{C}\Gamma\mathbf{u}(0)$$

......................................

$$\mathbf{C}\Phi^{n-1}\mathbf{x}(0) = \mathbf{y}(n-1) - \mathbf{C}\Phi^{n-2}\Gamma\mathbf{u}(0) - ....\mathbf{C}\Gamma\mathbf{u}(n-2)$$

Can we uniquely estimate the initial state by
Solving above set of linear algebraic equations?

Automation Lab
IIT Bombay

# Estimation of Initial State

Combining the equations, we have

$$\underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{C\Phi} \\ \mathbf{C\Phi}^2 \\ \cdots \\ \mathbf{C\Phi}^{n-1} \end{bmatrix}}_{\mathbf{A}} \underbrace{\mathbf{x}(0)}_{(n\times 1)} = \underbrace{\begin{bmatrix} \mathbf{y}(0) \\ \mathbf{y}(1) - \mathbf{C\Gamma u}(0) \\ \mathbf{y}(2) - \mathbf{C\Phi\Gamma u}(0) - \mathbf{C\Gamma u}(1) \\ \cdots\cdots \\ \mathbf{y}(n-1) - \mathbf{C\Phi}^{n-2}\mathbf{\Gamma u}(0) - \ldots \mathbf{C\Gamma u}(n-2) \end{bmatrix}}_{\mathbf{b}}$$

Known quantity

$$\underbrace{\mathbf{A}}_{(nr)\times n} \mathbf{x}(0) = \underbrace{\mathbf{b}}_{(nr)\times 1}$$

A unique solution $\mathbf{x}(0)$ can be found only if matrix $\mathbf{A}$ has rank equal to 'n'

Automation Lab
IIT Bombay

# Observability

**Observability:** System is said to be observable if initial state can be **uniquely** estimated from output observations

Initial state can be uniquely estimated from measurements of inputs and outputs if following rank condition holds

Observability Matrix

$$rank \begin{bmatrix} \mathbf{C} \\ \mathbf{C\Phi} \\ \cdots \\ \mathbf{C\Phi}^{n-1} \end{bmatrix} = n = \text{state dimension}$$

## CSTR: Continuous Perturbation Model

Continuous time linear state space model

$$\mathbf{x}(t)=\begin{bmatrix} C_A(t)-\bar{C}_A \\ T(t)-\bar{T} \end{bmatrix} \; ; \; \mathbf{u}(t)=\begin{bmatrix} F(t)-\bar{F} \\ F_c(t)-\bar{F}_c \end{bmatrix} \; ; \mathbf{d}(t)=C_{Ai}(t)-\bar{C}_{Ai}$$

$$\frac{d\mathbf{x}}{dt}=\begin{bmatrix} -7.56 & -0.09 \\ 852.72 & 5.77 \end{bmatrix}\mathbf{x}(t)+\begin{bmatrix} 0 & 1.735 \\ -6.07 & -70.95 \end{bmatrix}\mathbf{u}(t)+\begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{d}(t)$$

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

Discrete time linear state space model

Sampling Time (T) = 0.1 min

$$\mathbf{x}(k+1)=\begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix}\mathbf{x}(k)+\begin{bmatrix} 0.005 & 0.13 \\ -0.73 & -1.8 \end{bmatrix}\mathbf{u}(k)+\begin{bmatrix} 0.06 \\ 3.9 \end{bmatrix}\mathbf{d}(k)$$

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





## Observability: CSTR Example

Can we estimate concentrations
from measurements of temperature ?

$$\Phi=\begin{bmatrix} 0.185 & -0.008 \\ 73.492 & 1.333 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$rank\begin{bmatrix} C \\ C\Phi \end{bmatrix}=\text{rank}\begin{bmatrix} 0 & 1 \\ 73.492 & 1.333 \end{bmatrix}=2$$

Linear Perturbation model for CSTR is observable

Let x(0)= (0.1,1) and u(0)=(0,0),
Then we get x(1) = (0.0104, 8.682) and
Temperature measurements are
y(0) = 1, y(1) = 8.682.

Estimated initial state from measurements: (0.1,1)

# Quadruple Tank System

Discrete Time State Space Model
Sampling Time T = 5 sec

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

$$\Phi = \begin{bmatrix} 0.9233 & 0 & 0.1813 & 0 \\ 0 & 0.9462 & 0 & 0.1493 \\ 0 & 0 & 0.8112 & 0 \\ 0 & 0 & 0 & 0.8465 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0.4001 & 0.02276 \\ 0.01209 & 0.3055 \\ 0 & 0.2159 \\ 0.1438 & 0 \end{bmatrix}$$





# Measurement Structure Selection

Question: We have only two level sensors and we would like to place them such that the state is observable from the level measurements. How to place the sensors?

Observability can be used as a basis for placing the sensors

Structure 1 : $h_1$ and $h_2$

$$\mathbf{C} = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \end{bmatrix}$$

Observability Matrix Rank = 4

Structure 2 : $h_3$ and $h_4$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix}$$

Observability Matrix Rank = 2

Automation Lab
IIT Bombay

## Measurement Structure Selection

Structure 3 : $h_1$ and $h_3$

$$\mathbf{C} = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \end{bmatrix}$$

Observability Matrix Rank = 2

Structure 4 : $h_2$ and $h_4$

$$\mathbf{C} = \begin{bmatrix} 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix}$$

Observability Matrix Rank = 2

Structure 5 : $h_1$ and $h_4$

$$\mathbf{C} = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix}$$

Observability Matrix Rank = 3

Structure 6 : $h_2$ and $h_3$

$$\mathbf{C} = \begin{bmatrix} 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \end{bmatrix}$$

Observability Matrix Rank = 3

Thus, only structure 1, i.e. $h_1$ and $h_2$ measured, permits state observability.



Automation Lab
IIT Bombay

## Quadruple Tank System

Rank of observability matrix = 4
only when $h_1$ and $h_2$ are measured

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{C\Phi} \\ \mathbf{C\Phi}^2 \\ \mathbf{C\Phi}^3 \end{bmatrix} = \begin{bmatrix} 0.5000 & 0 & 0 & 0 \\ 0 & 0.5000 & 0 & 0 \\ 0.4617 & 0 & 0.0906 & 0 \\ 0 & 0.4731 & 0 & 0.0746 \\ 0.4263 & 0 & 0.1572 & 0 \\ 0 & 0.4476 & 0 & 0.1338 \\ 0.3936 & 0 & 0.2048 & 0 \\ 0 & 0.4235 & 0 & 0.1800 \end{bmatrix}$$

Levels in Tank 3 and Tank 4 can be estimated using level measurements of Tank 1 and Tank 2

# Measurements with errors

What if measurements have errors ?

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

Collect larger sample of size N >> n

Perform least square estimation

True Value

Measurement Noise

$$\min_{\hat{x}(0)} \sum_{k=0}^{N} \hat{v}(k)^T R^{-1} \hat{v}(k)$$

*subject to*

$$\hat{v}(k) = y(k) - \left[ C\Phi^k \hat{x}(0) + \sum_{j=1}^{k-1} C\Phi^{j-1}\Gamma u(k-j) \right]$$

R: Measurement Noise Covariance





# CSTR Example (Contd.)

Let the process initial state be (0.1,1) and input sequence be u(0) = u(1) =…u(5) = (0,0)

Suppose we collect following 6 temperature measurements corrupted with measurement noise
Ym = (0.957, 8.516, 12.353, 11.498, 6.975, 1.291)

Least square estimate of state vector

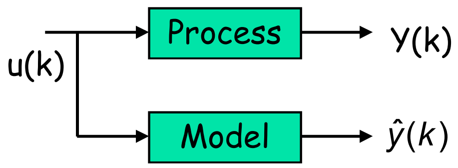
$$\hat{\mathbf{x}}(0) = \begin{bmatrix} 0.1003 \\ 0.924 \end{bmatrix}$$

Estimate improves if more measurements are added.

Difficulty in on-line implementation:
Optimization problem size grows with time!

# On-line State Observer

On-line recursive estimation of states
from measured data and mathematical model




*Note*

$\hat{\mathbf{x}}(0) \neq \mathbf{x}(0)$

True Process

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) \quad ....(1)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

"Open-Loop" State Estimator

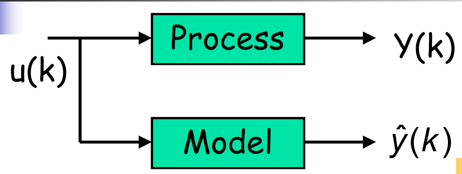
$$\hat{\mathbf{x}}(k+1) = \Phi\hat{\mathbf{x}}(k) + \Gamma\mathbf{u}(k) \quad ....(2)$$
$$\hat{\mathbf{y}}(k) = \mathbf{C}\hat{\mathbf{x}}(k)$$

Difficulty : Initial State $\hat{x}(0)$ is not know exactly.





# On-line State Observer




Difficulty:
Cannot be used
If process is marginally
Stable or unstable.

Defining Estimation error

$$\varepsilon(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k)$$

Subtracting (2) from (1), we have

$$\varepsilon(k+1) = \Phi\varepsilon(k) \Rightarrow \varepsilon(k) = \Phi^k\varepsilon(0)$$

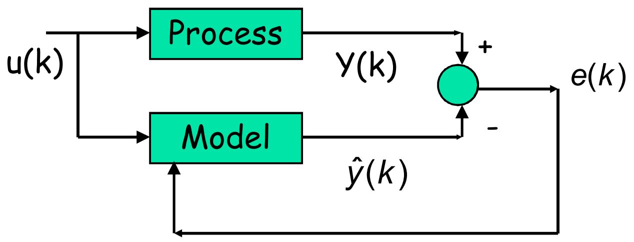
If process is stable, i.e., $\rho(\Phi) < 1$,
then $\varepsilon(k) \to 0$ as $k \to \infty$

Even when $\rho(\Phi) < 1$
$\rho(\Phi)$ decides rate of
convergence of $\varepsilon(k)$
Can we accelerate
the convergence?

## "Closed Loop" State Observer

Open Loop Observer: Difficulties

1. Not applicable to unstable systems
2. Rate of convergence governed by spectral radius of $\Phi$ matrix




Use of output prediction error to

1. Stabilize estimator for unstable processes
2. Improve rate of convergence for stable systems





## Recursive Estimation

**Recursive On-line State Estimator**

$$\hat{\mathbf{x}}(k+1) = \Phi\hat{\mathbf{x}}(k) + \Gamma\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k)$$

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k)$$

$$= \mathbf{y}(k) - \hat{\mathbf{y}}(k)$$

Estimation error

Feedback Correction

True process dynamics (deterministic case)

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

How to choose estimator gain matrix L such that estimation error reduces to zero as quickly as possible?

## Estimator Error Dynamics

Automation Lab
IIT Bombay

Estimation Error $\boldsymbol{\varepsilon}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k)$

$$\boldsymbol{\varepsilon}(k+1) = (\Phi - \mathbf{LC})\boldsymbol{\varepsilon}(k)$$

$$or \quad \boldsymbol{\varepsilon}(k) = (\Phi - \mathbf{LC})^k \boldsymbol{\varepsilon}(0)$$

Choose observer L gain such that

$$\max_i \left| \lambda_i (\Phi - \mathbf{LC}) \right| < 1$$

$\lambda_i(.) : i'th\ eigenvalue\ of\ matrix\ (\Phi - \mathbf{LC})$

The above choice ensure $\|\varepsilon(k)\| \to 0$ *as* $k \to \infty$

*as* $(\Phi - \mathbf{LC})^k \to$ *Null Matrix as* $k \to \infty$

**irrespective of choice of** $\hat{\mathbf{x}}(0)$ **i.e.** $\boldsymbol{\varepsilon}(0)$



## Single Output System (SOS): Luenberger Observer

Automation Lab
IIT Bombay

Deterministic Observer Design:

Choose observer gain matrix L such that matrix $\Phi - \mathbf{LC}$ has poles at the desired locations (Pole Placement)

Choice of observer poles: Compromise between decay of estimation error and sensitivity to measurement noise/modeling errors

Choice of poles so as to systematically account for Measurement noise and Unmeasured Disturbances is difficult

Consequence: sub-optimal performance in presence of stochastic disturbances

# Pole Placement Design

Consider CSTR model with

$$\Phi = \begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

Let the observer gain matrix be

$$\mathbf{L} = \begin{bmatrix} a & b \end{bmatrix}^T$$

$$\Phi - \mathbf{LC} = \begin{bmatrix} 0.185 & -0.01\text{-}a \\ 73.49 & 1.33\text{-}b \end{bmatrix}$$

The characteristic equation of $\Phi - \mathbf{LC}$ is

$$\det[\lambda \mathbf{I} - (\Phi - \mathbf{LC})] = (\lambda - 0.185)(\lambda - (1.33\text{-}b)) + 73.49(0.01 + a)$$
$$= \lambda^2 - (1.515\text{-}b)\lambda + (73.49a - 0.185b + 0.9809)$$





# Pole Placement Design

Suppose we want to 'place' poles of $\Phi - \mathbf{LC}$ at

$$\lambda_1 = 0.5, \lambda_2 = 0.25$$

i.e. desired characteristic polynomial of $\Phi - \mathbf{LC}$ is

$$(\lambda - 0.5)(\lambda - 0.25) = \lambda^2 - 0.75\lambda + 0.125$$

Comparing coefficients of characteristic polynomials,
the poles can be placed at the desired location if

$$(1.515\text{-}b) = 0.75$$
$$(73.49a - 0.185b + 0.9809) = 0.125$$

i.e. if se set $a = -0.0097$ and $b = 0.765$

Difficulty: This 'raw approach' of placing poles becomes cumbersome to use for systems of higher dimension.
Remedy: Use variable transformations for pole placement design

# SOS Luenberger Observer

Coordinate Transformation : $\eta(k) = Tx(k)$

Original Model

$$x(k+1) = \Phi x(k) + \Gamma u(k) \quad .......(I)$$
$$y(k) = Cx(k)$$

Transfer Fundtion

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

$\longrightarrow$

Observable Cannonical Form

$$\eta(k+1) = \Phi_o \eta(k) + \Gamma_o u(k) \quad .......(II)$$
$$y(k) = C_o \eta(k)$$

$$\Phi_o = \begin{bmatrix} -a_1 & 1 & 0.... & 0 \\ -a_2 & 0 & 1.... & 0 \\ .... & .... & .... & .... \\ -a_n & 0 & .... & 0 \end{bmatrix} \quad \Gamma_o = \begin{bmatrix} b_1 \\ b_2 \\ .... \\ b_n \end{bmatrix}$$

$$C_o = \begin{bmatrix} 1 & 0 & ..... & 0 \end{bmatrix}$$

Design Procedure

- Transform the model to observable canonical form
- In transformed coordinates, choose observer vector such that poles of are placed at desired location
- Express the observer gain matrix in the original co-ordinate system





# SOS Luenberger Observer

Design observer in transformed coordinates

$$\hat{\boldsymbol{\eta}}(k+1) = \Phi_o \hat{\boldsymbol{\eta}}(k) + \Gamma_o \mathbf{u}(k) + \mathbf{L}_o \mathbf{C}_o [\boldsymbol{\eta}(k) - \hat{\boldsymbol{\eta}}(k)]$$

$$\Phi_o - \mathbf{L}_0 \mathbf{C}_o = \begin{bmatrix} -a_1 - l_{o,1} & 1 & 0.... & 0 \\ -a_2 - l_{o,2} & 0 & 1.... & 0 \\ .... & .... & .... & .... \\ -a_n - l_{o,n} & 0 & .... & 0 \end{bmatrix}$$

$$\Rightarrow \det[\lambda \mathbf{I} - (\Phi_o - \mathbf{L}_0 \mathbf{C}_o)] = \lambda^n + (a_1 + l_{o,1})\lambda^{n-1} + ....... + (l_{o,n} + a_n)$$

Let the desired observer characteristic polynomial be

$$P(\lambda) = \lambda^n + p_1 \lambda^{n-1} + ....... + p_n$$

where polynomial on R.H.S. has poles at the desired location

Equating coefficients of $\det[\lambda I - (\Phi_o - \mathbf{L}_0 \mathbf{C}_o)]$ with $P(\lambda)$, we have

$$p_i = a_i + l_{o,i} \Rightarrow l_{o,i} = p_i - a_i \quad \text{for } i = 1,2...n$$

# SOS Luenberger Observer

Transform the observer $L_o$ back to original state space as

$$\mathbf{L} = \mathbf{T}^{-1}\mathbf{L}_o$$

Coordinate Transformation

$$\boldsymbol{\eta} = \mathbf{T}\mathbf{x}$$

$$\mathbf{T} = \left[\tilde{\mathbf{W}}_{OBS}\right]^{-1}\mathbf{W}_{OBS}$$

$$\tilde{\mathbf{W}}_{OBS} = \begin{bmatrix} \mathbf{C}_o \\ \mathbf{C}_o\boldsymbol{\Phi}_o \\ \ldots\ldots \\ \mathbf{C}_o\boldsymbol{\Phi}_o^{n-1} \end{bmatrix} \; ; \; \mathbf{W}_{OBS} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\boldsymbol{\Phi} \\ \ldots\ldots \\ \mathbf{C}\boldsymbol{\Phi}^{n-1} \end{bmatrix}$$

Note that the above coordinate transformation is possible only if the original system is observable, i.e. Rank( $W_{OBS}$) = n





# CSTR Example

Linearized (Original) State Space Model

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix}\mathbf{x}(k) + \begin{bmatrix} 0.005 & 0.13 \\ -0.73 & -1.8 \end{bmatrix}\mathbf{u}(k)$$

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

Observable Canonical form

$$\boldsymbol{\eta}(k+1) = \begin{bmatrix} 1.518 & 1 \\ -0.836 & 0 \end{bmatrix}\boldsymbol{\eta}(k) + \begin{bmatrix} -0.7335 & -1.797 \\ 0.3256 & -10.18 \end{bmatrix}\mathbf{u}(k)$$

$$\mathbf{y}(k) = \begin{bmatrix} 1 & 0 \end{bmatrix}\boldsymbol{\eta}(k)$$

$$\mathbf{L}_p = \mathbf{T}^{-1}\begin{bmatrix} p_1 + 1.518 \\ p_2 - 0.836 \end{bmatrix}$$

$$\mathbf{T} = \left[\tilde{\mathbf{W}}_{OBS}\right]^{-1}\mathbf{W}_{OBS} = \begin{bmatrix} 1 & 0 \\ 1.518 & 1 \end{bmatrix}^{-1}\begin{bmatrix} 0 & 1 \\ 73.492 & 1.333 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 73.492 & -0.185 \end{bmatrix}$$

## Prediction Estimation

The observer we have designed
corresponds to "prediction estimation"

$$\hat{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k-1) + \Gamma\mathbf{u}(k) + \mathbf{L}_P\left[y(k) - \mathbf{C}\hat{\mathbf{x}}(k|k-1)\right]$$

$\hat{\mathbf{x}}(k+1|k)$ : Prediction estimate of state
at time instant $(k+1)$
based on information up to time instant (k)

Can be employed if sampling time is very small and
time for estimator calculations is significant relative to the sampling interval.
$\hat{\mathbf{x}}(k|k-1)$ calculations can be carried out during intersample period
and used for controller implementation at the k'th insiant.
Disadvantage : Unit information delay



Automation Lab
IIT Bombay

## Prediction Estimation and Current State Estimation

Current state estimator
Prediction Step

$$\hat{\mathbf{x}}(k|k-1) = \Phi\hat{\mathbf{x}}(k-1|k-1) + \Gamma\mathbf{u}(k-1)$$

Measurement Update

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{L}_C\left[\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k|k-1)\right]$$

Estimation error dynamics

$$\varepsilon(k+1|k) = \Phi\varepsilon(k|k)$$

$$\varepsilon(k|k) = [\mathbf{I} - \mathbf{L}_C\mathbf{C}]\varepsilon(k|k-1)$$

$$\Rightarrow \varepsilon(k+1|k) = \Phi[\mathbf{I} - \mathbf{L}_C\mathbf{C}]\varepsilon(k|k-1)$$

Prediction estimator and Current state estimator
gain matrices are related as

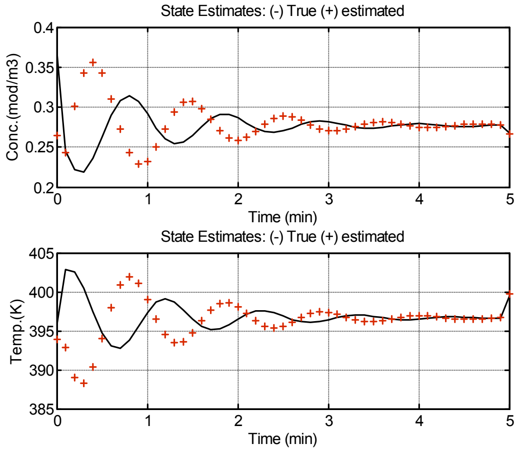
$$\mathbf{L}_P = \mathbf{\Phi}\mathbf{L}_c \quad or \quad \mathbf{L}_C = \mathbf{\Phi}^{-1}\mathbf{L}_P$$

Automation Lab
IIT Bombay

# CSTR: "Open Loop" Observer

"Open Loop" Observer" : no Measurement based Correction

## Linear Plant- Linear "Open Loop" Observer



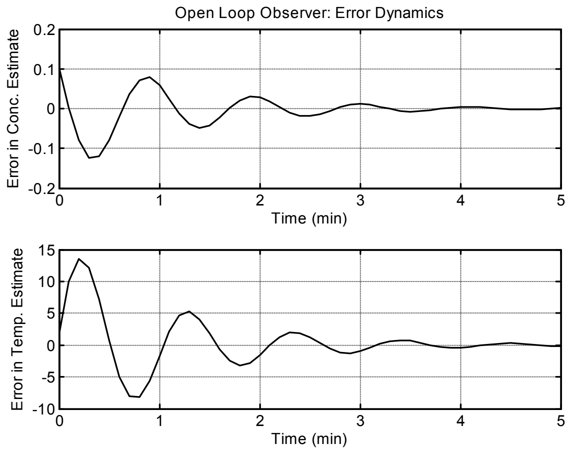



Automation Lab
IIT Bombay

# CSTR: "Open Loop" Observer

"Open Loop" Observer" : no Measurement based Correction

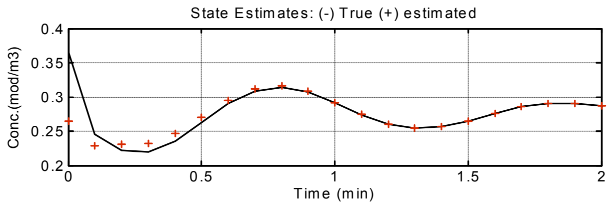
## Linear Plant- Linear "Open Loop" Observer

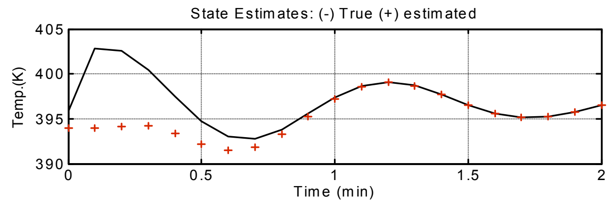


# CSTR: Luenberger Observer

Observer poles: Both poles placed at 0.5

## Linear Plant- Linear Observer



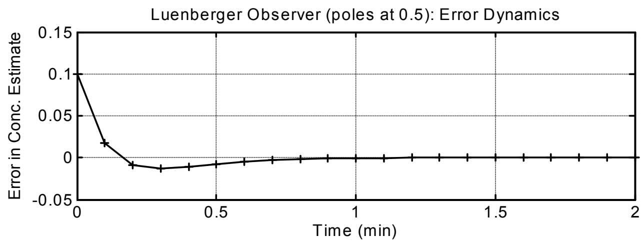



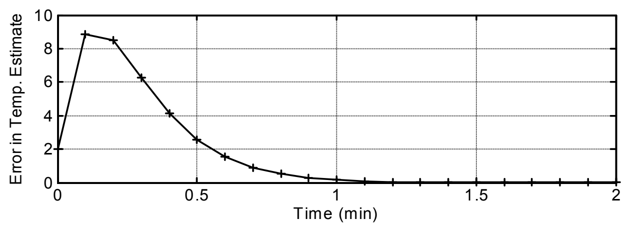





# CSTR: Luenberger Observer

Observer poles: Both poles placed at 0.5



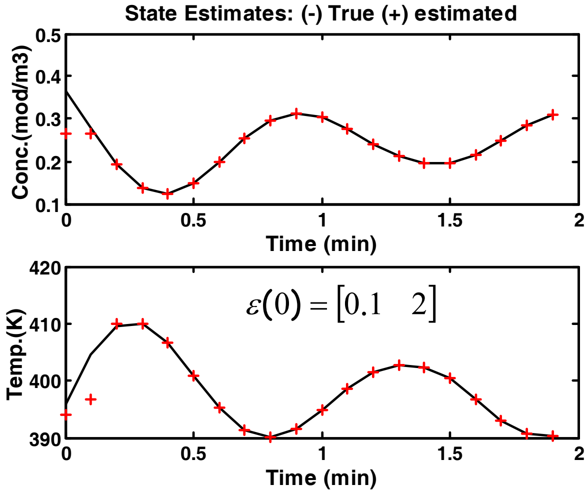



Automation Lab
IIT Bombay

## CSTR Example: Dead-beat Observer

### Linear Plant- Linear Observer

**State Estimates: (-) True (+) estimated**



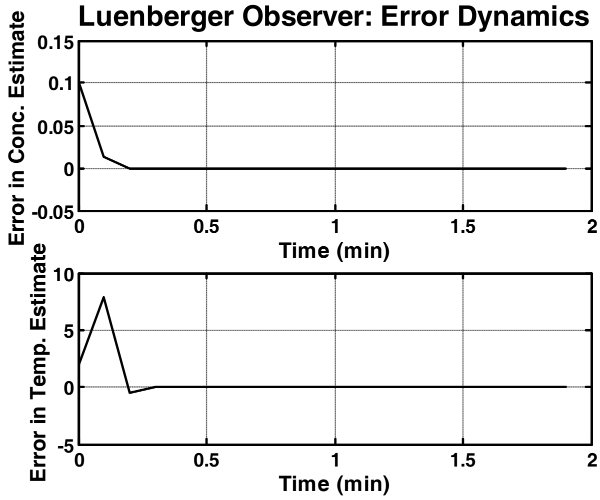



Automation Lab
IIT Bombay

## CSTR Example: Dead-beat Observer

### Linear Plant- Linear Observer

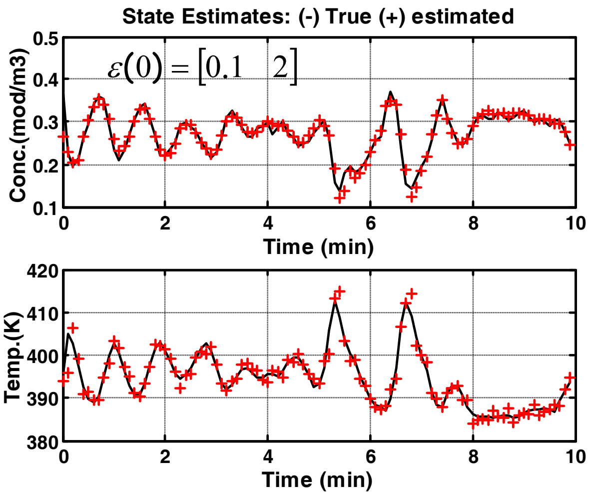
**Luenberger Observer: Error Dynamics**




Dead-beat Observer Poles at (0,0)

# CSTR Example: Dead-beat Observer

## Non-Linear Plant- Linear Observer

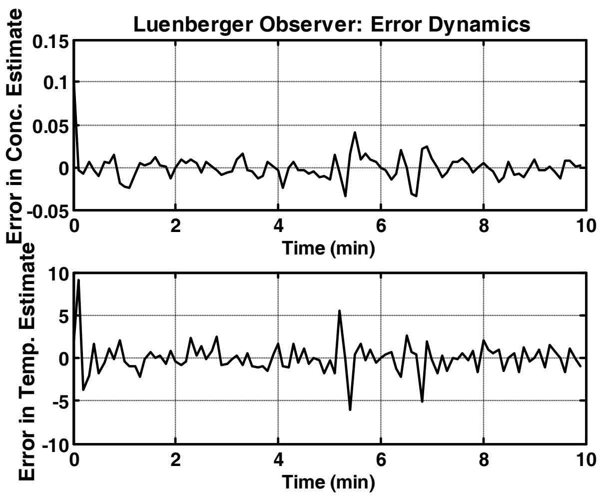







# CSTR Example: Dead-beat Observer

## Non-Linear Plant- Linear Observer



# Estimation Error Variances

| Luenberger | Conc. | Temp. |
|---|---|---|
| Linear Plant | $3.993 \times 10^{-5}$ | 1.112 |
| Nonlinear Plant | $2.534 \times 10^{-4}$ | 3.3303 |

| Kalman predictor | Conc. | Temp. |
|---|---|---|
| Linear Plant | $3.984 \times 10^{-5}$ | 1.113 |
| Nonlinear Plant | $2.547 \times 10^{-4}$ | 3.341 |





# A Difficulty

Consider CSTR model with modified observation matrix

$$\Phi = \begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Since two measurements are available, the $\mathbf{L}$ is of the form

$$\mathbf{L} = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}^T$$

$$\Phi - \mathbf{LC} = \begin{bmatrix} 0.185 - \alpha & -0.01-\beta \\ 73.49 - \gamma & 1.33-\delta \end{bmatrix}$$

The $\det[\lambda \mathbf{I} - (\Phi - \mathbf{LC})]$ is function of $(\alpha, \beta, \gamma, \delta)$

Comparing coefficients with $\lambda^2 + p_1\lambda + p_2$ yields only 2 equations in 4 unonowns. $\Rightarrow$ There is no unique solution to the design problem.

## Difficulty with Multiple Output Systems

In general, for a system with r measurements, the observer gain matrix is a matrix with (r x n) unknowns

Comparison of the characteristic equation of the observer error dynamics with desired characteristic polynomial yields only n equations in (r x n) unknowns

Thus, (r-1) x n unknowns have to be determined by some other means. Also, for a large dimensional system it is difficult to place poles optimally.





## Unmeasured Disturbances

What if there are unknown disturbances influencing state dynamics?

What if the measurements are corrupted with measurement noise?

Suppose we have stochastic models for time evolution of these unmeasured disturbances and measurement noise, then

can we use these models to design a state estimator, which filters out the measurement noise but compensates for the unmeasured disturbances?

It is difficult to carry out pole placement based on these noise models such that the desired goal is achieved.

## Unmeasured Disturbances

Consider Continuous Time Linear Perturbation Model obtained through linearization of a mechanistic model

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{H}\mathbf{d}(t)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

Perturbation variables

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

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

Computer Controlled Systems

Manipulated inputs are piecewise constant

$$\mathbf{u}(t) = \mathbf{u}(k)$$

$$for \quad t = kT \le t < (k+1)T$$

Difficulty
Disturbance inputs d(t) are NOT piecewise constant functions!
How to develop a discrete time model?





## Unmeasured Disturbances

Simplifying Assumption 1 :
Sampling interval (T) is small enough so that disturbance inputs can be approximated as piecewise constant functions during the sampling interval

$$\mathbf{d}(t) = \mathbf{d}(k) \quad for \quad t = kT \le t < (k+1)T$$

Under the simplifying assumption 1, we can write

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \Psi\mathbf{d}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

where $\Phi = \exp(\mathbf{A}T)$

$$\Gamma = \int_0^T \exp(\mathbf{A}\tau)\mathbf{B}d\tau \quad \text{and} \quad \Psi = \int_0^T \exp(\mathbf{A}\tau)\mathbf{H}d\tau$$

# Unmeasured Disturbances

Simplifying Assumption 2 :

$\mathbf{d}(k)$ : zero mean white noise process

with $\text{Cov}[\mathbf{w}(k)] = \text{E}[\mathbf{w}(k)\mathbf{w}(k)^T] = \mathbf{Q}_d$

Simplifying Assumption 3 :

Measurements are corrupted

with zero mean white noise process

$\{\mathbf{v}(\text{k})\}$ with $\text{Cov}[\mathbf{v}(\text{k})] = \text{E}[\mathbf{v}(\text{k})\mathbf{v}(\text{k})^T] = \mathbf{R}$

Define $\mathbf{w}(k) = \Psi \mathbf{d}(k)$

$$E\{\mathbf{w}(k)\} = \Psi E\{\mathbf{d}(k)\} = \bar{\mathbf{0}}$$

$$Cov\{\mathbf{w}(k)\} = E\{\mathbf{w}(k)\mathbf{w}(k)^T\} = \Psi E\{\mathbf{d}(k)\mathbf{d}(k)^T\}\Psi^T = \Psi \mathbf{Q}_d \Psi^T$$

Let $\mathbf{Q} = \Psi \mathbf{Q}_d \Psi^T$





# Unmeasured Disturbances

Thus, we consider a general model of the form

$$\mathbf{x}(k+1) = \Phi \mathbf{x}(k) + \Gamma \mathbf{u}(k) + \mathbf{w}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

where

$\{\mathbf{w}(k)\}$ is a zero mean white noise with $Cov(\mathbf{w}(k)) = \mathbf{Q}$

$\{\mathbf{v}(k)\}$ is a zero mean white noise with $Cov(\mathbf{w}(k)) = \mathbf{R}$

Additional source of uncertainty: unknown initial state

Simplifying Assumption 4

Initial State at k = 0 is a Random Variable such that

$\text{E}[\mathbf{x}(0)] = \text{E}[\hat{\mathbf{x}}(0 \mid 0)] \quad \text{Cov}[\mathbf{x}(0)] = \mathbf{P}(0)$

$\Rightarrow \text{E}[\mathbf{x}(0) - \hat{\mathbf{x}}(0 \mid 0)] = \text{E}[\varepsilon(0 \mid 0)] = \bar{\mathbf{0}}$

## CSTR: Continuous Perturbation Model

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix}\mathbf{x}(k) + \begin{bmatrix} 0.005 & 0.13 \\ -0.73 & -1.8 \end{bmatrix}\mathbf{u}(k) + \begin{bmatrix} 0.06 \\ 3.9 \end{bmatrix}\mathbf{d}(k)$$

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

State Noise

$E[d(k)] = 0$ and $Cov[d(k)] = (0.05)^2$

$$\Rightarrow Q = Cov[w(k)] = \Psi Q_d \Psi^T = \begin{bmatrix} 0.06 \\ 3.9 \end{bmatrix}(0.05)^2\begin{bmatrix} 0.06 \\ 3.9 \end{bmatrix}^T$$

$$= (0.05)^2\begin{bmatrix} 0.0036 & 0.234 \\ 0.234 & 15.21 \end{bmatrix}$$

Measurement Noise

$E[v(k)] = 0$ and $Cov\{v(k)\} = R = (0.5)^2$





## Optimal State Estimation

Thus, given stochastic state space model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \mathbf{w}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

where **w**(k) and **v**(k) are uncorrelated (in time and with each other) random sequences with zero mean and know variances

$$E\left[\mathbf{w}(k)\mathbf{w}(k)^T\right] = \mathbf{Q} \quad ; \quad E\left[\mathbf{v}(k)\mathbf{v}(k)^T\right] = \mathbf{R}$$

**Q** quantify uncertainties in state dynamics and/or due to modeling errors/unmeasured disturbances
**R** quantifies variability of measurement errors

Given measurements {y(k)}, inputs (u(k)} and the model, how to construct optimal state estimate?

Automation Lab
IIT Bombay

# Optimal State Estimation

- Since {w(k)} and {v(k)} are stochastic processes, the state sequence {x(k)} is also a stochastic process
- Notice that through the difference equation, x(k) and x(k-j) are correlated. Thus, even when the sequences {w(k)} and {v(k)} are white noise processes, {x(k)} is a correlated stochastic variable.
- Two important statistical measures that can be used to characterize the stochastic process {x(k)} are its mean and covariance functions, which are related to characteristics of {w(k)} and {v(k)}.

Automation Lab
IIT Bombay

# Preliminaries

Define set

$$\mathbf{Y}^k \equiv \{(\mathbf{y}(0),\mathbf{u}(0)),(\mathbf{y}(1),\mathbf{u}(1)),\ldots\ldots,(\mathbf{y}(k),\mathbf{u}(k))\}$$

- Under weak conditions, the best (i.e. optimal) estimate is the conditional (or a posteriori) mean

$$\hat{\mathbf{x}}(k\,|\,k) \equiv E\left[\mathbf{x}(k)\,|\,\mathbf{Y}^k\right]$$

Prediction Step

$$E\left[\mathbf{x}(k)\,|\,\mathbf{Y}^{k-1}\right] = E\left[\Phi\mathbf{x}(k-1)+\Gamma\mathbf{u}(k-1)+\mathbf{w}(k-1)\,|\,\mathbf{Y}^{k-1}\right] \quad 0$$

$$= \Phi E\left[\mathbf{x}(k-1)\,|\,\mathbf{Y}^{k-1}\right]+\Gamma\mathbf{u}(k-1)+E\left[\mathbf{w}(k-1)\right]$$

OR $\hat{\mathbf{x}}(k\,|\,k-1) = \Phi\hat{\mathbf{x}}(k-1\,|\,k-1)+\Gamma\mathbf{u}(k-1)$

Automation Lab
IIT Bombay

## Preliminaries

$$\text{Cov}\left[\mathbf{x}(k) \mid \mathbf{Y}^{k-1}\right] = \text{E}\left[(\mathbf{x}(k) - \overline{\mathbf{x}}(k))(\mathbf{x}(k) - \overline{\mathbf{x}}(k))^T \mid \mathbf{Y}^{k-1}\right]$$
$$\overline{\mathbf{x}}(k) = \text{E}\left[\mathbf{x}(k) \mid \mathbf{Y}^{k-1}\right]$$

Subtracting the equation governing the mean

$$\hat{\mathbf{x}}(k \mid k-1) = \Phi\hat{\mathbf{x}}(k-1 \mid k-1) + \Gamma\mathbf{u}(k-1)$$

from the equation governing the system dynamics

$$\mathbf{x}(k) = \Phi\mathbf{x}(k-1) + \Gamma\mathbf{u}(k-1) + \mathbf{w}(k-1)$$

we have

$$\varepsilon(k \mid k-1) = \Phi\varepsilon(k-1 \mid k-1) + \mathbf{w}(k-1)$$

Prediction Error
$$\varepsilon(k \mid k-1) \equiv \mathbf{x}(k) - \hat{\mathbf{x}}(k \mid k-1)$$

Estimation Error
$$\varepsilon(k-1 \mid k-1) \equiv \mathbf{x}(k-1) - \hat{\mathbf{x}}(k-1 \mid k-1)$$



Automation Lab
IIT Bombay

## Preliminaries

Update Step

$$\hat{\mathbf{x}}(k \mid k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{L}(k)\mathbf{e}(k)$$
$$\mathbf{e}(k) = \left[\mathbf{y}(k) - \hat{\mathbf{y}}(k \mid k-1)\right]$$

(with an **arbitrary** gain matrix L(k))

where "innovation" $\mathbf{e}(k)$
is related to state estimation error as follows

$$\begin{aligned}\mathbf{e}(k) &= \mathbf{y}(k) - \hat{\mathbf{y}}(k \mid k-1)\\ &= \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k) - \mathbf{C}\hat{\mathbf{x}}(k \mid k-1)\\ &= \mathbf{C}\varepsilon(k \mid k-1) + \mathbf{v}(k)\end{aligned}$$

Prediction and estimation errors are related as follows

$$\hat{\mathbf{x}}(k \mid k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{L}(k)\mathbf{e}(k)$$
$$\Rightarrow \left[\mathbf{x}(k) - \hat{\mathbf{x}}(k \mid k)\right] = \left[\mathbf{x}(k) - \hat{\mathbf{x}}(k \mid k-1)\right] - \mathbf{L}(k)\mathbf{e}(k)$$
$$\Rightarrow \varepsilon(k \mid k) = \left[\mathbf{I} - \mathbf{L}(k)\mathbf{C}\right]\varepsilon(k \mid k-1) - \mathbf{L}(k)\mathbf{v}(k)$$

## Mean Values of Estimation Errors

Error Dynamics

$$\boldsymbol{\varepsilon}(k \mid k-1) = \Phi\boldsymbol{\varepsilon}(k-1 \mid k-1) + \mathbf{w}(k-1)$$
$$\boldsymbol{\varepsilon}(k \mid k) = [\mathbf{I} - \mathbf{L}(k)\mathbf{C}]\boldsymbol{\varepsilon}(k \mid k-1) - \mathbf{L}(k)\mathbf{v}(k)$$

Combining

$$\boldsymbol{\varepsilon}(k \mid k) = [\mathbf{I} - \mathbf{L}(k)\mathbf{C}][\Phi\boldsymbol{\varepsilon}(k-1 \mid k-1) + \mathbf{w}(k-1)] - \mathbf{L}(k)\mathbf{v}(k)$$

Simplifying Assumption 4

Initial State at k = 0 is a Random Variable such that

$$E[\mathbf{x}(0)] = E[\hat{\mathbf{x}}(0 \mid 0)] \quad Cov[\mathbf{x}(0)] = \mathbf{P}(0)$$
$$\Rightarrow E[\mathbf{x}(0) - \hat{\mathbf{x}}(0 \mid 0)] = E[\boldsymbol{\varepsilon}(0 \mid 0)] = \bar{\mathbf{0}}$$





## Mean Values of Estimation Errors

$$E[\boldsymbol{\varepsilon}(1 \mid 1)] = [\mathbf{I} - \mathbf{L}(1)\mathbf{C}]E[\Phi\boldsymbol{\varepsilon}(0 \mid 0) + \mathbf{w}(0)] - \mathbf{L}(1)E[\mathbf{v}(1)] = \bar{\mathbf{0}}$$
$$\Downarrow$$
$$E[\boldsymbol{\varepsilon}(2 \mid 2)] = [\mathbf{I} - \mathbf{L}(2)\mathbf{C}]E[\Phi\boldsymbol{\varepsilon}(1 \mid 1) + \mathbf{w}(1)] - \mathbf{L}(2)E[\mathbf{v}(2)] = \bar{\mathbf{0}}$$

.........

$$\Downarrow$$
$$E[\boldsymbol{\varepsilon}(k \mid k)] = [\mathbf{I} - \mathbf{L}(k)\mathbf{C}]E[\Phi\boldsymbol{\varepsilon}(k-1 \mid k-1) + \mathbf{w}(k-1)] - \mathbf{L}(k)E[\mathbf{v}(k)] = \bar{\mathbf{0}}$$
$$\Downarrow$$
$$\Rightarrow E[\boldsymbol{\varepsilon}(k \mid k-1)] = E[\Phi\boldsymbol{\varepsilon}(k-1 \mid k-1) + \mathbf{w}(k-1)] = \bar{\mathbf{0}}$$

Thus, the proposed linear observer is unbiased

# Estimation Errors: Covariance Matrices

Define

$$\mathbf{P}(k|k-1) \equiv \text{Cov}[\varepsilon(k|k-1)] = \text{E}\left[\varepsilon(k|k-1)\varepsilon(k|k-1)^T\right]$$

$$\mathbf{P}(k-1|k-1) \equiv \text{Cov}[\varepsilon(k-1|k-1)] = \text{E}\left[\varepsilon(k-1|k-1)\varepsilon(k-1|k-1)^T\right]$$

Now

$$\varepsilon(k|k-1)\varepsilon(k|k-1)^T = [\Phi\varepsilon(k-1|k-1) + \mathbf{w}(k-1)][\Phi\varepsilon(k-1|k-1) + \mathbf{w}(k-1)]^T$$

Taking expectation on both the sides and noting

$\varepsilon(k-1|k-1)$ and $\mathbf{w}(k-1)$ are uncorrelated

i.e. $\text{E}\left[\varepsilon(k-1|k-1)\mathbf{w}(k-1)^T\right] = \bar{\mathbf{0}}$

it follows that

$$\mathbf{P}(k|k-1) = \Phi\mathbf{P}(k-1|k-1)\Phi^T + \mathbf{Q}$$

(Recursive equation for update of prediction covariance)





# Prediction Error

The innovation

$$\begin{aligned}\mathbf{e}(k) &= \mathbf{y}(k) - \hat{\mathbf{y}}(k|k-1) \\ &= \mathbf{C}\varepsilon(k|k-1) + \mathbf{v}(k) \\ &= \mathbf{C}[\Phi\varepsilon(k-1|k-1) + \mathbf{w}(k-1)] + \mathbf{v}(k)\end{aligned}$$

contains information about $\mathbf{w}(k-1)$ and $\mathbf{v}(k)$

It is desired to compensate the state estimate for w(k) while filtering v(k) out

Update Step can be viewed as

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \hat{\mathbf{w}}(k-1|k)$$

$$\hat{\mathbf{w}}(k-1|k) = \mathbf{L}(k)\mathbf{e}(k)$$

$\mathbf{L}(k)$ : decides the "portion of $\mathbf{e}(k)$" used for disturbance compensation.

## Means and Covariance of Errors

Mean of the innovation

$$E[\mathbf{e}(k)] = \mathbf{C}E[\boldsymbol{\varepsilon}(k \mid k-1)] + E[\mathbf{v}(k)] = \bar{\mathbf{0}}$$

Covariance of Innovations

$$\begin{aligned}\mathbf{P}_e(k) = \mathsf{E}\left[\mathbf{e}(k)\mathbf{e}(k)^T\right] &= \mathsf{E}\left[(\mathbf{C}\boldsymbol{\varepsilon}(k \mid k-1) + \mathbf{v}(k))(\mathbf{C}\boldsymbol{\varepsilon}(k \mid k-1) + \mathbf{v}(k))^T\right] \\ &= \mathbf{C}\mathsf{Cov}[\boldsymbol{\varepsilon}(k \mid k-1)]\mathbf{C}^T + \mathsf{Cov}[\mathbf{v}(k)]; \\ &= \mathbf{C}\mathbf{P}(k \mid k-1)\mathbf{C}^T + \mathbf{R}\end{aligned}$$

Estimation Error

$$\boldsymbol{\varepsilon}(k \mid k) = \boldsymbol{\varepsilon}(k \mid k-1) - \mathbf{L}(k)\mathbf{e}(k)$$

$$\Rightarrow E[\boldsymbol{\varepsilon}(k \mid k)] = E[\boldsymbol{\varepsilon}(k \mid k-1) - \mathbf{L}(k)\mathbf{e}(k)] = \bar{\mathbf{0}}$$





## Means and Covariance of Errors

Estimation Error

$$\begin{aligned}\mathsf{Cov}[\boldsymbol{\varepsilon}(k \mid k)] &= \mathsf{Cov}[\boldsymbol{\varepsilon}(k \mid k-1)] + \mathbf{L}(k)\mathsf{Cov}[\mathbf{e}(k)]\mathbf{L}(k)^T \\ &\quad - \mathsf{E}\left[\boldsymbol{\varepsilon}(k \mid k-1)\mathbf{e}(k)^T\right]\mathbf{L}(k)^T - \mathbf{L}(k)\mathsf{E}\left[\mathbf{e}(k)\boldsymbol{\varepsilon}(k \mid k-1)^T\right]\end{aligned}$$

Defining

$$\mathbf{P}_{\varepsilon e}(k) \equiv \mathsf{E}\left[\boldsymbol{\varepsilon}(k \mid k-1)\mathbf{e}(k)^T\right]$$

$$\mathbf{P}_{\varepsilon e}(k) = \mathsf{E}\left[\boldsymbol{\varepsilon}(k \mid k-1)(\mathbf{C}\boldsymbol{\varepsilon}(k \mid k-1) + \mathbf{v}(k))^T\right] = \mathbf{P}(k \mid k-1)\mathbf{C}^T$$

we have

$$\mathbf{P}(k \mid k) = \mathbf{P}(k \mid k-1) + \mathbf{L}(k)\mathbf{P}_e(k)\mathbf{L}(k)^T - \mathbf{L}(k)\mathbf{P}_{\varepsilon e}(k)^T - \mathbf{P}_{\varepsilon e}(k)\mathbf{L}(k)^T$$

## Minimum Variance Design

Find gain matrix L(k) such that estimation error variance in minimum

Minimum Variance Design

$$\underset{\mathbf{L}(k)}{\text{Min}} \; tr[\mathbf{P}(k \mid k)]$$

Necessary Condition for Optimality

$$\frac{\partial tr[\mathbf{P}(k \mid k)]}{\partial \mathbf{L}(k)} = [\mathbf{0}]$$

Note : Properties of Trace of a Matrix

$$tr(\mathbf{C} + \mathbf{D}) = tr(\mathbf{C}) + tr(\mathbf{D})$$

$$tr(\mathbf{C}) = tr(\mathbf{C}^T)$$





## Matrix Calculus

Consider $\mathbf{X}$ ($m \times n$ matrix) and $y = f(\mathbf{X})$, a scalar function of $\mathbf{X}$

$$\frac{\partial y}{\partial \mathbf{X}} \equiv \begin{bmatrix} \frac{\partial y}{\partial x_{11}} & \frac{\partial y}{\partial x_{12}} & \cdots & \frac{\partial y}{\partial x_{1n}} \\ \frac{\partial y}{\partial x_{21}} & \frac{\partial y}{\partial x_{22}} & \cdots & \frac{\partial y}{\partial x_{2n}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial y}{\partial x_{m1}} & \frac{\partial y}{\partial x_{m2}} & \cdots & \frac{\partial y}{\partial x_{mn}} \end{bmatrix}$$

Rules of Differentaition

$$\frac{\partial tr[\mathbf{AX}]}{\partial \mathbf{X}} = \frac{\partial tr[\mathbf{XA}]}{\partial \mathbf{X}} = \mathbf{A}^T$$

Let $\mathbf{B}$ represent a symmetric matrix

$$\frac{\partial tr[\mathbf{XBX}^T]}{\partial \mathbf{X}} = 2\mathbf{XB}$$

## Minimum Variance Observer

$$\frac{\partial tr\left[\mathbf{L}(k)\mathbf{P}_e(k)\mathbf{L}(k)^T\right]}{\partial \mathbf{L}(k)} = 2\mathbf{L}(k)\mathbf{P}_e(k)$$

$$\frac{\partial tr\left[\mathbf{L}(k)\mathbf{P}_{\varepsilon e}(k)^T + \mathbf{P}_{\varepsilon e}(k)\mathbf{L}(k)^T\right]}{\partial \mathbf{L}(k)} = 2\frac{\partial tr\left[\mathbf{L}(k)\mathbf{P}_{\varepsilon e}(k)^T\right]}{\partial \mathbf{L}(k)} = 2\mathbf{P}_{\varepsilon e}(k)$$

Thus, it follows that

$$\frac{\partial tr\left[\mathbf{P}(k \mid k)\right]}{\partial \mathbf{L}(k)} = 2\mathbf{L}(k)\mathbf{P}_e(k) - 2\mathbf{P}_{\varepsilon e}(k) = [\mathbf{0}]$$

$$\Rightarrow \mathbf{L}^*(k) = \left[\mathbf{L}(k)\right]_{OPT} = \mathbf{P}_{\varepsilon e}(k)\mathbf{P}_e(k)^{-1}$$

$$\Rightarrow \left[\mathbf{P}(k \mid k)\right]_{OPT} = \mathbf{P}(k \mid k-1) - \mathbf{L}^*(k)\mathbf{P}_e(k)^{-1}\mathbf{L}^*(k)^T$$
$$= \left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]\mathbf{P}(k \mid k-1)$$





## Kalman Filter: Summary

**Prediction**

$$\hat{\mathbf{x}}(k \mid k-1) = \Phi\hat{\mathbf{x}}(k-1 \mid k-1) + \Gamma\mathbf{u}(k-1)$$

$$\mathbf{P}(k \mid k-1) = \Phi\mathbf{P}(k-1 \mid k-1)\Phi^T + \mathbf{Q}$$

**Kalman Gain Computation**

$$\mathbf{L}^*(k) = \mathbf{P}_{\varepsilon e}(k)\mathbf{P}_e(k)^{-1}$$
$$= \mathbf{P}(k \mid k-1)\mathbf{C}^T\left[\mathbf{C}\mathbf{P}(k \mid k-1)\mathbf{C}^T + \mathbf{R}\right]^{-1}$$

**Update**

$$\mathbf{e}(k) = \left[\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k \mid k-1)\right]$$

$$\hat{\mathbf{x}}(k \mid k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{L}^*(k)\mathbf{e}(k)$$

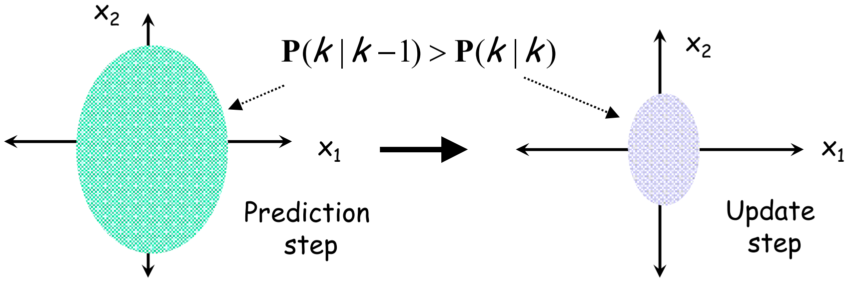
$$\mathbf{P}(k \mid k) = \left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]\mathbf{P}(k \mid k-1)$$

Automation Lab
IIT Bombay

# Interpretations

Covariance matrix quantifies uncertainty associated with the estimated state

$$\mathbf{P}(k \mid k-1) > \mathbf{P}(k \mid k)$$




$$[\mathbf{P}(k \mid k)] - \mathbf{P}(k \mid k-1) = -\mathbf{L}^*(k)\mathbf{P}_e(k)^{-1}\mathbf{L}^*(k)^T$$

$\mathbf{L}^*(k)\mathbf{P}_e(k)^{-1}\mathbf{L}^*(k)^T$ is $+$ve definite matrix

$$\Rightarrow \mathbf{P}(k \mid k) < \mathbf{P}(k \mid k-1)$$

Update step reduces covariance associated with the estimate

Automation Lab
IIT Bombay

# Gaussian Distributions

Why study multivariate Gaussian distribution?

- From Central Limit Theorem, it follows that sum of many independent and equally distributed random variables can be well approximated by Gaussian distribution. If unknown disturbances are assumed to be arising from many independent physical sources, then Gaussian distribution is appropriate for modeling their behavior
- Attractive mathematical properties: linear transformations of Gaussian distributions are still Gaussian distributed.
- For Gaussian distributed random variables, optimal estimated have a simple form.

## Multivariate Gaussian Distribution

Consider random variable $\mathbf{x} \in R^n$

Let $\overline{\mathbf{x}} \in R^n$ represent mean of $\mathbf{x}$ and

$\mathbf{P}$ represent $+$ve definite covarince matrix

$$p(\mathbf{x}) = \mathrm{N}(\overline{\mathbf{x}}, \mathbf{P}) \equiv \frac{1}{(2\pi)^{n/2}\sqrt{\det(\mathbf{P})}} \exp\left[-(\mathbf{x}-\overline{\mathbf{x}})^T \mathbf{P}^{-1}(\mathbf{x}-\overline{\mathbf{x}})\right]$$

Characterized completely by mean $(\overline{\mathbf{x}})$ and covariance $(\mathbf{P})$

If $\mathbf{x} \sim \mathrm{N}(\overline{\mathbf{x}}, \mathbf{P})$ is a random vector and $\mathbf{A}$ is a $(r \times n)$ matrix of rank r and $\mathbf{b}$ is a $(r \times 1)$ vector, then

$$\mathbf{z} = \mathbf{A}\mathbf{x} + \mathbf{b}$$

is also a Gaussian distributed $\mathbf{z} \sim \mathrm{N}(\mathbf{A}\overline{\mathbf{x}} + +\mathbf{b}, \mathbf{A}\mathbf{P}\mathbf{A}^T)$

Consequence: Linear filtering of a Gaussian distributed Inputs will generate a Gaussian distributed output



## Multivariate Gaussian Distribution

Consider two random variable $\mathbf{x}$ and $\mathbf{z}$

Let $\overline{\mathbf{x}}$ and $\overline{\mathbf{z}}$ represent means of $\mathbf{x}$ and $\mathbf{z}$, respectively

Random vector $\mathbf{x}$ and $\mathbf{z}$ are said to be uncorrelated if

$$E\left[(\mathbf{x}-\overline{\mathbf{x}})(\mathbf{z}-\overline{\mathbf{z}})^T\right] = [\mathbf{0}]$$

Random vector $\mathbf{x}$ and $\mathbf{z}$ are said to be independent if

$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x})p(\mathbf{z})$$

If vectors $\mathbf{x}$ and $\mathbf{z}$ are independent

$\Rightarrow \mathbf{x}$ and $\mathbf{z}$ are uncorrelated

If vectors $\mathbf{x}$ and $\mathbf{z}$ are uncorrelated and Gaussian

$\Rightarrow \mathbf{x}$ and $\mathbf{z}$ are independent

## Gaussian Noise and KF

Automation Lab
IIT Bombay

Let the process noise, the measurement noise and the initial state have Gaussian normal distributions, i.e.

$$\mathbf{w} \sim \mathrm{N}(\bar{\mathbf{0}}, \mathbf{Q}), \mathbf{v} \sim \mathrm{N}(\bar{\mathbf{0}}, \mathbf{R}) \text{ and } \mathbf{x}(0) \sim \mathrm{N}(\hat{\mathbf{x}}(0|0), \mathbf{P}(0))$$

then, from the properties of Gaussian distributions it follows that

$$p\left[\mathbf{x}(k) \mid \mathbf{Y}^{k-1}\right] \sim \mathrm{N}(\hat{\mathbf{x}}(k|k-1), \mathbf{P}(k|k-1))$$

and

$$p\left[\mathbf{x}(k) \mid \mathbf{Y}^{k}\right] \sim \mathrm{N}(\hat{\mathbf{x}}(k|k), \mathbf{P}(k|k))$$

Also, the innovation sequence is a Gaussian stochastic process

$$p\left[e(k) \mid \mathbf{Y}^{k}\right] \sim \mathrm{N}(\bar{0}, \mathbf{P}_{ee}(k))$$
$$\mathbf{P}_{ee}(k) = \mathbf{C}\mathbf{P}(k|k-1)\mathbf{C}^{T} + \mathbf{R}$$

## Gaussian Noise and KF

Automation Lab
IIT Bombay

When the process noise, the measurement noise and the initial state have Gaussian normal distributions, it can be shown that

$\hat{\mathbf{x}}(k|k)$ generated using Kalman filter maximizes $p\left[\mathbf{x}(k) \mid \mathbf{Y}^{k}\right]$
i.e. it is a "maximum a posteriori" or MAP estimate

$\hat{\mathbf{x}}(k|k)$ generated using Kalman filter maximizes log likelihood function i.e.

$$\log\left(p\left[\mathbf{x}(k) \mid \mathbf{Y}^{k}\right]\right) = \log\left(p\left[\mathbf{x}(k), \mathbf{Y}^{k}\right]\right) - \log\left(p\left[\mathbf{Y}^{k}\right]\right)$$

In other words, KF generates solution that minimizes

$$\hat{\mathbf{x}}(k|k) = \underset{\mathbf{x}(k)}{\mathrm{Min}} \left\|\mathbf{y}(k) - \mathbf{C}\mathbf{x}(k)\right\|^{2}_{\mathbf{R}^{-1}} + \left\|\mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1)\right\|^{2}_{\mathbf{P}(k|k-1)^{-1}}$$

Thus, Kalman Filter is a "Maximum Likelihood" (ML) Estimator

Automation Lab
IIT Bombay

## Kalman Filter: Advantages

- Generates the maximum likelihood (ML) and maximum a posteriori (MAP) estimates of the states when noises are Gaussian
- Kalman filter is the minimum variance estimator
- Requires only first and second moments of conditional densities of the states and the innovations
- Relatively easy to adapt to multi-rate and irregular sampling scenario
- Stability can be established using Lyapunov's 2'nd method (see Appendix)



Automation Lab
IIT Bombay

## Stationary Kalman Filter

Thus, as $k \to \infty$,

$\mathbf{P}(k|k-1) \to \widetilde{\mathbf{P}}_\infty, \; \mathbf{P}(k|k) \to \mathbf{P}_\infty$ and $\mathbf{L}^*(k) \to \mathbf{L}^*_\infty$

**Stationary Kalman Gain Computation using Algebraic Riccati Equation (ARE)**

$$\widetilde{\mathbf{P}}_\infty = \Phi \mathbf{P}_\infty \Phi^T + \mathbf{Q}$$

$$\mathbf{L}^*_\infty = \widetilde{\mathbf{P}}_\infty \mathbf{C}^T \left[ \mathbf{C} \widetilde{\mathbf{P}}_\infty \mathbf{C}^T + \mathbf{R} \right]^{-1}$$

$$\mathbf{P}_\infty = \left[ \mathbf{I} - \mathbf{L}^*_\infty \mathbf{C} \right] \widetilde{\mathbf{P}}_\infty$$

**Prediction and Update**

$$\hat{\mathbf{x}}(k|k-1) = \Phi \hat{\mathbf{x}}(k-1|k-1) + \Gamma \mathbf{u}(k-1)$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{L}^*_\infty \left[ \mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k|k-1) \right]$$

Automation Lab
IIT Bombay

## Example: Quadruple Tank System

True Initial State

$$\mathbf{x}(0) = \begin{bmatrix} 2 & -2 & 2 & -2 \end{bmatrix}^T$$

Kalman Filter Parameters

$$\text{Cov}[\mathbf{w}(k)] = \mathbf{Q} = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \quad \text{Cov}[\mathbf{v}(k)] = \mathbf{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

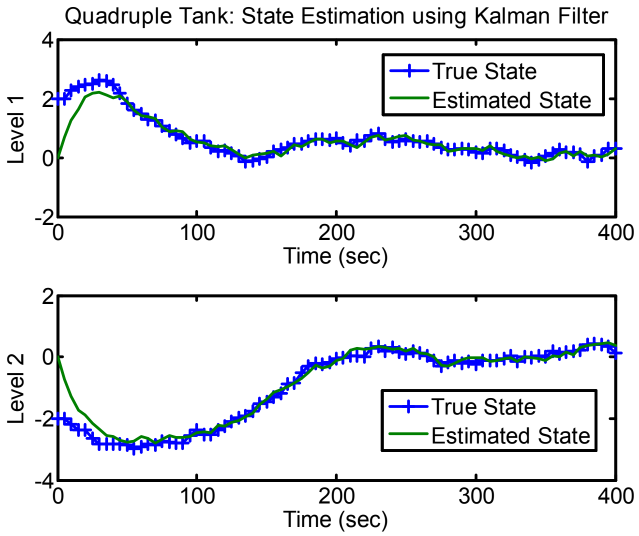
$$\hat{\mathbf{x}}(0|0) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \quad \text{and} \quad \mathbf{P}(0|0) = \mathbf{Q}$$

Stationary Kalman Filter Gain

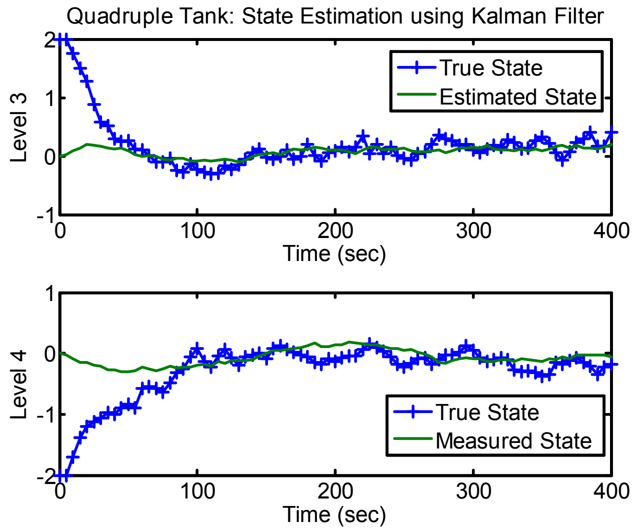
$$\mathbf{L}_\infty^* = \begin{bmatrix} 0.7825 & 0 \\ 0 & 0.7921 \\ 0.2212 & 0 \\ 0 & 0.2365 \end{bmatrix} \quad \text{Eigen values of } (\mathbf{I} - \mathbf{L}_\infty^* \mathbf{C}) = \begin{bmatrix} 0.6337 \\ 0.7195 \\ 0.6196 \\ 0.7806 \end{bmatrix}$$

Automation Lab
IIT Bombay

## Example: Quadruple Tank System



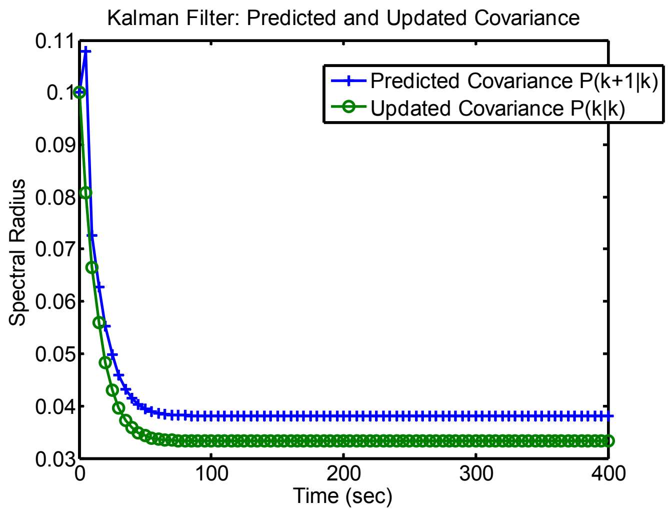
# Example: Quadruple Tank System




Automation Lab
IIT Bombay

# Example: Quadruple Tank System

$$\rho[\mathbf{P}(k|k-1)] > \rho[\mathbf{P}(k|k)]$$



Automation Lab
IIT Bombay

## Kalman Predictor: Summary

Initailization Step : Initail mean, $\hat{X}(0\,|\,-1)$,
Initial Covariance $P(0\,|\,-1)$

At Instant 'k'

Step 1 : Compute Kalman Gain $\mathbf{L}_p^*(k)$

$$\mathbf{L}_p^*(k) = \Phi\mathbf{P}(k \mid k-1)\mathbf{C}^T\left[\mathbf{R} + \mathbf{C}\mathbf{P}(k \mid k-1)\mathbf{C}^T\right]^{-1}$$

Step 2 : Recursive Prediction Estimator

$$\mathbf{e}(k) = \left[\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k \mid k-1)\right]$$

$$\hat{\mathbf{x}}(k+1 \mid k) = \Phi\hat{\mathbf{x}}(k \mid k-1) + \Gamma\mathbf{u}(k) + \mathbf{L}_p^*(k)\mathbf{e}(k)$$

Step 3 : Update Covariance matrix

$$\mathbf{P}(k+1 \mid k) = \Phi\mathbf{P}(k \mid k-1)\Phi^T + \mathbf{Q} - L_p(k)\mathbf{C}\mathbf{P}(k \mid k-1)\Phi^T$$



Automation Lab
IIT Bombay

## Kalman Predictor: CSTR Example

$$x(k+1) = \begin{bmatrix} 0.185 & -0.01 \\ 73.49 & 1.33 \end{bmatrix} x(k) + \begin{bmatrix} 0.005 & 0.13 \\ -0.73 & -1.8 \end{bmatrix} u(k) + \begin{bmatrix} 0.06 \\ 3.9 \end{bmatrix} d(k)$$

$$y(k) = [0 \quad 1]x(k) + v(k)$$

$$Q_d = (0.05)^2$$

$$Q = \Psi Q_d \Psi^T = (0.05)^2 \begin{bmatrix} 0.0036 & 0.234 \\ 0.234 & 15.21 \end{bmatrix}$$

$$Cov\{v(k)\} = R = (0.5)^2$$

Apriori estimate of initial state

$$x(0 \mid -1) = [0 \quad 0]^T$$

Initial State Covariance Estimate (selected arbitrarily large)

$$P(0 \mid -1) = 1 \times 10^3 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

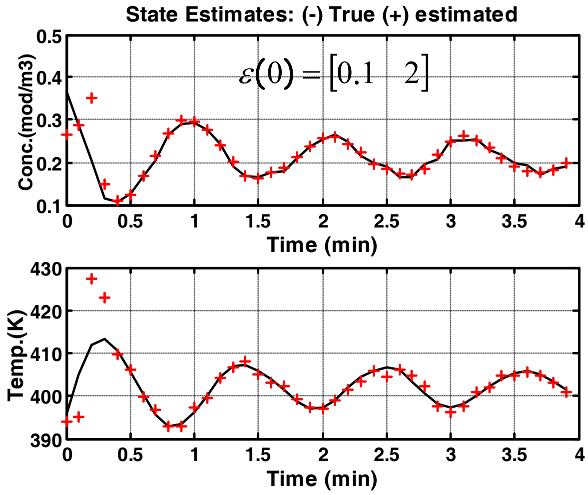
After about 20 iterations, Kalman (Predictor) Gain settles to following steady State Values

$$L_{P\infty} = [-0.00516 \quad 0.696]^T$$

Automation Lab
IIT Bombay

# CSTR Example: Kalman Predictor

## Linear Plant- Linear Observer

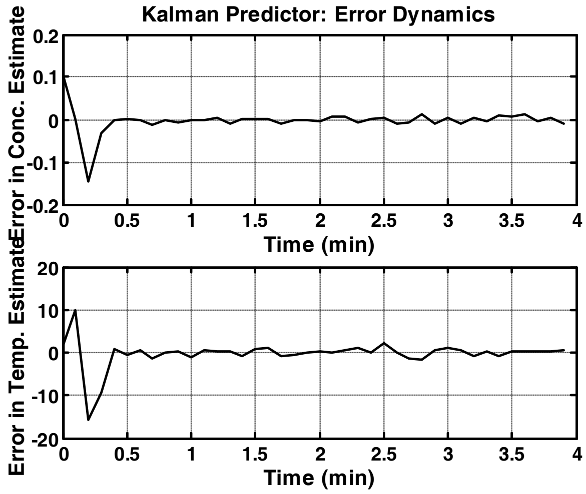





Automation Lab
IIT Bombay

# CSTR Example: Kalman Predictor

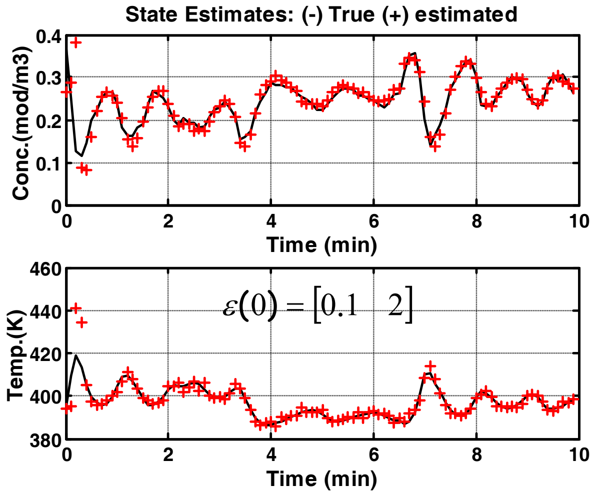
## Linear Plant- Linear Observer



Automation Lab
IIT Bombay

# CSTR Example: Kalman Predictor

## Non-Linear Plant- Linear Observer

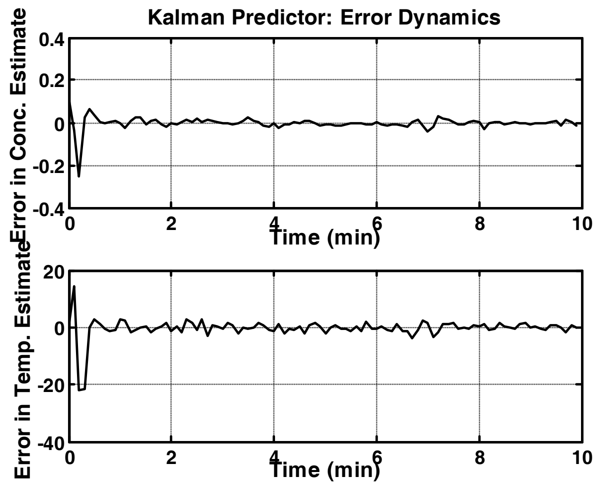





Automation Lab
IIT Bombay

# CSTR Example: Kalman Predictor

## Non-Linear Plant- Linear Observer



## "Steady State" Kalman Predictor

As $k \to \infty$, under weak conditions
the optimal estimator will be time invariant

Theorem

Assume pair $(\Phi, \sqrt{\mathbf{Q}})$ is stabilizable and the pair $(\Phi, \mathbf{C})$ is detectable
Then the solution of the Riccati equation $\mathbf{P}(k|k-1) \to \mathbf{P}_\infty > 0$
where $\mathbf{P}_\infty$ denotes solution of the Algebraic Riccati Equation

$$\mathbf{P}_\infty = \Phi \mathbf{P}_\infty \Phi^T + \mathbf{Q} - \mathbf{L}^*_{p,\infty} \mathbf{C} \mathbf{P}_\infty \Phi^T$$

$$\mathbf{L}^*_{p,\infty} = \Phi \mathbf{P}_\infty \mathbf{C}^T \left[\mathbf{R} + \mathbf{C}\mathbf{P}_\infty \mathbf{C}^T\right]^{-1}$$

Lemma

Assume pair $(\Phi, \sqrt{\mathbf{Q}})$ is controllable and $\mathbf{R}$ is non-singular
Then all eigen values of $(\Phi - \mathbf{L}^*_{p,\infty}\mathbf{C})$ are inside the unit circle.
(Dynamics governing the estimation error $\varepsilon(k|k-1)$ is asymptotically stable)





## "Steady State" Kalman Predictor

As $k \to \infty$, $\mathbf{P}(k|k-1) \to \mathbf{P}_\infty$
where $\mathbf{P}_\infty$ denotes solution of the Algebraic Riccati Equation

$$\mathbf{P}_\infty = \Phi \mathbf{P}_\infty \Phi^T + \mathbf{Q} - \mathbf{L}^*_{p,\infty} \mathbf{C} \mathbf{P}_\infty \Phi^T$$

$$\mathbf{L}^*_{p,\infty} = \Phi \mathbf{P}_\infty \mathbf{C}^T \left[\mathbf{R} + \mathbf{C}\mathbf{P}_\infty \mathbf{C}^T\right]^{-1}$$

Recursive Prediction Estimator

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k|k-1)$$

$$\hat{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k-1) + \Gamma\mathbf{u}(k) + \mathbf{L}^*_{p,\infty}\mathbf{e}(k)$$

The above "steady state observer" can be written as

$$\hat{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k-1) + \Gamma\mathbf{u}(k) + \mathbf{L}^*_{p,\infty}\mathbf{e}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\hat{\mathbf{x}}(k|k-1) + \mathbf{e}(k)$$

$E[\mathbf{e}(k)] = \bar{\mathbf{0}}$ and $Cov[\mathbf{e}(k)] = \mathbf{R} + \mathbf{C}\mathbf{P}_\infty\mathbf{C}^T$

## Connection with Time Series Models

Stationary form of Kalman predictor
is also known as **Innovation form of State Space** Model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{e}(k)$$
$$E[\mathbf{e}(k)] = \overline{\mathbf{0}} \text{ and } Cov[\mathbf{e}(k)] = \mathbf{P}_e$$

Taking q-transform, we can write

$$\mathbf{y}(k) = \mathbf{G}(q)\mathbf{u}(k) + \mathbf{H}(q)\mathbf{e}(k)$$
$$\mathbf{G}(q) = \mathbf{C}[q\mathbf{I} - \Phi]^{-1}\Gamma \; ; \; \mathbf{H}(q) = \mathbf{I} + \mathbf{C}[q\mathbf{I} - \Phi]^{-1}\mathbf{L}$$

Thus, stationary form of Kalman predictor is equivalent to **ARMAX type time series model**





## Connection with Time Series Models

Conversely, a time series model (ARX/ARMAX/BJ) estimated from the input output data

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

through a cannonical state space realization,
can be expressed as an innovation form of
the state space model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{e}(k)$$
$$E[\mathbf{e}(k)] = \overline{\mathbf{0}} \text{ and } Cov[\mathbf{e}(k)] = \mathbf{P}_e$$

Thus, identifying an ARX/ARMAX/BJ model is equivalent to identifying Stationary form of Kalman predictor

## Connection with Time Series Models

- Thus, stationary form of Kalman predictor can be identified directly from input output data using ARX / ARMAX / Box-Jenkins parameterization and converting into state space realization.
- Advantage: No need to model the state noise, **w**(k), and the measurement noise, **v**(k)
- Disadvantage: states do not have physical meaning

State realization of a ARMAX/BJ model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \mathbf{w}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$
$$\mathbf{w}(k) = \mathbf{L}\mathbf{e}(k) \quad \text{and} \quad \mathbf{v}(k) = \mathbf{e}(k)$$

$$E[\mathbf{w}(k)] = E[\mathbf{e}(k)] = \bar{\mathbf{0}}$$
$$Cov[\mathbf{w}(k)] = \mathbf{L}\mathbf{P}_e\mathbf{L}^T \quad \text{and} \quad Cov[\mathbf{v}(k)] = \mathbf{P}_e$$
$$Cov[\mathbf{w}(k), \mathbf{v}(k)] = E\left[\mathbf{w}(k)\mathbf{v}(k)^T\right] = \mathbf{L}\mathbf{P}_e$$



## Connection with Time Series Models

Thus, given a state realization of a ARMAX/BJ model

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{e}(k)$$

we directly can develop a state estimator as follows

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k)$$
$$\hat{\mathbf{x}}(k+1) = \Phi\hat{\mathbf{x}}(k) + \Gamma\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k)$$

and further use for implementing

a state feedback control law

$$\mathbf{u}(k) = \mathbf{G}\left(\mathbf{x}_s(k) - \hat{\mathbf{x}}(k)\right)$$

Note: The states are observable by construction

## Dealing with Non-stationary Disturbances

Augment state space model with extra artificial states (equal to no. of outputs), which behave as integrated white noise sequence and can capture drifting

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k) + \Gamma_\eta \boldsymbol{\eta}(k) + \mathbf{w}(k)$$
$$\boldsymbol{\eta}(k+1) = \boldsymbol{\eta}(k) + \mathbf{w}_\eta(k)$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k)$$

State Noise Covarance : $\begin{bmatrix} \mathbf{Q} & [0] \\ [0] & \mathbf{Q}_\eta \end{bmatrix}$

Tuning Parameter

Choice of $\Gamma_\eta$ matrix

Bias in Input Model : $\Gamma_\eta = \Gamma$

Mean shift in diaturbance : $\Gamma_\eta = \Psi$

Design Kalman Filter / Predictor using augmented model

Fast changing disturbance: use high values of co-variance $Q_\eta$





## Notation

**Mechanistic Model**

State Dynamics

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{d},\theta)$$

Measurement Model

$$\mathbf{y} = H[\mathbf{x}]$$

**Assumptions**

Manipulated inputs and piecewise constant

$\mathbf{u}(t) = \mathbf{u}(k)$ for $t_k \le t < t_{k+1}$

Unmeasured disturbances are random fluctuations in the neighborhood of mean value

$\mathbf{d}(t) = \bar{\mathbf{d}} + \mathbf{w}(k)$

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{f}(\mathbf{x}(\tau),\mathbf{u}(k),\bar{\mathbf{d}} + \mathbf{w}(k),\boldsymbol{\theta})d\tau$$

$t_k = kT \quad t_{k+1} = (k+1)T \quad T: \textit{Sampling Time}$

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \int_{t_k}^{t_{k+1}} \mathbf{f}(\mathbf{x}(\tau),\mathbf{u}(k),\bar{\mathbf{d}} + \mathbf{w}(k),\theta)d\tau$$
$$= F[\mathbf{x}(k),\mathbf{u}(k),\mathbf{w}(k),\boldsymbol{\theta}]$$

**Control Relevant Discrete Time Representation**

$$\mathbf{x}(k+1) = F[\mathbf{x}(k),\mathbf{u}(k),\mathbf{w}(k),\boldsymbol{\theta}]$$
$$\mathbf{y}(k) = H[\mathbf{x}(k)]$$

# Extended Kalman Filter: Summary

**Successive Local Linearization**

1. Compute local Jacobian matrices

$$\mathbf{A}(t_{k-1}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right]_{(\bullet)} \quad \text{and} \quad \mathbf{B_d}(t_{k-1}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{d}}\right]_{(\bullet)}$$

$$\text{at } (\bullet) \equiv \left[\hat{\mathbf{x}}(k-1\,|\,k-1), \mathbf{u}(k-1), \bar{\mathbf{0}}\right]$$

2. Compute matrices $\mathbf{\Phi}(\mathrm{k},\mathrm{k}-1) = \exp\left[\mathbf{A}(t_{k-1})T\right]$ and

$$\mathbf{\Gamma}_d(k,k-1) = \left[\mathbf{\Phi}(\mathrm{k},\mathrm{k}-1) - \mathbf{I}\right]\left[\mathbf{A}(t_{k-1})\right]^{-1}\mathbf{B}_d(t_{k-1})$$

**Prediction Step**

$$\hat{\mathbf{x}}(k\,|\,k-1) = \mathbf{F}\left[\hat{\mathbf{x}}(k-1\,|\,k-1), \mathbf{u}(k-1), \bar{0}\right]$$

$$\mathbf{P}(k\,|\,k-1) = \Phi(k,k-1)\mathbf{P}(k-1\,|\,k-1)\Phi(k,k-1)^T + \Gamma_d(k,k-1)\mathbf{Q}_d\Gamma_d(k,k-1)^T$$





# Extended Kalman Filter: Summary

**Kalman Gain Computation**

$$\text{Compute } \mathbf{C}(k) = \left[\frac{\partial H}{\partial \mathbf{x}}\right]_{(\bullet)} \quad \text{at } \hat{\mathbf{x}}(k\,|\,k-1) \text{ and}$$

$$\mathbf{L}(k) = \mathbf{P}_{\varepsilon e}(k)\mathbf{P}_e(k)^{-1}$$

$$= \mathbf{P}(k\,|\,k-1)\mathbf{C}(k)^T\left[\mathbf{C}(k)\mathbf{P}(k\,|\,k-1)\mathbf{C}(k)^T + \mathbf{R}\right]^{-1}$$

**Update Step**

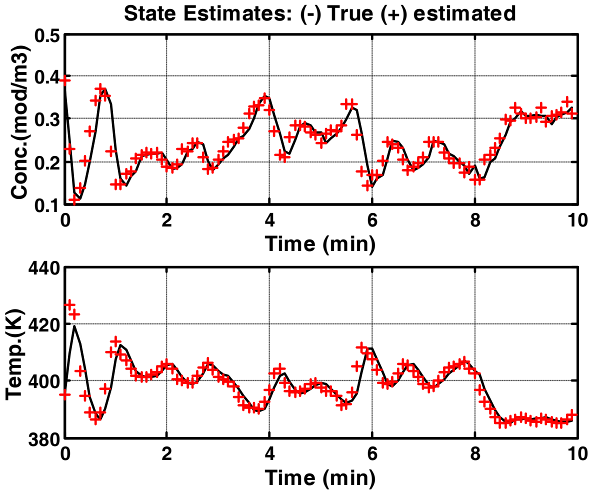
$$\mathbf{e}(k) = \left[\mathbf{y}(k) - H\left(\hat{\mathbf{x}}(k\,|\,k-1)\right)\right]$$

$$\hat{\mathbf{x}}(k\,|\,k) = \hat{\mathbf{x}}(k\,|\,k-1) + \mathbf{L}(k)\mathbf{e}(k)$$

$$\mathbf{P}(k\,|\,k) = \left[\mathbf{I} - \mathbf{L}(k)\mathbf{C}(k)\right]\mathbf{P}(k\,|\,k-1)$$

Approach originally derived for state estimation of a discrete linear system used for state estimation of a nonlinear system

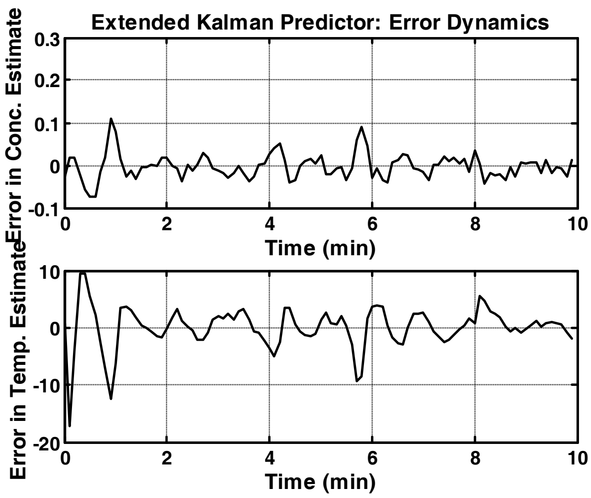
# EKF : CSTR Example




Automation Lab
IIT Bombay

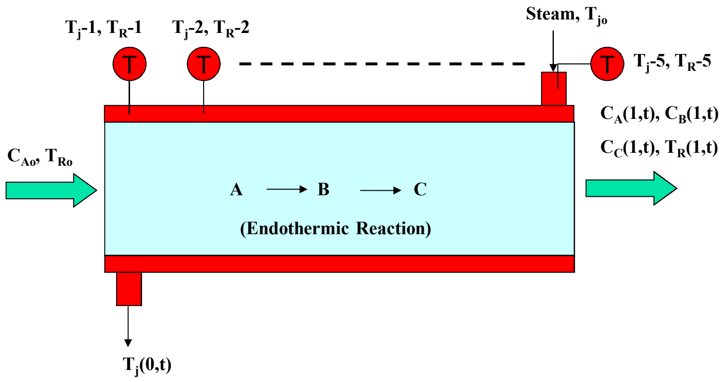
# EKF : CSTR Example

## Estimation Error Dynamics



Automation Lab
IIT Bombay

## EKF : Plug Flow (Tubular) Reactor (PFR)




State Estimation Problem
Estimate concentration profile inside the reactor using few temperature measurements along the length

Automation Lab
IIT Bombay

## Fixed Bed Reactor

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

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

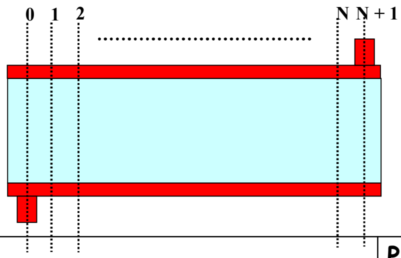
$$\frac{\partial C_B}{\partial t} = -v_l \frac{\partial C_B}{\partial z} + k_{10} e^{-E_1/RT_r} C_A - k_{20} e^{-E_2/RT_r} C_B$$ ........Product B

- **Energy Balances**

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

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

# PDE To ODE Model (Finite Differencing)




| | Plant | Model |
|---|---|---|
| No. of internal discretization points | 19 | 4 |
| No. of states | 80 | 20 |
| No. of jacket side temp. measurements | 3 | 3 |
| No. of reactor side temp. measurements | 3 | 3 |

Automation Lab
IIT Bombay

# State Estimation using EKF

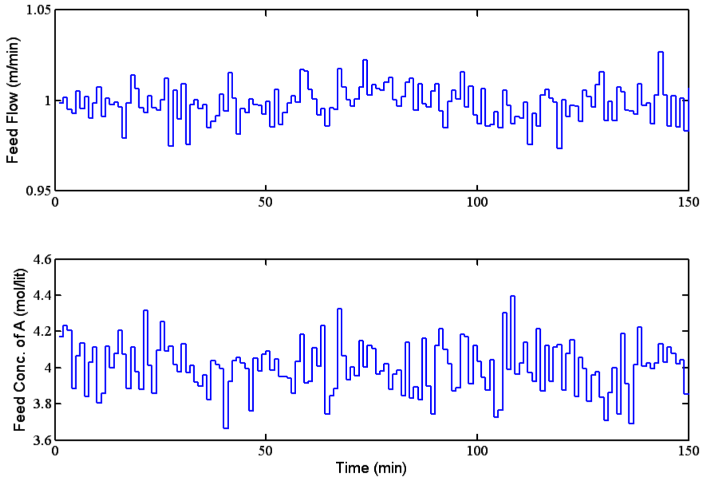
## Simulation Parameters

| Variable | Nominal Value | Fluctuations added |
|---|---|---|
| Feed Flow | 1 m/min | 0.01 m/min |
| Feed Concentration | 4 mol/lit | 0.14 mol/lit |
| Temperature measurements | - | 0.4 K |
| Steam flow rate | 1 m/min | - |

- Performance of EKF under the effect of feed flow and feed concentration fluctuations was studied
- The estimated concentration approaches the true concentration within 5 minutes

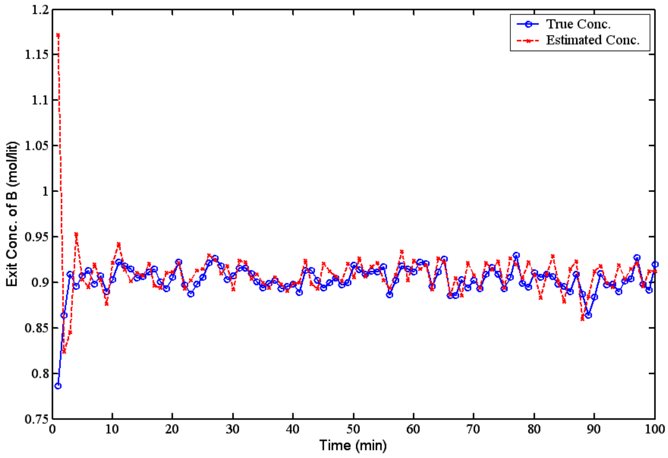
Automation Lab
IIT Bombay

## Fluctuations in Feed Flow and Feed Concentration



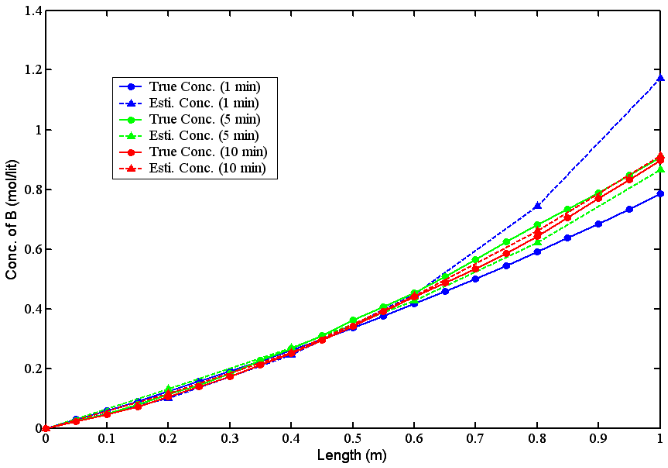

Automation Lab
IIT Bombay

## Actual and Estimated Exit Concentration of B



Automation Lab
IIT Bombay

## Simulation Result: Concentration profiles of product B at different time instants




Automation Lab
IIT Bombay

## State and Parameter Estimation

Estimation of deterministic changes in unmeasured disturbances / model parameters

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \int_{kT}^{(k+1)T} \mathbf{f}\left[\mathbf{x}(\tau), \mathbf{u}(k), \overline{\mathbf{d}} + \mathbf{w}(k), \boldsymbol{\theta}(k)\right] d\tau$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \mathbf{w}_{\boldsymbol{\theta}}(k)$$

$$\mathbf{y}(k) = \mathbf{H}\left[\mathbf{x}(k)\right] + \mathbf{v}(k)$$

Augment the model with fictitious discrete evolution equation

$\boldsymbol{\theta}(k)$ : Vector containing slow drifting model parameters / unmeasured disturbances to be estimated with states

# State and Parameter Estimation

**Prediction step for augmented model**

$$\begin{bmatrix} \hat{\mathbf{x}}(k|k-1) \\ \hat{\boldsymbol{\theta}}(k|k-1) \end{bmatrix} = \begin{bmatrix} \mathbf{F}\left[\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1), \bar{0}, \hat{\boldsymbol{\theta}}(k-1|k-1)\right] \\ \hat{\boldsymbol{\theta}}(k-1|k-1) \end{bmatrix}$$

**Update Step for augmented model**

$$\begin{bmatrix} \hat{\mathbf{x}}(k|k) \\ \hat{\boldsymbol{\theta}}(k|k) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}(k|k-1) \\ \hat{\boldsymbol{\theta}}(k|k-1) \end{bmatrix} + \mathbf{L}_a(k)\left[\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k|k-1)\right]$$

**Predicted Covariance Update step**

$$\mathbf{A}(t_{k-1}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{X}}\right]_{(\bullet)} \; ; \quad \mathbf{B}_{\boldsymbol{\theta}}(t_{k-1}) = \left[\frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}}\right]_{(\bullet)} \quad ; \; \mathbf{B}_{\mathbf{d}}(t_{k-1}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{d}}\right]_{(\bullet)}$$

$$(\bullet) \equiv (\mathbf{x}(k-1|k-1), \mathbf{u}(k-1), \mathbf{d}, \boldsymbol{\theta}(k-1|k-1))$$





# State and Parameter Estimation

**Predicted Covariance Update** : using augmented matrices

$$\Gamma_\theta(k,k-1) = \int_0^T \exp[A(t_{k-1})\tau] B_\theta(t_{k-1}) d\tau$$

$$\Phi_a(k,k-1) = \begin{bmatrix} \Phi(k,k-1) & \Gamma_\theta(k,k-1) \\ [0] & \mathbf{I} \end{bmatrix} \text{ and } \Gamma_a(k,k-1) = \begin{bmatrix} \Gamma_d(k,k-1) \\ \mathbf{I} \end{bmatrix}$$

$$\text{State Noise Covarance} : \mathbf{Q}_a = \begin{bmatrix} \mathbf{Q}_d & [0] \\ [0] & \mathbf{Q}_\theta \end{bmatrix}$$

Fast changing parameter/disturbance :
use high values of covariance

Tuning Parameter

$$\mathbf{P}_a(k|k-1) = \Phi_a(k,k-1)\mathbf{P}_a(k-1|k-1)\Phi_a(k,k-1)^T$$
$$+ \Gamma_a(k,k-1)\mathbf{Q}_a\Gamma_a(k,k-1)^T$$

# Extended Kalman Filter: Summary

**Kalman Gain Computation**

$$\text{Compute } \mathbf{C}(k) = \left[ \left[ \frac{\partial H}{\partial \mathbf{x}} \right]_{(\bullet)} \quad [\mathbf{0}] \right] \text{ at } \hat{\mathbf{x}}(k|k-1) \text{ and } \hat{\boldsymbol{\theta}}(k|k-1)$$

$$\mathbf{L}_a(k) = \mathbf{P}_{\varepsilon e}(k)\mathbf{P}_e(k)^{-1}$$

$$= \mathbf{P}_a(k|k-1)\mathbf{C}_a(k)^T \left[ \mathbf{C}_a(k)\mathbf{P}_a(k|k-1)\mathbf{C}_a(k)^T + \mathbf{R} \right]^{-1}$$

**Updated Covariance**

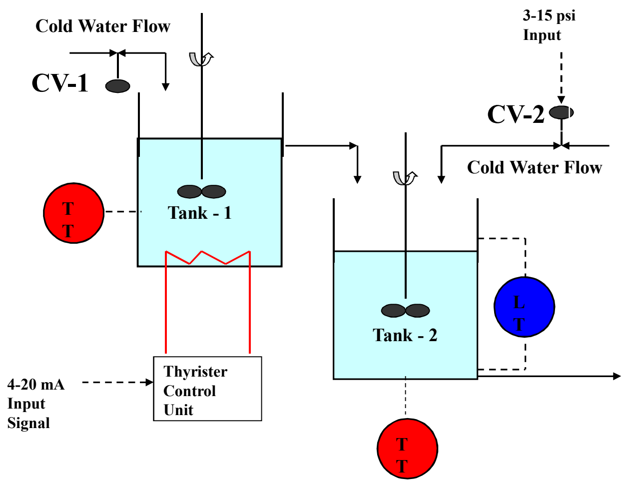
$$\mathbf{P}_a(k|k) = [\mathbf{I} - \mathbf{L}(k)\mathbf{C}_a(k)]\mathbf{P}_a(k|k-1)$$

Simultaneous state and parameter estimation can be used for
- Soft sensor for slowly changing parameters / unmeasured disturbances
- Faults in system, which are viewed as changing parameters





# Experiment: Combined State and Parameter Estimation on Heater-Mixer Setup



## 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



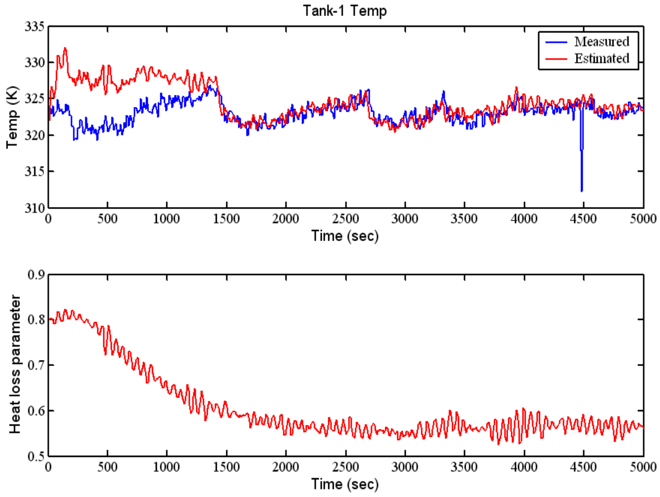
## Estimation of states and heat loss parameter using EKF – Experimental Conditions

- **Tank 1 temperature and heat loss parameter are to be estimated**
- **Tank 2 temperature and level are measured**
- **The system is kept in perturbed state by perturbing the inputs (heater input and tank 2 inlet flow)**
- **The flow to tank 1 is kept constant. This implies that overflow to tank 2 is also constant**
- **The parameter is initialized with a value of 0.8**

| | |
|---|---|
| No. of states estimated | 3 |
| No. of parameters estimated | 1 |
| No. of measurements | 2 |
| Measurement noise covariance | 0.01xeye(2,2) |
| State noise covariance | 0.1 |
| Initial guess for error covariance | 1 |

Automation Lab
IIT Bombay

## Experimental result: Tank 1 temperature and heat loss parameter estimates




Automation Lab
IIT Bombay

## Issues in State Estimation

- Robustness to plant-model mismatch: Model accuracy is critical to state estimation
- Noise Model Parameters: Measurement and state noise co-variances are difficult to estimate. These matrices are often treated as tuning parameters
- Number of extra states (unmeasured disturbances / parameters) estimated cannot exceed number of measurements
- Modifications necessary for multi-rate sampled data systems

## Summary

- Dynamic model based state observers can be used to reconstruct unmeasured states from frequently measured outputs
- Kalman filters generate state estimates with minimum estimation error variance, provided state and measurement noise models are known accurately
- Extended Kalman filtering can be used for estimating states of nonlinear systems
- **Note:** KF and EKF belong to a class of filters called Bayesian estimators, which are used in wide range of engineering applications (robotics, process control, target tracking, speech recognition, image reconstruction)





## References

- Soderstrom, T. Discrete Time Stochastic Systems, Springer, 2002.
- Gelb, A. Applied Optimal Estimation, MIT Press, 1974
- Astrom, K.J. and B. Wittenmark, Computer Controlled Systems, Prentice Hall, 1994.

# Appendix: Nominal Stability of Kalman Filter





## Convergence of Estimation Errors

Consider a KF as implemented on a linear deterministic system of the form

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma\mathbf{u}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

which is free of the state uncertainty and measurement noise

Kalman Gain Computation using **Riccati Equations**

$$\mathbf{P}(k \mid k-1) = \Phi\mathbf{P}(k-1 \mid k-1)\Phi^T + \mathbf{Q}$$

$$\mathbf{L}^*(k) = \mathbf{P}(k \mid k-1)\mathbf{C}^T\left[\mathbf{C}\mathbf{P}(k \mid k-1)\mathbf{C}^T + \mathbf{R}\right]^{-1}$$

$$\mathbf{P}(k \mid k) = \left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]\mathbf{P}(k \mid k-1)$$

where Q >0;R > 0 are tuning matrices

Automation Lab
IIT Bombay

## Convergence of Estimation Errors

Kalman Filter

$$\hat{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k) + \Gamma\mathbf{u}(k)$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{L}^*(k)[\mathbf{y}(k) - \mathbf{C}\hat{\mathbf{x}}(k|k)]$$

Under the nominal conditions, the only source of estimation error is the initial state $\hat{\mathbf{x}}(0|0)$

Error Dynamics

$$\varepsilon(k+1|k) = \Phi\varepsilon(k|k)$$

$$\varepsilon(k|k) = \left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]\varepsilon(k|k-1)$$

Combining

$$\varepsilon(k+1|k) = \Phi\left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]\varepsilon(k|k-1).........(3)$$

Equation (3) is a Linear Time Varying System
Stability Analysis cannot be carried out using eigenvalues

Automation Lab
IIT Bombay

## Convergence of Estimation Errors

Define matrices

$$\Pi(k|k-1) = [\mathbf{P}(k|k-1)]^{-1} \quad \text{and} \quad \Pi(k|k) = [\mathbf{P}(k|k)]^{-1}$$

Using **matrix inversion lemma**

$$\left[\mathbf{A}^{-1} + \mathbf{B}\right]^{-1} = \mathbf{A} - \mathbf{A}\left[\mathbf{A} + \mathbf{B}^{-1}\right]^{-1}\mathbf{A}$$

and Riccati equations, the following inequality can be proved

$$\Pi(k+1|k) \le [\Phi_c(k)]^{-T}\Pi(k|k-1)[\Phi_c(k)]^{-1} - [\Phi_c(k)]^{-T}\Omega(k)[\Phi_c(k)]^{-1}...........(4)$$

$$\Phi_c(k) = \Phi\left[\mathbf{I} - \mathbf{L}^*(k)\mathbf{C}\right]$$

$$\Omega(k) = \left[\Pi(k|k-1)\left(\Pi(k|k) + \Phi^T\mathbf{Q}^{-1}\Phi\right)^{-1}\Pi(k|k-1)\right]$$

Automation Lab
IIT Bombay

## Convergence of Estimation Errors

Define Lyapunov function

$$V(k) = \varepsilon(k|k-1)^T \Pi(k|k-1)\varepsilon(k|k-1)$$

Combining equation (3) with inequality (4)

$$V(k+1) - V(k) \leq -\varepsilon(k|k-1)^T \Omega(k)\varepsilon(k|k-1)$$

$$\Omega(k) = \left[\Pi(k|k-1)\left(\Pi(k|k) + \Phi^T \mathbf{Q}^{-1}\Phi\right)^{-1}\Pi(k|k-1)\right]$$

Since $\Omega(k)$ is always $+$ve definite

$$\varepsilon(k|k-1)^T \Omega(k)\varepsilon(k|k-1)^T > 0$$

and error dynamics given by equation (3) is Lyapunov stable

Automation Lab
IIT Bombay

## Convergence of Estimation Errors

Assumption: There exists $p_L, p_H > 0$ such that

$$p_L \mathbf{I} \leq \mathbf{P}(k|k-1) \leq p_H \mathbf{I} \quad \text{and} \quad p_L \mathbf{I} \leq \mathbf{P}(k|k) \leq p_H \mathbf{I}$$

$\Downarrow$

$$\frac{1}{p_H}\|\varepsilon(k|k-1)\| \leq V(k) \leq \frac{1}{p_L}\|\varepsilon(k|k-1)\|$$

$$\|\Omega(k)\| = \left\|\left[\Pi(k|k-1)\left(\Pi(k|k) + \Phi^T \mathbf{Q}^{-1}\Phi\right)^{-1}\Pi(k|k-1)\right]\right\| \leq \frac{1}{p_H^2\left[p_H + \left(\|\Phi\|^2 / \|\mathbf{Q}^{-1}\|\right)\right]}$$

$\Downarrow$

$$V(k+1) - V(k) \leq -\frac{1}{p_H^2\left[p_H + \left(\|\Phi\|^2 / \|\mathbf{Q}^{-1}\|\right)\right]}\|\varepsilon(k|k-1)\|^2$$

Thus, estimation error dynamics is asymptotically stable