Controlli Automatici T
Introduzione al Control System Toolbox in Matlab

Prof. Guido Carnevale

Department of Electrical, Electronic, and Information Engineering
Alma Mater Studiorum Università di Bologna
guido.carnevale@unibo.it

Definire un sistema LTI tempo continuo

Consideriamo un generico sistema LTI tempo continuo

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

Date le matrici A, B, C, D, possiamo creare l'oggetto "sistema" con la funzione ss (state space)

```
modello = ss(A, B, C, D);
```

Esempio singolo integratore con $x, u, y \in \mathbb{R}$:

$$\dot{x}(t) = u(t)$$
$$y(t) = x(t)$$

```
A = 0;
B = 1;
C = 1;
D = 0;
modello = ss(A, B, C, D);
```

Simulazione di un sistema lineare

```
help lsim
```

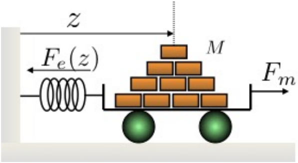
Per simulare l'evoluzione di un sistema LTI su Matlab possiamo usare la funzione `lsim`.

```
tt = 0:0.1:10;                 % intervallo temporale
uu = ones(length(tt), 1);  % ingresso a 'gradino' unitario
x0 = 5;                        % condizione iniziale
[Y, TT, X] = lsim(modello, uu, tt, x0);
plot(TT, X);
```

- `modello` oggetto sistema LTI creato con la funzione `ss`
- `uu` vettore contenente i valori dell'ingresso di controllo (es.: gradino unitario)
- `tt` vettore degli istanti temporali (es.: $\begin{bmatrix}0 & 0.1 & 0.2 & \dots & 10\end{bmatrix}$)
- `x0` condizione iniziale (es.: 5)
- `Y` traiettoria dell'uscita $y(t)$
- `TT` vettore degli istanti temporali
- `X` traiettoria dello stato $x(t)$

Esempio carrello

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) + \frac{1}{M}u(t)$$
$$y(t) = x_1(t)$$



x_1 posizione centro di massa
x_2 velocità centro di massa

$k = 1\,\mathrm{N/m}$ costante elastica (tempo invariante)
$M = 0.5\,\mathrm{kg}$ massa

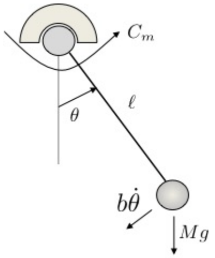
Esercizio: graficare traiettoria di stato per $0 \leq t \leq 10$ con condizione iniziale $x(0) = [0, 1]^\top$ e $u(t) = 0$ (evoluzione libera) e poi con condizione iniziale nulla e $u(t) = \sin(\frac{2\pi}{T}t)$ con $T = 5$ (evoluzione forzata). Infine confrontare la somma delle due traiettorie ottenute con quella ottenuta usando sia condizione iniziale $x(0) = [0, 1]^\top$ che $u(t) = \sin(\frac{2\pi}{T}t)$ nella stessa simulazione.

Esempio: pendolo

Equazione della dinamica

$$\ddot{\theta}(t) = -\frac{g}{\ell}\sin(\theta(t)) - \frac{b}{M\ell^2}\dot{\theta}(t) + \frac{1}{M\ell^2}C_{\mathsf{m}}(t)$$

M, ℓ, b parametri fisici del pendolo
$C_{\mathsf{m}}(t)$ momento torcente (input) applicato al pendolo



Esercizio: graficare traiettoria di stato per $0 \le t \le 10$ con condizione iniziale $x(0) = [\frac{\pi}{6}, 0]^\top$ e $u(t) = 0$ (evoluzione libera) e poi con condizione iniziale nulla e $u(t) = 2$. Infine confrontare la somma delle due traiettorie con quella ottenuta usando sia condizione iniziale $x(0) = [\frac{\pi}{6}, 0]^\top$ che $u(t) = 2$ nella stessa simulazione.

Esempio carrello con e senza attrito

Carrello senza attrito

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) + \frac{1}{M}u(t)$$

La matrice di stato ha autovalori puramente immaginari (parte reale nulla) e infatti l'evoluzione libera oscilla senza convergere né divergere.

Esempio carrello con e senza attrito

Carrello senza attrito

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) + \frac{1}{M}u(t)$$

La matrice di stato ha autovalori puramente immaginari (parte reale nulla) e infatti l'evoluzione libera oscilla senza convergere né divergere.

Carrello con attrito viscoso

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) - \frac{b}{M}x_2(t) + \frac{1}{M}u(t)$$

La matrice di stato ha autovalori con parte reale negativa e infatti l'evoluzione libera ha modi convergenti nell'origine.

Esempio carrello con e senza attrito

Carrello senza attrito

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) + \frac{1}{M}u(t)$$

La matrice di stato ha autovalori puramente immaginari (parte reale nulla) e infatti l'evoluzione libera oscilla senza convergere né divergere.

Carrello con attrito viscoso

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{M}x_1(t) - \frac{b}{M}x_2(t) + \frac{1}{M}u(t)$$

La matrice di stato ha autovalori con parte reale negativa e infatti l'evoluzione libera ha modi convergenti nell'origine.

Nel caso senza attrito, si può sfruttare l'ingresso di controllo $u(t)$ per imporre modi convergenti?

Retroazione dallo stato

Sistema lineare tempo invariante (LTI)

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

Supponendo di misurare l'intero stato, ovvero se $y(t) = x(t)$, allora possiamo progettare

$$u(t) = Kx(t) + v(t)$$

con $K \in \mathbb{R}^{m \times n}$ una matrice di guadagni e $v(t)$ un ulteriore ingresso per il sistema retroazionato

$$\dot{x}(t) = \underbrace{(A + BK)}_{\text{matrice di anello chiuso}} x(t) + Bv(t)$$

Se vogliamo il sistema in anello chiuso con modi convergenti allora dobbiamo progettare K tale che $(A + BK)$ abbia autovalori tutti a parte reale negativa.

Retroazione dallo stato

Sistema lineare tempo invariante (LTI)

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

Supponendo di misurare l'intero stato, ovvero se $y(t) = x(t)$, allora possiamo progettare

$$u(t) = Kx(t) + v(t)$$

con $K \in \mathbb{R}^{m \times n}$ una matrice di guadagni e $v(t)$ un ulteriore ingresso per il sistema retroazionato

$$\dot{x}(t) = \underbrace{(A + BK)}_{\text{matrice di anello chiuso}} x(t) + Bv(t)$$

Se vogliamo il sistema in anello chiuso con modi convergenti allora dobbiamo progettare K tale che $(A + BK)$ abbia autovalori tutti a parte reale negativa.

Nota: la possibilità di scegliere gli autovalori di $(A + BK)$ (e.g., per renderli tutti a parte reale negativa) dipende dalla coppia (A, B) ed è legata alla proprietà di raggiungibilità.

Retroazione dello stato

La funzione place permette di scegliere arbitrariamente gli autovalori della matrice in anello chiuso.

```
% progetta K tale che A-B*K abbia autovalori definiti da p
K = place(A,B,p);
```

Importante: la funzione considera il controllo $u(t) = -Kx(t)$ diversamente da quanto visto nella slide precedente e quanto vedrete nel modulo 1.

Retroazione dello stato

La funzione place permette di scegliere arbitrariamente gli autovalori della matrice in anello chiuso.

```
% progetta K tale che A-B*K abbia autovalori definiti da p
K = place(A,B,p);
```

Importante: la funzione considera il controllo $u(t) = -Kx(t)$ diversamente da quanto visto nella slide precedente e quanto vedrete nel modulo 1.

Esercizio: simulare l'evoluzione del carrello nell'intervallo $[0, 30]$ con condizioni iniziali $x(0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ e imponendo autovalori a parte reale negativa nella matrice di anello chiuso.