Lecture 5

Multivariable control: eigenvalue assignment

Giancarlo Ferrari Trecate[1]

[1]Dependable Control and Decision Group
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
giancarlo.ferraritrecate@epfl.ch

Outline of the lecture

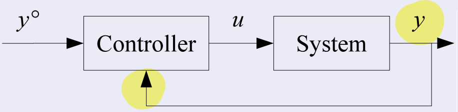
- Classification of control schemes
- The eigenvalue assignment (EA) problem
  - Systems with scalar input - the Ackermann's formula
- EA for MIMO systems
  - Approximate methods

Control schemes: output feedback

DT nonlinear system

$$x^+ = f(x, u)$$
$$y = h(x, u)$$

Output feedback

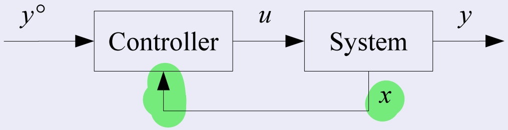


- $y^o(k)$: setpoint
- $u(k)$: control variable

Output feedback: the controller uses the setpoint and a measurement of the output to compute the control variable

Control schemes: state feedback

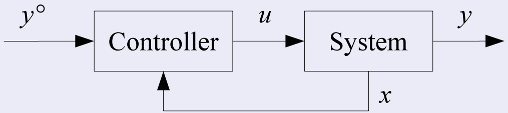
State feedback



State feedback: the controller uses the setpoint and a measurement of the state for computing the control variable

Control schemes: state feedback

State feedback



State feedback: the controller uses the setpoint and a measurement of the state for computing the control variable

Pros

Since $y = h(x, u)$ the output can only contain less information than the state. Therefore, state feedback usually guarantees better performances

Cons

The state must be measured and this is not always the case. Otherwise the state must be estimated from measurements of u and y

Control problems

Terminology

- *Regulation*: make a desired equilibrium state AS
- *Tracking*: make the system output track, according to given criteria, special classes of setpoints y^o

In both problems disturbances must be also attenuated or rejected.

Control problems

Terminology

- *Regulation*: make a desired equilibrium state AS
- *Tracking*: make the system output track, according to given criteria, special classes of setpoints y^o

In both problems disturbances must be also attenuated or rejected.

Taxonomy of controllers

- *Static*: the controller is a static system (e.g. proportional control $u(k) = \kappa(y(k) - y^o(k))$
- *Dynamic*: the controller is a dynamic system (e.g. PID controllers)

Topics that will be covered in this course

Static and dynamic controllers for LTI discrete-time systems

Stabilization of the origin

Regulation problem

$$x^+ = f(x, u)$$

Design the control law $u(k) = \kappa(x(k))$ $\kappa : \mathbb{R}^n \to \mathbb{R}$ such that the origin of the closed-loop system

is an AS equilibrium state $\quad x^+ = f(x, \kappa(x))$

Stabilization of the origin

Regulation problem

$$x^+ = f(x, u)$$

Design the control law $u(k) = \kappa(x(k))$ $\kappa : \mathbb{R}^n \to \mathbb{R}$ such that the origin of the closed-loop system

$$x^+ = f(x, \kappa(x))$$

is an AS equilibrium state

Remarks

- Several industrial systems are designed to work around a *nominal operation point* $(\bar{x}, \bar{u})$ that must be stabilized by the controller
- Linearization about this point produces an LTI system Σ_L with state $x - \bar{x} \to$ stabilisation of Σ_L about the origin often implies stabilisation of the original system about $\bar{x}$

Stabilization of the origin

Regulation problem

$$x^+ = f(x,u)$$

Design the control law $u(k) = \kappa(x(k))$ $\kappa : \mathbb{R}^n \to \mathbb{R}$ such that the origin of the closed-loop system

$$x^+ = f(x, \kappa(x))$$

is an AS equilibrium state

Remarks

- Several industrial systems are designed to work around a *nominal operation point* $(\bar{x}, \bar{u})$ that must be stabilized by the controller
- Linearization about this point produces an LTI system Σ_L with state $x - \bar{x}$ $\rightarrow$ stabilisation of Σ_L about the origin often implies stabilisation of the original system about $\bar{x}$
- Stabilization of the origin is also at the core of the design of controllers for tracking problems
- For the sake of simplicity, in most cases we will neglect the presence of disturbances

State-feedback controllers - LTI systems

Multi-input LTI system

$$x^+ = Ax + Bu, \quad x(k) \in \mathbb{R}^n, \; u(k) \in \mathbb{R}^m$$

Control law

Some books consider $u(k) = -Kx(k)$

$$u(k) = Kx(k), \quad K \in \mathbb{R}^{m \times n} \text{ to be designed for stabilizing } \bar{x} = 0$$

Closed-loop system: $$x^+ = (A + BK)\, x$$

Eigenvalue Assignment (EA) problem

Compute, if possible, K such that the eigenvalues of $A + BK$ take prescribed values (real or in complex conjugate pairs)

Solution to the EA problem

Theorem

The EA problem can be solved if and only if the LTI system is reachable

Review

The system $x^+ = Ax + Bu$ is reachable if and only if the matrix

$$M_r = \left[\ B \mid AB \mid A^2B \mid \cdots \mid A^{n-1}B\ \right]$$

has maximal rank.

- M_r: reachability matrix
- Terminology: the pair (A, B) is reachable

2h 24 ↓

Solution to the EA problem - single input

Definition

Let $u(k) \in \mathbb{R}$. The pair (A, B) is in the canonical controllability form if

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b \end{bmatrix}, \; b \neq 0$$

Solution to the EA problem - single input

Definition

Let $u(k) \in \mathbb{R}$. The pair (A, B) is in the canonical controllability form if

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b \end{bmatrix}, \quad b \neq 0$$

Remarks

- If (A, B) is the canonical controllability form, then M_r has maximal rank by construction
- Let $p_A(\lambda)$ be the characteristic polynomial of A. By construction, one has

$$p_A(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda + a_0$$

Solution to the EA problem - single input

- Structure of the canonical controllability form

$$\left.\begin{aligned} x_1^+ &= x_2 \\ x_2^+ &= x_3 \\ &\vdots \\ x_{n-1}^+ &= x_n \end{aligned}\right\} \leftarrow \text{ shift register storing the last } n-1 \text{ states}$$

$$x_n^+ = a(x) + bu \leftarrow \text{ the input acts on } x_n^+$$

where $a(x) = -a_0 x_1 - a_1 x_2 - \ldots - a_{n-1} x_n$

Idea

If the LTI system is in the canonical controllability form, choose

$$u = \underbrace{\frac{1}{b}(-a(x))}_{\text{this cancels } a(x)} + \frac{1}{b}\tilde{u}$$

such that the auxiliary input $\tilde{u}$ assigns the closed-loop eigenvalues

Solution to the EA problem - single input

Algorithm

Let (A, B) be in canonical controllability form

- For given desired closed-loop eigenvalues $\tilde{\lambda}_1, \tilde{\lambda}_2, \ldots, \tilde{\lambda}_n$, build up the polynomial

$$p^D(\lambda) = (\lambda - \tilde{\lambda}_1)(\lambda - \tilde{\lambda}_2)\cdots(\lambda - \tilde{\lambda}_n) = \lambda^n + \tilde{a}_{n-1}\lambda^{n-1} + \cdots + \tilde{a}_1\lambda + \tilde{a}_0$$

- Use

$$u = \frac{1}{b}(-a(x) + \tilde{a}(x))$$

where $\tilde{a}(x) = -\tilde{a}_0 x_1 - \tilde{a}_1 x_2 - \ldots - \tilde{a}_{n-1} x_n$.

Solution to the EA problem - single input

Closed-loop system

$$\left.\begin{aligned} x_1^+ &= x_2 \\ &\vdots \\ x_{n-1}^+ &= x_n \end{aligned}\right\} \text{ shift register storing the last } n-1 \text{ states}$$

$$x_n^+ = \tilde{a}(x)$$

The matrix $\tilde{A}$ of the closed-loop system $x^+ = \tilde{A}x$ is in the canonical controllability form: by construction $p^D(\lambda)$ is the closed-loop characteristic polynomial

Matrix K (gain matrix)

$$u = \frac{1}{b}(-a(x) + \tilde{a}(x)) =$$

$$= \frac{1}{b}((a_0 - \tilde{a}_0)\,x_1 + (a_1 - \tilde{a}_1)\,x_2 + \cdots + (a_{n-1} - \tilde{a}_{n-1})\,x_n) = Kx$$

with $K = \frac{1}{b}\begin{bmatrix}(a_0 - \tilde{a}_0) & (a_1 - \tilde{a}_1) & \cdots & (a_{n-1} - \tilde{a}_{n-1})\end{bmatrix}$

Solution to the EA problem - single input

How to solve the EA problem if the LTI system is not in the canonical controllability form ?

Solution to the EA problem - single input

How to solve the EA problem if the LTI system is not in the canonical controllability form ?

Lemma

If (A, B) is reachable, there is an invertible matrix T such that the equivalent system

$$\hat{x}^+ = \hat{A}\hat{x} + \hat{B}u, \quad \hat{A} = TAT^{-1}, \hat{B} = TB$$

where $\hat{x} = Tx$, is in the canonical controllability form with $b = 1$.

Solution to the EA problem - single input

How to solve the EA problem if the LTI system is not in the canonical controllability form ?

Lemma

If (A, B) is reachable, there is an invertible matrix T such that the equivalent system

$$\hat{x}^+ = \hat{A}\hat{x} + \hat{B}u, \quad \hat{A} = TAT^{-1}, \hat{B} = TB$$

where $\hat{x} = Tx$, is in the canonical controllability form with $b = 1$.

Computation of T

→ reachability matrix of (A,B)

$$\left.\begin{array}{l} M_r = \left[\begin{array}{c|c|c|c|c} B & AB & A^2B & \cdots & A^{n-1}B \end{array}\right] \\ \hat{M}_r = \left[\begin{array}{c|c|c|c|c} \hat{B} & \hat{A}\hat{B} & \hat{A}^2\hat{B} & \cdots & \hat{A}^{n-1}\hat{B} \end{array}\right] = TM_r \end{array}\right\} \rightarrow T = \hat{M}_r M_r^{-1}$$

↳ reachability matrix of $(\hat{A}, \hat{B})$

$\hat{A}^2\hat{B} = TAT^{-1}TAT^{-1}TB$

Solution to the EA problem - single input

Algorithm

Given A, B and the desired closed-loop characteristic polynomial

$$p^D(\lambda) = \lambda^n + \tilde{a}_{n-1}\lambda^{n-1} + \cdots + \tilde{a}_1\lambda + \tilde{a}_0$$

1. compute M_r and verify that (A, B) is reachable
2. compute
$$p_A(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda + a_0$$
3. build[a] $\hat{A}$, $\hat{B}$ and $\hat{M}_r$. Compute $T = \hat{M}_r M_r^{-1}$
4. build[b] $\hat{K} = \begin{bmatrix}(a_0 - \tilde{a}_0) & (a_1 - \tilde{a}_1) & \cdots & (a_{n-1} - \tilde{a}_{n-1})\end{bmatrix}$
5. compute $K = \hat{K}T$ an set $u = Kx$

[a] $\hat{A}$ and $\hat{B}$ are in the canonical controllability form with $b = 1$. For the computation it is enough to know $p_A(\lambda)$.

[b] Controller design in the coordinates $\hat{x}$.

Ackermann's formula

In the previous algorithm one can avoid the use of $\hat{x}$ coordinates and design directly the controller K as a function of A and B.

Theorem

Let (A, B) be a reachable pair and let

$$p^D(\lambda) = \lambda^n + \tilde{a}_{n-1}\lambda^{n-1} + \cdots + \tilde{a}_1\lambda + \tilde{a}_0$$

be the desired closed-loop polynomial. Then, the controller $u = Kx$ such that the characteristic polynomial of $A + BK$ is $p^D(\lambda)$ is given by

$$K = -\begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix} M_r^{-1} p^D(A) \tag{1}$$

↳ reachability matrix of (A, B)

Equation (1) is called the Ackermann's formula

Proof of the Ackermann's formula

Being $\hat{A}$ in in the canonical controllability form, one can verify that the first row of $\hat{A}^i$, $1 \leq i < n$ is composed by zero entries except the entry in position $(1, i+1)$ that is 1

$$\hat{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ x & x & x & \cdots & x & x \end{bmatrix} \quad \hat{A}^2 = \begin{bmatrix} 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ x & x & x & \cdots & x & x \\ x & x & x & \cdots & x & x \end{bmatrix}$$

"don't care" elements

$$\hat{A}^{n-1} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \\ x & x & x & \cdots & x & x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x & x & x & \cdots & x & x \\ x & x & x & \cdots & x & x \\ x & x & x & \cdots & x & x \end{bmatrix}$$

Proof of the Ackermann's formula

→ coeff. of the characteristic polynomial of $\hat{A}$

Since from the Cayley-Hamilton theorem one has
$\hat{A}^n + a_{n-1}\hat{A}^{n-1} + \cdots + a_1\hat{A} + a_0 I = 0$, it follows that

$$p^D(\hat{A}) = p^D(\hat{A}) - 0 = \hat{A}^n + \tilde{a}_{n-1}\hat{A}^{n-1} + \cdots + \tilde{a}_1\hat{A} + \tilde{a}_0 I$$

$$-\hat{A}^n - a_{n-1}\hat{A}^{n-1} - \cdots - a_1\hat{A} - a_0 I =$$

$$(\tilde{a}_{n-1} - a_{n-1})\hat{A}^{n-1} + \cdots + (\tilde{a}_0 - a_0)I$$

$$p^D(\hat{A}) = \begin{bmatrix} (\tilde{a}_0 - a_0) & (\tilde{a}_1 - a_1) & (\tilde{a}_2 - a_2) & \cdots & (\tilde{a}_{n-1} - a_{n-1}) \\ x & x & x & \cdots & x \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x & x & x & \cdots & x \\ x & x & x & \cdots & x \\ x & x & x & \cdots & x \end{bmatrix}$$

and therefore the controller $\hat{K}$ we have computed before is given by

$$\hat{K} = -\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} p^D(\hat{A})$$

Proof of the Ackermann's formula

Since $\hat{A} = TAT^{-1}$, $T = \hat{M}_r M_r^{-1}$, $K = \hat{K}T$ one has

$$K = -\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} p^D(\hat{A})T = \tag{2}$$

$$= -\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} Tp^D(A)T^{-1}T = \tag{3}$$

$$= -\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \hat{M}_r M_r^{-1} p^D(A) \tag{4}$$

Proof of the Ackermann's formula

Since $\hat{A} = TAT^{-1}$, $T = \hat{M}_r M_r^{-1}$, $K = \hat{K}T$ one has

$$\begin{aligned} K &= -\begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix} p^D(\hat{A})T = && (2)\\ &= -\begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix} Tp^D(A)T^{-1}T = && (3)\\ &= -\begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix} \hat{M}_r M_r^{-1} p^D(A) && (4)\end{aligned}$$

For getting rid of $\hat{M}_r$, we observe that, since $\hat{A}$ and $\hat{B}$ are in canonical controllability form, one has

$$\hat{M}_r = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1\\ 0 & 0 & 0 & \cdots & 1 & x\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 1 & \cdots & x & x\\ 0 & 1 & x & \cdots & x & x\\ 1 & x & x & \cdots & x & x \end{bmatrix}$$

Therefore, $-\begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix}\hat{M}_r = -\begin{bmatrix}0 & 0 & \cdots & 1\end{bmatrix}$.

Example

Problem

$$x_1^+ = x_1 + x_2 + u$$
$$x_2^+ = u$$

Compute a state-feedback controller such that the closed-loop system has all eigenvalues equal to $\frac{1}{2}$

Example

Problem

$$x_1^+ = x_1 + x_2 + u$$
$$x_2^+ = u$$

Compute a state-feedback controller such that the closed-loop system has all eigenvalues equal to $\frac{1}{2}$

Desired closed-loop characteristic polynomial

$$p^D(\lambda) = (\lambda - \frac{1}{2})^2 = \lambda^2 + \underbrace{(-1)}_{\tilde{a}_1}\lambda + \underbrace{\frac{1}{4}}_{\tilde{a}_0}$$

Computation of M_r

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \; B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow M_r = [\; B \mid AB \;] = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix}$$

M_r is full rank $\Rightarrow$ EA problem can be solved

Example

Computation of $p_A(\lambda)$

$$p_A(\lambda) = \det\left(\begin{bmatrix} \lambda - 1 & -1 \\ 0 & \lambda \end{bmatrix}\right) = \lambda^2 + \underbrace{(-1)}_{a_1}\lambda + \underbrace{0}_{a_0}$$

Build $\hat{A}$, $\hat{B}$, $\hat{M}_r$ and T

$$\hat{A} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix}$$

$$\hat{A} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \hat{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Rightarrow \hat{M}_r = \left[\ \hat{B} \mid \hat{A}\hat{B}\ \right] = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

$$T = \hat{M}_r M_r^{-1} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Build $\hat{K}$

$$\hat{K} = \left[(a_0 - \tilde{a}_0) \quad (a_1 - \tilde{a}_1)\right] = \left[0 - \frac{1}{4} \quad -1 + 1\right] = \left[-\frac{1}{4} \quad 0\right]$$

Example

Build K

$$K = \hat{K}T = \begin{bmatrix} -\frac{1}{8} & \frac{1}{8} \end{bmatrix}$$

Check the result

$$A + BK = \begin{bmatrix} \frac{7}{8} & \frac{9}{8} \\ -\frac{1}{8} & \frac{1}{8} \end{bmatrix}$$

Eigenvalues of $A + BK$: $\lambda_1 = \lambda_2 = \frac{1}{2}$

Example

Using Ackermann's formula

$$K = -\begin{bmatrix} 0 & 1 \end{bmatrix} M_r^{-1} p^D(A)$$

$$p^D(A) = A^2 - A + \frac{1}{4}I = \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{4} \end{bmatrix}$$

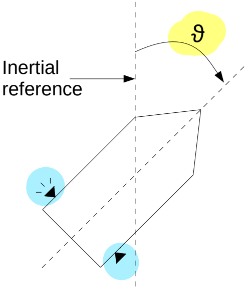
$$M_r = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} \quad \Rightarrow \quad M_r^{-1} = \begin{bmatrix} 0 & 1 \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

$$K = -\begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{4} \\ \frac{1}{8} & -\frac{1}{8} \end{bmatrix} = \begin{bmatrix} -\frac{1}{8} & \frac{1}{8} \end{bmatrix}$$

1h 45

Example : Single-axis satellite attitude control

Attitude control = proper orientation of the satellite antenna with respect to earth.



$$I\ddot{\theta} = M_C + M_D$$

I = moment of inertia of the satellite (about the mass center)

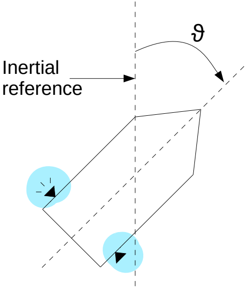
M_C = control torque applied by thrusters

M_D = disturbance torque

θ = angle of satellite

Example : Single-axis satellite attitude control

Attitude control = proper orientation of the satellite antenna with respect to earth.



- Model with normalized inputs:

$$u = \frac{M_C}{I}, \quad w = \frac{M_D}{I}$$

$$\ddot{\theta} = u + w$$

State-space models

- CT LTI models $x_1 = \theta$, $x_2 = \dot{\theta}$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w$$

$$y = \theta = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Double integrator dynamics

State-space models

- CT LTI models $x_1 = \theta$, $x_2 = \dot{\theta}$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w$$

$$y = \theta = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

  Double integrator dynamics

- DT LTI model (exact discretization, sampling time $T > 0$)

$$\begin{bmatrix} x_1^+ \\ x_2^+ \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix}}_{B} (u + w)$$

Control design

Goal

Design $u = Kx$ such that the closed-loop eigenvalues are $z_{1,2} = 0.8 \pm j0.25$

Control design

Goal

Design $u = Kx$ such that the closed-loop eigenvalues are $z_{1,2} = 0.8 \pm j0.25$

- Desired closed-loop polynomial

$$p^D(\lambda) = (\lambda - z_1)(\lambda - z_2) = \lambda^2 - 1.6\lambda + 0.7$$

- Closed-loop polynomial for $u = \begin{bmatrix} \kappa_1 & \kappa_2 \end{bmatrix} x$

$$p^K(\lambda) = \det\left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \left(\underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A} + \underbrace{\begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix}}_{B} \begin{bmatrix} \kappa_1 & \kappa_2 \end{bmatrix}\right)\right) =$$

$$\lambda^2 + \left(-T\kappa_2 - \frac{T^2}{2}\kappa_1 - 2\right)\lambda - \frac{T^2}{2}\kappa_1 + T\kappa_2 + 1$$

Idea for design: equate the coefficients of p^K and p^D $\Rightarrow$ simple equations for $n = 1, 2$ (even easier than using Ackermann's formula)

Control design

Equating the coefficients of the two polynomials for $T = 0.1$

$$\begin{cases} -T\kappa_2 - \frac{T^2}{2}\kappa_1 - 2 = -1.6 \\ -\frac{T^2}{2}\kappa_1 + T\kappa_2 + 1 = 0.7 \end{cases} \rightarrow \begin{cases} \kappa_1 = -\frac{0.1}{T^2} = -10 \\ \kappa_2 = -\frac{0.35}{T} = -3.5 \end{cases}$$

Control design

Equating the coefficients of the two polynomials for $T = 0.1$

$$\begin{cases} -T\kappa_2 - \frac{T^2}{2}\kappa_1 - 2 = -1.6 \\ -\frac{T^2}{2}\kappa_1 + T\kappa_2 + 1 = 0.7 \end{cases} \rightarrow \begin{cases} \kappa_1 = -\frac{0.1}{T^2} = -10 \\ \kappa_2 = -\frac{0.35}{T} = -3.5 \end{cases}$$

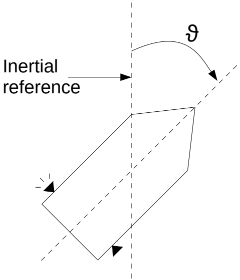
Same results through Ackermann's formula

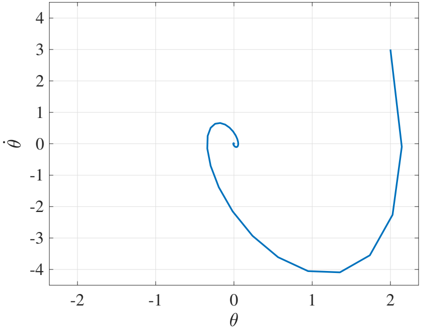
- Matlab code

```
T = 0.1
A = [1 T ; 0 1], B = [T^2/2 ; T]
p = [0.8+i*0.25 ; 0.8-i*0.25]
K = -acker(A, B, p)
```

$u = -Kx \rightarrow u = +Kx$

Simulations





Eigenvalue assignment for MIMO systems

Problems

- If $m > 1$, there is no Ackermann's formula
- Possible to find a change of variables $\tilde{x} = Tx$ such that $\tilde{A}_D$ and $\tilde{B}_D$ are in a suitable "canonical form" simplifying the computation of $\tilde{K}$ (and then K) $\rightarrow$ Hard to compute T
  - not covered in this class

In MatLab: `K = -place(A, B, p)`

Eigenvalue assignment for MIMO systems

Alternative approach

1. Compute the desired closed-loop characteristic polynomial
$$p^D(\lambda) = \lambda^n + \tilde{a}_{n-1}\lambda^{n-1} + \cdots + \tilde{a}_1\lambda^1 + \tilde{a}_0$$
2. Compute the characteristic polynomial $p^K(\lambda)$ of $A + BK$, where entries of
$$K = \begin{bmatrix} K_{11} & \cdots & K_{1n} \\ \vdots & \ddots & \vdots \\ K_{m1} & \cdots & K_{mn} \end{bmatrix}$$
are free parameters
3. Choose K_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$ so as to make each coefficient of $p^K(\lambda)$ equal to the corresponding coefficient of $p^D(\lambda)$

$\hookrightarrow$ Solve a system of nonlinear equations (can be difficult)

Simplified methods for MIMO systems

Next: two simplified algorithms - but they cannot be always used

Method 1 : feedback on a scalar channel

$$x^+ = Ax + Bu \quad B = \left[\begin{array}{c|c|c|c} b_1 & b_2 & \cdots & b_m \end{array}\right] \in \mathbb{R}^{n\times m}$$

Assumption : system reachable from a single input

- Can be u_1, without loss of generality, i.e. (A, b_1) is reachable.

Idea: use only u_1 for assigning the eigenvalues.

Simplified methods for MIMO systems

Next: two simplified algorithms - but they cannot be always used

Method 1 : feedback on a scalar channel

$$x^+ = Ax + Bu \quad B = \left[\, b_1 \mid b_2 \mid \cdots \mid b_m \,\right] \in \mathbb{R}^{n \times m}$$

Assumption : system reachable from a single input

- Can be u_1, without loss of generality, i.e. (A, b_1) is reachable.

Idea: use only u_1 for assigning the eigenvalues.

1. Set $u(k) = K_1 v(k)$, $K_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{m \times 1}$ where $v(k) \in \mathbb{R}$ is an auxiliary input.
   Closed-loop system
   $$x^+ = Ax + BK_1 v = Ax + b_1 v$$
2. Set $v(k) = K_2 x(k)$ and use Ackermann's formula for assigning the eigenvalues of
   $$(A + b_1 K_2) = (A + BK_1 K_2)$$

Simplified methods for MIMO systems

Feedback gain

$$K = K_1 K_2 = \begin{bmatrix} \kappa_1 & \kappa_2 & \cdots & \kappa_n \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

Simplified methods for MIMO systems

Feedback gain

$$K = K_1 K_2 = \begin{bmatrix} \kappa_1 & \kappa_2 & \cdots & \kappa_n \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

Drawbacks

- Only a single input is used, all others are set to zero
  - Can be a nonsense if inputs are physical variables that cannot be set to zero
- If the system is reachable from multiple scalar inputs, the choice of the channel is arbitrary

Simplified methods for MIMO system

Method 2 - Probabilistic approach

1. Parametrize the control law as

$$u(k) = K_2 x(k) + K_3 v(k) \quad K_2 \in \mathbb{R}^{m\times n}, K_3 \in \mathbb{R}^{m\times 1}$$

where $v(k) \in \mathbb{R}$ is an auxiliary input
Partial closed-loop system

$$x^+ = (A + BK_2)x + BK_3 v$$

Simplified methods for MIMO system

Method 2 - Probabilistic approach

1) Parametrize the control law as

$$u(k) = K_2 x(k) + K_3 v(k) \quad K_2 \in \mathbb{R}^{m\times n}, K_3 \in \mathbb{R}^{m\times 1}$$

where $v(k) \in \mathbb{R}$ is an auxiliary input
Partial closed-loop system

$$x^+ = (A + BK_2)x + BK_3 v$$

Lemma

By choosing randomly K_2 and K_3, the pair $(A + BK_2, BK_3)$ is reachable with probability one

2) Use Ackermann's formula for designing K_1, such that the closed-loop system

$$x^+ = (A + BK_2 + BK_3 K_1)x$$

has the desired eigenvalues.

Simplified methods for MIMO systems

Feedback gain

$$K = K_2 + K_3 K_1$$

Simplified methods for MIMO systems

Feedback gain

$$K = K_2 + K_3 K_1$$

Drawbacks

- Same problems as in method 1
- The random choice of K_2, K_3 is independent of the system physics and can be meaningless