PID Control
& Estimation

ELEC 3004: **Digital Linear Systems**: Signals & Controls
Dr. Surya Singh

Lecture 12

elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/

May 27, 2014

http://elec3004.com

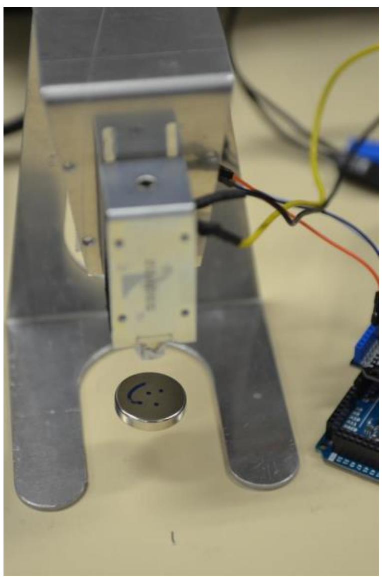
Lecture Schedule:

Week	Date	Lecture Title
1	4-Mar	Introduction & Systems Overview
	6-Mar	[Linear Dynamical Systems]
2	11-Mar	Signals as Vectors & Systems as Maps
	13-Mar	[Signals]
3	18-Mar	Sampling & Data Acquisition & Antialiasing Filters
	20-Mar	[Sampling]
4	25-Mar	System Analysis & Convolution
	27-Mar	[Convolution & FT]
5	1-Apr	Frequency Response & Filter Analysis
	3-Apr	[Filters]
6	8-Apr	Discrete Systems & Z-Transforms
	10-Apr	[Z-Transforms]
7	15-Apr	Introduction to Digital Control
	17-Apr	[Feedback]
8	29-Apr	Digital Filters
	1-May	[Digital Filters]
9	6-May	Digital Control Design
	8-May	[Digitial Control]
10	13-May	Stability of Digital Systems
	15-May	[Stability]
11	20-May	State-Space
	22-May	Controllability & Observability
12	27-May	PID Control & System Identification
	29-May	Digitial Control System Hardware
13	3-Jun	Applications in Industry & Information Theory & Communications
	5-Jun	Summary and Course Review

Announcements:

- **PS 3 Grading:**
  - Due Wednesday at 11:59pm
  - We're working on it
- **PS 4:**
  - Working on it too!
- **Lab 4:**
  - Working on it too too!!
- Final Exam Logistics:
  - Saturday 21/6/2014 at 4:30pm
  - Location: TBA
  - Closed-book
  - Practice exam will be posted soon (Working on it too too too!!)

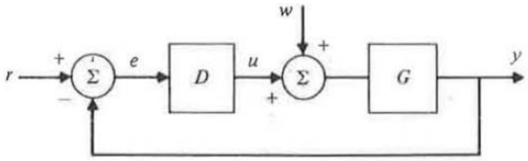
Lab 4



PID

- Three basic types of control:
  - Proportional
  - Integral, and
  - Derivative

- The next step up from lead compensation
  - Essentially a combination of proportional and derivative control



Proportional Control

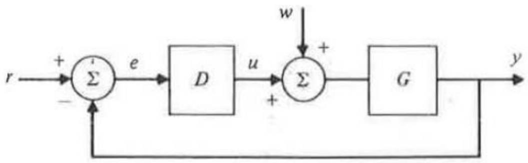
A discrete implementation of proportional control is identical to continuous; that is, where the continuous is

$$u(t) = K_p e(t) \quad \Rightarrow \quad D(s) = K_p,$$

the discrete is

$$u(k) = K_p e(k) \quad \Rightarrow \quad \boxed{D(z) = K_p}$$

where $e(t)$ is the error signal as shown in Fig 5.2.



Derivative Control

For continuous systems, derivative or rate control has the form

$$u(t) = K_p T_D \dot{e}(t) \quad \Rightarrow \quad D(s) = K_p T_D s$$

where T_D is called the *derivative time*. Differentiation can be approximated in the discrete domain as the first difference, that is,

$$u(k) = K_p T_D \frac{(e(k) - e(k-1))}{T} \quad \Rightarrow \quad \boxed{D(z) = K_p T_D \frac{1 - z^{-1}}{T} = K_p T_D \frac{z-1}{Tz}.}$$

In many designs, the compensation is a sum of proportional and derivative control (or PD control). In this case, we have

$$D(z) = K_p \left(1 + \frac{T_D(z-1)}{Tz}\right).$$

or, equivalently,

$$\boxed{D(z) = K \frac{z - \alpha}{z}}$$

Derivative Control [2]

- Similar to the lead compensators
  - The difference is that the pole is at $z = 0$

    [Whereas the pole has been placed at various locations along the z-plane real axis for the previous designs.]

- In the continuous case:
  - pure derivative control represents the ideal situation in that there is no destabilizing phase lag from the differentiation
  - the pole is at $s = -\infty$

- In the discrete case:
  - z=0
  - However this has phase lag because of the necessity to wait for one cycle in order to compute the first difference

Integral Control

For continuous systems, we integrate the error to arrive at the control,

$$u(t) = \frac{K_p}{T_I}\int_{t_o}^{t} e(t)dt \;\Rightarrow\; D(s) = \frac{K_p}{T_I s},$$

where T_I is called the *integral*, or *reset time*. The discrete equivalent is to sum all previous errors, yielding

$$u(k) = u(k-1) + \frac{K_p T}{T_I} e(k) \;\Rightarrow\; \boxed{D(z) = \frac{K_p T}{T_I(1 - z^{-1})} = \frac{K_p T z}{T_I(z-1)}.} \quad (5.60)$$

Just as for continuous systems, the primary reason for integral control is to reduce or eliminate steady-state errors, but this typically occurs at the cost of reduced stability.

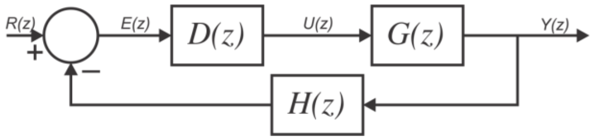
PID Control

$$\boxed{D(z) = K_p\left(1 + \frac{Tz}{T_I(z-1)} + \frac{T_D(z-1)}{Tz}\right).}$$

The user simply has to determine the best values of

- K_p
- T_D and
- T_I

PID as Difference Equation



$$\frac{U(z)}{E(z)} = D(z) = K_p + K_i\left(\frac{Tz}{z-1}\right) + K_d\left(\frac{z-1}{Tz}\right)$$

$$\boxed{u(k) = \left[K_p + K_iT + \left(\frac{K_d}{T}\right)\right]\cdot e(k) - [K_dT]\cdot e(k-1) + [K_i]\cdot u(k-1)}$$

State-Space Control

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x}$$

(That can not be all of it? There has to be more to it than this…)

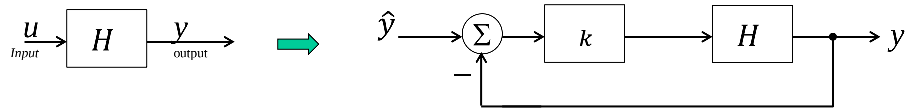
State-Space Control

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u}$$

Benefits:

- Characterises the process by systems of coupled, first-order differential equations
- More general mathematical model
  - MIMO, time-varying, nonlinear
- Mathematically esoteric (who needs practical solutions)
- Yet, well suited for digital computer implementation
  - That is: based on vectors/matrices (think LAPACK ➔ MATLAB)

Difference Equations & Feedback



- Start with the Open-Loop:

$$y = Hu$$

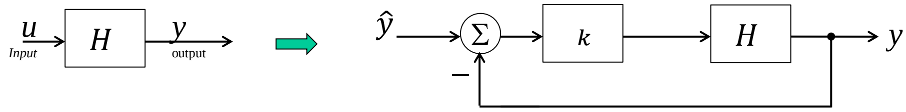
- Close the loop:

$$u = ke = k(\hat{y} - y) \rightarrow y = H[k(\hat{y} - y)]$$

$$\rightarrow y = \frac{Hk}{1+Hk}\hat{y}$$

- All easy! (yesa!)

Difference Equations & Feedback



- Now add delay (image the plant is a replica with a delay τ)

$$y(t) = u(t-\tau)$$

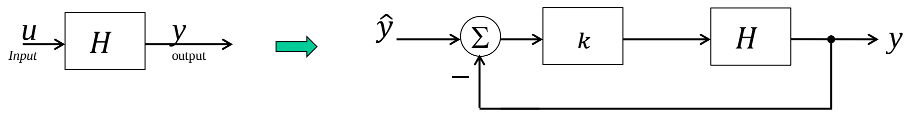
- Close the loop:

$$u(t-\tau) = ke(t-\tau) = k\,[\hat{y}(t-\tau) - \mathrm{y}(t-\tau)]$$

$$\Rightarrow \mathrm{y(t)} = k\,[\hat{y}(t-\tau) - \mathrm{y}(t-\tau)]$$

- Notice we have a difference equation!

Difference Equations & Feedback



- What happens with a single delay and a unit step?

$$u(t) = k \text{ for } 0<\mathrm{t}<\tau$$

$$y(t) = u(t-\tau) \text{ for } \tau<\mathrm{t}<2\tau$$

- Then with feedback we get:

$$u(t) = k(1-k) = k - k^2$$

$$y(t) = k - k^2 + k^3 + \cdots + (-1)^{n-1}k^{n-1}$$

- If k<1: then:

$$\Rightarrow \lim \mathrm{y(t)} = \frac{k}{1+k}$$

Example 0: PID

Great, so how about control?

- Given $\dot{\boldsymbol{x}} = \mathbf{F}\boldsymbol{x} + \mathbf{G}u$, if we know $\mathbf{F}$ and $\mathbf{G}$, we can design a controller $u = -\mathbf{K}\boldsymbol{x}$ such that

$$\mathrm{eig}(\mathbf{F} - \mathbf{GK}) < 0$$

- In fact, if we have full measurement and control of the states of $\boldsymbol{x}$, we can position the poles of the system in arbitrary locations!

Example: PID control

- Consider a system parameterised by three states:
  - x_1, x_2, x_3
  - where $x_2 = \dot{x}_1$ and $x_3 = \dot{x}_2$

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 1 & & \\ & 1 & \\ & & -2 \end{bmatrix} \boldsymbol{x} - \mathbf{K}u$$

$$y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \boldsymbol{x} + 0u$$

x_2 is the output state of the system;

x_1 is the value of the integral;

x_3 is the velocity.

- We can choose **K** to move the eigenvalues of the system as desired:

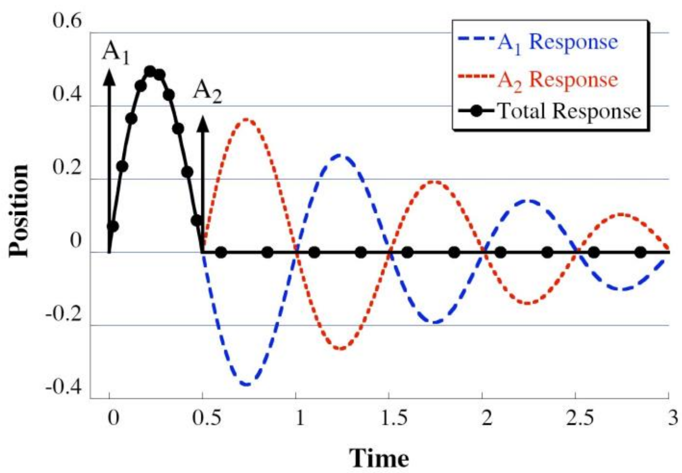
$$\det \begin{bmatrix} 1 - K_1 & & \\ & 1 - K_2 & \\ & & -2 - K_3 \end{bmatrix} = \mathbf{0}$$

All of these eigenvalues must be positive.

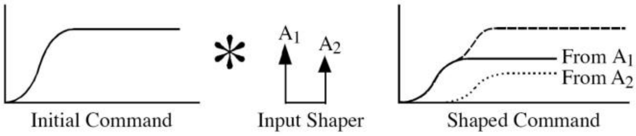
It's straightforward to see how adding derivative gain K_3 can stabilise the system.

Example 1:
Command Shaping

Command Shaping



Command Shaping



- Zero Vibration (ZV)

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+K} & \frac{K}{1+K} \\ 0 & \frac{T_d}{2} \end{bmatrix} \qquad K = e^{\left(\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}\right)}$$

- Zero Vibration and Derivative (ZVD)

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{(1+K)^2} & \frac{2K}{(1+K)^2} & \frac{K^2}{(1+K)^2} \\ 0 & \frac{T_d}{2} & T_d \end{bmatrix}$$

Can you use this for more than Control?

- Yes

Frequency Response in State Space

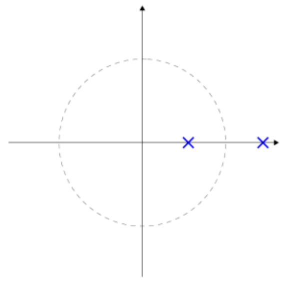
$$H(z) = C(zI - A)^{-1}B + D = \frac{1}{100z^2 - 200z + 80}$$

Poles at $\approx 0.55, 1.45$.

Eigenvalues of A:

$1, 1, 1.45, .55$

What are the (physical) implications?

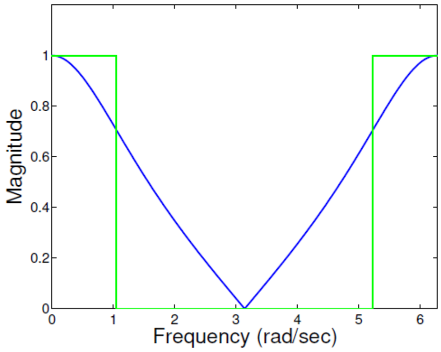


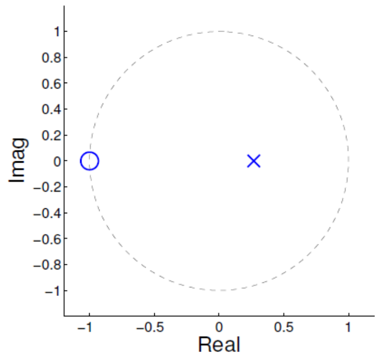
The Approach:

- Formulate the goal of control as an **optimization** (e.g. minimal impulse response, minimal effort, ...).
- You've already seen some examples of optimization-based design:
  - Used least-squares to obtain an FIR system which matched (in the least-squares sense) the desired frequency response.
  - Poles/zeros lecture: Butterworth filter

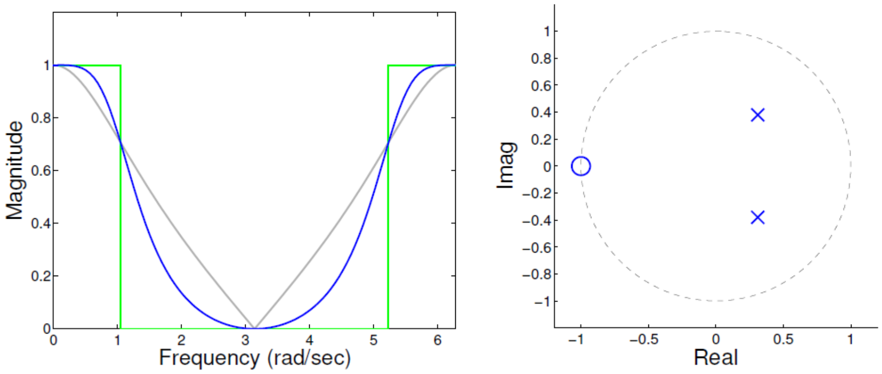
Discrete Time Butterworth Filters

"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.

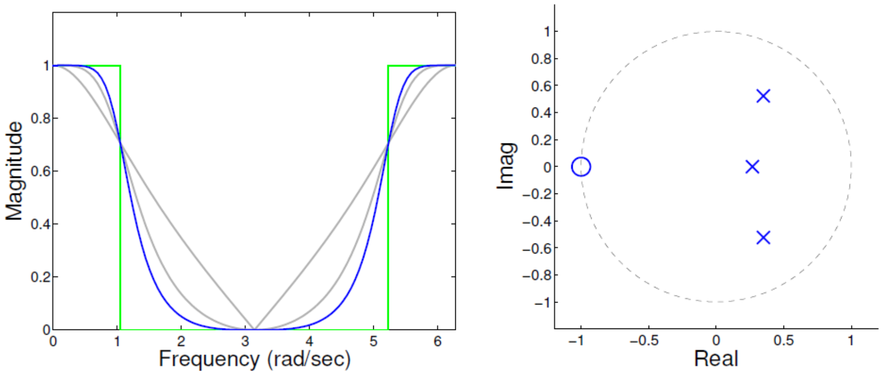




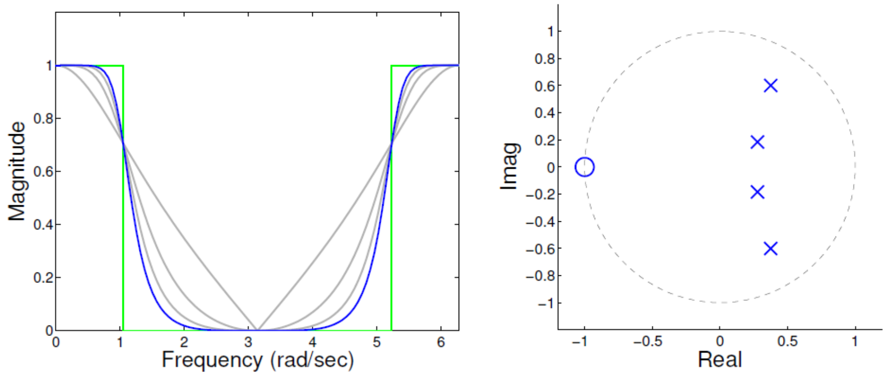
"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.



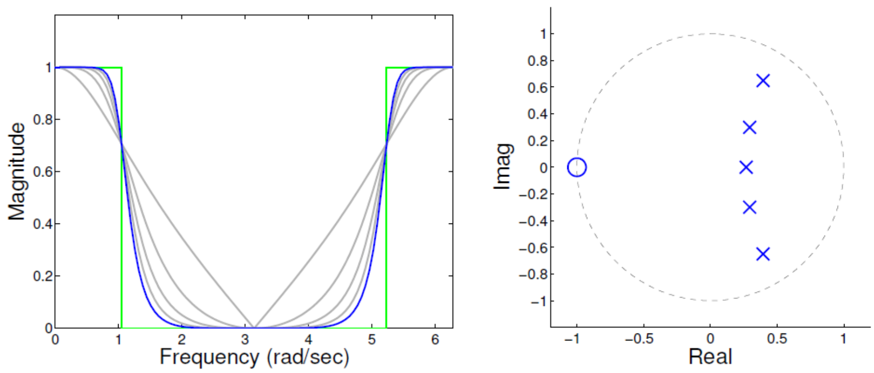
"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.



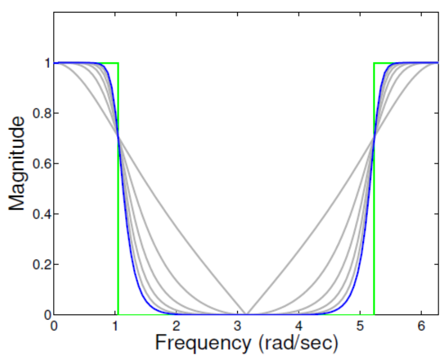
"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.

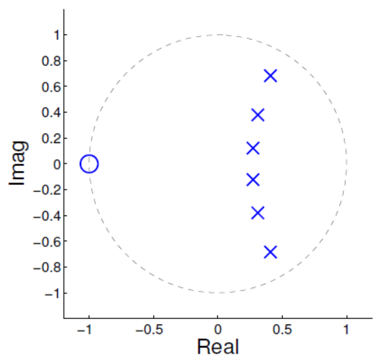


"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.

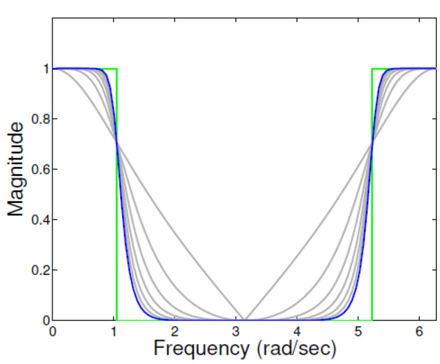


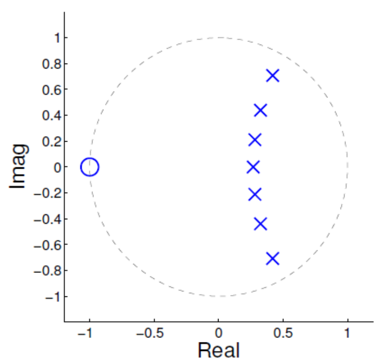
“Maximally-flat filter”. Sacrifice sharpness to have flat response in pass band and stop band.





“Maximally-flat filter”. Sacrifice sharpness to have flat response in pass band and stop band.





How?

- Constrained Least-Squares …

One formulation: Given $x[0]$

$$\underset{u[0],u[1],\ldots,u[N]}{\text{minimize}} \quad ||\vec{u}||^2, \quad \text{where } \vec{u} = \begin{bmatrix} u[0] \\ u[1] \\ \vdots \\ u[N] \end{bmatrix}$$

$$\text{subject to} \quad x[N] = 0.$$

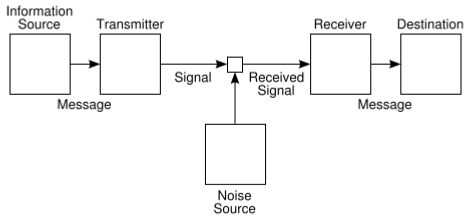
Note that

$$x[n] = A^n x[0] + \sum_{k=0}^{n-1} A^{(n-1-k)} B u[k],$$

so this problem can be written as

$$\underset{x_{ls}}{\text{minimize}} \, ||A_{ls} x_{ls} - b_{ls}||^2 \quad \text{subject to} \quad C_{ls} x_{ls} = D_{ls}.$$

Shannon Information Theory



On the transmission of information over a noisy channel:

- An information source that produces a message
- A transmitter that operates on the message to create a signal which can be sent through a channel
- A channel, which is the medium over which the signal, carrying the information that composes the message, is sent
- A receiver, which transforms the signal back into the message intended for delivery
- A destination, which can be a person or a machine, for whom or which the message is intended

Additional Use II: Estimation

Along multiple dimensions

State Space

- We collect our set of uncertain variables into a vector …
  $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$
- The set of values that **x** might take on is termed the *state space*
- There is a *single* true value for **x**,
  but it is unknown

State Space Dynamics

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

$$H(s) = C(sI - A)^{-1}B$$

Measured versus True

- Measurement errors are inevitable
- So, add Noise to State...
  - State Dynamics becomes:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + w$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} + v$$

- Can represent this as a "Normal" Distribution

$$\mathcal{N}\left(x;\mu,\sigma\right) = \frac{1}{\left(\sqrt{2\pi}\right)\sigma}\exp\left(-\frac{\left(x-\mu\right)^2}{2\sigma^2}\right)$$

Recovering The Truth

- Numerous methods
- Termed "Estimation" because we are trying to estimate the truth from the signal
- A strategy discovered by Gauss
- Least Squares in Matrix Representation

$$\begin{bmatrix} p_0 \\ p_1 \end{bmatrix} = \begin{bmatrix} n & \sum_1^n t_i \\ \sum_1^n t_i & \sum_1^n t_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_1^n z_i \\ \sum_1^n t_i z_i \end{bmatrix}$$

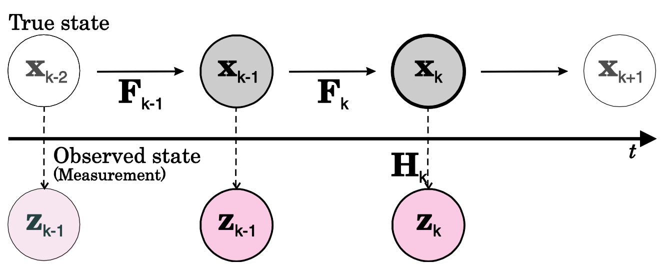
Recovering the Truth: Terminology

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u} + \mathbf{w}$$

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v}$$

- $\mathbf{x}$: the state vector
- $\mathbf{x}_{A|B}$: the state of $\mathbf{x}$ at time A based on data taken up to time B
- $\hat{\mathbf{x}}$: estimate of the true state vector
- $\mathbf{F}$: system dynamics matrix in continuous time (equivalent to $\mathbf{A}$ in Eq. 1)
- $\mathbf{G}$: system control matrix relating deterministic input, $\mathbf{u}$, to the state (equivalent to $\mathbf{B}$ in Eq. 1)
- $\mathbf{H}$: measurement matrix in continuous time (equivalent to $\mathbf{C}$ in Eq. 2)
- $\mathbf{F}_i$: system model in **discrete** time at $t = t_i$
- $\mathbf{H}_i$: measurement model in **discrete** time at $t = t_i$
- $\mathbf{P}_i$: estimate covariance in **discrete** time at $t = t_i$
- $\mathbf{w}$: process uncertainty (noise) vector (of type $\mathcal{N}(0, s)$)
- $\mathbf{Q}$: process noise matrix, $\mathbf{Q} = E\left[ww^{\mathsf{T}}\right]$
- $\mathbf{Q}_i$: $\mathbf{Q}$ in discrete time at $t = t_i$
- $\mathbf{v}$: measurement noise vectors (of type $\mathcal{N}(0, \sigma)$)
- $\mathbf{R}_i$: the measurement variance matrix, $\mathbf{R} = E\left[vv^{\mathsf{T}}\right]$, in discrete time at $t = t_i$

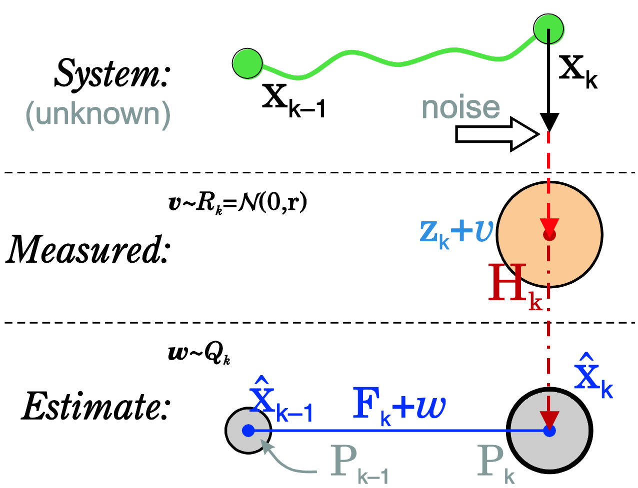
General Problem…



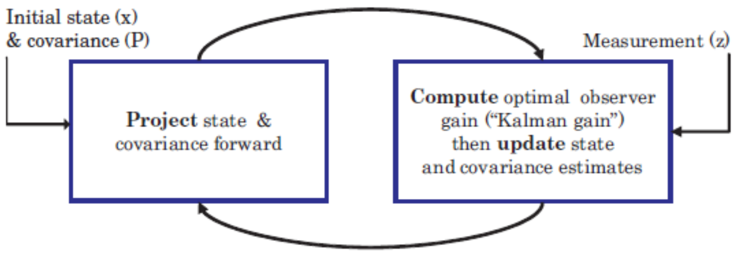
Duals and Dual Terminology

	Estimation		Control
Model:	$\dot{x} = Fx$ (discrete: $x = F_k x$)	$\leftrightarrow$	$\dot{x} = Ax$, $A = F^T$
Regulates:	P (covariance)	$\leftrightarrow$	M (performance matrix)
Minimized function:	Q (or GQG^T)	$\leftrightarrow$	V
Optimal Gain:	K	$\leftrightarrow$	G
Completeness law:	Observability	$\leftrightarrow$	Controllability

Estimation Process in Pictures



Kalman Filter Process



KF Process in Equations

Prediction: $\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1},$ (state prediction)

$\mathbf{P}_{k|k-1} = \mathbf{Q}_{k-1} + \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}^{\mathsf{T}}{}_{k-1},$ (covariance prediction)

Kalman Gain: $\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^{\mathsf{T}}[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^{\mathsf{T}} + \mathbf{R}_k]^{-1},$

Update: $\mathbf{P}_{k|k} = [\mathbf{I} - \mathbf{K}_k\mathbf{H}]\mathbf{P}_{k|k-1},$ (covariance update)

$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1})$ (state update)

KF Considerations

$$
\begin{aligned}
\underbrace{\hat{\mathbf{x}}_{k|k-1}}_{n\times 1} &= \underbrace{\mathbf{F}_{k-1}}_{n\times n}\hat{\mathbf{x}}_{k-1|k-1} + \underbrace{\mathbf{G}_{k-1}}_{n\times j}\underbrace{\mathbf{u}_{k-1}}_{j\times 1} \\
\underbrace{\mathbf{P}_{k|k-1}}_{n\times n} &= \underbrace{\mathbf{Q}_{k-1}}_{n\times n} + \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}^{\mathsf{T}}{}_{k-1} \\
\underbrace{\mathbf{K}_k}_{n\times m} &= \mathbf{P}_{k|k-1}\underbrace{\mathbf{H}^{\mathsf{T}}}_{n\times m}\underbrace{\left[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^{\mathsf{T}} + \mathbf{R}_k\right]^{-1}}_{m\times m} \\
\mathbf{P}_{k|k} &= [\mathbf{I} - \mathbf{K}_k\mathbf{H}]\mathbf{P}_{k|k-1} \\
\hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\underbrace{\mathbf{z}_k}_{m\times 1} - \underbrace{\mathbf{H}}_{m\times n}\hat{\mathbf{x}}_{k|k-1} - \mathbf{H}\mathbf{G}_k\mathbf{u}_{k-1})
\end{aligned}
$$

Ex: Kinematic KF: Tracking

- Consider a System with Constant Acceleration

$$
\begin{aligned}
\ddot{y} &= -g \\
\dot{y} &= gt + p_1 \\
y &= p_0 + p_1 t + \tfrac{gt^2}{2}
\end{aligned}
$$

$$
\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} y \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ g \end{bmatrix}
$$

$$
\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{F_k} = \begin{bmatrix} 0 & t_s & \frac{t_s{}^2}{2} \\ 0 & 0 & t_s \\ 0 & 0 & 0 \end{bmatrix}
$$

$$
\hat{\mathbf{x}}_k = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1})
$$

In Summary

- KF:
  - The true state (*x*) is separate from the measured (*z*)
  - Lets you **combine** prior controls knowledge with measurements to filter signals and find the <u>truth</u>
  - It **regulates** the covariance (P)
    - As P is the scatter between *z* and *x*
    - So, if P → 0, then *z* → *x* (measurements → truth)
- EKF:
  - Takes a Taylor series approximation to get a local “F” (and “G” and “H”)