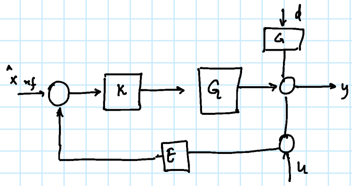
Lecture 8 - Eigenstructures (but lecture 10)

Tuesday 14 November 2023 09:26



$\Delta u = -k \Delta x$

choosing k:

(1) pole placement

(2) LQR

EIGENSTRUCTURE ASSIGNMENT

LQR: Review:

"very close to MPC"

It solves the problem

$\Delta \dot{x} = A \Delta x + B \Delta u \rightarrow$ find k:

$\Delta u = -k \Delta x$ minimizes the following cost: → linear controller is best soln. (almost always)!

$$J = \int_0^\infty \left(x^T Q x + \bar{u}^T R \bar{u} \right) dt$$

In MATLAB: $k = LQR(A, B, Q, R)$ → directly 6 dB
→ a lot-of trial & error

Pole placement: Review

Problem: Given a linear plant

$$\begin{bmatrix} \Delta \dot{x} = A \Delta x + B \Delta u \\ \Delta y = C \Delta x + D \Delta u \end{bmatrix} \qquad \Delta x \in \mathbb{R}^n \text{ \& } \Delta u \in \mathbb{R}^1$$

(+)

$u = -k\Delta x$ ⇒ Find $k \in \mathbb{R}^{1 \times n} = [k_1 \ k_2, \text{----} \ k_n]$ such that $eig(A - Bk) = 0$

Desired eigen values

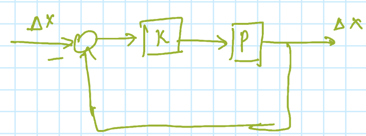
$\{\lambda_i\}_{i=1}^n \Rightarrow n$ eigen values

(II)

Desired $p(\lambda) = (s - \lambda_1)(s - \lambda_2) \text{......}$ → (1)

the closed loop will be

$$\Delta \dot{x} = A \Delta x - B k \Delta x$$
$$= \underbrace{(A - Bk)}_{A_0} \Delta x$$



In MATLAB; place(A, B, [poles])

[how to compute by hand?]

To solve that problem:

notice that the eigen values of a system are invariant to state transformations to

Notice that the eigen values of a system are invariant to state transformations to state transformations! Meaning

$$\bar{x}_c = T \cdot \bar{x}$$

$\hookrightarrow$ invertible!

$$\Rightarrow \left[x = T^{-1} \bar{x}_c \right]$$

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

$$\Rightarrow T^{-1}\dot{x}_c = AT^{-1}x_c + Bu$$

$$\Rightarrow \dot{x}_c = \left[TAT^{-1} \right] x_c + TBu$$

Instead of working with A, we will work with its canonical controllability form:

in MATLAB: A' = canon(sys, 'companion')

$$\dot{x}_c = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -d^{(0)} & -d^{(1)} & -d^{(2)} & -d^{(3)} \end{bmatrix}}_{A} x_c + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$B_k = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} k_1 & k_2 & k_3 & k_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ k_1 & k_2 & k_3 & k_4 \end{pmatrix}$$

$$A - Bk = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -d^{0} - k_1 & -d^{(1)} - k_2 & -d^{(2)} - k_3 & -d^{(3)} - k_4 \end{pmatrix}$$

Cofactor!

$$\det\left(sI - (A - Bk)\right) = \begin{vmatrix} s & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & s & -1 \\ \boxed{d^{0} + k_1} & d^{(1)} + k_2 & d^{(2)} + k_3 & s + d^{(3)} + k_4 \end{vmatrix}$$

$*s \quad *s^2 \quad *s^3$

using Laplace expansion to compute determinant

"its great if we have a bunch of 0s"

$$d^{(0)} + k_1 + d^{(1)}s + k_2 s + d^{(2)}s^2 + k_3 s^2 + s^4 + d^{(3)}s^3 + k_4 s^3 = r(s)$$

$$\rightarrow \det(\cdot) = (-1)^{i+j} \begin{vmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \end{vmatrix} r(s)$$

$$\rightarrow \det(\cdot) = (-1)^{i+j} \begin{vmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & s & -1 \end{vmatrix} r(s)$$

$$= r(s)$$

Eg:

$$\begin{array}{ccc} (1,1) & (1,2) & (1,3) \end{array}$$
$$\begin{vmatrix} 1 & 1 & 1 \\ 2 & 1 & 2 \\ 0 & 1 & 0 \end{vmatrix} = 2 - 2 = 0$$

Laplace expansion

$$1(-1)^{1+1} \begin{vmatrix} 1 & 2 \\ 1 & 0 \end{vmatrix} + 1(-1)^{1+2} \begin{vmatrix} 2 & 2 \\ 0 & 0 \end{vmatrix}$$
$$+ 1(-1) \begin{vmatrix} 2 & 1 \\ 0 & 1 \end{vmatrix} = 0.$$

Going back to result ①,

$$r(s) = s^4 + a_3 s^2 + a_1 s^1 + a_0$$

↳ Requirements!

$$\left.\begin{aligned} a_3 &= d^{(3)} + k_4 \\ a_2 &= d^{(2)} + k_3 \\ a_1 &= d^{(1)} + k_2 \\ a_0 &= d^{(0)} + k_1 \end{aligned}\right\} \Rightarrow \boxed{k_j = a_{j-1} - d^{(j-1)}}$$

$$u = -k x_c = -k^T x$$

↳ This is what we were looking for

if we're doing pole placement, we require n poles for an n^{th} order system.

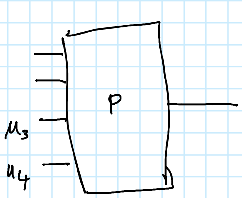
① K is unique

② $\{ \lambda_i \}_{i=1}^{4} \longrightarrow$ define the eigenvector of closed loop system A!

↓ Cannot be set in SISO pole placement

③ Also with LQR, eigenvectors cannot be explicitly defined!

pole placement: for MIMO



u_3 + entire space! but we also have u_4 which can do some extra controlling

u_3 + entire space! but we also have u_4 which can do some extra controlling

⚠ perhaps we could use the extra degrees of freedom in MIMO to add additional requirements

Analysis of a system through $(\underbrace{\{x_i, u_i\}}_{\text{Eigen structure}})$ ⟺ Modal analysis

Eigen structure → eg: phugoid, short p., etc

$$\bar{x}(t) = e^{At} \cdot \bar{x}(0) + \int_0^t e^{A(t-\tau)} B \bar{u}(\tau) \, d\tau$$

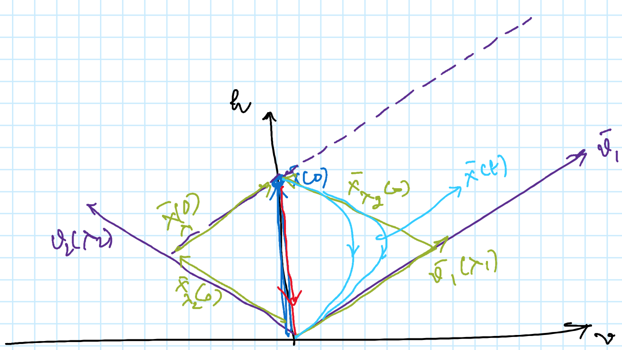
$$\Rightarrow \bar{x}(t) = e^{(T^{-1} \lambda T) t} \bar{x}(0)$$

→ diagonalized form

$$\Rightarrow T^{-1} e^{\lambda t} T \bar{x}(0)$$

$$\Rightarrow T \bar{x}(t) = e^{\lambda t} T \bar{x}(0)$$

Graphically,



$$\bar{x}(0) = \begin{pmatrix} 0 \\ \delta h \end{pmatrix}$$

$$\Longrightarrow \bar{x}_\lambda(0) = \begin{pmatrix} x_{\lambda_1}(0) \\ x_{\lambda_2}(0) \end{pmatrix}$$

More -ve λ is, faster the exp will die out

Lets say we have $0 < \lambda_1 << \lambda_2$

lo Hi

if $\lambda_1 = \lambda_2$

this can induce extra dynamics on the system.

Pilot induced oscillations

→ can be controlled by using the additional dofs

→ say increase alt of a/c

→ say increase alt of a/c

$L = \frac{1}{2} \rho u^2 S C_L(\alpha, h, Re, \ldots)$

↳ ↓ing α ⇒ Short period mode
but on reaching reqd alt, we reach a state where v is low.

→ we can use MIMO, to maintain higher v, ~~while maint~~ while still ↓ing alt.

↳ TECS : Total energy Control Systems

↳ just rep eqn of system to obtain/solve this problem using TECS.

Thm: Let $\{\lambda_i\}$, $\{\bar{v}_i\}$ be desired (Eigenvector + Eigenvalue) Eigenstructure;

$\exists\, k$ such that $(A - BK)\bar{v}_i = \lambda \bar{v}_i$

iff

$\bar{v}_i \in \text{Span}\{N_{\lambda_i}\}$, where:

$$S_{\lambda_i} = \Big[\underbrace{(\lambda_i I - A)}_{\substack{n \text{ columns} \\ (\text{states})}} \Big| \underbrace{-B}_{\substack{m \text{ columns} \\ \text{inputs}}}\Big]$$

$R_\lambda \triangleq \left[\dfrac{N_\lambda}{m_\lambda}\right]$, where columns of R_λ form a basis of $\ker(S_\lambda)_B$

Computationally,

$(A - BK)\bar{v}_i = \lambda_i \bar{v}_i$

$\Leftrightarrow \quad A\bar{v}_i - BK\bar{v}_i - \lambda_i \bar{v}_i = 0$

$\Leftrightarrow \quad (A - \lambda_i I)\bar{v}_i - B(K\bar{v}_i) = 0$

$$\underbrace{[(A - \lambda_i I) \,|\, -B]}_{S_{\lambda_i}} \left(\frac{\bar{v}_i}{K\bar{v}_i}\right) = 0$$

concatenate

$$\underbrace{\quad S_{\lambda_i} \quad}_{\text{Concatenate with } B} \qquad \begin{pmatrix} \\ k\bar{v}_i \end{pmatrix} =$$

$$\lambda_1 \longrightarrow \text{Ker } S_{\lambda_1} \xrightarrow[\text{basis!}]{\text{pick}} \left\{ \begin{pmatrix} \bar{v}_1^{(1)} \\ \hline k\bar{v}_1^{(1)} \end{pmatrix}, \begin{pmatrix} \bar{v}_1^{(2)} \\ \hline k\bar{v}_1^{(2)} \end{pmatrix}, \ \ldots\ldots \right\}$$

$\bar{m}_{\lambda_1}^{(2)}$

$$\Rightarrow k \cdot v_n^{(j)} = \bar{M}_{\lambda_i}^{(j)}$$

We do this comp. for all λ_n.

$$\vdots$$

$$\lambda_n \longrightarrow \text{Ker } S_{\lambda_n} \xrightarrow[\text{basis}]{\text{pick}} \left\{ \begin{pmatrix} v_n^{(1)} \\ \hline k\bar{v}_n^{(1)} \end{pmatrix}, \begin{pmatrix} v_n^{(2)} \\ \hline k\bar{v}_n^{(2)} \end{pmatrix}, \ \ldots \right\}$$

$$k \underbrace{\left[\bar{v}_1^{(2)} \ \ldots\ldots \ \bar{v}_n^{(2)} \right]}_{V} = \underbrace{\left[\bar{m}_{\lambda_1}^{(2)} \ \ldots\ldots \ m_{\lambda_n}^{(2)} \right]}_{M}$$

$$\Rightarrow \boxed{K = MV^{-1}}$$