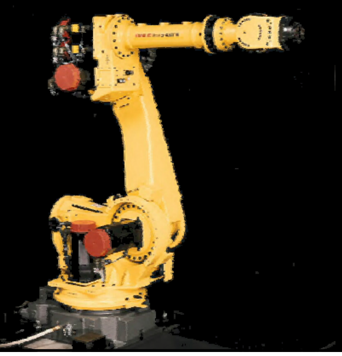
Automatic Control



MEG 421 Automatic Control

GOALS: To provide advanced students in mechanical engineering with a solid background in dynamic system modeling and analysis and to enable them to analyze and design linear control systems.

MEG 421 Automatic Control

FORMAT:

Lecture: 3 credits

Lab: 1 credit

You must enroll in both MEG 421 and MEG 421L

MEG 421 Automatic Control

MEG 421:

Prerequisites by Topic:

1. Electrical Circuits
2. Mathematics for Engineers.
3. Analysis of Dynamic Systems

Automatic Control

–Most General Definition:

To Produce a Desired Result

"Trust is good, control is better."

Definition of Automation

- “Having the capability of starting, operating, moving, etc., independently.” [1]
- “The use of machines to perform tasks that require decision making.” [2]

Automatic Control

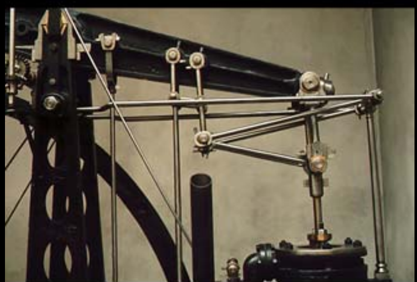
- Technical Control Systems

Open Loop Control Systems

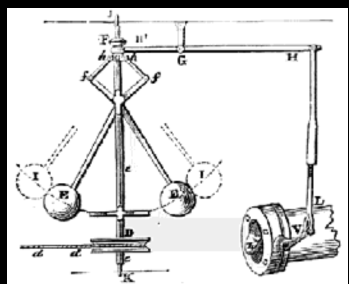
Open Loop Control Systems Example: Batch Filling

Block Diagram for Feedback (or Closed Loop) Control

England - Eighteenth Century AD Watt Steam Engine



England - Eighteenth Century AD



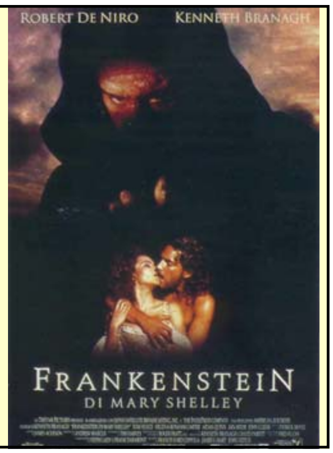
The accelerating technological change of the 19th century was reflected in literature and art.

In Jacques Offenbach's opera 'Les contes d'Hoffmann' the hero falls in love with Olympia, a mechanical doll. Olympia can sing and dance. She needs rewinding every 5 minutes or so.

Olympia on Youtube:

http://www.youtube.com/watch?v=sXK3pUdBRGA

Another famous example is Mary Shelley's ever popular **Frankenstein** (1831).

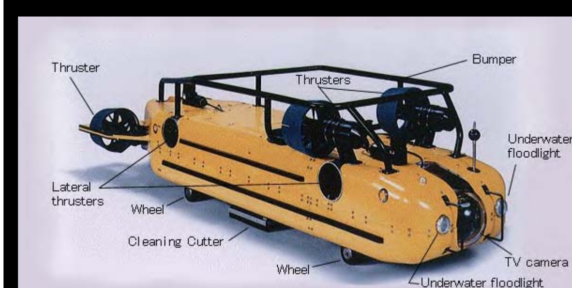


Automation Today

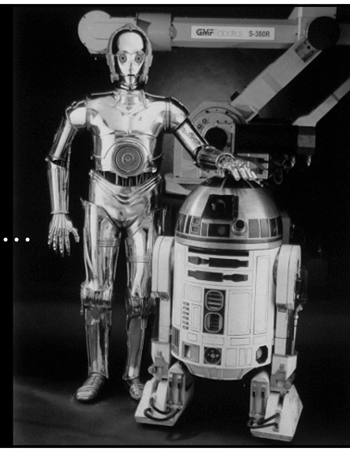
The world around us

Rapid Growth of Machine Intelligence

Robots for Hazardous Areas



Around the House



Warm
And
Cuddly …

Control Systems

Closed-loop control.
Benefits:

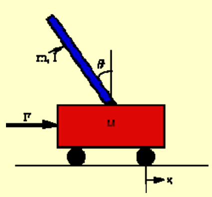
- System corrects “errors” (e.g. your fridge corrects for temperature variations due to door openings and other events.)
- Labor saving

Drawbacks:

- More expensive and complex.
- Need for sensors
- System can become unstable

Control System Example: Inverted Pendulum

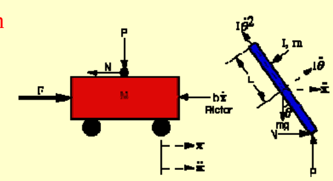
The Problem: The cart with an inverted pendulum is "bumped" with an impulse force, F. Determine the dynamic equations of motion for the system, and find a controller to stabilize the system.



Control System Example: Inverted Pendulum

Force analysis and system equations

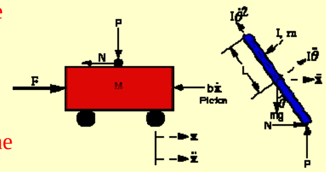
At right are the two *Free Body Diagrams* of the system.



Control System Example: Inverted Pendulum

Equation of motion for the cart:

$$M\ddot{x}+b\dot{x}+N=F$$

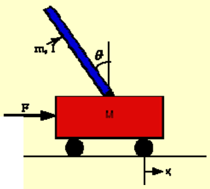


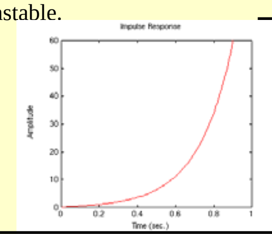
Equation of motion for the pendulum:

$$N=m\ddot{x}+ml\ddot{\theta}\cos\theta-ml\dot{\theta}^2\sin\theta$$

Control System Example: Inverted Pendulum

Without control we get the velocity response shown below, i.e. the pendulum falls to one side and the system is unstable.





Control System Example: Inverted Pendulum

The equation at right describes the four "States" of the system:

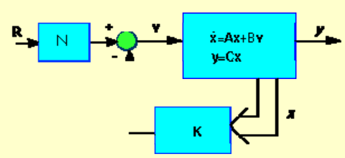
Cart Pos. **x**
Cart Vel **x-dt**
Angle **Phi**
Ang. Vel. **Phi-dt**

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{\phi}\\ \ddot{\phi}\end{bmatrix}=\begin{bmatrix}0 & 1 & 0 & 0\\ 0 & \frac{-(I+ml^2)b}{I(M+m)+Mml^2} & \frac{m^2gl^2}{I(M+m)+Mml^2} & 0\\ 0 & 0 & 0 & 1\\ 0 & \frac{-mlb}{I(M+m)+Mml^2} & \frac{mgl(M+m)}{I(M+m)+Mml^2} & 0\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ \phi\\ \dot{\phi}\end{bmatrix}+\begin{bmatrix}0\\ \frac{I+ml^2}{I(M+m)+Mml^2}\\ 0\\ \frac{ml}{I(M+m)+Mml^2}\end{bmatrix}u$$

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

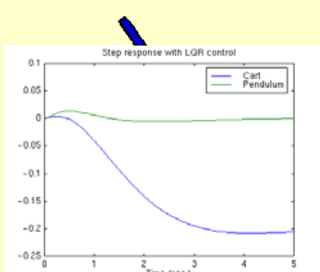
Control System Example: Inverted Pendulum

Control Loop Schematic:
R = 'reference' = desired state
K = 'state controller'



Control System Example: Inverted Pendulum

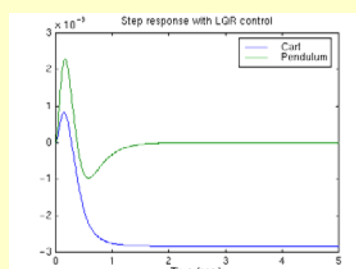
After some mathematical analysis, the controller stabilizes the inverted pendulum:



Control System Example: Inverted Pendulum

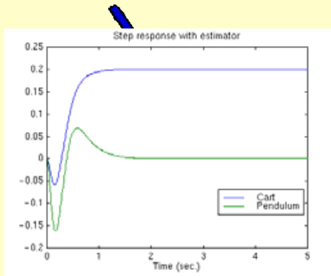
We can make the system respond faster, but it will oscillate more:

If we drive the controller gain too high, the system will become unstable.



Control System Example: Inverted Pendulum

There is still a small problem:
The cart has wandered off too far. So we add a requirement to return to (almost) where it started:



I-clicker Question 1

Closed-loop control is used to

(A) Read measurement data from a process
(B) Keep Temperatures constant
(C) Maintain a process variable as closely as possible to a desired reference.
(D) Compare a process output with a reference

The Airplane as Computer Peripheral

The Future of Aviation

No lab the first week

Contact hours: MW after class

HW: Submit in class
I-Clicker question Topics : Closed loop definitions, Transfer function, incl. computation.
Practice!

The Future: More Automation. Manufacturing

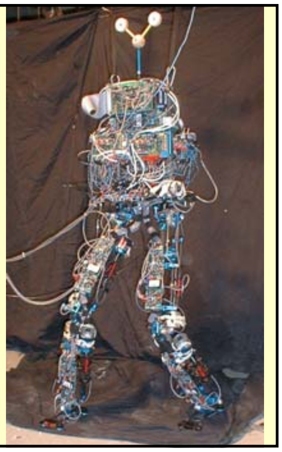
Households and Service Industries: Repetitive Jobs will be automated.

Robotic mower for Golf courses. (Carnegie-Mellon)

Medicine:
Robodoc (Surgical Robot for hip replacement)

Medicine
Prosthetics

Mobility Assistance



Artificial Intelligence
With better brains and sensors, robots will interact better with humans, and perform more functions.

Sony's 'Aibo'

…Remember the poor poet who fell in love with the robot doll?

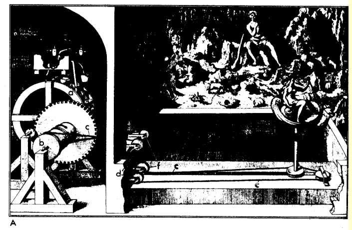
'Love' is reality for many Aibo owners who seem to think that their robot loves them.

Quoted from: NY Times, May2, 2002

DIANE wasn't well. Her owner, Harry Brattin, placed a white muffler around her neck. She sat quietly on a metal desk in the meeting room while the others scampered around the floor playing.
"I get very sad when one of my dogs gets ill," said Mr. Brattin, 63, a motorcycle dealer from San Diego. "When Diane's head stopped moving I felt bad. I truly felt grief."
Diane is an Aibo, a computer-controlled robot made by Sony, and D.H.S. is Droopy Head Syndrome, which is caused when a clutch wears out (it's repairable by replacing the head). Weird, perhaps, but not unusual.

Exploration

Control Systems in Entertainment



A One of Salomon de Caus's theatrical sets at Hellbrunn Castle, in which movable figures were placed in mountain caves. Via a gear transmission, a water-wheel (*a*) drove a shaft (*b*) to which both ends of a rope (*c*) were attached. The rope ran over pulleys (*d*) and round a wheel (*e*) at the bottom of a pillar, which carried a mythological figure. Thereupon, the rope ran over more pulleys (*f*) and back to the shaft.

Control Systems in Entertainment

Control Systems in Entertainment

Control Systems in Entertainment

Control

Open-Loop,

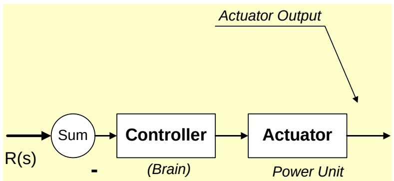
Benefit:

- Simple, always stable
- Widely used in well-defined situations, e.g. Batch filling

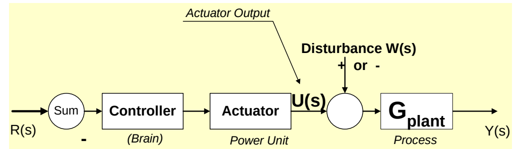
Closed-Loop

- Maintains desired output in the presence of disturbances
- Can become unstable

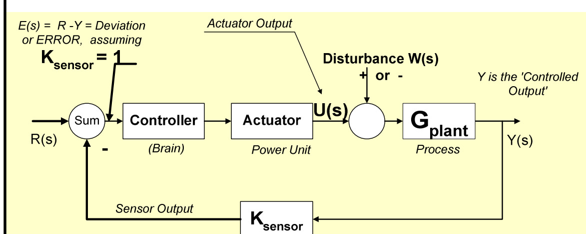
Feedback Control



Feedback Control



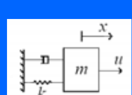
Feedback Control



Chapter 2:
Dynamic Models

Differential Equations in
State-Variable Form

State-Variables: Example



$$\begin{cases} m\dot{v} + bv + kx = u \\ \dot{x} = v \end{cases}$$

- x is the variable that describes any arbitrary position of the system (also called **system variable**)
- x and $\dot{x}$ are the **state-variables** of the system.
- Since $\dot{x} = v$, the state-variables can be defined as x and v

State-Variable Form

Deriving differential equations in state-variable form consists of writing them as a **vector equation** as follows:

$$\dot{X} = F\ X + G\,u$$

where $y = H\ X + J\ u$ is the output

and u is the input

Definitions

- X is the state vector. It contains n elements for an nth-order system, which are the n state-variables of the system.

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Important

- $\dot{X}$ is called **state of the system**.

$$\dot{X} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix}$$

You can forget that

- The constant J is called direct transmission term

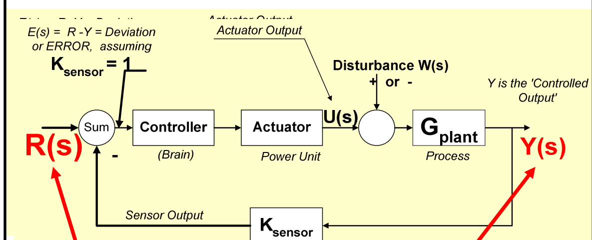
Deriving the State Variable Form requires to specify F, G, H, J for a given X and u

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \underbrace{\begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ f_{n1} & \cdots & \cdots & f_{nn} \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix}}_{=G} u$$

$$\text{where } y = \underbrace{\begin{bmatrix} h_1 & h_2 & \cdots & h_n \end{bmatrix}}_{=H} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + J\,u \quad \text{is the output}$$

and u is the input

Feedback Terminology



In Block diagrams, we use not the time domain variables, but their Laplace Transforms. Always denote Transforms by **(s)**!

I-clicker Question 1

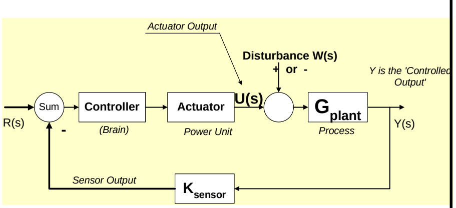
A necessary part of any Closed-loop control is

(A) Negative Feedback of the controlled variable, y
(B) Positive Feedback of the controlled variable, y
(C) Negative Feedback of the Disturbance, w
(D) Positive Feedback of the Disturbance, w

I-clicker Question 2

In a Closed-loop feedback system, the error is

(A) r - y
(B) y - w
(C) w - r
(D) y - w



Each element or 'Block' has **one** input and **one** output variable

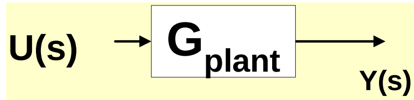
Transfer Function

$$G(s) = \frac{\text{Output(s)}}{\text{Input(s)}} \quad \text{e.g.} = \frac{Y(s)}{U(s)}$$

Each element or ‘Block’ has **one** input and **one** output variable

Transfer Function

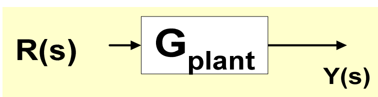
$$G(s) = \frac{Output(s)}{Input(s)} \quad \text{e.g.} = \frac{Y(s)}{U(s)}$$



For instance, the plant in the preceding block diagram can be modeled as:

$$G_{plant}(s) = {}^{Y(s)}/_{U(s)}$$

Transfer Function



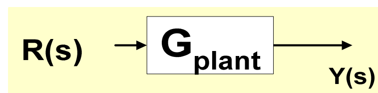
Example: System with Input var. r(t), Output var. y(t):

$4\dot{y} + y(t) = r(t)$ Laplace operator: s = d/dt

4 *s*Y(s) + Y(s) = R(s)

Regroup:

Transfer Function



4 *s*Y(s) + Y(s) = R(s)

Regroup:

Y(s)*(4s+1) = R(s)

Regroup:

$$\frac{Y(s)}{R(s)} = \frac{1}{4s + 1}$$

I-clicker Question 3

A Transfer function, T(s) is defined as

(A) the initial values of a dynamic system
(B) the amplification or gain of a system
(C) The ratio Output/Input of the Laplace transform of an input-output differential equation
(D) The Laplace transform of the reference transmitted from the input to the output.
(E) The Laplace transform of the Feedback from the output to the Input

I-clicker Question 4

Compare a differential equation (DE) with input u and output y, and its Transfer function, T(s):

(A) T(s) can be obtained from the DE, but we cannot reconstruct the DE from T(s)
(B) T(s) can be obtained from the DE, and we can reconstruct the DE from T(s)
(C) we can reconstruct the DE from T(s), but we must redefine the input and output variables
(D) T(s) must be inverted using Laplace transform rules. The result is the solution y(t).

i-CLICKER QUESTION 2.1

An Input-Output Diff. equ. (DE) is converted to State Variable format by

(A) Choosing one state each for the input and the output.

(B) Choosing as many states as the order of the DE, and writing a first order DE for each state.

(C) Grouping the output variable so that the input appears as a matrix

i-CLICKER QUESTION 2.2

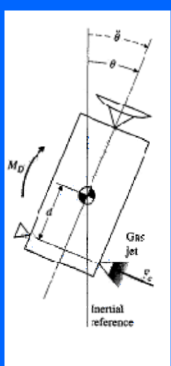
The main advantage of State Variable DE's is:

(A) Neater Appearance.

(B) You can see the matrix symmetries better

(C) Easier Solution with Computers

(D) There is no advantage



Satellite Altitude Control Example

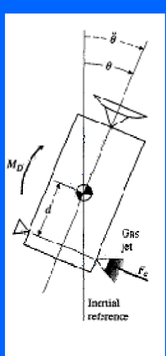
Assumptions:

- ω is the angular velocity
- The desired system output is θ

Q: Dynamic model in state-variable form**?**

Strategy (recommended but not required)

1. Derive the dynamic model.
2. Identify the input control variable, denoted by u.
3. Identify the output variable, denoted by y.
4. Define a state vector, X, having for elements the system variables and their first derivative.
5. Determine $\dot{X}$
6. Determine F and G, in manner that $\dot{X} = FX + Gu$
7. Determine H and J, in manner that $y = HX + Ju$



Ex 1: Dynamic Model

- Applying Newton's law for 1-D rotational motion leads to: $F_c\, d + M_D = I\ddot{\theta}$

=> $$\ddot{\theta} = \frac{F_C\, d + M_D}{I} \quad (1)$$

Example 1 (cont'd)

Given:

- The control input, denoted by *u, is given by:* $u = F_C\, d + M_D$
- The output, denoted by *y, is the displacement angle*: $y = \theta$

Assumption:

- The state vector, denoted by *X, is defined as*:

$$X = \begin{bmatrix} \theta \\ \omega \end{bmatrix} \text{ with } \omega = \dot{\theta}$$

Known: The dynamic model

$$\ddot{\theta} = \frac{F_C\, d + M_D}{I} \quad (1)$$

Required: Rewrite (1) as:

$$\underbrace{\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} \quad & \quad \\ \quad & \quad \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} \theta \\ \omega \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \quad \\ \quad \end{bmatrix}}_{=G} u$$

$$\text{where } y = \underbrace{\begin{bmatrix} \quad & \quad \end{bmatrix}}_{=H} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{\quad}_{=J} \times u$$

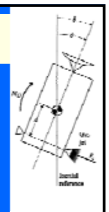
$$\text{and } u =$$

I-clicker question 2.3

The first row of the F-matrix at left is:

(A) 1 1
(B) 1 0
(C) 0 1
(D) 0 0

$$X = \begin{bmatrix} \theta \\ \omega \end{bmatrix} \; with\ \omega = \dot{\theta}$$



Known: The dynamic model

$$\ddot{\theta} = \frac{F_C\, d + M_D}{I} \quad (1)$$

Required: Rewrite (1) as:

$$\underbrace{\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} & \\ & \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} \theta \\ \omega \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \\ \end{bmatrix}}_{=G} u$$

$$where\ y = \underbrace{[\quad]}_{=H} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{\quad}_{=J} \times u$$

$$and\ u =$$

I-clicker question 2.4

The bottom row of the F-matrix at left is:

(A) 1 1
(B) 1 0
(C) 0 1
(D) 0 0

$$X = \begin{bmatrix} \theta \\ \omega \end{bmatrix} \; with\ \omega = \dot{\theta}$$

Known: The dynamic model

$$\ddot{\theta} = \frac{F_C\, d + M_D}{I} \quad (1)$$

Required: Rewrite (1) as:

$$\underbrace{\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} & \\ & \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} \theta \\ \omega \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \\ \end{bmatrix}}_{=G} u$$

$$where\ y = \underbrace{[\quad]}_{=H} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{\quad}_{=J} \times u$$

$$and\ u =$$

Example 1 (cont'd)

- By definition: $\dot{\theta} = \omega$. Thus, $\ddot{\theta} = \dot{\omega}$
- Expressing the dynamic model:

$$\ddot{\theta} = \frac{F_C\, d + M_D}{I} \quad (1)$$

as a function of ω and u (with $u = F_c d + M_D$) yields:

$$\dot{\omega} = \frac{u}{I}$$

Example 1 (cont'd)

Available equations:

- From the dynamic model: $\begin{cases} \dot{\theta} = \omega \\ \dot{\omega} = u/I \end{cases}$
- The output y is defined as: $y = \theta$
- The input u is given by: $u = F_C\, d + M_D$

Equivalent form of available eq:

$$\begin{cases} \dot{\theta} = 0\times\theta + 1\times\omega + 0\times u \\ \dot{\omega} = 0\times\theta + 0\times\omega + (1/I)\times u \end{cases}$$

$$y = 1\times\theta + 0\times\omega + 0\times u$$

where $u = F_C\, d + M_D$

$$\underbrace{\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix}}_{=\dot{X}} \quad \underbrace{\begin{matrix} 1 \\ 0 \end{matrix}\Bigg]}_{=F} \underbrace{\begin{bmatrix} \theta \\ \omega \end{bmatrix}}_{=X} \quad \underbrace{\begin{bmatrix} 0 \\ 1/I \end{bmatrix}}_{=G} u$$

$$where\ y \quad \underbrace{0\,]}_{=H} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{0}_{=J} \times u$$

$$and\ \quad M_D$$

Example 1: Dynamic Model in State-Variable Form

By defining X and u as:

$$X = \begin{bmatrix} \theta \\ \omega \end{bmatrix}$$

$$u = F_C\, d + M_D$$

input

state-variables

The state-variable form is given by:

$$F = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad G = \begin{bmatrix} 0 \\ 1/I \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad J = 0$$

Analysis in Control Systems

- Step 1: Derive a dynamic model
- Step 2: Specify the dynamic model for software by writing it
  - either → in STATE-VARIABLE form
  - or → in terms of its TRANSFER FUNCTION (see chapter 3)

Example 2: Cruise Control Step Response

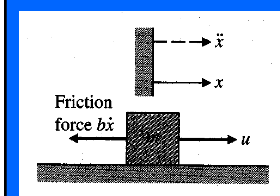
- **Q1**: Rewrite the equation of motion in state-variable form where the output is the car velocity v **?**

- **Q2**: Use MATLAB to find the step response of the velocity of the car **?**
  Assume that the input jumps from being $u(t) = 0\ N$ at time $t = 0\ sec$ to a constant $u(t) = 500\ N$ thereafter.

Reminder: Strategy

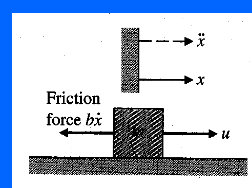
1. Derive the dynamic model.
2. Identify the input control variable, denoted by u.
3. Identify the output variable, denoted by y.
4. Define a state vector, X, having for elements the system variables and their first derivative.
5. Determine $\dot{X}$
6. Determine F and G, in manner that $\dot{X} = FX + Gu$
7. Determine H and J, in manner that $y = HX + Ju$

Ex 2, Clicker Q2.5:Dynamic Model



- Applying Newton's law for translational motion yields:
- (a) $b\dot{x} + u = m\ddot{x}$
- (b) $b\dot{x} = m\ddot{x} + u$
- (c) $-b\dot{x} + u = m\ddot{x}$
- (d) $-b\dot{x} - u = m\ddot{x}$

Ex 2, Q1:Dynamic Model



- Applying Newton's law for translational motion yields:
- (c) $-b\dot{x} + u = m\ddot{x}$

=> $$\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m}$$
(2)

Example 2, Question 1 (cont'd)

Given:

- The input (= external force applied to the system) is denoted by u
- The output, denoted by y, *is the car's velocity*:

$$y = v$$

Assumption:

- The state-vector is defined as: $X = \begin{bmatrix} x \\ v \end{bmatrix}$ *with* $v = \dot{x}$

Known: The dynamic model

$$\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m} \quad (2)$$

Required: Rewrite (2) as:

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x \\ v \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=G} u$$

where $y = \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=H} \begin{bmatrix} x \\ v \end{bmatrix} + \underbrace{\quad}_{=J} \times u$

and $u =$

I-clicker question 2.6

The top row of the F-matrix at left is:

(A) 1 1
(B) 1 0
(C) 0 1
(D) 0 0

$X = \begin{bmatrix} x \\ v \end{bmatrix}$ *with* $v = \dot{x}$

Known: The dynamic model

$$\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m} \quad (2)$$

Required: Rewrite (2) as:

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x \\ v \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=G} u$$

where $y = \underbrace{\begin{bmatrix} \quad \end{bmatrix}}_{=H} \begin{bmatrix} x \\ v \end{bmatrix} + \underbrace{\quad}_{=J} \times u$

and $u =$

I-clicker question 2.7

The bottom row of the F-matrix at left is:

(A) 1 -1

(B) 1 0

(C) 0 -1

(D) 0 0

(E) 0 1

$$X = \begin{bmatrix} x \\ v \end{bmatrix} \text{ with } v = \dot{x}$$

Known: The dynamic model

$$\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m} \quad (2)$$

Required: Rewrite (2) as:

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} & \\ & \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x \\ v \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \\ \end{bmatrix}}_{=G} u$$

$$\text{where } y = \underbrace{\begin{bmatrix} & \end{bmatrix}}_{=H} \begin{bmatrix} x \\ v \end{bmatrix} + \underbrace{\ }_{=J} \times u$$

$$\text{and } u =$$

I-clicker question 2.8

The H-matrix (row vector) at left is:

(A) 1 1

(B) 1 0

(C) 0 1

(D) 0 0

$$X = \begin{bmatrix} x \\ v \end{bmatrix} \text{ with } v = \dot{x}$$

The output y is defined as:

$$y = v$$

Known: The dynamic model $\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m}$

$$(2) \quad \underbrace{\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} & \\ & \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x \\ v \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \\ \end{bmatrix}}_{=G} u$$

Required: Rewrite (2) as:

$$\text{where } y = \underbrace{\begin{bmatrix} & \end{bmatrix}}_{=H} \begin{bmatrix} x \\ v \end{bmatrix} + \underbrace{\ }_{=J} \times u$$

$$\text{and } u =$$

Example 2 (cont'd)

- By definition: $\dot{x} = v$. As a result, $\ddot{x} = \dot{v}$
- Expressing the dynamic model:

$$\ddot{x} = -\frac{b}{m}\dot{x} + \frac{u}{m} \quad (2)$$

as a function of v and u leads to:

$$\dot{v} = -\frac{b}{m}v + \frac{u}{m}$$

Ex 2, Q1 (cont'd)

Available equations:

- From the dynamic model: $\begin{cases} \dot{x} = v \\ \dot{v} = -\frac{b}{m}v + \frac{u}{m} \end{cases}$
- The output y is defined as:

$$y = v$$

- The input is the step function u

Equivalent form of available eq:

$$\begin{cases} \dot{x} = 0 \times x + 1 \times v + 0 \times u \\ \dot{v} = 0 \times x + \left(\frac{-b}{m}\right) \times v + \left(\frac{1}{m}\right) \times u \end{cases}$$

$$y = 0 \times x + 1 \times v + 0 \times u$$

where

Therefore:

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix}}_{=\dot{X}} = \underbrace{\begin{bmatrix} & \\ & \end{bmatrix}}_{=F} \underbrace{\begin{bmatrix} x \\ v \end{bmatrix}}_{=X} + \underbrace{\begin{bmatrix} \\ \end{bmatrix}}_{=G} u$$

$$\text{where } y = \underbrace{\begin{bmatrix} & \end{bmatrix}}_{=H} \begin{bmatrix} x \\ v \end{bmatrix} + \underbrace{\ }_{=J} \times u$$

Example 2, Question 1: Dynamic Model in State-Variable Form

By defining X as:

$$X = \begin{bmatrix} x \\ v \end{bmatrix}$$

The state-variable form results as:

$$F = \begin{bmatrix} 0 & 1 \\ 0 & -b/m \end{bmatrix} \quad G = \begin{bmatrix} 0 \\ 1/m \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad J = 0$$

Example 2, Question2: Step Response using MATLAB?

Assumptions: $m = 1000\ kg$ and $b = 50\ N.sec/m$.

$$F = \begin{bmatrix} 0 & 1 \\ 0 & -0.05 \end{bmatrix} \quad G = \begin{bmatrix} 0 \\ 0.001 \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad J = 0$$

Ex 2, Q2: Step Response with MATLAB?

- The step function in MATLAB calculates the time response of a linear system to a unit **step input**.
- In the problem at hand, the input u is a step function of amplitude 500 N:

  u = 500 * unity step function.

- Because the system is linear ($\dot{X} = FX + Gu$):

  $G * u$ = (500 * G) * unity step function

G * Step 0 to 500 *N* $\longrightarrow$ **500*G * Step 0 to 1 *N***

MATLAB Statements

F = [0 1;0 -0.05];	% defines state variable matrices
G = [0;0.001];	% defines system by its state-space matrices
H = [0 1];	
J = 0;	
sys = ss(F, 500*G, H, J);	% setup time vector ($dt = 0.2\ sec$)
t = 0:0.2:100;	% computes the response to a unity step response
y = step(sys,t);	
plot (t,y)	% plots output (*i.e.*, step response)

Response of the car velocity to a step input u of amplitude 500 N