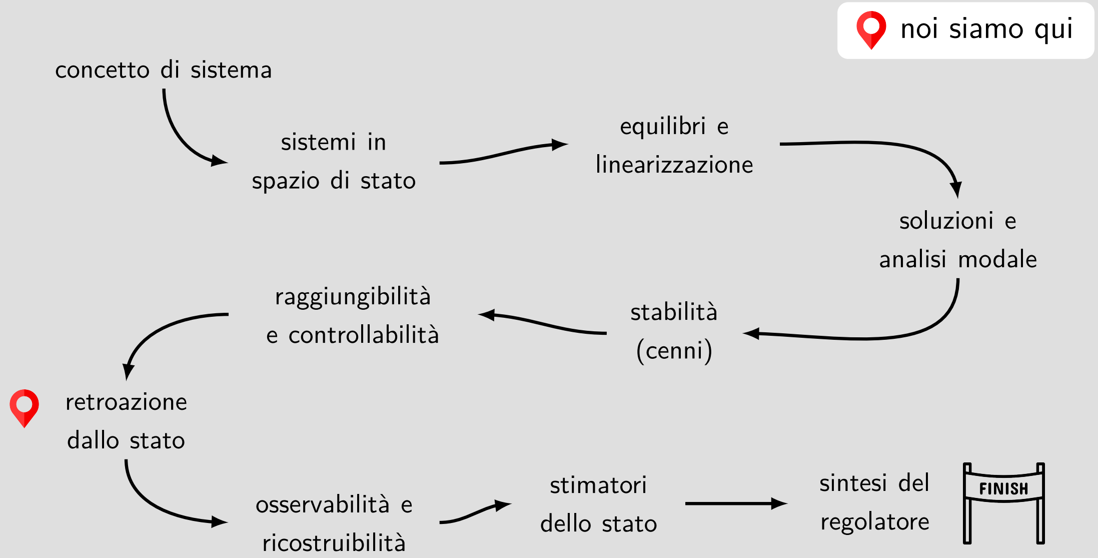
Teoria dei Sistemi e Controllo Ottimo e Adattativo (C. I.)
Teoria dei Sistemi (Mod. A)

Docente: Giacomo Baggio

Lez. 18: Controllo in retroazione dallo stato (parte 1)

Corso di Laurea Magistrale in Ingegneria Meccatronica

A.A. 2021-2022


noi siamo qui
concetto di sistema
sistemi in spazio di stato
equilibri e linearizzazione
soluzioni e analisi modale
stabilità (cenni)
raggiungibilità e controllabilità
retroazione dallo stato
osservabilità e ricostruibilità
stimatori dello stato
sintesi del regolatore
FINISH

In questa lezione

- ▷ Proprietà di sistemi lineari retroazionati dallo stato
- ▷ Controllo in retroazione dallo stato: caso $m = 1$
- ▷ Comandi Matlab®

Retroazione dallo stato ed equivalenza algebrica

$$\Sigma^{(K)} : x(t+1) = (F + GK)x(t) + Gv(t)$$

$$\boxed{\text{Come si modificano le matrici del sistema per effetto di un cambio di base } T?}$$

$$F' = T^{-1}FT, \quad G' = T^{-1}G, \quad K' = KT$$

Forma di Kalman del sistema retroazionato dallo stato

$$\Sigma^{(K)} : x(t+1) = (F + GK)x(t) + Gv(t)$$

$$F_K \triangleq T^{-1}FT = \begin{bmatrix} F_{11} & F_{12} \\ 0 & F_{22} \end{bmatrix}, \quad G_K \triangleq T^{-1}G = \begin{bmatrix} G_1 \\ 0 \end{bmatrix}, \quad K_K \triangleq KT = \begin{bmatrix} K_1 & K_2 \end{bmatrix}$$

$$T^{-1}x = \begin{bmatrix} x_R(t+1) \\ x_{NR}(t+1) \end{bmatrix} = \begin{bmatrix} F_{11} + G_1K_1 & F_{12} + G_1K_2 \\ 0 & F_{22} \end{bmatrix} \begin{bmatrix} x_R(t) \\ x_{NR}(t) \end{bmatrix} + \begin{bmatrix} G_1 \\ 0 \end{bmatrix} v(t)$$

Il sottosistema non raggiungibile non è influenzato dalla retroazione !

Controllo in retroazione per sistemi a singolo ingresso ($m = 1$)

$$\Sigma: \quad x(t+1) = Fx(t) + gu(t), \quad g \in \mathbb{R}^{n\times 1}$$

$$\Sigma^{(K)}: \quad x(t+1) = (F+gK)x(t) + gv(t)$$

$\boxed{\text{Quando è possibile assegnare a } F+gK \text{ degli autovalori desiderati?}}$

Teorema: Per ogni polinomio

$$p(\lambda) = \lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_1\lambda + p_0, \;\; p_i \in \mathbb{R},$$

esiste una matrice di retroazione $K \in \mathbb{R}^{1\times n}$ tale che $\Delta_{F+gK}(\lambda) = p(\lambda)$ se e solo se il sistema Σ è raggiungibile.

Allocazione degli autovalori ($m = 1$): metodo diretto

$$\Sigma: \quad x(t+1) = Fx(t) + gu(t), \quad g \in \mathbb{R}^{n\times 1}, \ \Sigma \text{ raggiungibile}$$

$$\Sigma^{(K)}: \quad x(t+1) = (F+gK)x(t) + gv(t)$$

$$\boxed{\text{Come fare ad assegnare a } F+gK \text{ degli autovalori desiderati?}}$$

$$p(\lambda) = \lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_1\lambda + p_0 = \text{polinomio con autovalori desiderati}$$

$$\boxed{\text{Risolvere } \Delta_{F+gK}(\lambda) = \det(\lambda I - F - gK) = p(\lambda) \text{ con incognita } K}$$

$$\Downarrow$$

Sistema di equazioni lineari con incognite $k_1, \dots, k_n$, $K = \begin{bmatrix} k_1 & \cdots & k_n \end{bmatrix}$!

Esempio

$$x(t+1) = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} u(t)$$

Retroazione K^* tale che il sistema retroazionato abbia autovalori $\lambda_1 = 0$, $\nu_1 = 3$?

$$K^* = \begin{bmatrix} -\frac{1}{2} & -\frac{3}{2} & -\frac{1}{2} \end{bmatrix}$$

Allocazione autovalori ($m = 1$): osservazioni

1. Il procedimento permette di allocare gli autovalori di $F + gK$ a nostro piacimento! L'unico vincolo è la raggiungibilità di Σ e il fatto che se un autovalore è complesso deve esserci anche il suo complesso coniugato.

2. Se il sistema Σ non è raggiungibile allora possiamo cambiare tramite retroazione solo gli autovalori di F_{11} (matrice di stato del sottosistema raggiungibile).

3. Se tutti gli autovalori vengono allocati in zero ($p(\lambda) = \lambda^n$) tutti i modi del sistema retroazionato convergono a zero in tempo finito. Il controllore in questo caso viene detto controllore dead-beat!

4. Il procedimento rimane invariato per sistemi a tempo continuo, ma in questo caso non si possono avere controllori dead-beat.

Comandi Matlab® – Control System Toolbox

`K = place(F,G,v)`	calcola matrice di retroazione K tale che $F + GK$ ha come autovalori gli elementi del vettore v (**N.B.** numericamente robusto ma non funziona per autovalori multipli);
`K = acker(F,G,v)`	calcola matrice di retroazione K tale che $F + GK$ ha come autovalori gli elementi del vettore v (**N.B.** numericamente instabile ma funziona anche per autovalori multipli);