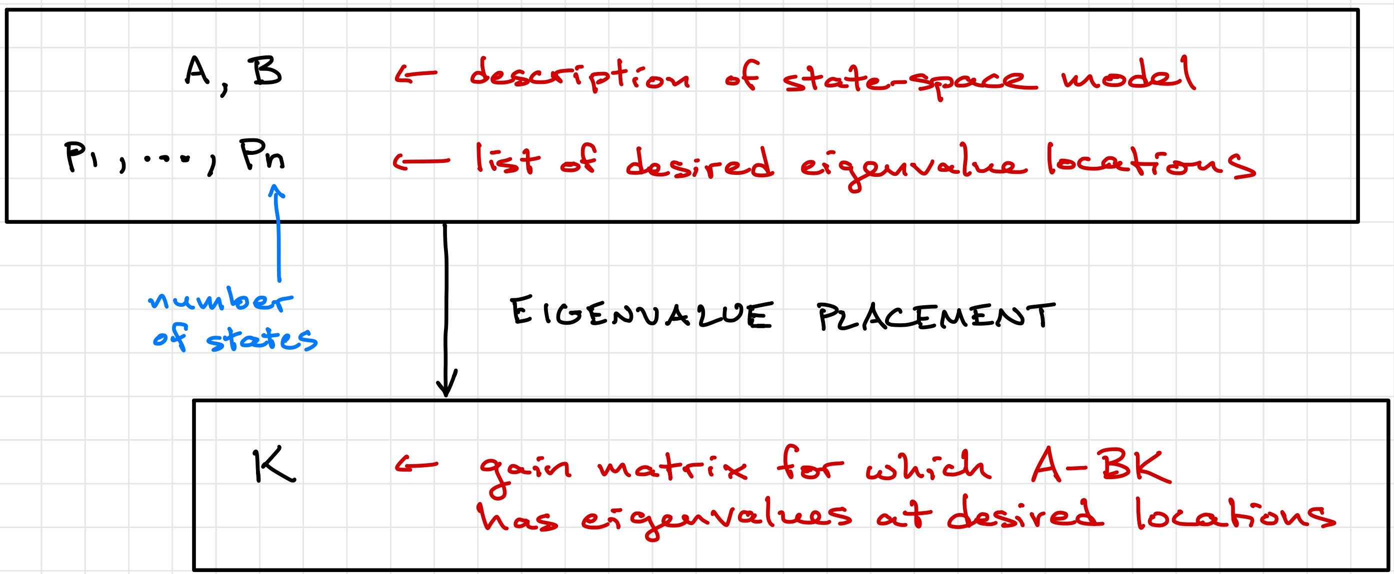
Ackermann's method (part 2)

AE353

Spring 2025

Bretl



A, B ← description of state-space model
$p_1, \ldots, p_n$ ← list of desired eigenvalue locations
number of states
EIGENVALUE PLACEMENT
K ← gain matrix for which $A-BK$ has eigenvalues at desired locations

WHAT CHARACTERISTIC POLYNOMIAL HAS ROOTS AT $p_1, \ldots, p_n$?

$$(s - p_1) = s - p_1$$

$$(s-p_1)(s-p_2) = s^2 - (p_1+p_2)s + p_1 p_2$$

$$(s-p_1)(s-p_2)(s-p_3) = s^3 - (p_1+p_2+p_3)s^2 + (p_1p_2+p_2p_3+p_3p_1)s - p_1p_2p_3$$

$$\vdots$$

$$= s^n + r_1 s^{n-1} + r_2 s^{n-2} + \ldots + r_{n-1} s + r_n$$

$\left.\begin{array}{l}\text{It is easy to compute the coefficients} \\ \quad r_1, \ldots, r_n \\ \text{given the eigenvalue locations} \\ \quad p_1, \ldots, p_n.\end{array}\right\}$ NUMERIC !!!

np.poly(p)

A, B ← description of state-space model

$r_1, \dots, r_n$ ← list of desired coefficient

number of states

EIGENVALUE PLACEMENT

K ← gain matrix for which the characteristic polynomial of $A-BK$ has desired coefficients

STRATEGY

1. Find K in special case when it is easy
2. Transform general case to special case

IF:

$$A = \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad K = \begin{bmatrix} k_1 & k_2 \end{bmatrix}$$

THEN:

$$A - BK = \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix}\begin{bmatrix} k_1 & k_2 \end{bmatrix} = \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} k_1 & k_2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} -a_1-k_1 & -a_2-k_2 \\ 1 & 0 \end{bmatrix}$$

$$\mathrm{def}\,(sI-(A-BK)) = \det\left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} -a_1-k_1 & -a_2-k_2 \\ 1 & 0 \end{bmatrix}\right) = \det\left(\begin{bmatrix} s+(a_1+k_1) & a_2+k_2 \\ -1 & s \end{bmatrix}\right)$$

$$= s^2 + (a_1+k_1)s + (a_2+k_2)$$

SO:

if you want $s^2 + r_1 s + r_2$

then $k_1 = r_1 - a_1 \qquad k_2 = r_2 - a_2$ ← easy! no symbolic computation!

Controllable Canonical Form (CCF)

$$A = \begin{bmatrix} \begin{bmatrix} -a_1 & \cdots & -a_n \end{bmatrix} \\ \begin{bmatrix} I_{(n-1)\times(n-1)} \end{bmatrix} \begin{bmatrix} 0_{(n-1)\times 1} \end{bmatrix} \end{bmatrix} \qquad B = \begin{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \\ \begin{bmatrix} 0_{(n-1)\times 1} \end{bmatrix} \end{bmatrix}$$

Facts

$$\det(sI - A) = s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n$$

$$A - BK = \begin{bmatrix} \begin{bmatrix} -a_1 - k_1 & \cdots & -a_n - k_n \end{bmatrix} \\ \begin{bmatrix} I \end{bmatrix} \begin{bmatrix} 0 \end{bmatrix} \end{bmatrix}$$

$$\det(sI - (A - BK)) = s^n + (a_1 + k_1) s^{n-1} + \cdots + (a_{n-1} + k_{n-1}) s + (a_n + k_n)$$

Consequence

if you want $\quad s^n + r_1 s^{n-1} + \cdots + r_{n-1} s + r_n$

no symbolic computation!

then $\quad k_1 = r_1 - a_1 \quad \cdots \quad k_n = r_n - a_n$

If we could put a system in CCF...

$$\dot{x} = Ax + Bu$$

$$\Big\downarrow \quad x = Vz$$

$$\dot{z} = A_{CCF}\, z + B_{CCF}\, u$$

$$V\dot{z} = AVz + Bu$$

$$\dot{z} = \underbrace{\boxed{V^{-1}AV}}_{A_{CCF}} z + \underbrace{\boxed{V^{-1}B}}_{B_{CCF}} u$$

Then...

$$u = -\overbrace{K_{CCF}}^{\text{easy to find}}\, z = -K_{CCF}V^{-1}x$$

$$= -\underbrace{(K_{CCF}V^{-1})}_{K \text{ (what we want)}}\, x$$

How to find A_{CCF}? (B_{CCF} is always the same)

eigenvalues are invariant to coordinate transformation

$$\det(sI - A_{CCF}) = \det(sI - V^{-1}AV)$$

$$= \det(sV^{-1}V - V^{-1}AV) \quad \leftarrow V^{-1}V = I$$

$$= \det(V^{-1}(sI-A)V)$$

$$= \det(V^{-1})\det(sI-A)\det(V) \quad \leftarrow \det(MN) = \det(M)\det(N)$$

$$= \cancel{\det(V)^{-1}}\det(sI-A)\cancel{\det(V)} \quad \leftarrow \det(M^{-1}) = \det(M)^{-1}$$

$$= \det(sI-A)$$

$\downarrow$

$a_1, \ldots, a_n$ are the coefficients of the characteristic polynomial of A

$\leftarrow$ np.poly(A)

How to find V?

$$\underbrace{A_{CCF} = V^{-1}AV \qquad B_{CCF} = V^{-1}B}_{\text{solve for } V^{-1}}$$

(that's what we need to find K given K_{CCF})

$$B_{CCF} = V^{-1}B$$

$$A_{CCF}B_{CCF} = V^{-1}AV\,V^{-1}B = V^{-1}AB$$

$$A_{CCF}^2 B_{CCF} = V^{-1}A\cancel{V\,V^{-1}}A\cancel{V\,V^{-1}}B = V^{-1}A^2B$$

$$\vdots$$

$$A_{CCF}^{n-1}B_{CCF} = V^{-1}A^{n-1}B$$

$$\underbrace{\begin{bmatrix} B_{CCF} & A_{CCF}B_{CCF} & A_{CCF}^2B_{CCF} & \cdots & A_{CCF}^{n-1}B_{CCF} \end{bmatrix}}_{W_{CCF}} = V^{-1}\underbrace{\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}}_{W}$$

W_{CCF} ← a square $n \times n$ matrix

W ← a square $n \times n$ matrix

$$\boxed{V^{-1} = W_{CCF}W^{-1}}$$

↑ works as long as W is invertible

ACKERMANN'S METHOD

NUMERIC !!

- Compute the characteristic equation that we want:

$$(s - p_1) \cdots (s - p_n) = s^n + r_1 s^{n-1} + \cdots + r_{n-1} s + r_n$$

- Compute the characteristic equation that we have:

$$\det(sI - A) = s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n$$

- Compute the controllability matrix of the original system (and check that $\det(W) \neq 0$):

$$W = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}$$

- Compute the controllability matrix of the transformed system:

$$W_{\text{ccf}} = \begin{bmatrix} B_{\text{ccf}} & A_{\text{ccf}} B_{\text{ccf}} & \cdots & A_{\text{ccf}}^{n-1} B_{\text{ccf}} \end{bmatrix}$$

where

$$A_{\text{ccf}} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \qquad B_{\text{ccf}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Compute the gains for the transformed system:

$$K_{\text{ccf}} = \begin{bmatrix} r_1 - a_1 & \cdots & r_n - a_n \end{bmatrix}$$

- Compute the gains for the original system:

$$K = K_{\text{ccf}} W_{\text{ccf}} W^{-1}$$

The system

$$\dot{x} = Ax + Bu$$

is controllable if

$$W = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}$$

"n" is the number of states (i.e., the size of x)

has full rank.

If W is square (i.e., if there is only one input, so B is a column matrix), then "full rank" is equivalent to "invertible" and you can simply check that

$$\det(W) \neq 0.$$

If W is not square (i.e., if there is more than one input, so B has more than one column), then W has no inverse and so you must check the "full rank" condition.

Our derivation assumed one input only, but this result generalizes (with more work) to any number of inputs.