# کنترل پیش بین
# Model Predictive Control

ارائه کننده: امیرحسین نیکوفرد
مهندسی برق و کامپیوتر دانشگاه خواجه نصیر

# Dynamic State Estimation for Dynamical Systems

- Introduce dynamic state estimation (DSE)
- Discuss classes of observers/estimators + Applications Briefly
- discuss stochastic estimators — Kalman filter
- Deterministic observers
- State observer for MPC

# Introduction

- What is dynamic state estimation (DSE)?

–Accurately depicting what's happening inside a system

- Precisely: estimating internal system states

–In circuits: voltages and currents

–Water networks: amount of water flowing

–Chemical plants: concentrations

–Robots and UAVs: location & speed

–Humans: temperature, blood pressure, glucose level

- So how does having estimates help me?

–Well, if you have estimates, you can do control

–And if you do good control, you become better off!

- In power systems: DSE can tell me what's happening to generators & lines⇒ Preventing/Predicting Blackouts!

# Introduction

- **Dynamic observer:** dynamical system that observes the internal system state, given a set of input & output measurements
- **State estimator:** estimates the system's states under different assumptions Estimators: utilized for state estimation and parametric identification
- **Observers:** used for deterministic systems, Estimators: for stochastic dynamical systems
- If statistical information on process and measurement is available, stochastic estimators can be utilized
- This assumption is strict for many dynamical systems
- Quantifying distributions of measurement and process noise is very challenging

# Stochastic estimators

- Stochastic estimators:
  - Extended Kalman Filter (EKF)
  - Unscented Kalman filter (UKF)
  - Square-root Unscented Kalman filter (SRUKF)
  - Cubature Kalman Filter (CKF)
- Stochastic estimators used if distributions of measurement & process noise are available
- System dynamics:

$$x_k = f\,(x_{k-1}, u_{k-1}) + w_{k-1}$$

$$y_k = h(x_k, u_k) + v_k$$

  - $w_{k-1} \sim N\,(0, Q_{k-1})$ and $v_k \sim N\,(0, R_k)$: process & measurement noise
  - $Q_{k-1}$ and $R_k$ : covariance of $q_{k-1}$ & $r_k$

# Stochastic Estimator: The Extended Kalman Filter

- Most stochastic estimators have two main steps: predictions & updates
- EKF (=KF+Nonlinearities) algorithm:

(1) Prediction:

State esimate prediction: $\hat{\boldsymbol{x}}_{k|k-1} = f(\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_{k-1})$

Predicted covariance estimate: $\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_{k-1}^{\top} + \boldsymbol{Q}_{k-1}$

(2) Update:

Innovation or measurement residual: $\tilde{\boldsymbol{y}}_k = \boldsymbol{z}_k - h(\hat{\boldsymbol{x}}_{k|k-1})$

Innovation (or residual) covariance: $\boldsymbol{S}_k = \boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^{\top} + \boldsymbol{R}_k$

Near-optimal Kalman gain: $\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^{\top}\boldsymbol{S}_k^{-1}$

Updated covariance estimate: $\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_{k|k-1}$

Updated state estimate: $\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\tilde{\boldsymbol{y}}_k$

$$\boldsymbol{F}_{k-1} = \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_{k-1}}, \quad \boldsymbol{H}_k = \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k|k-1}}$$

# Determinstic Estimators (Observers)

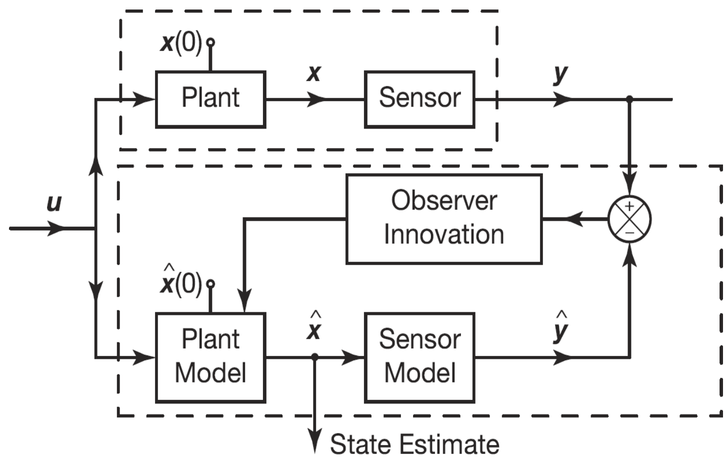
- Deterministic observers for:

–LTI systems
–LTI systems + Unknown Inputs
–LTI systems + Unknown Inputs + Measurement Noise
–Nonlinear systems (bounded nonlinearity)
–Nonlinear systems + Unknown Inputs
–Nonlinear systems + Unknown Inputs + Measurement
–LTI delayed systems
–LTI delayed systems + Unknown Inputs
–Hybrid systems
–… and many more

*Deterministic estimators used if measurement and process noise distributions are not available

# Determinstic Estimators (Observers)

- Controllers often need values for the full state-vector of the plant
- This is nearly impossible in most complex systems
- Why? You simply can't put sensors everywhere, and some states are inaccessible
- **Observer:** a dynamical system that estimates the states of the system based on the plant's inputs and outputs
- Who introduced observers? David Luenberger in 1963, Ph.D. dissertation



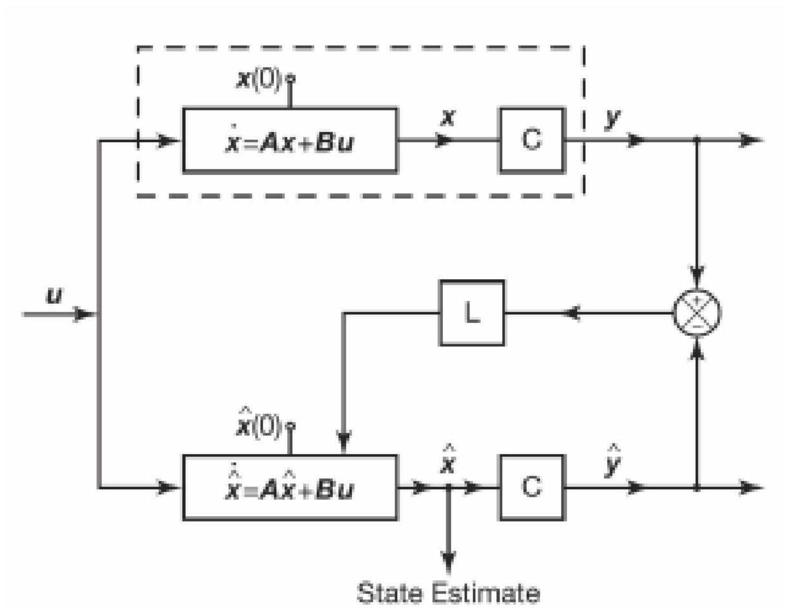
# LuenbergerObserver and Plant Dynamics

- Plant Dynamics: $\begin{cases} \dot{x} = Ax + Bu \\ y = Cx, \;\; x(0) \text{ not given} \end{cases}$

Observers Dynamics: $\begin{cases} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \leftarrow Innovation \\ \dot{\hat{x}} &= A\hat{x} + Bu + LC(x - \hat{x}) \end{cases}$

- Error dynamics [2]:

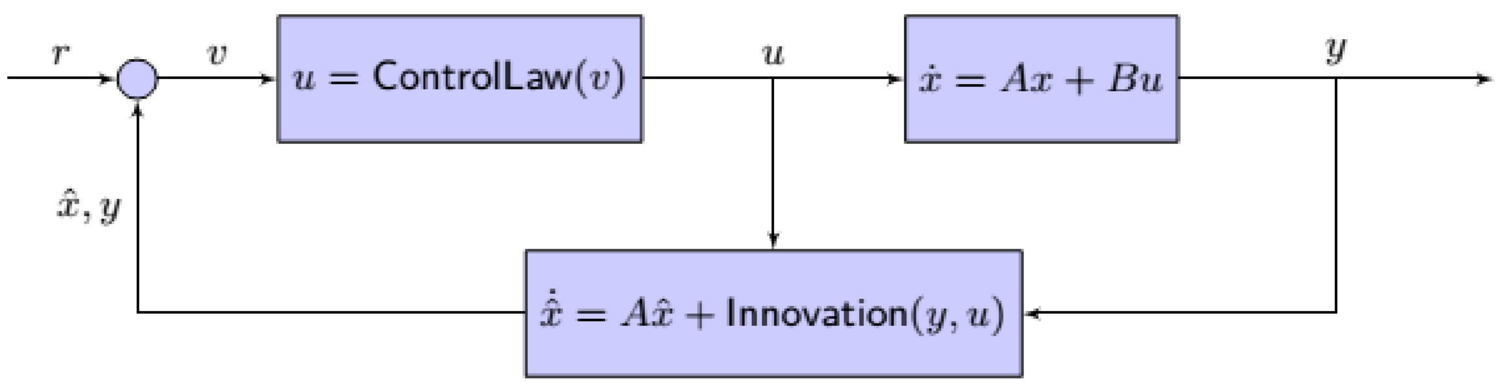
$$\dot{e} = \dot{x} - \dot{\hat{x}} = (A - LC)(x - \hat{x}) \to 0, \text{ as } t \to \infty, \textbf{iff } \lambda_i(A - LC) < 0$$



# Observer-Based Control

- After designing an observer for an LTI system, obtain state estimates $(\hat{x}(t))$
- What to do with $\hat{x}(t)$? Well, use it for control $\Rightarrow$ *Observer-Based Control!*
- OBC dynamics:

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + \text{Innovation}(y, u) \\ u = \text{ControlLaw}(v), \quad v = \begin{bmatrix} \hat{x} & y & r \end{bmatrix} \end{cases}$$



# Observer-Based Control

- Closed-loop dynamics:

$$\begin{aligned} \dot{x} &= Ax - BK\hat{x} \\ \dot{\hat{x}} &= A\hat{x} + L(y - \hat{y}) - BK\hat{x} \end{aligned}$$

- Or

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & -BK \\ LC & A - LC - BK \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix}$$

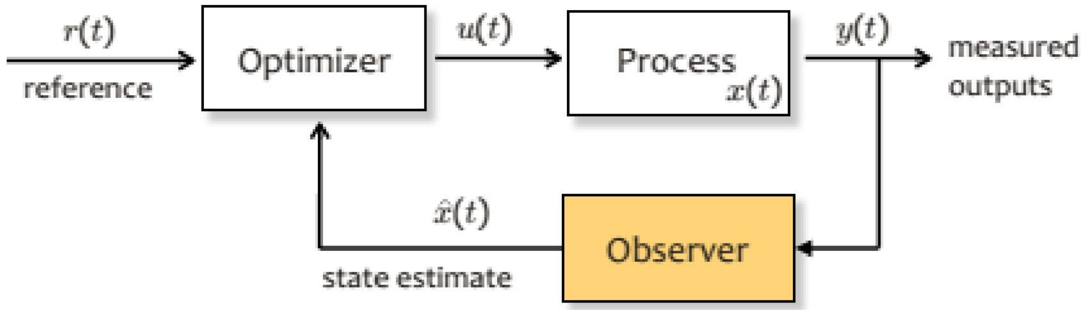
- Transformation: $\begin{bmatrix} x \\ e \end{bmatrix} = \begin{bmatrix} x \\ x - \hat{x} \end{bmatrix} = \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix}$

- Hence, we can write:

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \underbrace{\begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix}}_{A_{\mathrm{cl}}} \begin{bmatrix} x \\ e \end{bmatrix}$$

- If the system is controllable & observable $\Rightarrow$ $\mathtt{eig}(A_{\mathrm{cl}})$ can be arbitrarily assigned by proper $K$ and $L$
- What if the system is stabilizable and detectable?

# State observer for MPC

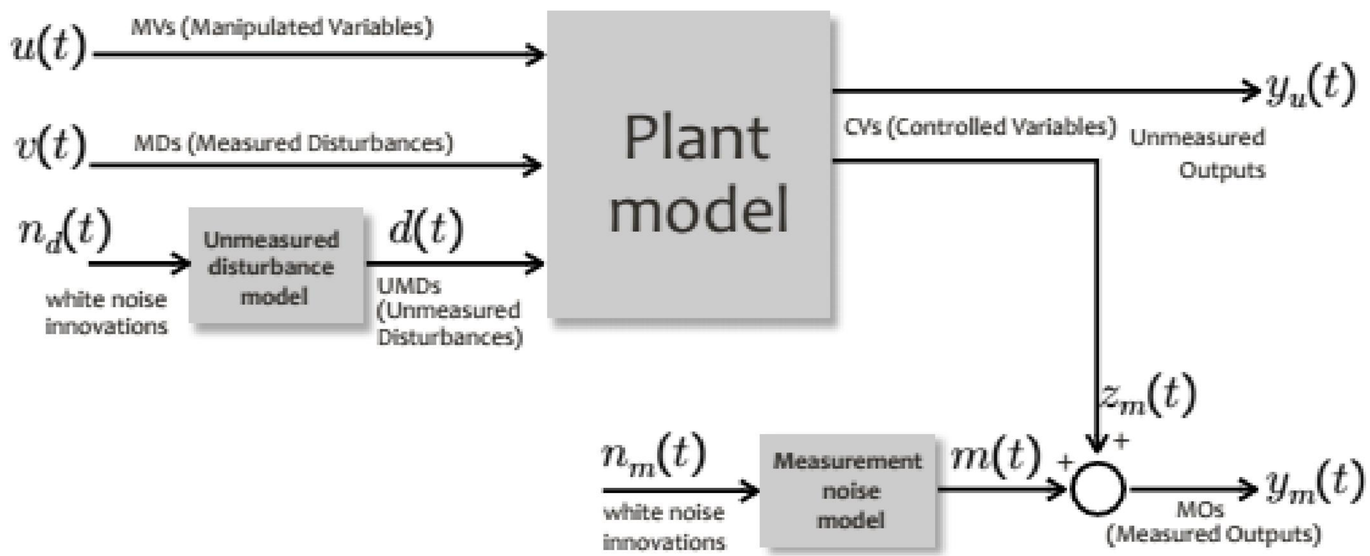



- Full state $x(t)$ of process may not be available, only outputs $y(t)$
- Even if $x(t)$ is available, noise should be filtered out
- Prediction and process models may be quite different
- The state $x(t)$ may not have any physical meaning (e.g., in case of model reduction or subspace identification)

**We need to use a state observer**

- Example: Luenberger observer $\hat{x}(t+1) = A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t))$

# Extended model for observer design




**unmeasured disturbance model**

$$\begin{cases} x_d(t+1) = \bar{A}x_d(t) + \bar{B}n_d(t) \\ d(t) = \bar{C}x_d(t) + \bar{D}n_d(t) \end{cases}$$

**measurement noise model**

$$\begin{cases} x_m(t+1) = \tilde{A}x_m(t) + \tilde{B}n_m(t) \\ m(t) = \tilde{C}x_m(t) + \tilde{D}n_m(t) \end{cases}$$

- Note: the measurement noise model is not needed during optimization

# Kalman filter design

Rudolf Emil Kalman
(1930-2016)

- Plant model

$$\begin{cases} x(t+1) &= Ax(t) + B_u u(t) + B_v v(t) + B_d d(t) \\ y(t) &= Cx(t) + D_v v(t) + D_d d(t) \end{cases}$$

- Full model for designing Kalman filter

$$\begin{aligned} \begin{bmatrix} x(t+1) \\ x_d(t+1) \\ x_m(t+1) \end{bmatrix} &= \begin{bmatrix} A & B_d\bar{C} & 0 \\ 0 & \bar{A} & 0 \\ 0 & 0 & \tilde{A} \end{bmatrix} \begin{bmatrix} x(t) \\ x_d(t) \\ x_m(t) \end{bmatrix} + \begin{bmatrix} B_u \\ 0 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} B_v \\ 0 \\ 0 \end{bmatrix} v(t) + \\ &\quad \begin{bmatrix} B_d\bar{D} \\ \bar{B} \\ 0 \end{bmatrix} n_d(t) + \begin{bmatrix} 0 \\ 0 \\ \tilde{B} \end{bmatrix} n_m(t) + \begin{bmatrix} B_u \\ 0 \\ 0 \end{bmatrix} n_u(t) \\ y_m(t) &= \begin{bmatrix} C_m & D_{dm}\bar{C} & \tilde{C} \end{bmatrix} \begin{bmatrix} x(t) \\ x_d(t) \\ x_m(t) \end{bmatrix} + D_{vm}v(k) + \bar{D}_m n_d(t) + \tilde{D}_m n_m(t) \end{aligned}$$

- $n_d(k)$ = source of **modeling errors**
- $n_m(k)$ = source of **measurement noise**
- $n_u(k)$ = white noise on input $u$ (added to compute the Kalman gain)

# I/O feedthrough

- We always assumed **no feedthrough from $u$ to measured $y$**

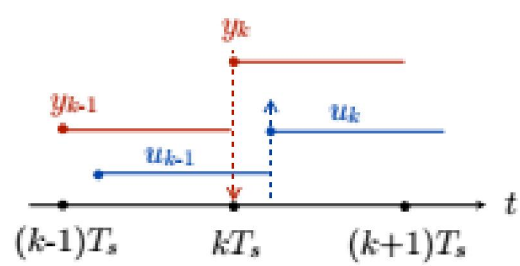
$$y_k = Cx_k + Du_k + D_v v_k + D_d d_k, \quad D_m = 0$$

- This avoids **static loops** between state observer, as $\hat{x}(t|t)$ depends on $u(t)$ via $\hat{y}_m(t|t-1)$, and MPC ($u(t)$ depends on $\hat{x}(t|t)$)

- Often $D = 0$ is not a limiting assumption as
  - often **actuator dynamics** must be considered ($u$ is the set-point to a low-level controller of the actuators)
  - most physical models described by ordinary differential equations are **strictly causal**, and so is the discrete-time version of the model

- In case $D \neq 0$, we can assume a **delay** in executing the commanded $u$

$$y_k = C_m x_k + Du_{k-1}$$




and treat $u(t-1)$ as an extra state

- Not an issue for **unmeasured** outputs