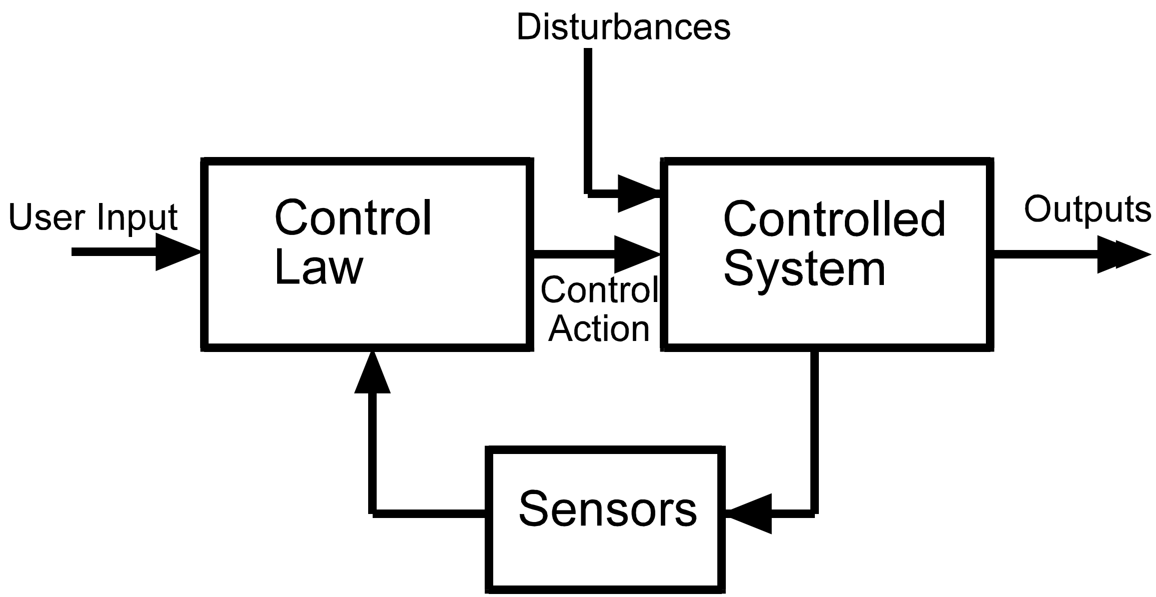
41014 Sensors and Control of Mechatronic Systems

Control Part 1:
Linear Continuous-Time Systems

Dr. Liang Zhao

Centre for Autonomous Systems
School Mechanical and Mechatronic Systems
Faculty of Engineering and Information Technology

Introduction



- **Objective of the control law is to manipulate one or more of the system inputs to achieve desired outputs**
- **Systems with control laws that make use of the measurements obtained through sensors are called "Closed-Loop Control Systems"**

Linear Continuous-Time State Space Models

A continuous-time linear time-invariant state space model takes the form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \qquad x(t_0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

where $x \in \mathrm{R}^n$ is the state vector, $u \in \mathrm{R}^m$ is the control signal, $y \in \mathrm{R}^p$ is the output, $x_0 \in \mathrm{R}^n$ is the state vector at time $t = t_0$ and A, B, C, **and** D **are matrices of appropriate dimensions.**

State space models use first order Ordinary Differential Equations (ODE)

Why State Space model?

There are many alternative model formats that can be used for linear dynamic systems.

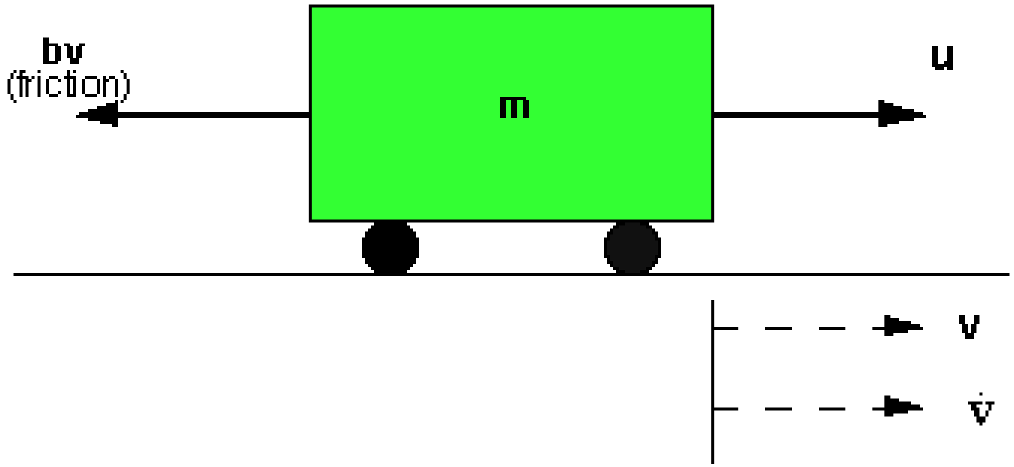
Advantages of state space models:

- Naturally come from many physical systems
- Easily deal with multivariable case
- Suitable for studying nonlinear models

Example 1: Simplified Dragging Control

- **Objective: Control the speed v**
- **Input: Force u**
- **Output: Speed v**

Assume the friction force is proportional to the speed, the model of the vehicle motion is



$$\boxed{m\dot{v} = u - bv.}$$

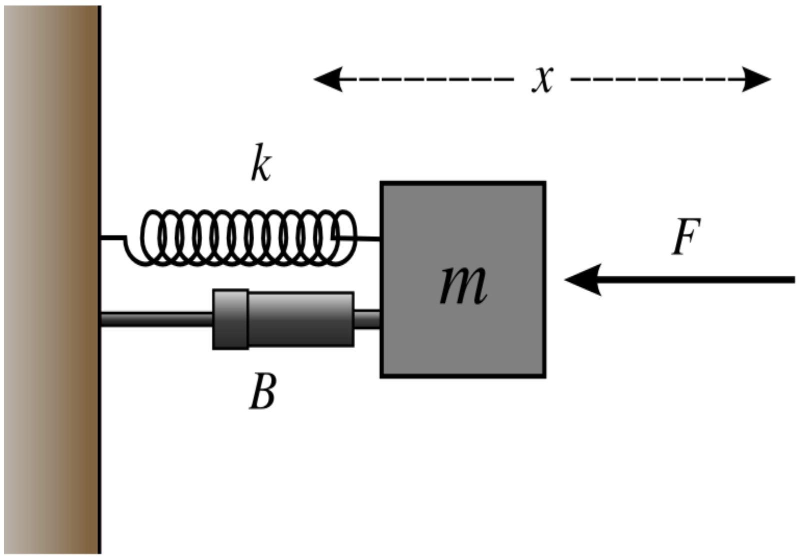
State space model: State -- v

$$\begin{aligned} \dot{v} &= -\frac{b}{m}v + \frac{1}{m}u \\ y &= v \end{aligned}$$

$$A = -\frac{b}{m}, \quad B = \frac{1}{m}, \quad C = 1, \quad D = 0$$

Example 2: Mass-spring-damper system

- **Objective: Control the position x**
- **Input: Force F**
- **Output: position x**



Oscillatory force from spring

$$F_s = -kx$$

where k is the spring constant.
Damping force

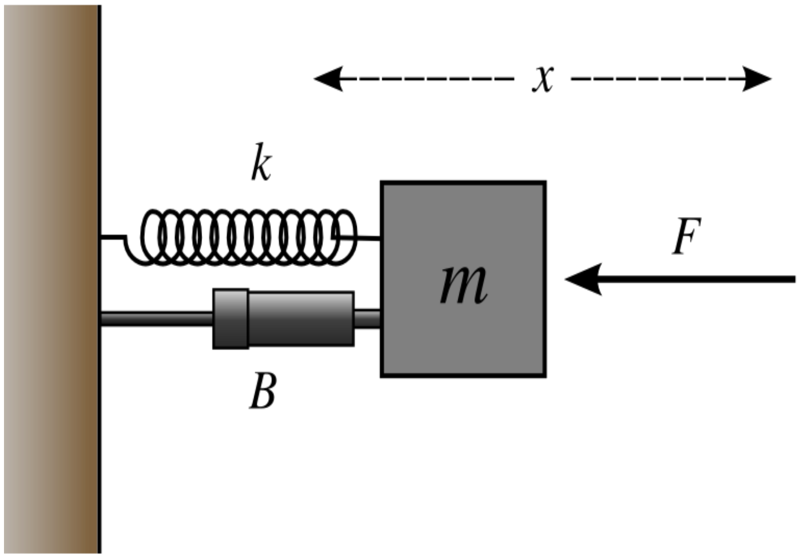
$$F_d = -cv = -c\frac{dx}{dt} = -c\dot{x}$$

where c is damping coefficient.
Applying Newton's second law,

$$m\ddot{x} + c\dot{x} + kx = F. \longrightarrow \ddot{x} = -\frac{c}{m}\dot{x} - \frac{k}{m}x + \frac{1}{m}F.$$

Example 2: Mass-spring-damper system

- **Objective: Control the position x**
- **Input: Force F**
- **Output: position x**



$$\ddot{x} = -\frac{c}{m}\dot{x} - \frac{k}{m}x + \frac{1}{m}F.$$

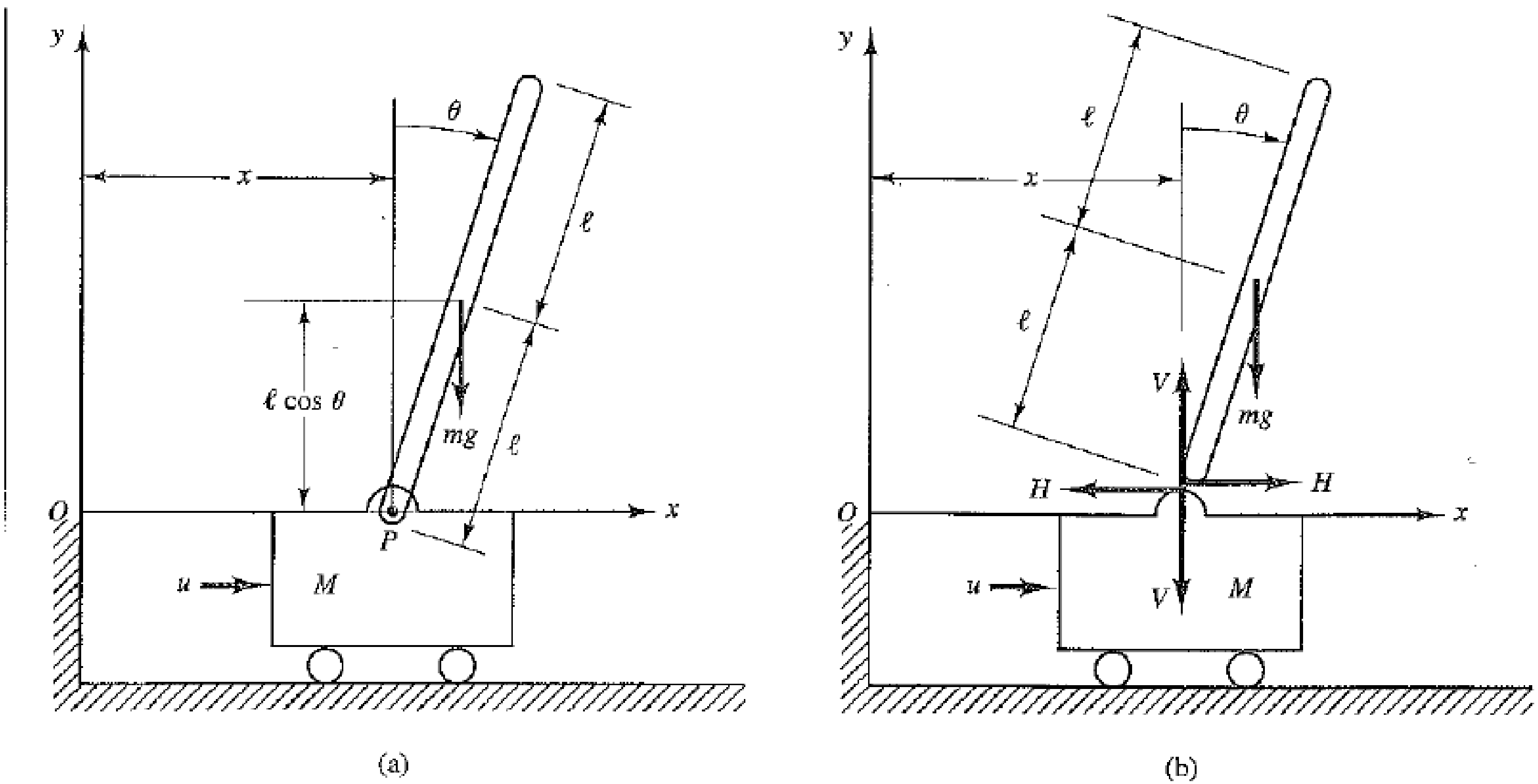
Introduce state vector $X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$

State space model:

$$\boxed{\begin{aligned} \dot{X} &= AX + BF \\ y &= CX \end{aligned}}$$

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}, \quad C = [1 \quad 0], \quad D = 0$$

Example 4: Inverted Pendulum



Model of the system ---- refer to the notes for details

Example 4: Inverted Pendulum

Define state variables x_1, x_2, x_3, and x_4 by

$$
\begin{aligned}
x_1 &= \theta \\
x_2 &= \dot{\theta} \\
x_3 &= x \\
x_4 &= \dot{x}
\end{aligned}
$$

We consider θ and x as the outputs of the system,

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \theta \\ x \end{bmatrix} = \begin{bmatrix} x_1 \\ x_3 \end{bmatrix}$$

Example 4: Inverted Pendulum

The state space model is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{M+m}{Ml}g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{m}{M}g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{Ml} \\ 0 \\ \frac{1}{M} \end{bmatrix} u$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

State space models

More models for some interesting systems are available at

http://www.engin.umich.edu/class/ctms/index.htm

(together with MATLAB tutorials)

Solution to the state space model

State space model

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \qquad x(t_0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

The solution is

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$$

where

$$e^{At} = \sum_{k=0}^{\infty} \frac{A^k t^k}{k!}$$

<u>One dimensional example</u>

$$\dot{x}(t) = -x(t), \; x(0) = x_0.$$

MATLAB simulation

Get the state space model:

```
% parameters
m = 1000;
b = 50;

% model
A = [-b/m];
B = [1/m];
C = [1];
D = 0;
drag=ss(A,B,C,D);
```

Simulate the system:

```
% simulation
t = 0:0.1:300;
u = 50*ones(size(t));
x0 = [0];
[Y,T,X]=lsim(cruise,u,t,x0)
```

MATLAB Examples

Stability

The system $\boxed{\dot{x}(t) = Ax(t) \qquad x(t_0) = x_0}$

MATLAB demo

is called asymptotically stable if for any initial state, the state x(t) converges to zero as t increases indefinitely.

Simple example 1

$$\dot{x} = -x, \quad x(0) = x_0$$

The solution is

$$x(t) = e^{-t}x_0$$

Simple example 2

$$\dot{x} = x, \quad x(0) = x_0$$

The solution is

$$x(t) = e^{t}x_0$$

Condition of stability: all the eigenvalues of A have negative real parts.
Eigenvalues of A are the solutions of the equation $|\lambda I - A| = 0$

Example: $A = \begin{bmatrix} -2 & 0 \\ 0 & -3 \end{bmatrix}$

Example: Pendulum

From Newton's second law,

$$ma = m\dot{v} = F_f - mg\sin\theta$$

where F_f is the friction force from the air given by

$$F_f = -kv$$

where v is the velocity.
Moreover

$$v = l\dot{\theta}.$$

Thus we have

$$ml\ddot{\theta} = -kl\dot{\theta} - mg\sin\theta.$$

That is

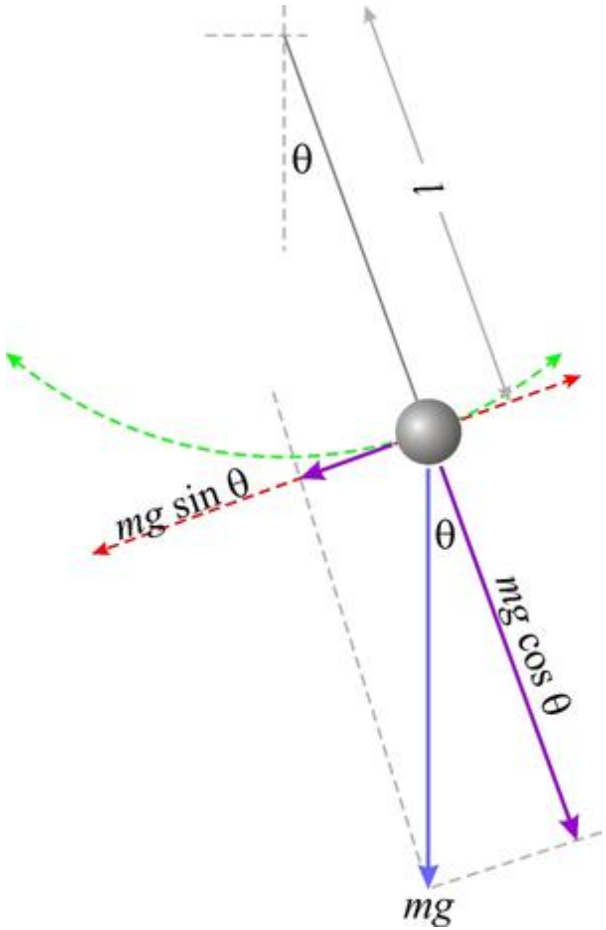
$$\ddot{\theta} = -\frac{k}{m}\dot{\theta} - \frac{g}{l}\sin\theta.$$

When θ is small, applying linearization

$$\sin\theta \approx \theta$$

we have

$$\ddot{\theta} = -\frac{g}{l}\theta - \frac{k}{m}\dot{\theta}.$$



Example: Pendulum

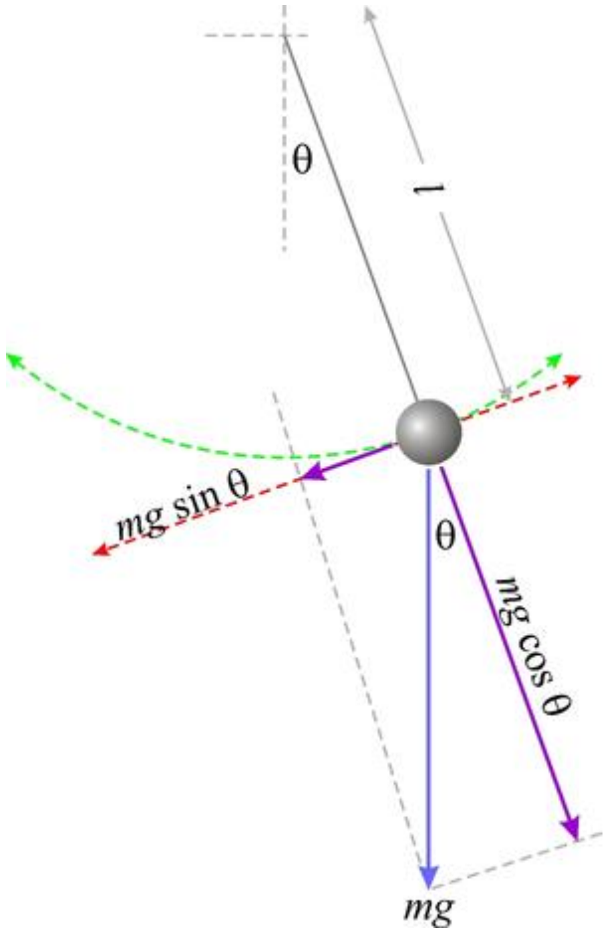
$$\ddot{\theta} = -\frac{g}{l}\theta - \frac{k}{m}\dot{\theta}$$

Denote state vector

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

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{k}{m} \end{bmatrix} X$$

<u>MATLAB demo</u>



Pole Placement (Pole Assignment)

Choose state feedback control law

$$u(t) = -Kx(t)$$

then the closed-loop system becomes

$$\dot{x}(t) = Ax(t) + Bu(t) = Ax(t) - BKx(t) = (A - BK)x(t)$$

We can design the closed-loop system to have good properties by assigning its poles --- the eigenvalues of matrix A-BK.

For example, if

$$\dot{x} = Ax$$

is not stable, we can choose K such that

$$\dot{x} = (A - BK)x$$

is stable.

Pole Placement

Example

$$A = \begin{bmatrix} -3 & 0 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Here

$$\dot{x} = Ax$$

is not stable.

For this example

$$K = [k_1 \quad k_2]$$

and

$$A - BK = \begin{bmatrix} -3 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ k_1 & k_2 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ -k_1 & 1 - k_2 \end{bmatrix}$$

Choose

$$k_1 = 0, \quad k_2 = 2$$

then

$$A - BK = \begin{bmatrix} -3 & 0 \\ 0 & -1 \end{bmatrix}$$

has eigenvalues -3 and -1.

Thus closed-loop system

$$\dot{x} = (A - BK)x$$

is stable.

MATLAB

K=place(A,B,P)

Demo

Pole Placement

We can design the closed-loop system to have good properties by assigning its poles --- the eigenvalues of matrix A-BK.

For example, if

$$\dot{x} = Ax$$

is not stable, we can choose K such that

$$\dot{x} = (A - BK)x$$

is stable.

Necessary and sufficient condition for <u>arbitrary pole placement</u> ---- the system is completely controllable.

Check for Controllability

Theorem: The state space model

$$\dot{x}(t) \;=\; Ax(t) + Bu(t)$$

is completely controllable if and only if the matrix

$$[B \; AB \; A^2B \; \cdots \; A^{n-1}B]$$

has full row rank.

Example

Consider the state space model

$$A = \begin{bmatrix} -3 & 1 \\ -2 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

The controllability matrix is given by

$$[B \ AB] = \begin{bmatrix} 1 & -4 \\ -1 & -2 \end{bmatrix}$$

Clearly, its rank is 2; thus, the system is completely controllable.

Example

Consider the state space model

$$A = \begin{bmatrix} -1 & 1 \\ 2 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

The controllability matrix is given by:

$$[B \ AB] = \begin{bmatrix} 1 & -2 \\ -1 & 2 \end{bmatrix}$$

Its rank = 1 < 2; thus, the system is not completely controllable.

Linear quadratic optimal control

A continuous-time linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$

design state feedback control $u(t) = -Kx(t)$

<u>Different ways to choose K, which way is the best?</u>

1D Example:

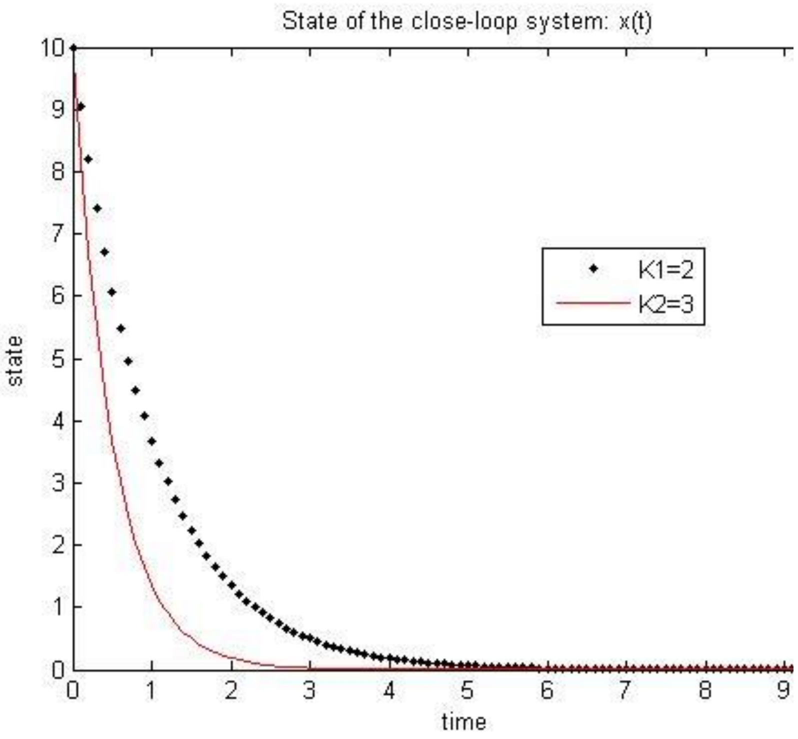
$$\dot{x}(t) = x(t) + u(t) \qquad K_1 = 2, K_2 = 3$$

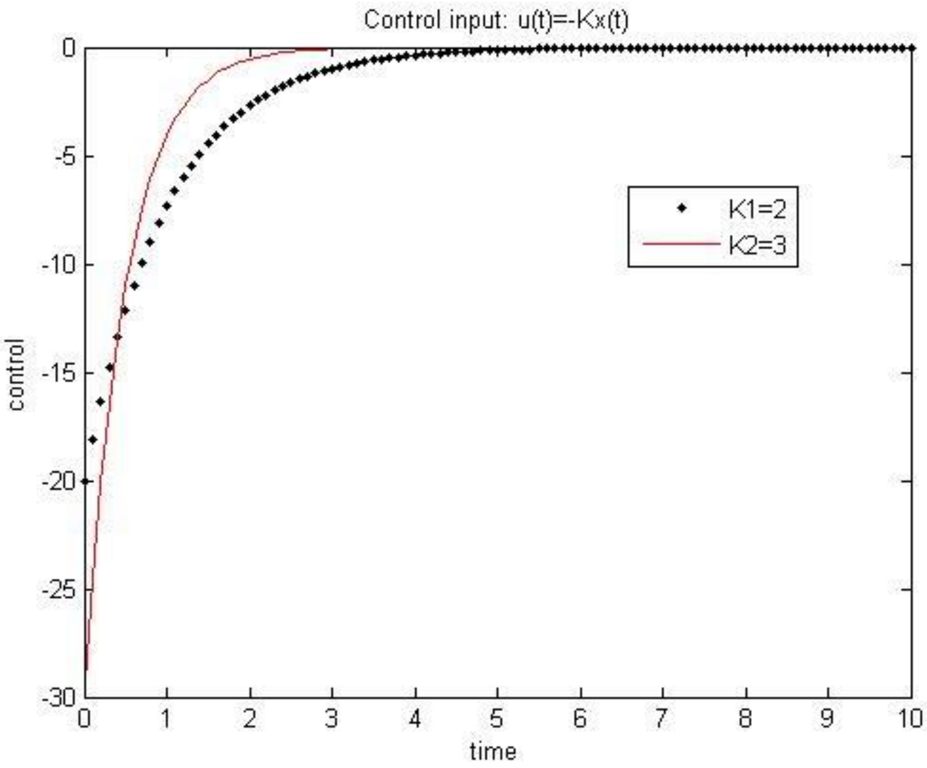
Both the two closed-loop systems are stable.

Linear quadratic optimal control

<u>Different ways to choose K, which way is the best?</u>

1D Example: $\dot{x}(t) = x(t) + u(t)$ $\quad K_1 = 2, K_2 = 3$ $\quad u(t) = -Kx(t)$





Linear quadratic optimal control

A continuous-time linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$

design state feedback control

$$u(t) = -Kx(t)$$

such that the Performance index:

$$J = \int_0^\infty (x^T Q x + u^T R u)dt$$

is minimized.

Example:

$$Q = 2, R = 3$$

$$J = \int_0^\infty (2x^2 + 3u^2)dt$$

What we want:
state close to 0
control close to 0

Linear Quadratic Regulator (LQR)

Optimal Solution = Optimal K:

Solve P from the Riccati equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

then let

$$K = R^{-1}B^T P$$

So the controller is

$$u = -R^{-1}B^T Px$$

and the minimal performance index

$$J = x^T(0)Px(0)$$

Example:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 1$$

Solve the Riccati equation,

$$P = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

Linear Quadratic Regulator (LQR)

Optimal Solution = Optimal K:

Solve P from the Riccati equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

then let

$$K = R^{-1}B^T P$$

So the controller is

$$u = -R^{-1}B^T P x$$

and the minimal performance index

$$J = x^T(0)Px(0)$$

MATLAB

[K P E] =lqr(A,B,Q,R)

P=care(A,B,Q,R)

Example:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 1$$

Summary

- Stability of the system can be studied from the eigenvalues of the matrix A.
- Pole placement is using feedback control law u=-Kx to place the poles of the closed-loop system --- the eigenvalues of A-BK.
- Linear optimal control is to find u=-Kx such that the Performance index

$$J = \int_0^\infty (x^T Q x + u^T R u) dt$$

is minimized.

Solution:

Solve P from the Riccati equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

then let

$$K = R^{-1}B^T P$$

So the controller is

$$u = -R^{-1}B^T Px$$

and the minimal performance index

$$J = x^T(0)Px(0)$$