Richiami di fondamenti di automatica e concetti fondamentali di controllo digitale

Sistemi lineari tempo continuo

Rappresentazione spazio di stato

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

$$x(0) = x_0$$

$$x(t) \in \mathbb{R}^n \qquad A \in \mathbb{R}^{n\times n}$$
$$u(t) \in \mathbb{R}^m \qquad B \in \mathbb{R}^{n\times m}$$
$$y(t) \in \mathbb{R}^p \qquad C \in \mathbb{R}^{p\times n}$$
$$D \in \mathbb{R}^{p\times m}$$

Caso SISO (singolo ingresso singola uscita)

$$\begin{cases} \begin{array}{rcll} \dot{x}_1(t) & = & a_{11}x_1(t) + \ldots + a_{1n}x_n(t) & +b_1u(t) \\ \dot{x}_2(t) & = & a_{21}x_1(t) + \ldots + a_{2n}x_n(t) & +b_2u(t) \\ \vdots & & \vdots & \vdots \\ \dot{x}_n(t) & = & a_{n1}x_1(t) + \ldots + a_{nn}x_n(t) & +b_nu(t) \\ y(t) & = & c_1x_1(t) + \ldots + c_nx_n(t) & +du(t) \end{array} \\ \\ x_1(0) = x_{10}, \quad \ldots \quad x_n(0) = x_{n0} \end{cases}$$

Equazioni differenziali di ordine n con ingresso

$$\frac{dy^{(n)}(t)}{dt^n} + a_{n-1}\frac{dy^{(n-1)}(t)}{dt^{n-1}} + \cdots + a_1\dot{y}(t) + a_0 y(t) = b_{n-1}\frac{du^{(n-1)}(t)}{dt^{n-1}} + b_{n-2}\frac{du^{(n-2)}(t)}{dt^{n-2}} + \cdots + b_1\dot{u}(t) + b_0 u(t)$$

equivale al sistema lineare di ordine n

$$\left\{\begin{array}{rcl} \dot{x}(t) & = & \begin{bmatrix} 0 & & & \\ \vdots & & \mathrm{I}_{n-1} & \\ 0 & & & \\ -a_0 & -a_1 & \ldots & -a_{n-1} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \\ y(t) & = & \begin{bmatrix} b_0 & b_1 & \ldots & b_{n-1} \end{bmatrix} x(t) \end{array}\right.$$

Sistemi lineari tempo continuo

Definizione: La *funzione di trasferimento* di un sistema lineare tempo continuo (A, B, C, D) è

$$G(s) = C(sI - A)^{-1}B + D$$

cioè il rapporto fra la trasf. di Laplace $Y(s)$ dell' uscita $y(t)$ e la trasf. di Laplace $U(s)$ dell' ingresso $u(t)$ per condizione iniziale nulla $x_0 = 0$.

Nel caso di eq. differenziali di ordine n:

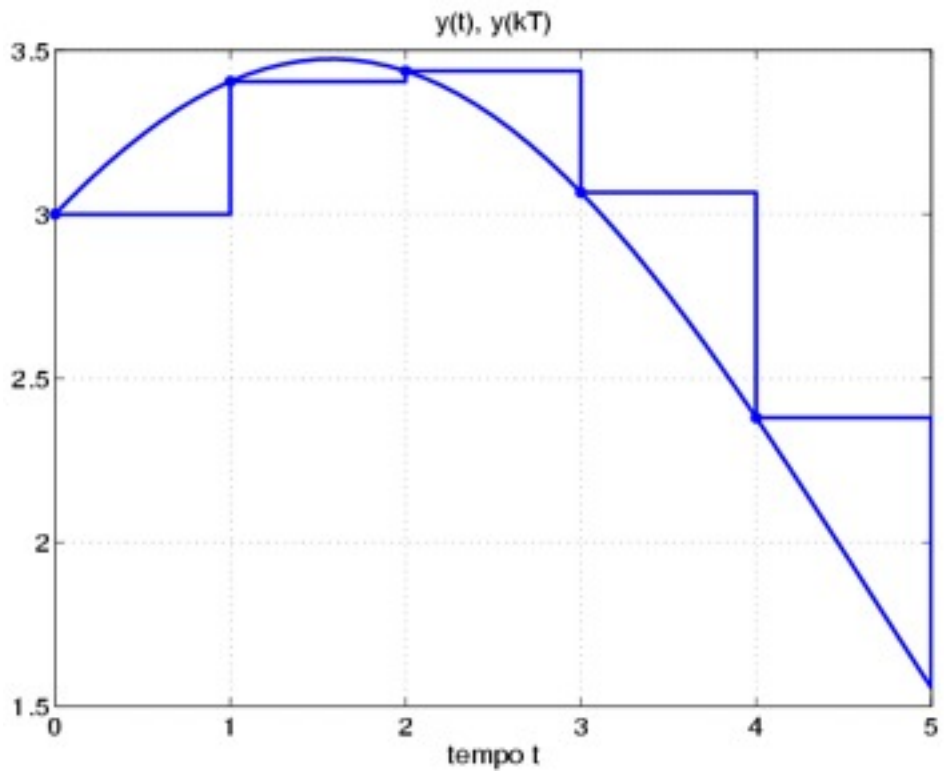
(condizioni iniziali nulle: $y(0),\ \dot{y}(0),\ \ldots,\ y^{(n-1)}(0) = 0$)

$$\frac{dy^{(n)}(t)}{dt^2} + a_{n-1}\frac{dy^{(n-1)}(t)}{dt^{n-1}} + \cdots + a_1\dot{y}(t) + a_0 y(t) =$$

$$b_{n-1}\frac{du^{(n-1)}(t)}{dt^{n-2}} + b_{n-2}\frac{du^{(n-2)}(t)}{dt^{n-2}} + \cdots + b_1\dot{u}(t) + b_0 u(t)$$

$$G(s) = \frac{b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$

Sistemi lineari tempo discreto



- Esprimono relazioni fra variabili *campionate* ad intervalli T: $x(kT)$, $u(kT)$, $y(kT)$, $k = 0, 1, \ldots$,
- Il segnale è $x(kT)$ è mantenuto costante durante l' *intervallo di campionamento* $[kT, (k+1)T)$.
- Il segnale può rappresentare il *campionamento* di un segnale *continuo* nel tempo, oppure essere intrinsecamente *discreto* nel tempo.

Rappresentazione spazio di stato

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

$$x(0) = x_0$$

$$x(k) \in \mathbb{R}^n \qquad A \in \mathbb{R}^{n \times n}$$
$$u(k) \in \mathbb{R}^m \qquad B \in \mathbb{R}^{n \times m}$$
$$y(k) \in \mathbb{R}^p \qquad C \in \mathbb{R}^{p \times n}$$
$$D \in \mathbb{R}^{p \times m}$$

Sistemi lineari tempo discreto

Soluzione:

$$x(k) = \underbrace{A^k x_0}_{\text{risposta libera}} + \underbrace{\sum_{i=0}^{k-1} A^i B u(k-1-i)}_{\text{risposta forzata}}$$

se la matrice A è diagonalizzabile:

$$A = T\Lambda T^{-1},\ T = [v_1|v_2|\ldots|v_n],\ \Lambda = \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix}$$

$$\Longrightarrow \quad A^k = T \begin{bmatrix} \lambda_1^k & 0 & \ldots & 0 \\ 0 & \lambda_2^k & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_n^k \end{bmatrix} T^{-1}$$

Risposta modale: simile al caso tempo continuo

Rappresentazioni di stato algebr. equivalenti: simile al caso t.continuo

Sistemi lineari tempo discreto

Equazioni alle differenze di ordine n con ingresso

$$\boxed{\begin{aligned}&a_n y(k-n) + a_{n-1} y(k-n+1) + \cdots + a_1 y(k-1) + y(k) = \\ &\qquad b_n u(k-n) + \cdots + b_1 u(k-1) + b_0 u(k)\end{aligned}}$$

equivale al sistema lineare di ordine n

$$\begin{cases} x(k+1) = \begin{bmatrix} 0 & & & \\ \vdots & & \mathrm{I}_{n-1} & \\ 0 & & & \\ -a_n & -a_{n-1} & \ldots & -a_1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k) \\ \\ y(k) = \begin{bmatrix} b_n - b_0 a_n & b_{n-1} - b_0 a_{n-1} & \ldots & b_1 - b_0 a_1 \end{bmatrix} x(k) + b_0 u(k) \end{cases}$$

Sistemi lineari tempo discreto

Funzione di trasferimento

Definizione: La *funzione di trasferimento* di un sistema lineare tempo discreto (A, B, C, D) è

$$G(z) = C(zI - A)^{-1}B + D$$

cioè il rapporto fra la trasf. zeta $Y(z)$ dell' uscita $y(k)$ e la trasf. zeta $U(z)$ dell' ingresso $u(k)$ <u>per condizione iniziale nulla</u> $x_0 = 0$.

Nel caso di eq. differenziali di ordine n:

$$a_n y(k-n) + a_{n-1}y(k-n+1) + \cdots + a_1 y(k-1) + y(k) = b_n u(k-n) + \cdots + b_1 u(k-1)$$

(condizioni iniziali nulle: $y(k) = u(k) = 0,\ \forall k < 0$)

$$G(z) = \frac{b_n z^{-n} + b_{n-1}z^{-n+1} + \cdots + b_1 z^{-1}}{a_n z^{-n} + a_{n-1}z^{-n+1} + \cdots + a_1 z^{-1} + 1} = \frac{b_1 z^{n-1} + \cdots + b_{n-1}z + b_n}{z^n + a_1 z^{n-1} + \cdots + a_{n-1}z + a_n}$$

Poli e zeri: simile al caso tempo continuo

Analisi nel discreto - Campionamento esatto

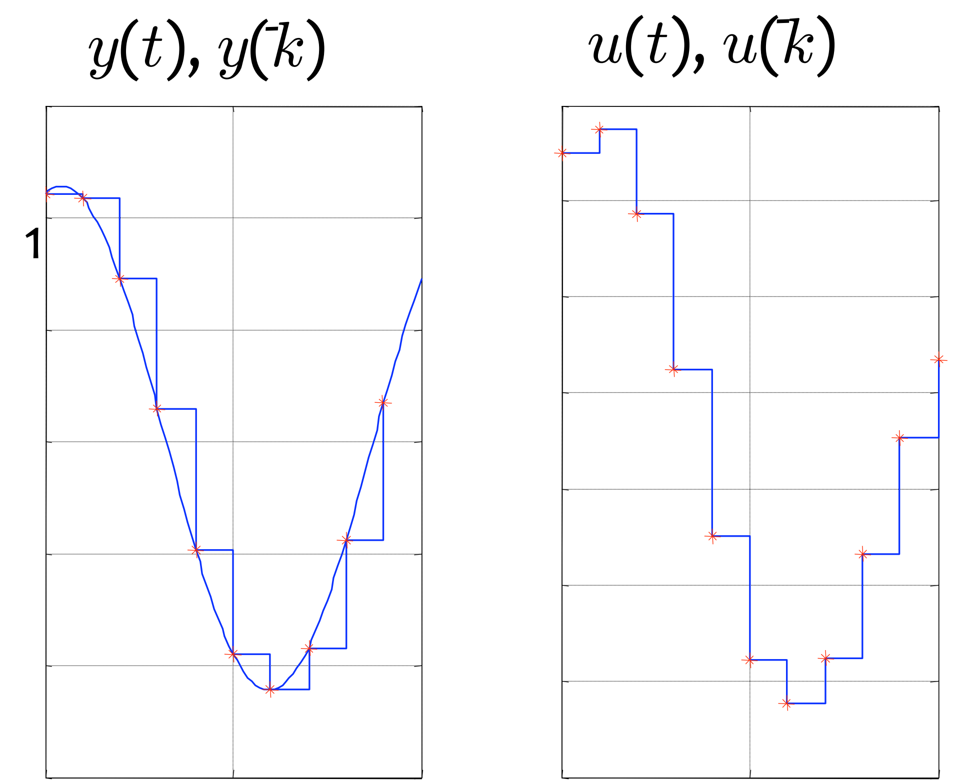
- Consideriamo un sistema a tempo continuo in forma di spazio di stato:

$$\begin{cases} \frac{dx}{dt}(t) & = & Ax(t) + Bu(t) \\ y(t) & = & Cx(t) + Du(t) \end{cases}$$
$$x(0) = x_0$$

- Vogliamo esprimerne l'evoluzione agli istanti di campionamento $t = 0,\ T,\ 2T,\ \ldots,\ kT,\ \ldots,$ supponendo che l'ingresso $u(t)$ sia costante durante ogni intervallo di campionamento:

$$\boxed{u(t) = \bar{u}(k),\ \ kT \leq t < (k+1)T}$$

- Siano $\bar{x}(k) \triangleq x(kT)$ e $\bar{y}(k) \triangleq y(kT)$ i campioni dello stato e dell'uscita, rispettivamente, all' istante di campionamento k-esimo.



Campionamento esatto

- Applichiamo la *formula di Lagrange* per integrare nel tempo il modello lineare del processo:

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$$

con $t = (k+1)T$, $t_0 = kT$, $x(t_0) = x(kT)$.

- Poiché l'ingresso è costante a tratti

$$u(\tau) \equiv \bar{u}(k),\ \ kT \leq \tau < (k+1)T$$

si ottiene:

$$x((k+1)T) = e^{AT}x(kT) + \left(\int_0^T e^{A(T-\tau)}d\tau\right)B\bar{u}(k)$$

e quindi

$$\bar{x}(k+1) = e^{AT}\bar{x}(k) + \left(\int_0^T e^{A\tau}d\tau\right)B\bar{u}(k)$$

Campionamento esatto

- Il sistema tempo-discreto a segnali campionati

$$\begin{cases} \bar{x}(k+1) &= \bar{A}\bar{x}(k) + \bar{B}\bar{u}(k) \\ \bar{y}(k) &= \bar{C}\bar{x}(k) + \bar{D}\bar{u}(k) \end{cases}$$

  è legato al sistema tempo continuo dalle relazioni

$$\boxed{\begin{array}{ll} \bar{A} \triangleq e^{AT} & \bar{B} \triangleq \int_0^T e^{A\tau} B d\tau \\ \bar{C} \triangleq C & \bar{D} \triangleq D \end{array}}$$

- Nota: in generale, affinchè il sistema a tempo discreto $(\bar{A}, \bar{B}, \bar{C}, \bar{D})$ e il sistema a tempo continuo (A, B, C, D) coincidano agli istanti di campionamento $t = kT$ occorre che l'ingresso $u(t)$ sia costante durante l'intervallo di campionamento.

In Matlab:
```
sys=ss(A,B,C,D);
sysd=c2d(sys,T);
[Ab,Bb,Cb,Db]=ssdata(sysd);
```

Stabilità dei sistemi lineari

sistema	*tempo continuo* $\dot{x}(t) = Ax(t)$	*tempo discreto* $x(k+1) = Ax(k)$
as. stabile	$Re(\lambda_i) < 0 \ \forall i = 1, \ldots, n$	$\lvert\lambda_i\rvert < 1$
instabile	$\exists i$ tale che $Re(\lambda_i) > 0$	$\lvert\lambda_i\rvert > 1$
stabile	1) $\forall i, \ldots, n,\ Re(\lambda_i) \leq 0$ 2) $\forall \lambda_i$ tale che $Re(\lambda_i) = 0$ molt(alg.)=molt(geom.)	$\lvert\lambda_i\rvert \leq 1$ $\lvert\lambda_i\rvert = 1$

Linearizzazione

- Considera il sistema non lineare

$$\begin{cases} \dot{x}(t) & = & f(x(t), u(t)) \\ y(t) & = & g(x(t), u(t)) \end{cases}$$

e sia (x_r, u_r) un equilibrio: $f(x_r, u_r) = 0$

- Obiettivo: studiare il sistema per piccole variazioni $\Delta u(t) \triangleq u(t) - u_r$e $\Delta x(0) \triangleq x(0) - x_r$.

- L'evoluzione di $\Delta x(t) \triangleq x(t) - x_r$ è data da

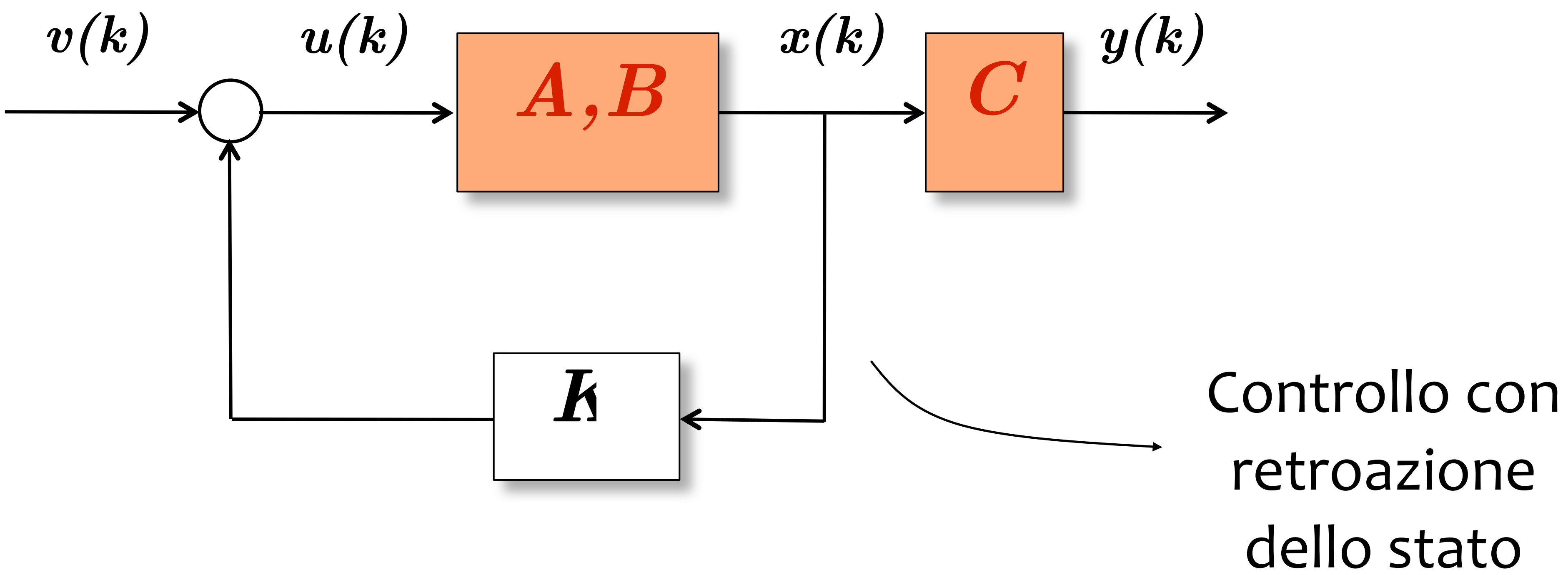
$$\begin{aligned} \dot{\Delta}x(t) &= \dot{x}(t) - \dot{x}_r = f(x(t), u(t)) \\ &= f(\Delta x(t) + x_r, \Delta u(t) + u_r) \\ &\approx \underbrace{\frac{\partial f}{\partial x}(x_r, u_r)}_{A} \Delta x(t) + \underbrace{\frac{\partial f}{\partial u}(x_r, u_r)}_{B} \Delta u(t) \end{aligned}$$

- In maniera simile,

$$\Delta y(t) \approx \underbrace{\frac{\partial g}{\partial x}(x_r, u_r)}_{C} \Delta x(t) + \underbrace{\frac{\partial g}{\partial u}(x_r, u_r)}_{D} \Delta u(t)$$

dove $\Delta y(t) \triangleq y(t) - g(x_r, u_r)$ è la deviazione dell'uscita dall'equilibrio.

Raggiungibilità: motivazione



- Primo obiettivo del controllo è la stabilizzazione.
- Si dispone delle misure di tutto lo stato.
- IDEA: utilizzare istantaneamente le misure dello stato per modificare la dinamica del sistema.

  È possibile determinare K tale che $A + BK$ è asintoticamente stabile?
- Essendo $u(k) = Kx(k) + v(k)$, risulta $x(k+1) = (A + BK)x(k) + Bv(k)$.
- La RAGGIUNGIBILITÀ affronta questo tipo di problema, dicendoci *quando* e *come* il problema può essere risolto

Raggiungibilità

$$x(k+1) = Ax(k)+Bu(k), \qquad x(0) = x_0 \qquad\qquad (x \in \mathbb{R}^n, u \in \mathbb{R}^m)$$

Soluzione: $$x(k) = A^k x_0 + \sum_{j=0}^{k-1} A^j Bu(k-1-j)$$

Definizione

Il sistema si dice *(completamente) raggiungibile* se per ogni x_1, $x_2 \in \mathbb{R}^n$ esistono $k \in \mathbb{N}$ e $u(0)$, $u(1)$, ..., $u(k-1) \in \mathbb{R}^m$ tali che

$$x_2 = A^k x_1 + \sum_{j=0}^{k-1} A^j Bu(k-1-j)\ .$$

Raggiungibilità

Si consideri il problema di determinare, se esiste, una sequenza di n ingressi che permette di portare lo stato dalla condizione iniziale x_1 alla condizione finale x_2. Essendo:

$$\underbrace{x_2 - A^n x_1}_{X} = \underbrace{\left[\, B\ AB\ \dots\ A^{n-1}B \,\right]}_{R} \underbrace{\begin{bmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(0) \end{bmatrix}}_{U}$$

il problema è equivalente a risolvere rispetto a U il sistema

$$X = RU \ ,$$

dove la matrice $R \in \mathbb{R}^{n \times nm}$ è detta matrice di raggiungibilità

- Il sistema ammette soluzione se e solo se $X \in \mathrm{Im}(R)$ (Teorema di Rouché-Capelli: $\mathrm{rank}([R\ X]) = \mathrm{rank}(R)$)

- Esiste una soluzione per ogni X se e solo se $\mathrm{rank}(R) = n$.

Raggiungibilità

TEOREMA

Il sistema è raggiungibile se e solo se $\text{rank}(R) = n$.

Dimostrazione. (necessità) Se il sistema è raggiungibile, allora scegliendo $x_1=0$ e $x_2=x$ si ha che $x = \sum_{j=0}^{k-1} A^j Bu(k-1-j)$. Se $k \leq n$, allora direttamente $x \in \text{Im}(R)$. D'altronde, se $k > n$, applicando il teorema di Hamilton-Cayley si ottiene ancora $x \in \text{Im}(R)$. Per l'arbitrarietà di x, segue che $\text{Im}(R) = \mathbb{R}^n$, e quindi $\text{rank}(R) = n$.

(sufficienza) Se $\text{rank}(R) = n$, allora $\text{Im}(R) = \mathbb{R}^n$, e quindi il sistema $X = RU$, dove $X = x_2 - A^n x_1$ e $U = [\, u(n-1)' \ \ldots \ u(1)' \, u(0)']'$, è risolvibile rispetto a U per ogni X. Dunque il sistema è raggiungibile.

Controllabilità

- Sotto l'ipotesi di raggiungibilità, si è visto che è possibile risolvere il problema di trovare una sequenza finita di ingressi che permette di portare lo stato dall'origine in un punto arbitrario $x \in \mathbb{R}^n$: $U = R^{\#}x$.

- Ci poniamo ora il problema inverso, ossia trovare una sequenza finita di ingressi che permette di portare lo stato da un punto arbitrario $x_0 \in \mathbb{R}^n$ nell'origine.

Definizione

Il sistema si dice *controllabile (all'origine) in k passi* se per ogni $x_0 \in \mathbb{R}^n$ esistono $u(0)$, $u(1)$, $\ldots$, $u(k-1) \in \mathbb{R}^m$ tali che $0 = A^k x_0 + \sum_{j=0}^{k-1} A^j B u(k-1-j)$.

Controllabilità

- Il sistema

$$-A^k x_0 = \underbrace{\left[\, B \; AB \; \ldots \; A^{k-1}B \, \right]}_{R_k} \begin{bmatrix} u(k-1) \\ u(k-2) \\ \vdots \\ u(0) \end{bmatrix}$$

ammette soluzione per ogni $x_0 \in \mathbb{R}^n$ se e solo se $\text{Im}(A^k) \subseteq \text{Im}(R_k)$.

Dunque il sistema è *controllabile (all'origine) in k passi* se e solo se

$$\text{Im}(A^k) \subseteq \text{Im}(R_k) \; .$$

- Dato che $\text{Im}(A^k) = \text{Im}(A^n)$ per ogni $k > n$, un sistema controllabile in n passi è controllabile in k passi per ogni $k > n$.

- D'altra parte, se un sistema è controllabile in k passi con $k < n$, allora è controllabile in n passi.

Posizionamento dei poli mediante retroazione dello stato

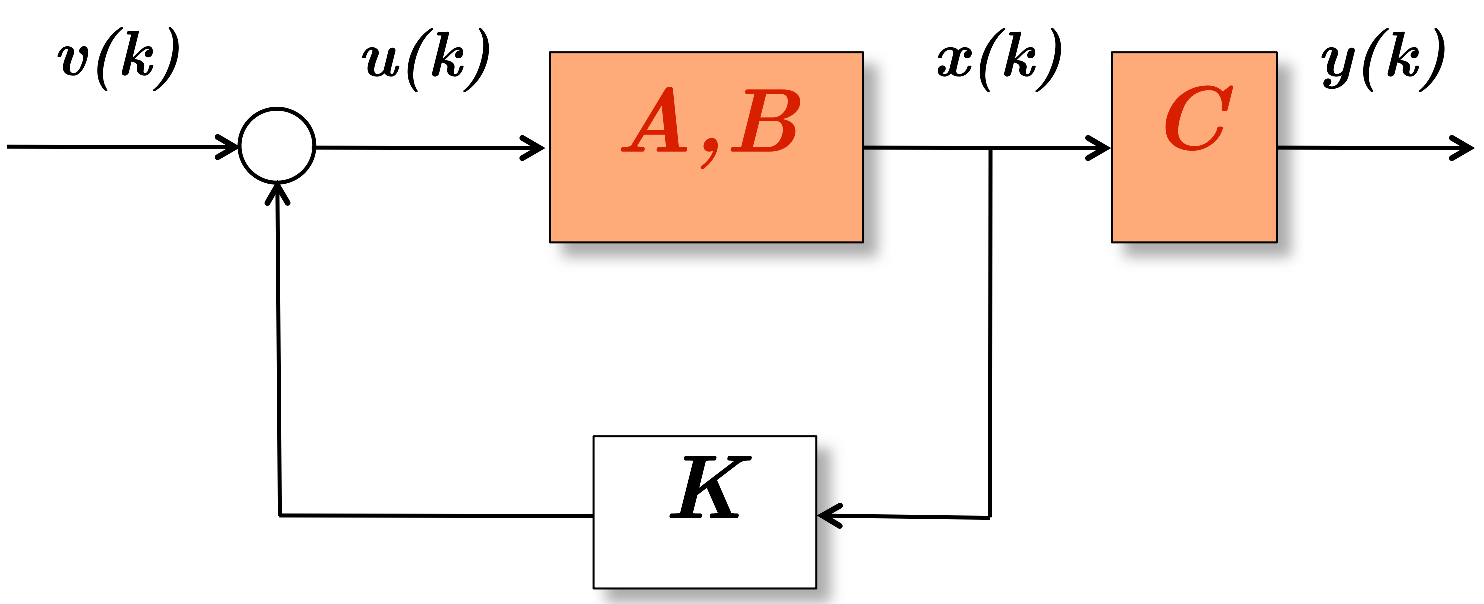
Controllo con retroazione dello stato

Problema: progettare un dispositivo che, connesso al sistema da controllare, renda asintoticamente stabile il sistema complessivo risultante.

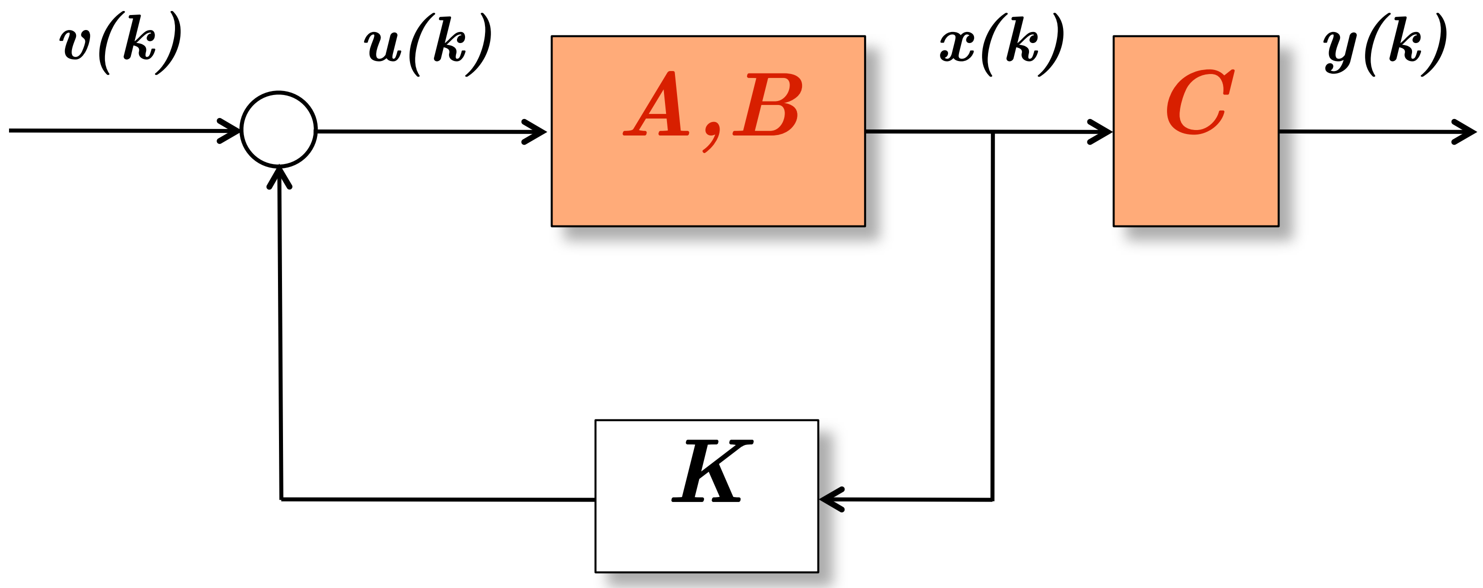
Soluzione con retroazione dello stato

Se sono disponibili le misure di tutto lo stato del sistema, possiamo generare l'ingresso di controllo moltiplicando le misure dello stato per un guadagno statico K=[k_1 k_2 ... k_n]:

$$u(k) = k_1 x_1(k) + k_2 x_2(k) + \ldots + k_n x_n(k) + v(k)$$



Controllo con retroazione dello stato



- Essendo $u(k) = Kx(k) + v(k)$, il sistema complessivo ha equazioni:

$$\begin{aligned} x(k+1) &= (A+BK)x(k) + Bv(k) \\ y(k) &= (C+DK)x(k) + Dv(k) \end{aligned}$$

TEOREMA
Se (A, B) è raggiungibile gli autovalori di $A + BK$ possono essere decisi arbitrariamente.

Assegnazione degli autovalori

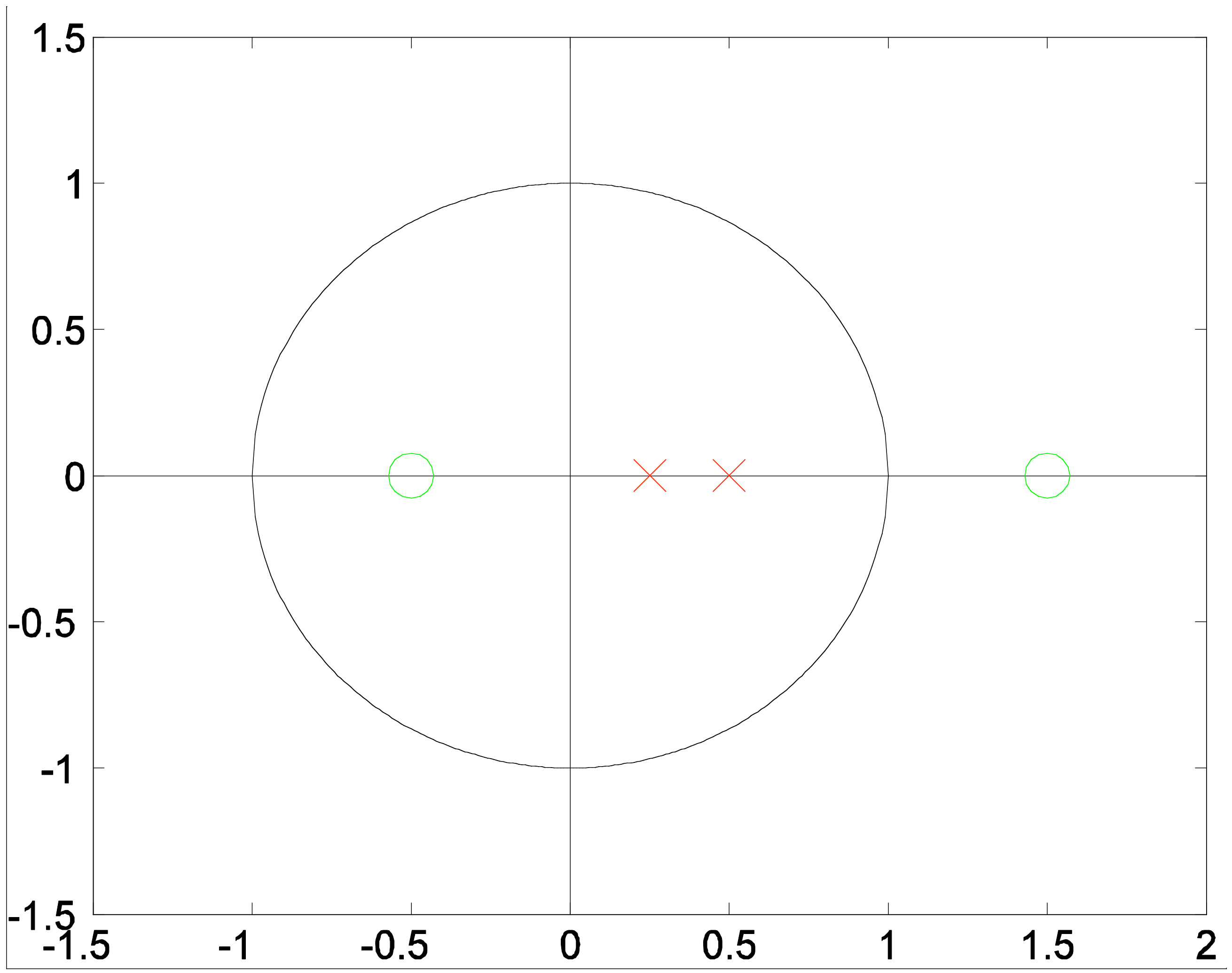
Formula di Ackermann:

- Siano:
  $p_A(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0$ il polinomio caratteristico di A
  $p_d(\lambda) = \lambda^n + d_{n-1}\lambda^{n-1} + \ldots + d_1\lambda + d_0$ è il polinomio caratteristico desiderato per la matrice $A + BK$ ad anello chiuso.
- Sia $p_d(A) = A^n + d_{n-1}A^{n-1} + \ldots + d_1 A + d_0 I$ (è una matrice $n \times n$)
- Allora

$$\boxed{K = -[0\ 0\ \ldots\ 0\ 1][B\ AB\ \ldots\ A^{n-1}B]^{-1}p_d(A)}$$

In MATLAB: **K=-acker(A,B,P); K=-place(A,B,P);**
dove P=[λ_1 λ_2 ... λ_n] sono i poli desiderati ad anello chiuso

Pole-placement: esempio



o = autovalori ad anello aperto

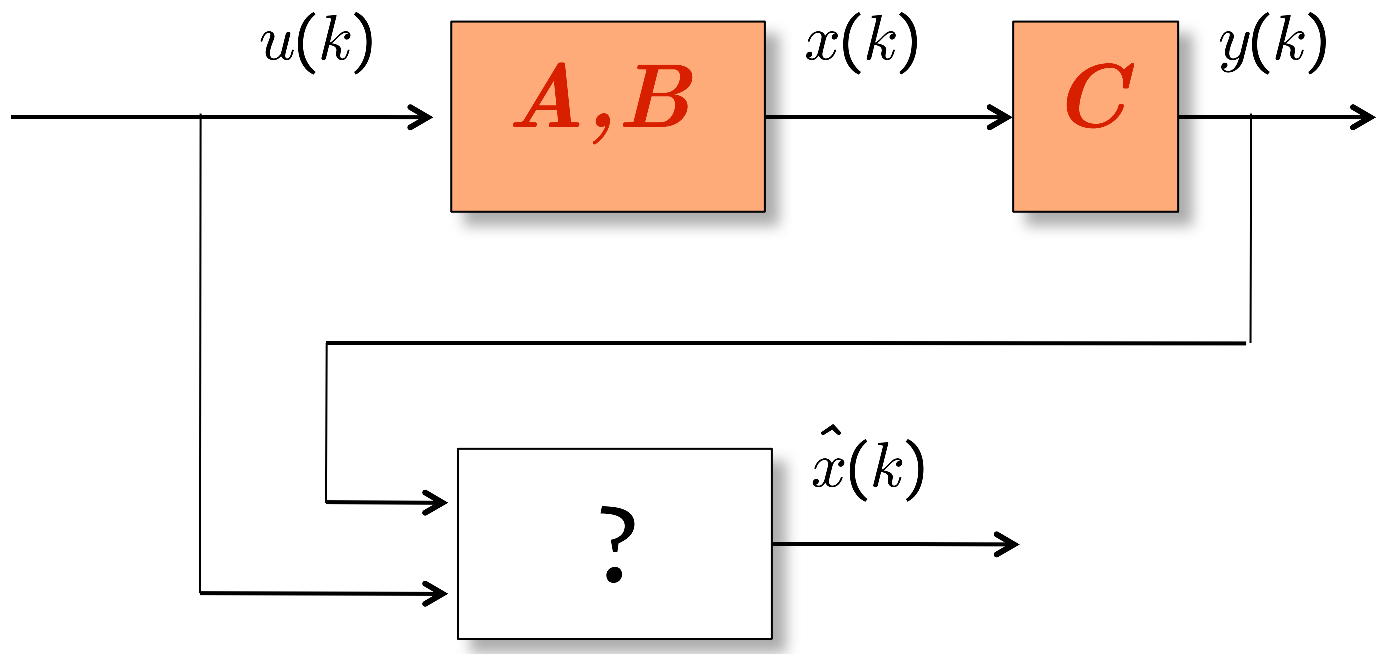
x = autovalori ad anello chiuso

<u>In MATLAB:</u>

```
» A=[ 0 -1/4 ; -3 1 ];
» B=[ 1 ; -1/2 ];
» K=-place(A,B,[ 1/2 1/4 ])
K =
    -0.0909   0.3182
```

Osservabilità e stima dello stato

Motivazione



- Osservazione: per implementare il controllo con retroazione dello stato $u = Kx$ si ha bisogno di tutto il vettore di stato x

- Problema: Spesso solo l'uscita y è disponibile dai sensori

- IDEA: è possibile ricostruire lo stato x del sistema a partire dalle misure di uscita y e degli ingressi u ?

- La OSSERVABILITÀ affronta questo tipo di problema, dicendoci *quando* e *come* il problema può essere risolto

Osservabilità

Si consideri il problema di ricostruire la condizione iniziale x_0 a partire da n misure dell'uscita, noti gli ingressi applicati.

$$\begin{cases} y(0) = Cx_0 + Du(0) \\ y(1) = CAx_0 + CBu(0) + Du(1) \\ \vdots \\ y(n-1) = CA^{n-1}x_0 + \sum_{j=1}^{n-2} CA^j Bu(n-2-j) + Du(n-1) \end{cases}$$

Posto:

$$Y = \begin{bmatrix} y(0) - Du(0) \\ y(1) - CBu(0) - Du(1) \\ \vdots \\ y(n-1) - \sum_{j=1}^{n-2} CA^j Bu(n-2-j) - Du(n-1) \end{bmatrix} \quad \Theta = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Osservabilità

... si deve risolvere (rispetto a x_0) il sistema:

$$Y = \Theta x_0 \ ,$$

dove la matrice $\Theta \in \mathbb{R}^{np \times n}$ è la matrice di osservabilità del sistema.

In assenza di rumore sulla misura dell'uscita, il sistema ha sempre soluzione. In particolare:

- la soluzione è unica se $\text{rank}(\Theta) = n$;
- esistono infinite soluzioni se $\text{rank}(\Theta) < n$. In questo caso, tutte le soluzioni sono date da $x_0 + ker(\Theta)$, essendo x_0 una soluzione particolare del sistema.

Una volta nota la condizione iniziale, e noti gli ingressi, si può prevedere l'evoluzione dello stato in tutti gli istanti futuri.

TEOREMA

Il sistema è osservabile se e solo se $\text{rank}(\Theta) = n$.

Dimostrazione. (necessità) Se il sistema è osservabile, si supponga per assurdo che $\text{rank}(\Theta)<n$. Dunque esiste $x\neq 0$ tale che $\Theta x = 0$, e quindi $Cx = 0$, $CAx = 0$, ..., $CA^{n-1}x = 0$. Per il teorema di Hamilton-Cayley segue che $CA^k x = 0$ per ogni k. Ma allora x è indistinguibile dall'origine $\Rightarrow$ contraddizione.

(sufficienza) Se $\text{rank}(\Theta) = n$, si supponga per assurdo che esistano $x_1 \neq x_2$ indistinguibili dall'uscita, e quindi tali che $CA^k x_1 = CA^k x_2$ per ogni k. Posto $x = x_1 - x_2$, segue che $Cx = 0$, $CAx = 0$, ..., $CA^{n-1}x = 0$, ossia $\Theta x = 0$, con $x \neq 0 \Rightarrow$ contraddizione.

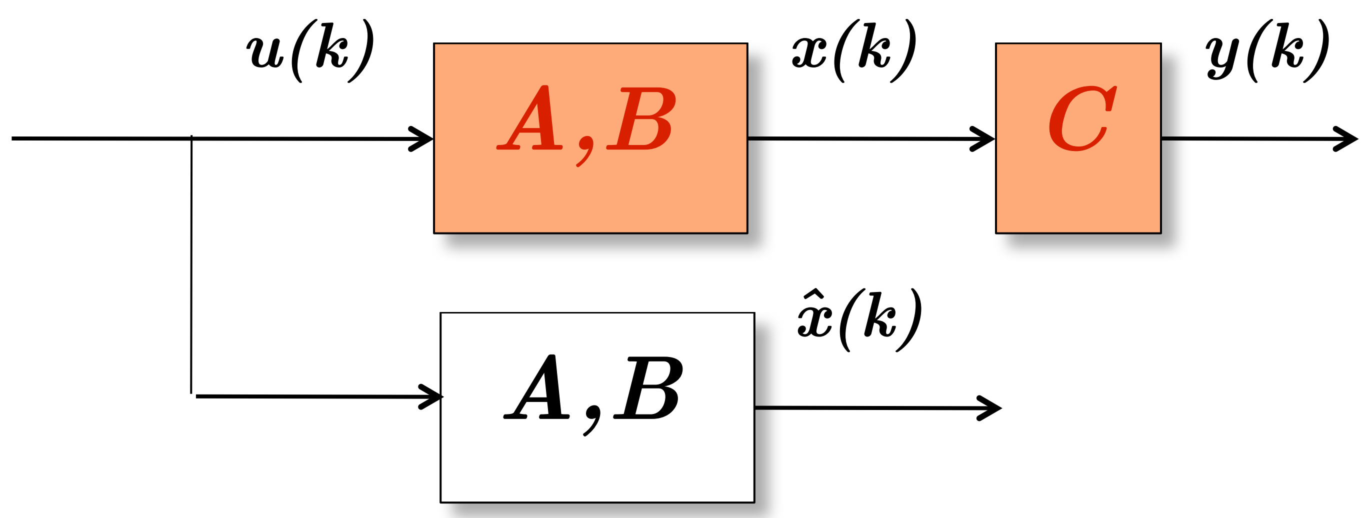
Osservatore (stimatore asintotico dello stato)

Stimatore asintotico dello stato

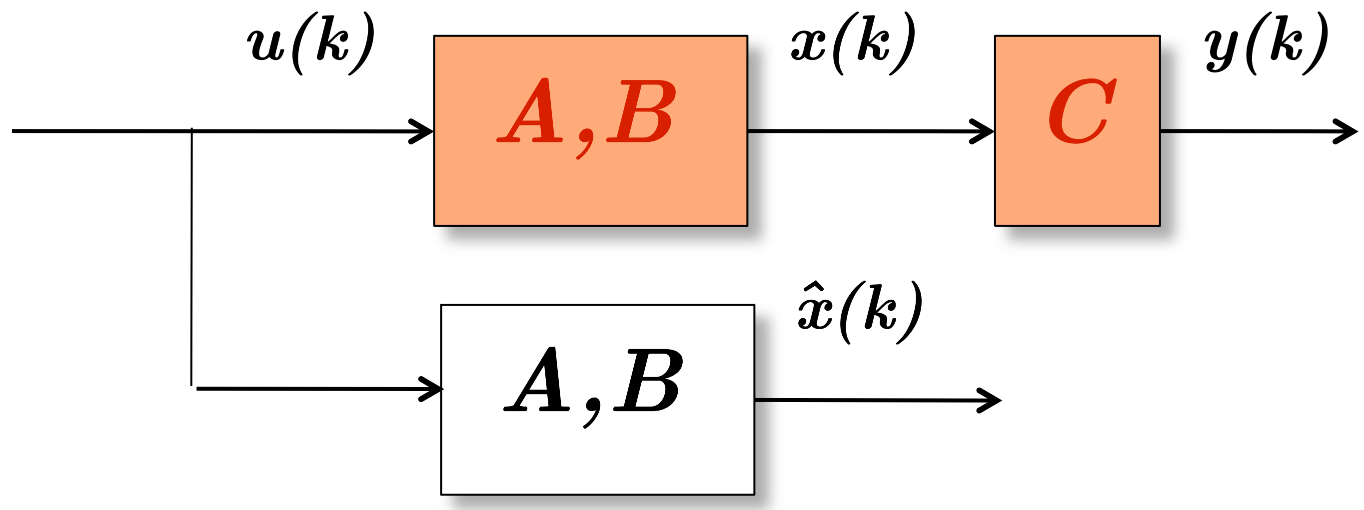
Problema: progettare un dispositivo che permetta di ricostruire una stima $\hat{x}(k)$ dello stato $x(k)$ del sistema quando questo non è direttamente misurabile

Soluzione #1:

Affianchiamo al sistema (reale) che genera i dati una copia (artificiale) comandata dagli stessi ingressi. In pratica, aggiungiamo un "simulatore" $\hat{x}(k+1) = A\hat{x}(k) + Bu(k)$ che riproduca il comportamento del sistema reale.



Stimatore asintotico dello stato



- Modello:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) \end{aligned}$$

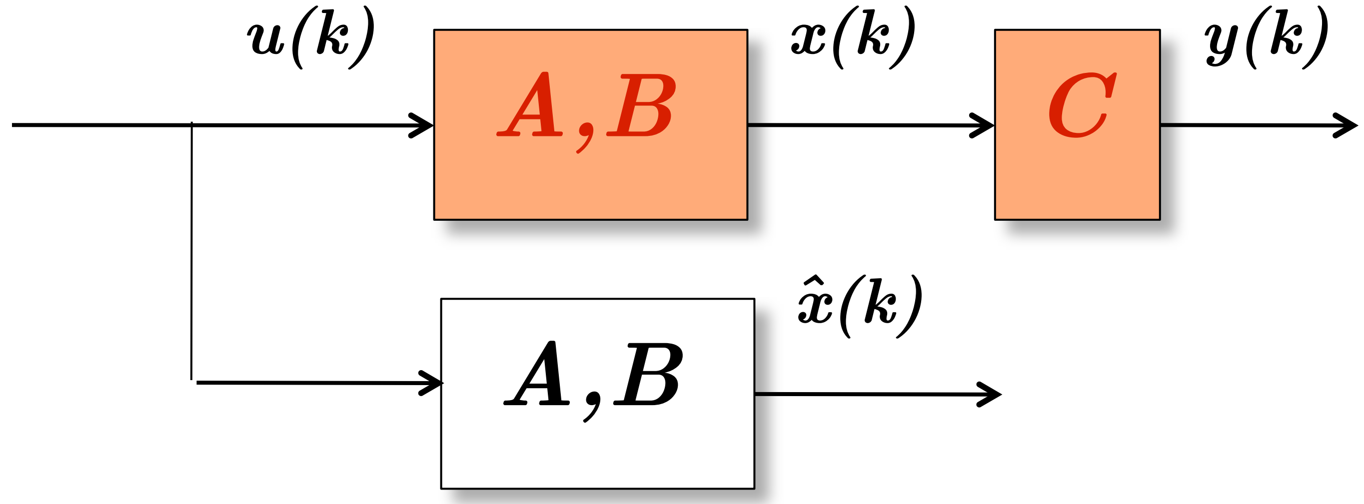
- Errore di stima: $\tilde{x}(k) \triangleq x(k) - \hat{x}(k)$

- Dinamica dell'errore di stima:

$$\tilde{x}(k+1) = Ax(k) + Bu(k) - A\hat{x}(k) - Bu(k) = A\tilde{x}(k)$$

e quindi $\tilde{x}(k) = A^k(x(0) - \hat{x}(0))$

Stimatore asintotico dello stato



Il fatto che la dinamica sia $\tilde{x}(k) = A^k(x(0) - \hat{x}(0))$ può comportare dei problemi:

- La dinamica dell'errore di stima non è modificabile, è dettata direttamente da A
- L'errore di stima si annulla asintoticamente se e solo se A è asintoticamente stabile !
- Nello stimare lo stato $\hat{x}(k)$ non si sta sfruttando minimamente la conoscenza dell'uscita misurata $y(k)$!!!

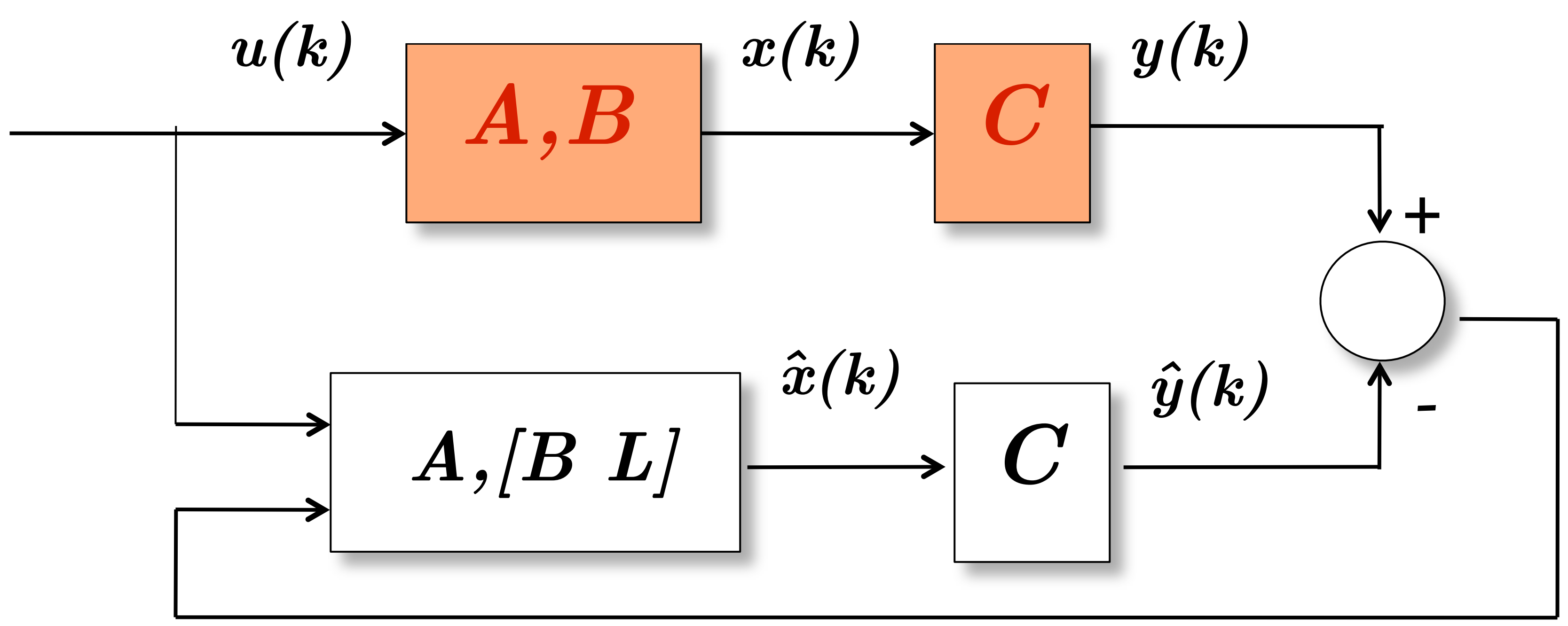
Stimatore asintotico dello stato

Soluzione #2:

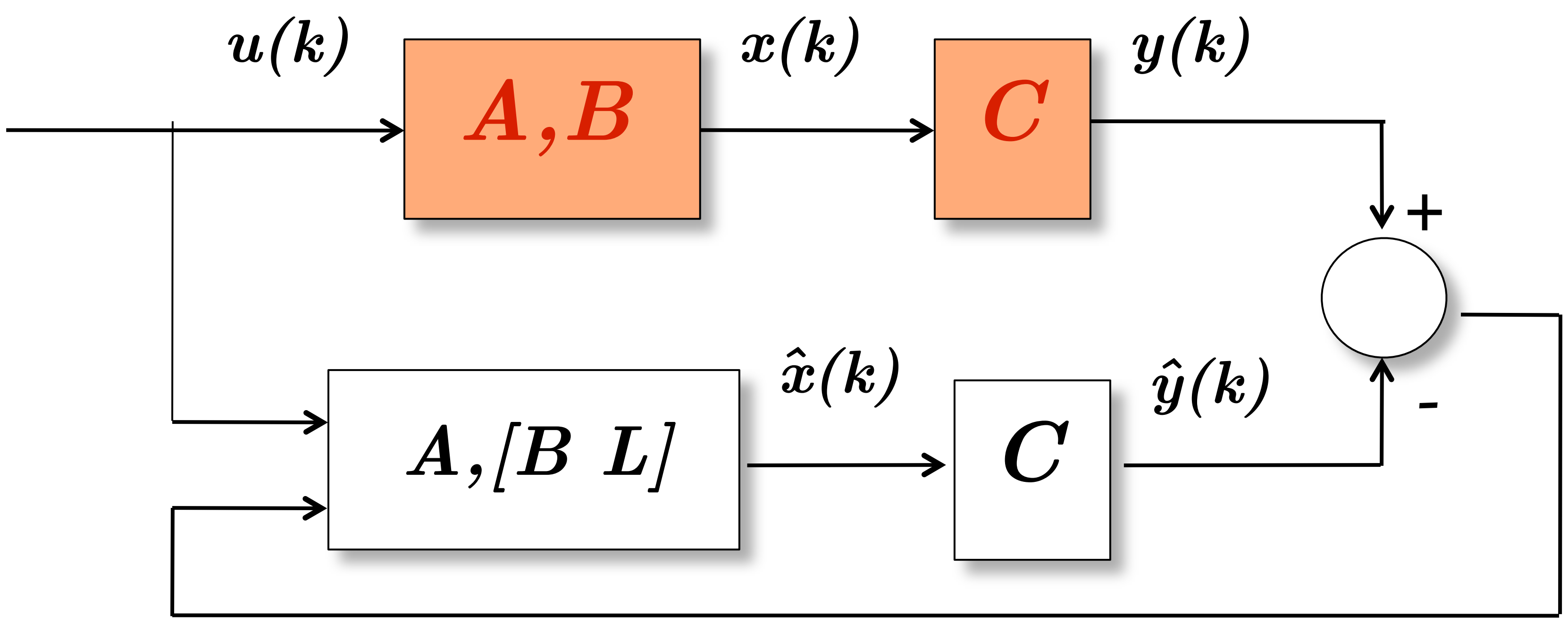
Aggiungiamo nell'equazione dello stimatore un termine che dipende dall'errore di stima

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + L(y(k) - C\hat{x}(k))$$

dove $L \in \mathbb{R}^{n \times p}$ è il guadagno dello stimatore.



Stimatore asintotico dello stato



La dinamica dell'errore di stima è

$$
\begin{aligned}
\tilde{x}(k+1) &= Ax(k) + Bu(k) - A\hat{x}(k) - Bu(k) - L[y(k) - C\hat{x}(k)] \\
&= (A - LC)\tilde{x}(k)
\end{aligned}
$$

e quindi $\tilde{x}(k) = (A - LC)^k(x(0) - \hat{x}(0))$

TEOREMA

Se (A, C) è osservabile gli autovalori di $(A - LC)$ possono essere decisi arbitrariamente

Stimatore asintotico dello stato

Esempio MATLAB:

```
» sys=tf([1 0],[1 2 1]);
```

$$G(s) = \frac{s}{s^2 + 2s + 1}$$

```
» sysd=c2d(sys,.1);
```

$$G(z) = 0.09048 \frac{z-1}{z^2 - 1.81z + 0.8187}$$

```
» [A,B,C,D]=ssdata(ss(sysd));
```

$$A = \begin{bmatrix} 1.8097 & -0.8187 \\ 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.1810 & -0.1810 \end{bmatrix} \quad D = \begin{bmatrix} 0 \end{bmatrix}$$

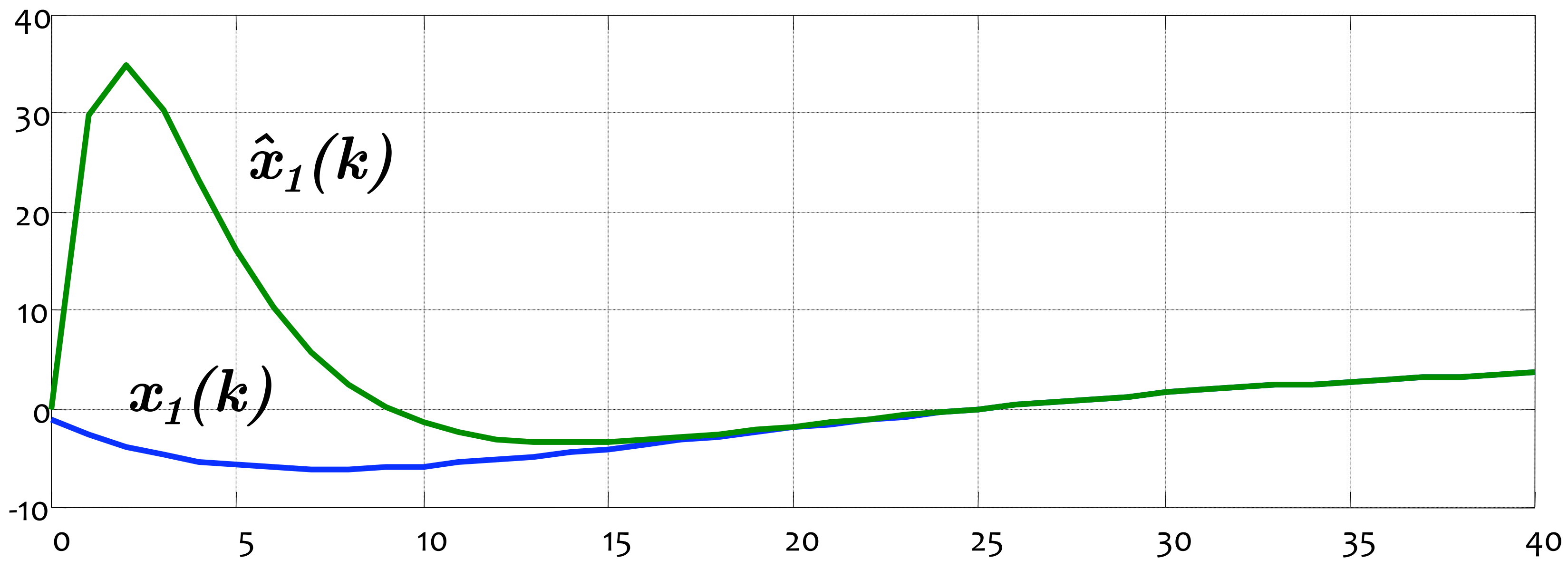
```
» L=place(A',C',[.5 .7])';
```

$$L = \begin{bmatrix} -82.6341 \\ -86.0031 \end{bmatrix}$$

```
» eig(A-L*C)

ans =

    0.7000
    0.5000
```

Stimatore asintotico dello stato

Segue esempio MATLAB:



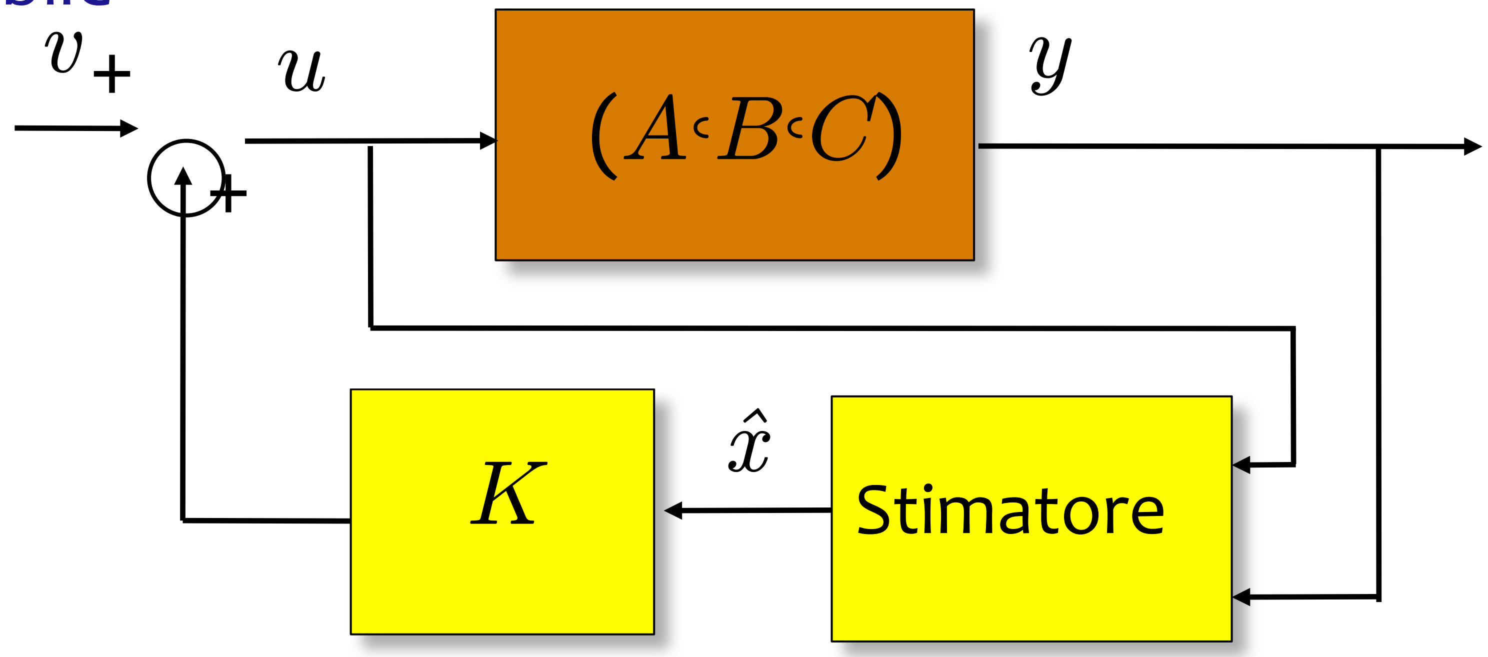
Condizione iniziale: $x(0) = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, $\hat{x}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

```
x=[-1;1];
xhat=[0;0];

XX=x;
XXhat=xhat;
T=40;

UU=.1*ones(1,T);

for k=0:T-1,
   u=UU(k+1);
   y=C*x+D*u;
   yhat=C*xhat+D*u;

   x=A*x+B*u;
   xhat=A*xhat+B*u+L*(y-yhat);

   XX=[XX,x];
   XXhat=[XXhat,xhat];
end

subplot(211)
plot(0:T,[XX(1,:);XXhat(1,:)]);
grid
title('x_1')
```

Compensatore dinamico

Compensatore dinamico

Ipotesi: sistema compl. raggiungibile
e compl. osservabile



$$\boxed{u(k) = K\hat{x}(k) + v(k)}$$

- Stimatore dello stato:

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + L(y(k) - C\hat{x}(k))$$

- Dinamica dell'errore $\tilde{x} = x - \hat{x}$:

$$\tilde{x}(k+1) = Ax(k) + \cancel{Bu(k)} - A\hat{x}(k) - \cancel{Bu(k)} + L(Cx(k) - C\hat{x}(k)) = (A - LC)\tilde{x}(k)$$

$$\Rightarrow \quad \boxed{\tilde{x}(k) = (A - LC)^k \tilde{x}(0)}$$

N.B.: non dipende
da $u(k)$, e quindi da K !

Sistema complessivo ad anello chiuso

Dinamica del sistema complessivo:

$$\begin{cases} x(k+1) &= Ax(k) + Bu(k) \\ \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) + L(y(k) - C\hat{x}(k)) \\ u(k) &= K\hat{x}(k) + v(k) \\ y(k) &= Cx(k) \end{cases}$$

Effettuiamo un cambio di coordinate:

$$\begin{bmatrix} x(k) \\ \tilde{x}(k) \end{bmatrix} = \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} x(k) \\ \hat{x}(k) \end{bmatrix}$$

La dinamica ad anello chiuso è descritta equivalentemente come

$$\begin{cases} \begin{bmatrix} x(k+1) \\ \tilde{x}(k+1) \end{bmatrix} = \begin{bmatrix} A+BK & -BK \\ 0 & A-LC \end{bmatrix} \begin{bmatrix} x(k) \\ \tilde{x}(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} v(k) \\ \\ y(k) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ \tilde{x}(k) \end{bmatrix} \end{cases}$$

Sistema complessivo ad anello chiuso

Funzione di trasferimento da v a y:

$$G(z) \triangleq \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} zI - A - BK & BK \\ 0 & zI - A + LC \end{bmatrix}^{-1} \begin{bmatrix} B \\ 0 \end{bmatrix} =$$

$$= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} (zI - A - BK)^{-1} & \star \\ 0 & (zI - A + LC)^{-1} \end{bmatrix} \begin{bmatrix} B \\ 0 \end{bmatrix} =$$

$$= C(zI - A - BK)^{-1}B = \frac{N(z)}{D_K(z)}$$

LA FUNZIONE DI TRASFERIMENTO AD ANELLO CHIUSO
È RIMASTA IDENTICA AL CASO STATE-FEEDBACK !!!

Pertanto, il comportamento ingresso uscita del sistema ad anello chiuso non dipende dal guadagno L dell'osservatore

Principio di separazione

PRINCIPIO DI SEPARAZIONE:

Poiché solo K influisce sul comportamento I/O dell'anello chiuso, e solo L influisce sull'evoluzione dell'errore di stima, posso progettare K e L indipendentemente l'uno dall'altro.

Attenzione: $G(z) = C(zI - A - BK)^{-1}B$ rappresenta solo il comportamento ingresso/uscita del sistema (condizioni iniziali nulle e/o transitorio esaurito) !

Poli del sistema ad anello chiuso:

$$\det(zI - \left[\begin{smallmatrix} A+BK & -BK \\ 0 & A-LC \end{smallmatrix}\right]) = \det(zI - A - BK)\det(zI - A + LC) = D_K(z)D_L(z)$$

Si ha quindi una cancellazione dei poli dell'osservatore:

$$G(z) = \begin{bmatrix} C & 0 \end{bmatrix}(zI - \left[\begin{smallmatrix} A+BK & -BK \\ 0 & A-LC \end{smallmatrix}\right])^{-1}\begin{bmatrix} B \\ 0 \end{bmatrix} = \frac{N(z)\cancel{D_L(z)}}{D_K(z)\cancel{D_L(z)}}$$

I poli del sistema complessivo sono rappresentati dall'unione dei poli dati dal regolatore K più quelli dati dallo stimatore L.

Scelta dello stimatore

La scelta di L sembra ininfluente.

Guardiamo però all' effetto delle condizioni iniziali $\begin{bmatrix} x(0) \\ \tilde{x}(0) \end{bmatrix}$ per $v(k) \equiv 0$:

$$\begin{aligned}
y(0) &= Cx(0) \\
y(1) &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} A-BK & -BK \\ 0 & A-LC \end{bmatrix} \begin{bmatrix} x(0) \\ \tilde{x}(0) \end{bmatrix} \\
&= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} (A+BK)x(0)-BK\tilde{x}(0) \\ (A-LC)\tilde{x}(0) \end{bmatrix} = C(A+BK)x(0) - CBK\tilde{x}(0) \\
y(2) &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} A-BK & -BK \\ 0 & A-LC \end{bmatrix} \begin{bmatrix} x(1) \\ \tilde{x}(1) \end{bmatrix} \\
&= C(A+BK)x(1) - CBK\tilde{x}(1) \\
&= C(A+BK)^2x(0) - C(A+BK)BK\tilde{x}(0) - CBK(A-LC)\tilde{x}(0)
\end{aligned}$$

La scelta di L influisce durante il transitorio !

Scelta dello stimatore

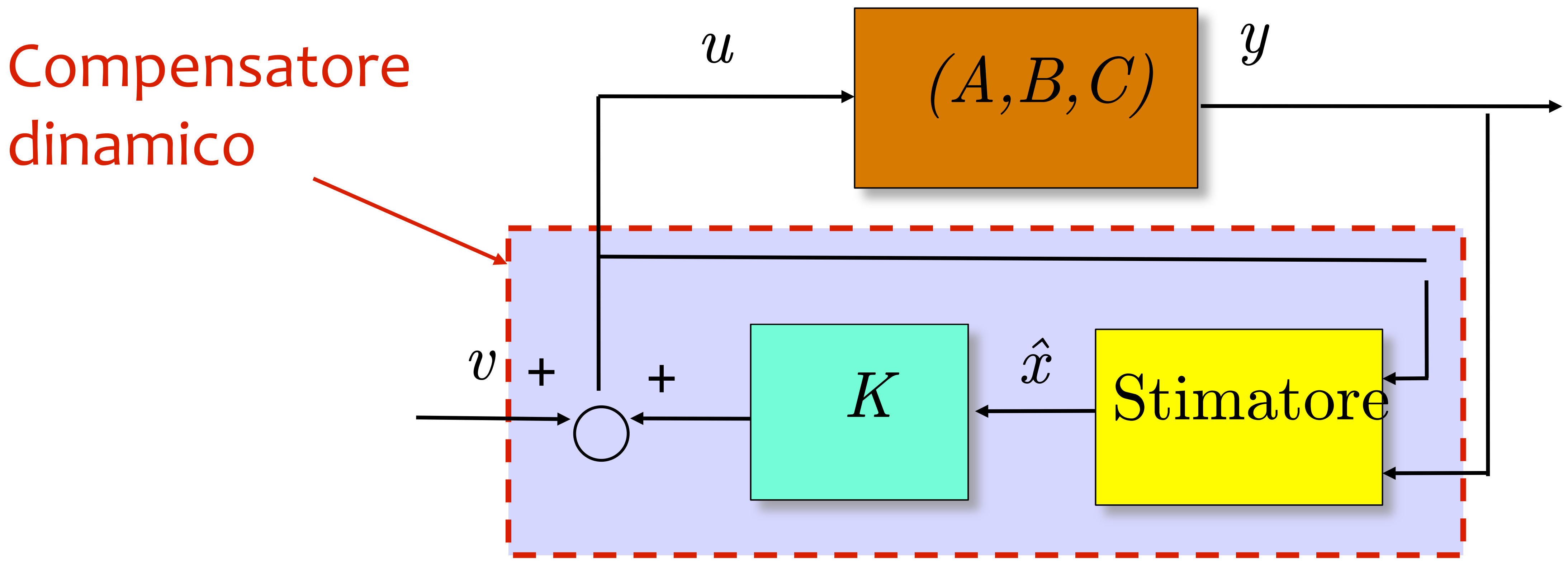
Intuitivamente: $u = K\hat{x}(t) + v$, dove $\hat{x}(t)$ dipende da L. Se $\hat{x}$ è una cattiva stima, anche il controllo ne deve risentire.

Regola pratica: scegliere i poli dell'osservatore $\simeq 10$ volte più veloci di quelli del controllore

(oppure: usare uno stimatore ottimo = filtro di Kalman – Vedi corso di Identificazione e Analisi dei Dati).

La scelta di L è quindi molto importante, soprattutto nei confronti di rumori additivi sull'ingresso e sull'uscita (vedi più avanti ...)

Compensatore Dinamico



Equazioni del compensatore dinamico:

$$\begin{cases} \hat{x}(k+1) = (A+BK-LC)\hat{x}(k) + Bv(k) + Ly(k) \\ u(k) = K\hat{x}(k) + v(k) \end{cases}$$

In termini di funzione di trasferimento:

$$u = (K(zI-A-BK+LC)^{-1}B+I)v + K(zI-A-BK+LC)^{-1}Ly$$

Esempio MATLAB

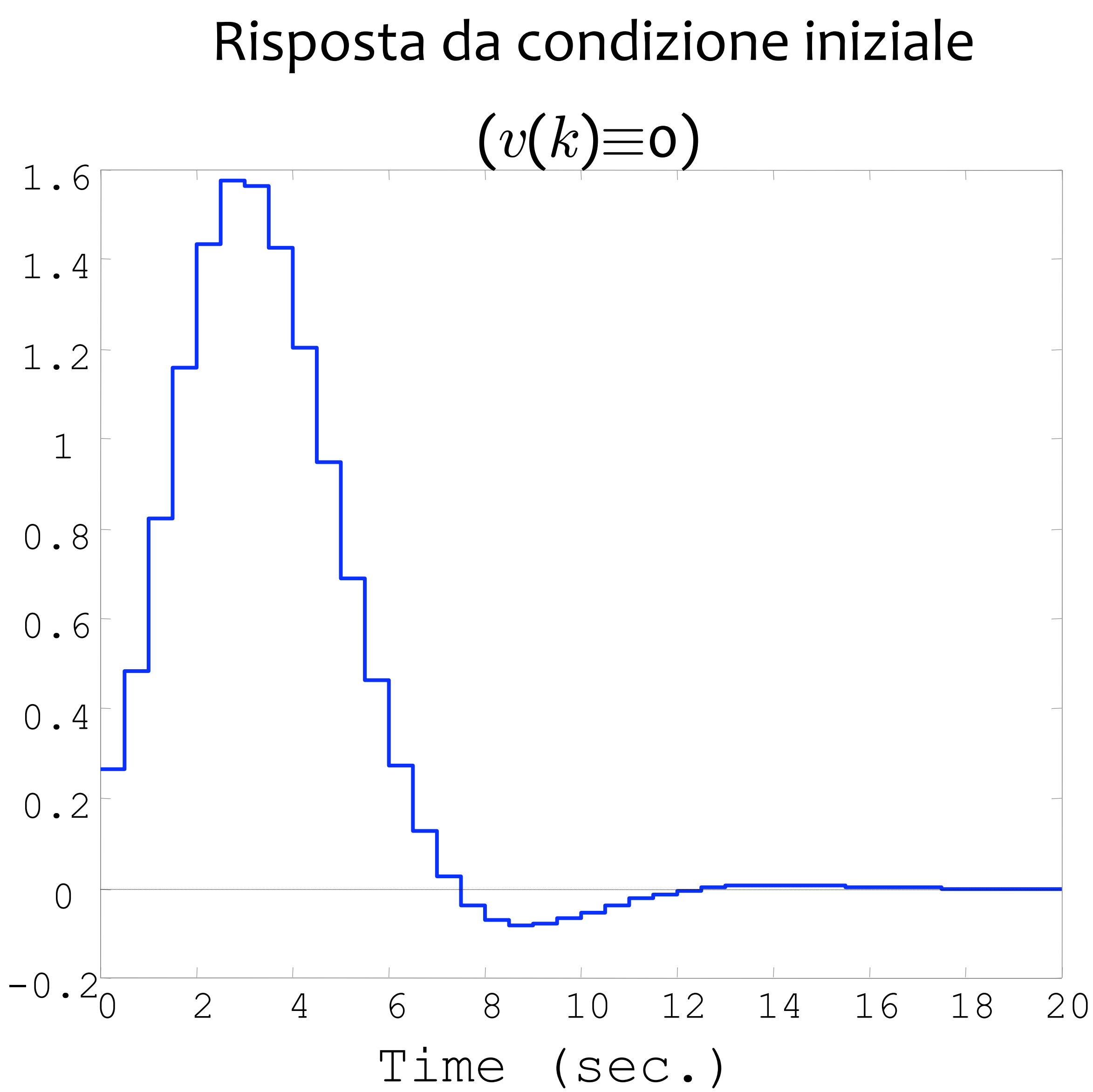
MOTORE DC

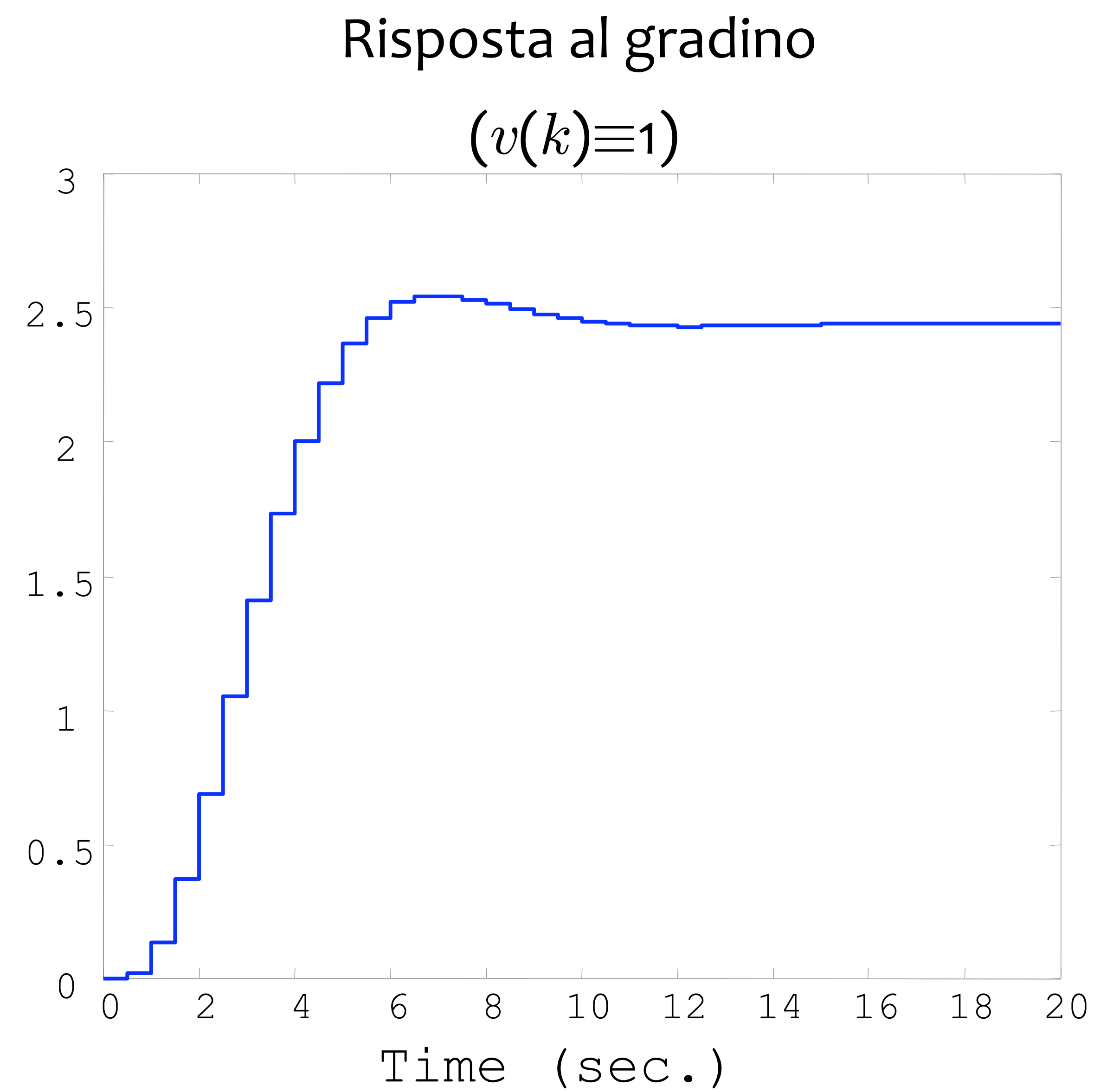
$$G(s) = \frac{K}{s^3 + \beta s^2 + \alpha s}$$

```
K=1;
beta=.3;
alpha=1;

G=tf(K,[1 beta alpha 0]);

ts=0.5;

Gd=c2d(G,ts);
sysd=ss(Gd);
[A,B,C,D]=ssdata(sysd);

% Controllore
policontinuo=[-1, -0.5+0.6*j,
-0.5-0.6*j];
polidiscreto=exp(ts*policontinuo);
K=-place(A,B,polidiscreto);

% Osservatore
policontinuo=[-10, -9, -8];
polidiscreto=exp(ts*policontinuo);
L=place(A',C',polidiscreto)';
```

```
%u=K*xhat+v
bigA=[A,B*K;L*C,A+B*K-L*C];
bigB=[B;B];
bigC=[C,zeros(1,3)];
bigD=0;

T=20;

clsys=ss(bigA,bigB,bigC,bigD,ts)
x0=[1 1 1]';
xhat0=[0 0 0]';
initial(clsys, [x0;xhat0],T);
pause

t=(0:ts:T)';
v=ones(size(t));
lsim(clsys,v);
```

Esempio MATLAB





Controllore PID digitale

PID - Parametri di base

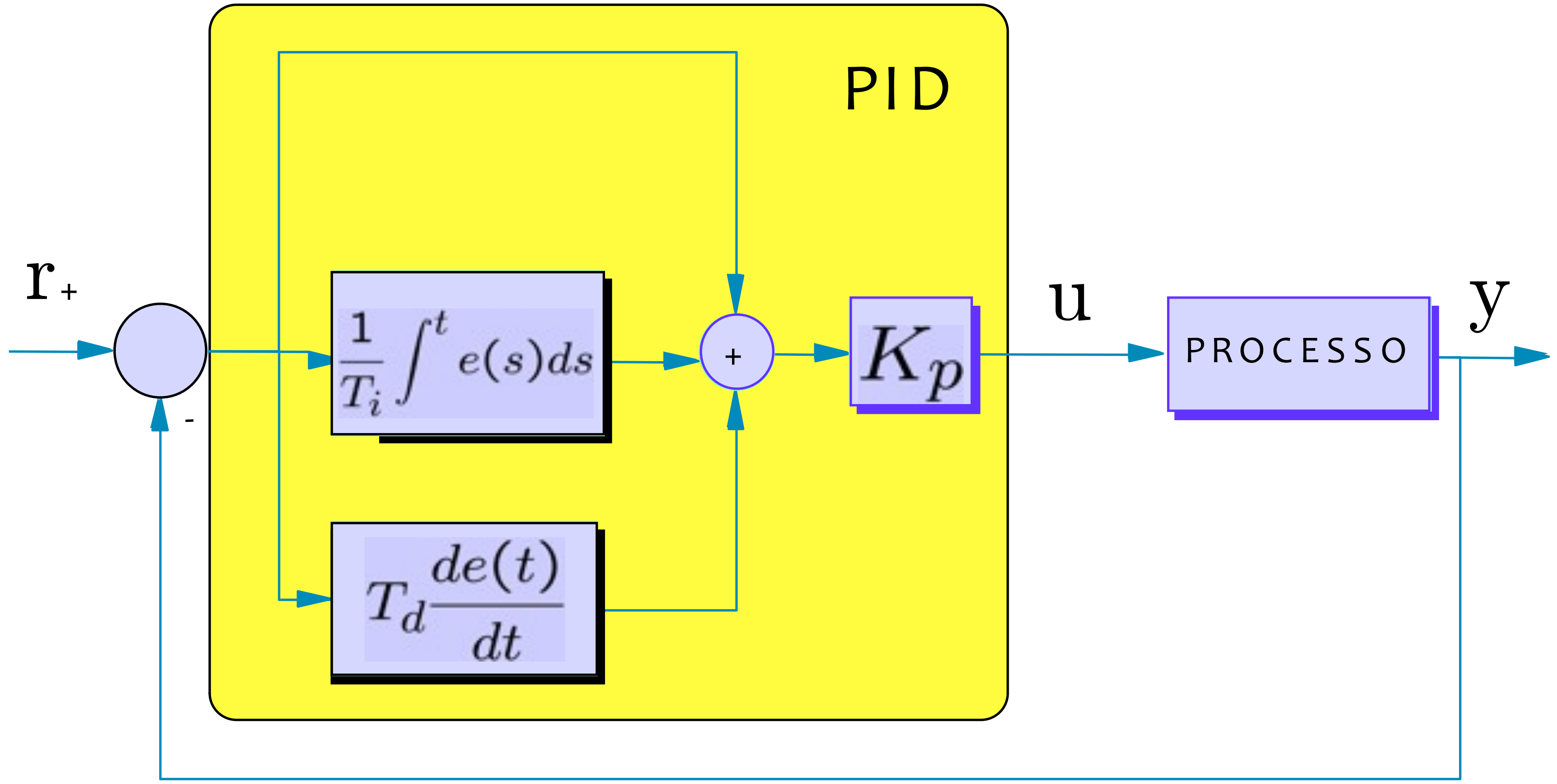
- Il controllore PID è tuttora la tecnica di controllo in retroazione (output feedback) più diffusa nelle applicazioni industriali

- A tempo continuo, il controllore PID si presenta nella forma

$$u(t) = K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{dt}\right]$$

  dove l'errore $e = r - y$ rappresenta la differenza tra il segnale di riferimento r (il set-point) e l'uscita del processo y (la variabile misurata e controllata) e

  - K_p rappresenta il guadagno del controllore, che determina l'aggressività del controllore stesso. È uno dei parametri di progetto.
  - T_i (*reset time*) è un parametro di progetto legato all'intensità dell'azione integrale.
  - T_d (*derivative time*) rappresenta invece il peso dell'azione derivatrice.



Azione P

$$u(t) = K_p \cdot e(t)$$

Azione PD

$$u(t) = K_p\left(e(t) + T_d\frac{de(t)}{dt}\right)$$

Azione PI

$$u(t) = K_p\left(e(t) + \frac{1}{T_i}\int^t e(s)ds\right)$$

Azione PID

$$u(t) = K_p\left[e(t) + \frac{1}{T_i}\int^t e(s)ds + T_d\frac{de(t)}{dt}\right]$$

PID industriale

Regolazione	Algoritmo	PID, PI, PD, P oppure On - Off
	Banda proporzionale (P)	0,5..1000%
	Tempo azione integrale (I)	0,1..100min., escludibile
	Tempo azione derivativa (D)	0,01..10min., escludibile
	Tempo del ciclo	1..200sec.
	Isteresi	0,1..10% (per regolazione on - off)
	Zona neutra	0..10% per regolazione a doppia azione (caldo-freddo)

L'operazione di discretizzazione può essere effettuata utilizzando le diverse metodologie viste (Eulero, Tustin, ecc.). Solitamente si usa la seguente tecnica:

- Parte proporzionale :

$$P(t) = K(br(t) - y(t))$$

Essendo una relazione di tipo statico, non richiede nessuna approssimazione.

- Parte integrale :

$$I(t) = \frac{K}{T_i} \int^t e(\tau)\, d\tau$$

viene approssimata mediante metodo di Eulero (approssimazione rettangolare)

$$I((k+1)T) = I(kT) + \frac{KT}{T_i} e(kT)$$

- Parte derivatrice :

$$\frac{T_d}{N}\frac{dD(t)}{dt} + D(t) = -KT_d\frac{dy(t)}{dt}$$

viene approssimata mediante la tecnica delle differenze all'indietro

$$D(kT) = \frac{T_d}{T_d + NT}D((k-1)T) - \frac{KT_dN}{T_d + NT}(y(kT) - y((k-1)T))$$

Nota: con questa approssimazione il polo discreto $z = \frac{T_d}{T_d+NT}$ è sempre all'interno del cerchio unitario.

Il segnale di controllo risulta quindi

$$u(kT) = P(kT) + I(kT) + D(kT)$$

Nota che questo tipo di approssimazione permette di calcolare separatamente le azioni proporzionale, derivatrice e integrale.

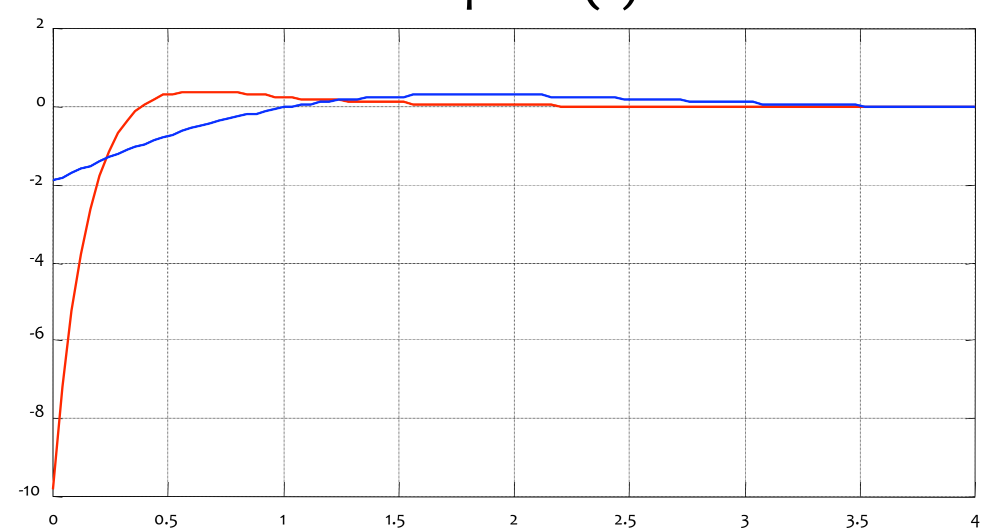
Controllo Ottimo Lineare Quadratico (LQR)

Problema LQR: Introduzione

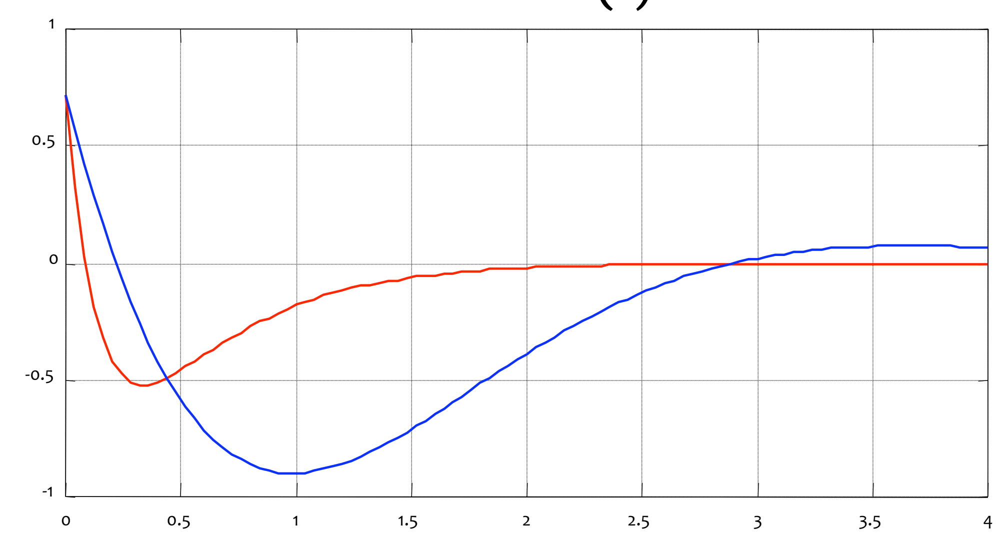
- Problema della scelta dei poli ad anello chiuso: dove posizionarli ?
- Obiettivi:
  - Rendere lo stato $x(k)$ “piccolo” (per regolarlo verso l’origine)
  - Utilizzare un ingresso $u(k)$ “piccolo” (per economizzare l’uso degli attuatori)

In generale sono obiettivi contrastanti !

input $u(t)$



state $x(t)$



- LQR: Tecnica che permette di piazzare i poli ad a.c. in maniera “ottima”

Controllo Ottimo LQ

- Controllo ottimo (su orizzonte temporale finito T):

Dato il sistema dinamico

$$x(k+1) = Ax(k) + Bu(k)$$

con condizione iniziale $x(0)$, si cerca la sequenza ottima di ingressi

$$U = \{u(0),\ u(1),\ \ldots,\ u(T-1)\}$$

che porta lo stato da $x(0)$ a verso l'origine e minimizza l'indice di prestazione

$$\boxed{J(x(0), U) = \sum_{k=0}^{T-1} x'(k)Qx(k) + u'(k)Ru(k) + x'(T)Q_T x(T)}$$

dove $Q = Q' \geq 0,\ R = R' > 0,\ Q_T = Q'_T \geq 0.$

Controllo Ottimo LQ

$$J(x(0), U) = \sum_{k=0}^{T-1} x'(k)Qx(k) + u'(k)Ru(k) + x'(T)Q_T x(T)$$

- T=orizzonte temporale (*time horizon*)
- Il primo termine misura la deviazione dello stato rispetto al valore desiderato $x = 0$
- Il secondo termine misura l'intensità dell' ingresso di controllo (*actuator authority*)
- Il terzo termine misura la deviazione dello stato finale rispetto allo 0
- Q, R, Q_T sono i parametri a disposizione del progettista (cfr. i parametri K_I, K_P, K_D del controllore PID), ed hanno un chiaro significato economico/fisico

Controllo Ottimo LQ

- Riconsidera il problema di controllabilità a zero dello stato con ingresso a energia minima

$$x(T) = 0, \qquad \min \left\| \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(T-1) \end{bmatrix} \right\|$$

- è un caso particolare di controllo ottimo LQ se poniamo:

  $R = I, \;\; Q = 0, \;\; Q_T = \infty I$ (in pratica: $Q_T = 10^8 I$, ad esempio)

Controllo ottimo: soluzione

$$J(x(0),U) = \sum_{k=0}^{T-1} x'(k)Qx(k) + u'(k)Ru(k) + x'(T)Q_T x(T)$$

- Sostituendo $x(k) = A^k x(0) + \sum_{i=0}^{k-1} A^i Bu(k-1-i)$ si ottiene

$$J(x(0),U) = \frac{1}{2}U'HU + x(0)'FU + \frac{1}{2}x(0)'Yx(0)$$

dove $H = H' > 0$ è una matrice definita positiva

- Il minimo lo si ottiene azzerando il gradiente: $\nabla_U J(x(0),U) = HU + F'x(0) = 0$ da cui ricaviamo

$$U^* = \begin{bmatrix} u^*(0) \\ u^*(1) \\ \vdots \\ u^*(T-1) \end{bmatrix} = -H^{-1}F'x(0)$$

Controllo ottimo: soluzione

Come ottenere H, F, Y:

$$J(x(0),U) = x'(0)Qx(0) + \begin{bmatrix} x'(1) & x'(2) & \dots & x'(T-1) & x'(T) \end{bmatrix} \overbrace{\begin{bmatrix} Q & 0 & 0 & \dots & 0 \\ 0 & Q & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & Q & 0 \\ 0 & 0 & \dots & 0 & Q_T \end{bmatrix}}^{\bar{Q}} \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(T-1) \\ x(T) \end{bmatrix} +$$

$$\begin{bmatrix} u'(0) & u'(1) & \dots & u'(T-1) \end{bmatrix} \underbrace{\begin{bmatrix} R & 0 & \dots & 0 \\ 0 & R & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & R \end{bmatrix}}_{\bar{R}} \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(T-1) \end{bmatrix}$$

$$\begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(T) \end{bmatrix} = \overbrace{\begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{T-1}B & A^{T-2}B & \dots & B \end{bmatrix}}^{\bar{S}} \begin{bmatrix} u(0) \\ u(1) \\ \dots \\ u(T-1) \end{bmatrix} + \underbrace{\begin{bmatrix} A \\ A^2 \\ \vdots \\ A^T \end{bmatrix}}_{\bar{T}} x(0)$$

Da cui:
$$\begin{aligned} J(x(0),U) &= x'(0)Qx(0) + (\bar{S}U + \bar{T}x(0))'\bar{Q}(\bar{S}U + \bar{T}x(0)) + U'\bar{R}U \\ &= \frac{1}{2}U' \underbrace{2(\bar{R} + \bar{S}'\bar{Q}\bar{S})}_{H} U + x'(0) \underbrace{2\bar{T}'\bar{Q}\bar{S}}_{F} U + \frac{1}{2}x'(0) \underbrace{2(Q + \bar{T}'\bar{Q}\bar{T})}_{Y} x(0) \end{aligned}$$

Controllo ottimo: soluzione

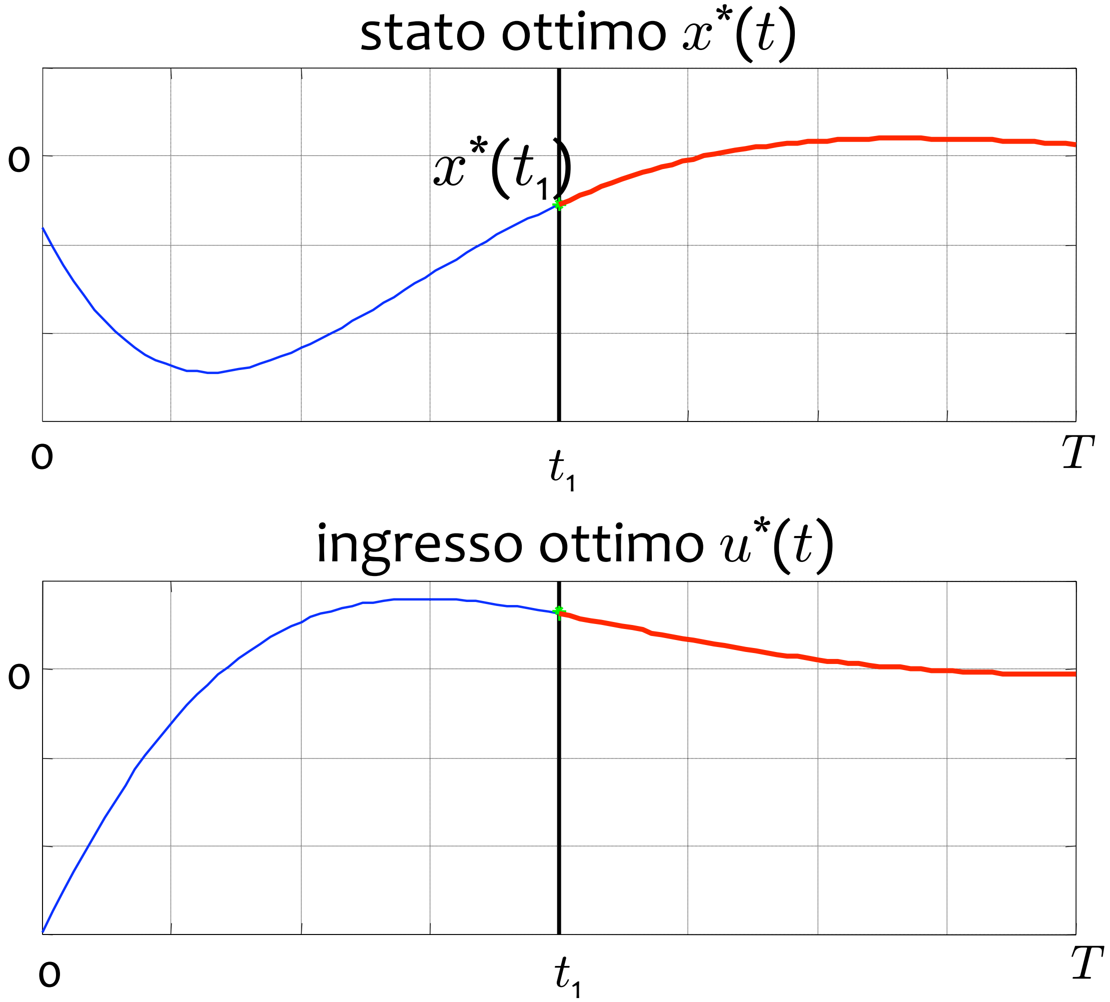
$$U^* = \begin{bmatrix} u^*(0) \\ u^*(1) \\ \vdots \\ u^*(T-1) \end{bmatrix} = -H^{-1}F'x(0)$$

- Problema: è una soluzione ad anello aperto
- Problema: la dimensione delle matrici H, F è proporzionale all'orizzonte temporale T
- Cerchiamo una soluzione migliore computazionalmente, più robusta (e più elegante) ...

Principio di Bellman

1. Principio di Bellman: data la sequenza ottima $U^* = [u^*(0), \dots, u^*(T-1)]$ (e la corrispondente traiettoria ottima $x^*(k)$), la sottosequenza $[u^*(t_1), \dots, u^*(T-1)]$ è ancora ottima per il problema su orizzonte $[t_1, T]$ a partire dallo stato ottimo $x^*(t_1)$

Richard Bellman (1920 - 1984)

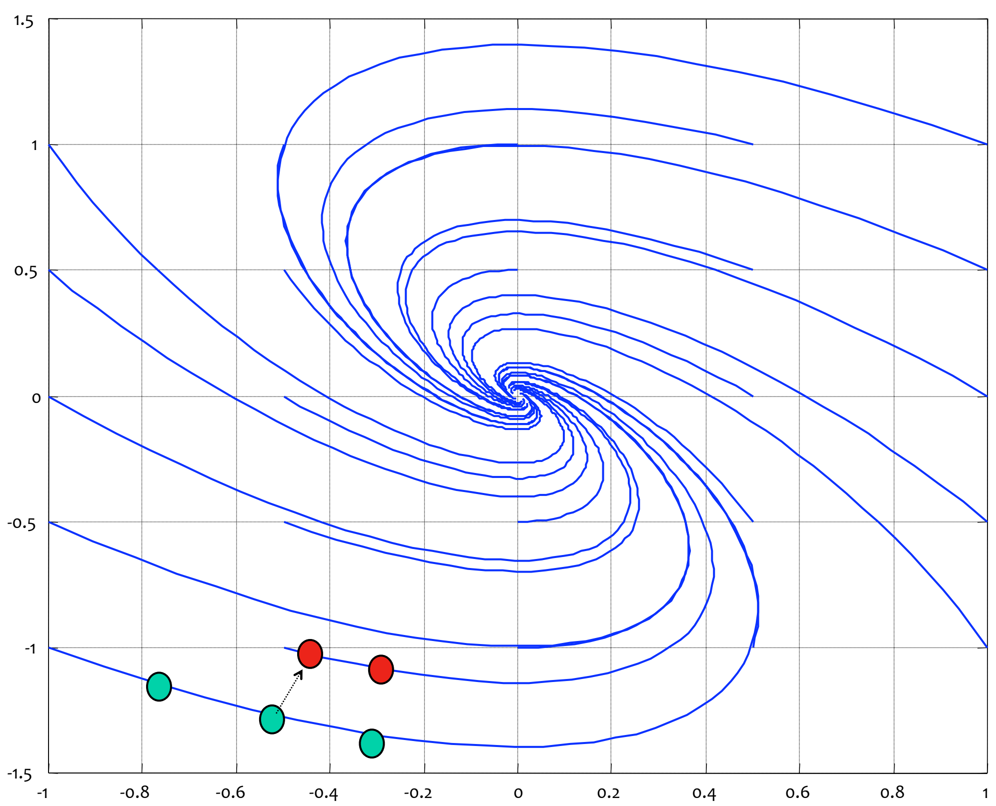


2. Inoltre la traiettoria ottima d'ingresso su un certo intervallo dipende unicamente dallo stato iniziale, in particolare la traiettoria ottima da t_1 a T dipende da $x^*(t_1)$

3. (1)+(2) implicano che ogni valore $u^*(t_1)$ della traiettoria ottima da 0 a T può essere espresso come funzione di $x^*(t_1)$, cioè in forma di retroazione dello stato (ottimo)

Principio di Bellman

traiettorie ottime



- Vale anche per sistemi non lineari e/o funzionali di costo non quadratici: **ogni legge di controllo ottimo può essere messa in forma di retroazione dello stato**:

$$u^*(t_1) = f_{t_1}(x^*(t_1)), \quad \forall t_1 = 0, \ldots, T-1$$

- Rispetto alla forma "ad anello aperto" $\{u^*(0), \ldots, u^*(T-1)\} = f(x(0))$ la forma di retroazione dello stato ha il vantaggio di essere più **robusta** rispetto alle perturbazioni (ad ogni istante si applica sempre la mossa ottima sul restante intervallo di tempo per la situazione in cui si trova il sistema)

Programmazione dinamica

- Per un generico istante t_1 e stato $x(t_1) = z$ consideriamo

$$V_{t_1}(z) = \min_{u(t_1),\ldots,u(T-1)} \left\{ \sum_{k=t_1}^{T-1} x'(k)Qx(k) + u'(k)Ru(k) + x'(T)Q_T x(T) \right\}$$

dove la funzione minimizzanda è detta *cost-to-go*, cioè il costo residuo sull'intervallo $[t_1, T]$ partendo dallo stato $x(t_1) = z$.

- $V_0(z)$ = costo minimo LQ partendo da condizione iniziale $x(0) = z$

- Principio della programmazione dinamica (DP, *dynamic programming*)

$$\begin{aligned} V_0(z) &= \min_{U \triangleq \{u(0),\ldots,u(T-1)\}} J(x(0), U) \\ &= \min_{u(0),\ldots,u(t_1-1)} \left\{ \sum_{k=0}^{t_1} x'(k)Qx(k) + u'(k)Ru(k) + V_{t_1}(x(t_1)) \right\} \end{aligned}$$

- In parole povere: il minimo del costo tra 0 e T a partire dallo stato $x(0)$ è uguale al minimo del (costo speso fino al passo t_1 + minimo del cost-to-go tra t_1 e T a partire dallo stato $x(t_1)$)

Soluzione LQ mediante progr. dinamica

- Partiamo dall'istante finale T: per un generico $x(T)$

$$V_T(x(T)) = x'(T)\ \underbrace{Q_T}_{P(T)}\ x(T) \quad \text{(non dipende da nessun ingresso!)}$$

- All'istante $T-1$: per un generico $x(T-1)$

$$\begin{aligned}
V_{T-1}(x(T-1)) &= \min_{u(T-1)} \left\{x'(T-1)Qx(T-1) + u'(T-1)Ru(T-1) + x(T)'Q_T x(T)\right\} \\
&= x'(T-1)Qx(T-1) + \min_{u(T-1)} \left\{u'(T-1)Ru(T-1) + \right. \\
&\quad \left.(Ax(T-1) + Bu(T-1))'Q_T(Ax(T-1) + Bu(T-1))\right\} \\
&= x'(T-1)(A'Q_TA + Q)x(T-1) + \min_{u(T-1)} \left\{u'(T-1)(R + \right. \\
&\quad \left. B'Q_TB)u(T-1) + 2x'(T-1)A'Q_TBu(T-1))\right\}
\end{aligned}$$

da cui ricaviamo l'ingresso ottimo:

$$u^*(T-1) = -(R + B'Q_TB)^{-1}B'Q_TAx(T-1)$$

e quindi, sostituendo,

$$V_{T-1}(x(T-1)) = \ldots = x'(T-1)\underbrace{\left[Q - A'Q_TB(R + B'Q_TB)^{-1}B'Q_TA + A'Q_TA\right]}_{P(T-1)} x(T-1)$$

Soluzione LQ mediante programm. dinamica

- All'istante $T-2$: per un generico $x(T-2)$

$$\begin{aligned} V_{T-2}(x(T-2)) &= \min_{u(T-2)} \left\{x'(T-2)Qx(T-2) + u'(T-2)Ru(T-2) + V_{T-1}(x(T-1))\right\} \\ &= x'(T-2)Qx(T-2) + \min_{u(T-2)} \left\{u'(T-2)Ru(T-2) + \right. \\ &\quad \left. x(T-1)'P(T-1)x(T-1)\right\} \end{aligned}$$

  ha la stessa forma del problema al passo $T-1$!!!

- l'ingresso ottimo $u^*(T-2)$ è pertanto

$$u^*(T-2) = -(R + B'P(T-1)B)B'P(T-1)Ax(T-2)$$

  e quindi, sostituendo,

$$V_{T-2}(x(T-2)) = x'(T-2)P(T-2)x(T-2)$$

  dove

$$P(T-2) = Q - A'P(T-1)B(R + B'P(T-1)B)^{-1}B'P(T-1)A + A'P(T-1)A$$

Iterazioni di Riccati

- Iterazioni di Riccati:

  1. Inizializza $P(T) = Q_T$

  2. Per $j = T, \ldots, 1$:

  $$P(j-1) = Q - A'P(j)B(R + B'P(j)B)^{-1}B'P(j)A + A'P(j)$$

  3. Definisci

  $$K(j) = -(R + B'P(j+1)B)^{-1}B'P(j+1)$$

  4. L'ingresso ottimo

  $$u^*(j) = K(j)x(j)$$

Jacopo Francesco Riccati (1676 - 1754)

- Nota: l'ingresso ottimo è calcolato in forma di retroazione dello stato !

LQR (orizzonte infinito)

- Per processi che operano su un orizzonte temporale molto lungo, un orizzonte temporale finito T non è sufficiente

- Mandiamo $T \to \infty$:

$$V^\infty(x(0)) = \min_{u(0),u(1),\ldots} \sum_{k=0}^{\infty} x'(k)Qx(k) + u'(k)Ru(k)$$

- Risultato: se (A,B) è stabilizzabile, esiste ed è unica la soluzione P_∞ dell' *equazione algebrica di Riccati* (ARE)

$$P_\infty = A'P_\infty A + Q - A'P_\infty B(B^T P_\infty B + R)^{-1} B' P_\infty A$$

- Nota: il costo ottimo su orizzonte infinito è $V^\infty(x(0)) = x'(0)P_\infty x(0)$

LQR (orizzonte infinito)

- Ritorniamo alle iterazioni di Riccati: partendo da $P(\infty) = P_\infty$ ed andando all'indietro, otteniamo $P(j) = P_\infty \ \forall j \geq 0$
- Di conseguenza:

$$K(j) = -(R + B'P_\infty B)^{-1}B'P_\infty A \triangleq K_{LQ}, \quad \forall j = 0, 1, \ldots$$

- La legge di controllo LQR è lineare e non dipende dall'indice temporale j
- in Matlab:

```
[KLQ,P∞,E] = -DLQR(A,B,Q,R)
```

  dove E=modi del sistema ad anello chiuso (cioè autovalori di $(A + BK_{LQ})$)

- È un metodo universale (e ottimo) di piazzare i poli !
- Sistemi lineari a tempo continuo: vale un risultato analogo (in Matlab: `LQR`)

LQR - Peso sull'uscita

- Spesso ci interessa pesare solo le uscite: $y'(k)Q_y y(k)$

- Equivale a porre $Q = C'Q_y C$

- Vale il seguente risultato: Sia (A, B) stabilizzabile, (A, C) rivelabile e $Q_y > 0$ (in generale: $Q \geq 0$, $(A, Q^{\frac{1}{2}})$ rivelabile, dove $Q = Q^{\frac{1}{2}\prime}Q^{\frac{1}{2}}$). Allora l'anello chiuso sotto la regolazione $u(k) = K_{LQ}x(k)$ è asintoticamente stabile:

$$\lim_{t\to\infty} x(t) = 0, \quad \lim_{t\to\infty} u(t) = 0$$

- Spiegazione intuitiva: solo la parte osservabile influisce sul costo, e quindi deve necessariamente andare a zero perchè il costo minimo sia finito. La parte non osservabile invece non ha influenza, e può pertanto non convergere a zero.

LQR: Esempio

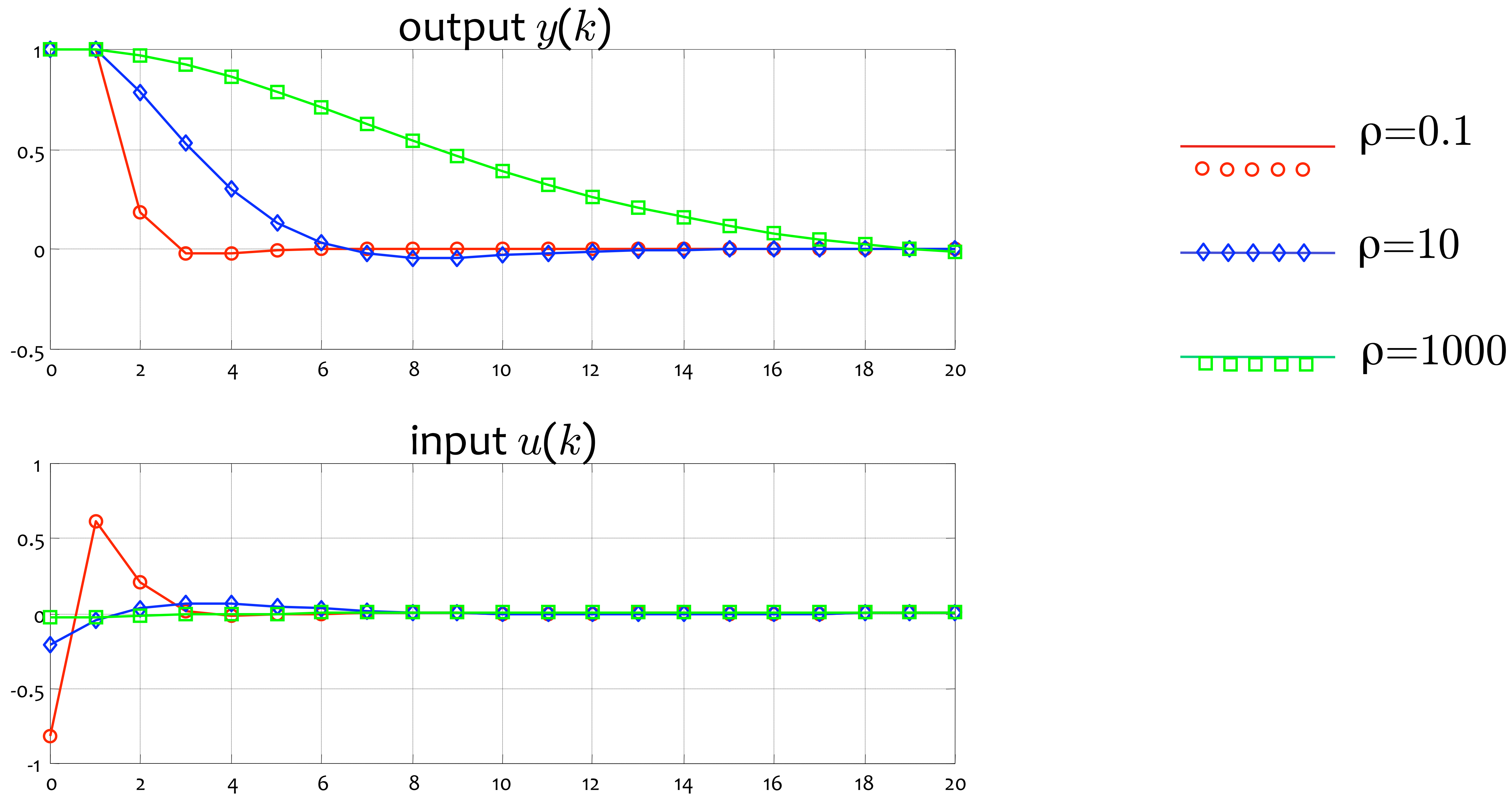
- Sistema a due stati, singolo ingresso singola uscita (SISO) (doppio integratore)

$$
\begin{aligned}
x(k+1) &= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \\
y(k) &= \begin{bmatrix} 1 & 0 \end{bmatrix} x(k)
\end{aligned}
$$

- LQR (orizzonte infinito)

$$
V^{\infty}(x(0)) = \min_{u(0),u(1),\ldots} \sum_{k=0}^{\infty} y^2(k) + \rho u^2(k)
$$

- Pesi: $Q = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] \cdot 1 \cdot [1\ 0] = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right]$, $R = \rho > 0$



Stato iniziale: $x(0) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$$V^\infty(x(0)) = \min_{u(0),u(1),\ldots} \sum_{k=0}^{\infty} y^2(k) + \rho u^2(k)$$