



**UNIVERSITÀ
DEGLI STUDI
DI BERGAMO**

Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Corso di laurea magistrale in Ingegneria Informatica

CORSO DI CONTROLLO AVANZATO MULTIVARIABILE

CONTROLLO DI UN SERVOMECCANISMO

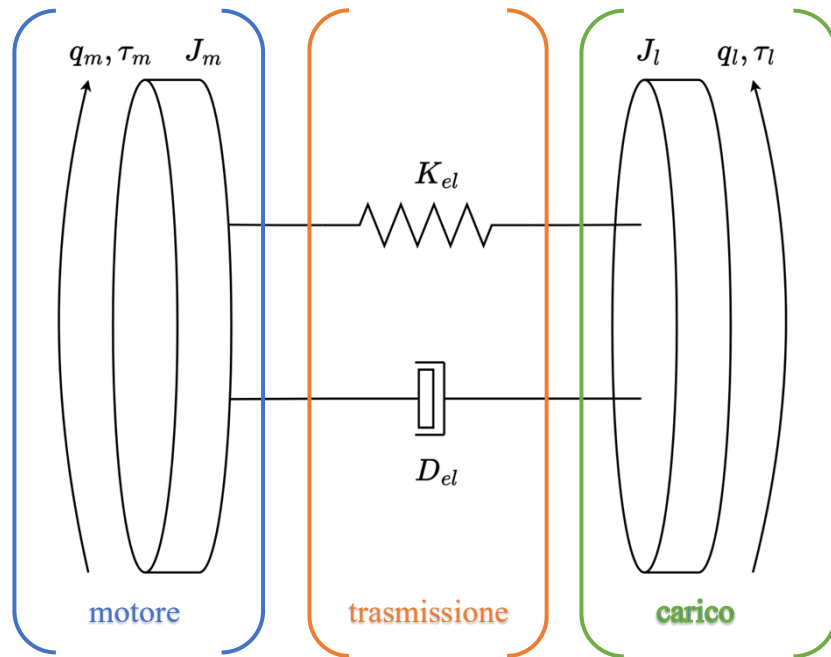
Fabio Filippo Mandalari
Matricola: 1047426

A.A. 2022/2023

DESCRIZIONE DEL PROBLEMA

Un servomeccanismo è un sistema che trasforma potenza elettrica in potenza meccanica in modo controllato. Ai fini della modellistica, si può considerare un servomeccanismo come se fosse composto dai seguenti elementi:

- *Motore*. Il motore, per effetto della corrente, genera una coppia meccanica che mette in rotazione l'albero del motore;
- *Trasmissione*. La trasmissione accoppia l'albero del motore all'albero del carico mettendo in rotazione quest'ultimo;
- *Carico*. La rotazione dell'albero del carico mette in rotazione il carico.



OBIETTIVO

L'obiettivo è quello di progettare un regolatore MPC che sia in grado di portare il sistema dalla condizione iniziale $(q_m; q_l; \dot{q}_m; \dot{q}_l) = (0; 0; 0; 0)$ all'equilibrio $(q_m; q_l; \dot{q}_m; \dot{q}_l) = (425,0286; 1,7; 0; 0)$ rispettando i vincoli.

Col fine di valutare le prestazioni del controllore progettato, bisogna effettuare delle simulazioni del sistema in anello chiuso al variare dei seguenti parametri:

- Ingredienti terminali (vincolo terminale di uguaglianza vs costo terminale e vincolo terminale di disuguaglianza);
- Q e R del costo quadratico;
- Orizzonte di predizione N ;
- Tempo di campionamento $T_s \geq 0,1s$.

MODELLISTICA

Per modellistica si intende la scrittura dei modelli matematici che consentono di effettuare il controllo.

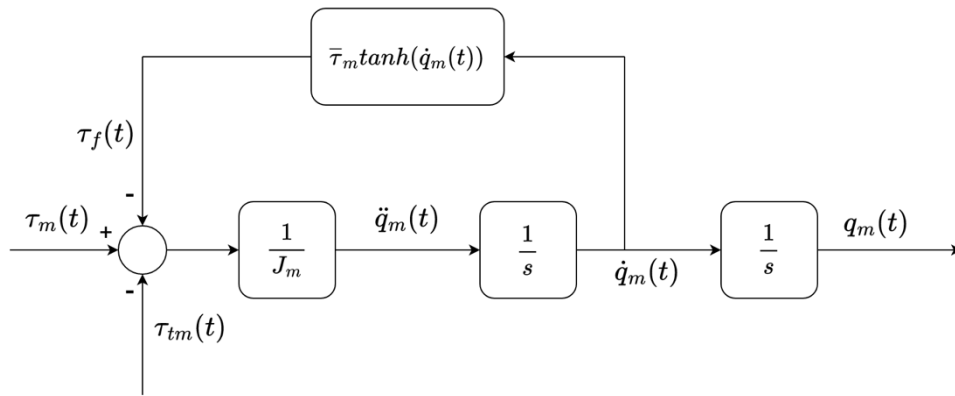
MODELLO DEL MOTORE

La dinamica del servomeccanismo è descritta dalla seguente equazione:

$$J_m \ddot{q}_m(t) = \tau_m(t) - \tau_f(t) - \tau_{tm}(t)$$

$$s^2 J_m q_m(t) = \tau_m(t) - \tau_f(t) - \tau_{tm}(t)$$

$$q_m(t) = \frac{1}{s^2 J_m} [\tau_m(t) - \tau_f(t) - \tau_{tm}(t)]$$



MODELLO DELLA TRASMISSIONE

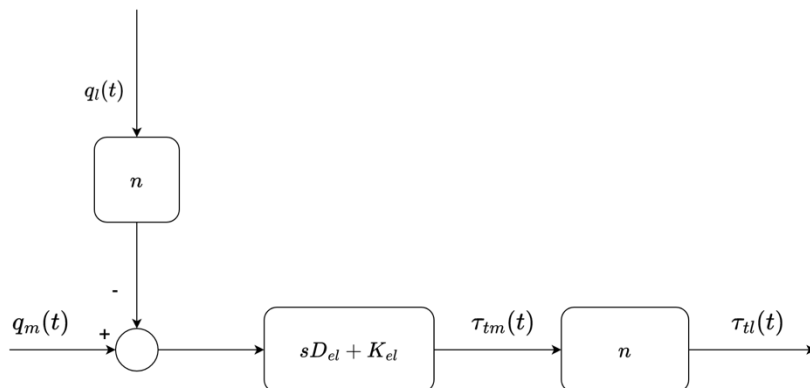
La dinamica della trasmissione è descritta dalla seguente equazione:

$$\tau_{tm}(t) = K_{el}[q_m(t) - nq_l(t)] + D_{el}[\dot{q}_m(t) - n\dot{q}_l(t)]$$

$$\tau_{tl}(t) = n\tau_{tm}(t) = n[K_{el}[q_m(t) - nq_l(t)] + D_{el}[\dot{q}_m(t) - n\dot{q}_l(t)]] =$$

$$= n[K_{el}[q_m(t) - nq_l(t)] + sD_{el}[q_m(t) - nq_l(t)]] =$$

$$= n[(q_m(t) - q_l(t))(sD_{el} + K_{el})]$$



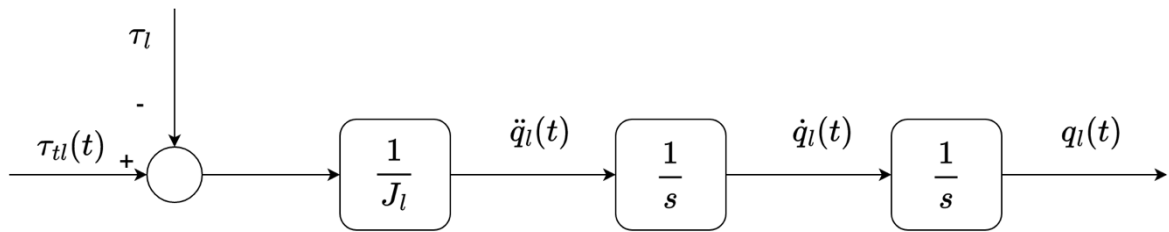
MODELLO DEL CARICO

La dinamica del carico è descritta dalla seguente equazione:

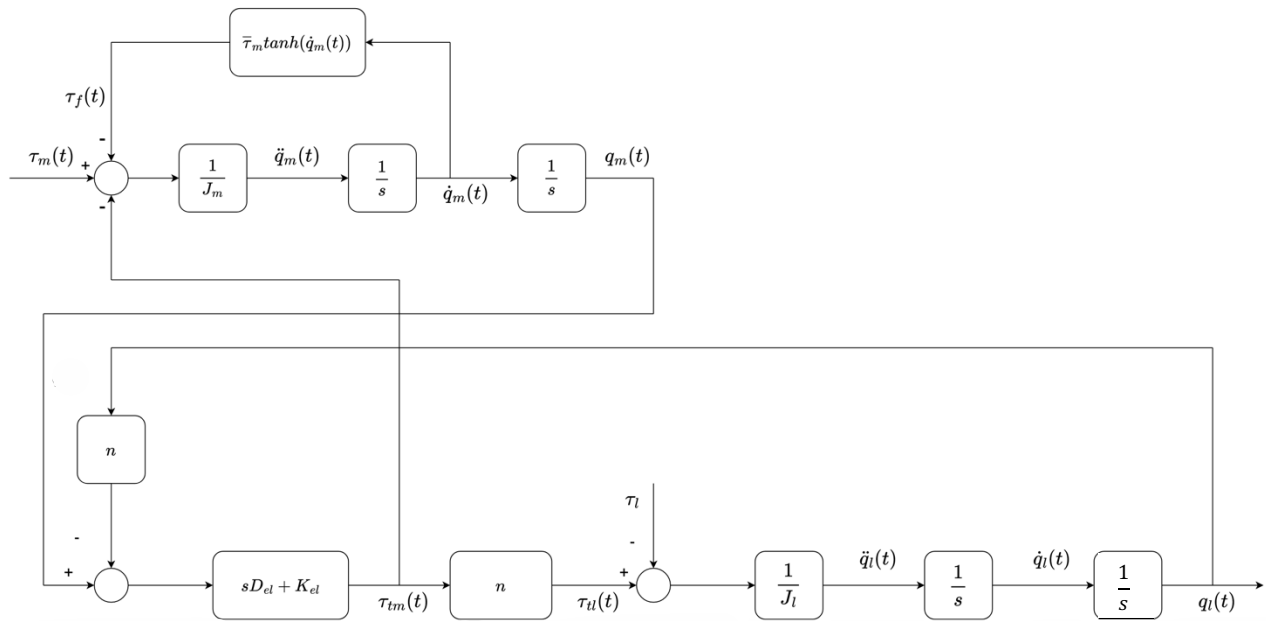
$$J_l \ddot{q}_l(t) = \tau_{tl}(t) - \tau_l$$

$$s^2 J_l q_l(t) = \tau_{tl}(t) - \tau_l$$

$$q_l(t) = \frac{1}{s^2 J_l} [\tau_{tl}(t) - \tau_l]$$



MODELLO COMPLESSIVO



MODELLO IN VARIABILI DI STATO

Per poter elaborare il modello in variabili di stato bisogna mettere a sistema tutte le equazioni riportate nel capitolo precedente:

$$\begin{cases} J_m \ddot{q}_m(t) = \tau_m(t) - \tau_{tm}(t) - \tau_f(t) \\ \tau_{tm}(t) = K_{el}[q_m(t) - nq_l(t)] + D_{el}[\dot{q}_m(t) - n\dot{q}_l(t)] \\ J_l(t) \ddot{q}_l(t) = \tau_{tl}(t) - \tau_l \\ \tau_{tl}(t) = n\tau_{tm}(t) \end{cases}$$

Con l'obiettivo di isolare le quantità $\ddot{q}_m(t)$ e $\ddot{q}_l(t)$, effettuo delle semplici operazioni algebriche:

$$\begin{cases} \ddot{q}_m(t) = -\frac{K_{el}}{J_m}q_m(t) + \frac{nK_{el}}{J_m}q_l(t) - \frac{D_{el}}{J_m}\dot{q}_m(t) - \frac{\bar{\tau}_m}{J_m}\tanh[\dot{q}_m(t)] + \frac{nD_{el}}{J_m}\dot{q}_l(t) + \frac{1}{J_m}\tau_m(t) \\ \ddot{q}_l(t) = \frac{nK_{el}}{J_l}q_m(t) - \frac{n^2K_{el}}{J_l}q_l(t) + \frac{nD_{el}}{J_l}\dot{q}_m(t) - \frac{n^2D_{el}}{J_l}\dot{q}_l(t) - \frac{\tau_l}{J_l} \end{cases}$$

Effettuo un cambiamento di variabili del sistema in variabili di stato sfruttando la seguente notazione:

Variabile del sistema	Variabile di stato	Significato
$q_m(t)$	$x_1(t)$	Posizione angolare del motore
$q_l(t)$	$x_2(t)$	Posizione angolare del carico
$\dot{q}_m(t)$	$x_3(t)$	Velocità angolare del motore
$\dot{q}_l(t)$	$x_4(t)$	Velocità angolare del carico
$\tau_m(t)$	$u_1(t)$	Coppia generata dal motore
τ_l	u_2	Coppia di carico

Si ottiene così il modello in variabili di stato:

$$\begin{cases} \dot{x}_1(t) = x_3(t) \\ \dot{x}_2(t) = x_4(t) \\ \dot{x}_3(t) = -\frac{K_{el}}{J_m}x_1(t) + \frac{nK_{el}}{J_m}x_2(t) - \frac{D_{el}}{J_m}x_3(t) - \frac{\bar{\tau}_m}{J_m}\tanh[x_3(t)] + \frac{nD_{el}}{J_m}x_4(t) + \frac{1}{J_m}u_1(t) \\ \dot{x}_4(t) = \frac{nK_{el}}{J_l}x_1(t) - \frac{n^2K_{el}}{J_l}x_2(t) + \frac{nD_{el}}{J_l}x_3(t) - \frac{n^2D_{el}}{J_l}x_4(t) - \frac{1}{J_l}u_2 \end{cases}$$

I valori dei parametri sono forniti dalla traccia del progetto:

Parametro	Valore	Significato
J_m	0,001	Momento di inerzia del motore
J_l	9	Momento di inerzia del carico
K_{el}	0,7	Costante elastica della trasmissione
D_{el}	0,03	Costante viscosa della trasmissione
$\bar{\tau}_f$	0,2	Costante numerica
n	250	Rapporto di riduzione del motore
τ_l	5	Coppia di carico

LINEARIZZAZIONE

Il sistema è non lineare a causa della presenza del termine $\tanh[x_3(t)]$ nell'espressione di $\dot{x}_3(t)$, dunque, per poter ricavare le matrici A , B , C e D , bisogna preventivamente linearizzarlo. La linearizzazione, effettuata con l'ausilio di Matlab, porta ad ottenere il seguente sistema:

$$\begin{cases} 0 = \bar{x}_3 \\ 0 = \bar{x}_4 \\ 0 = -700\bar{x}_1 + 175.000\bar{x}_2 - 30\bar{x}_3 + (200 \tanh(\bar{x}_3)^2 - 230)\bar{x}_3 + 7.500\bar{x}_4 + 1.000\bar{u}_1 \\ 0 = \frac{175}{9}\bar{x}_1 - \frac{43.750}{9}\bar{x}_2 + \frac{5}{6}\bar{x}_3 - \frac{625}{3}\bar{x}_4 - \frac{1}{9}\bar{u}_2 \end{cases}$$

Sapendo che il punto di equilibrio intorno al quale linearizzare il sistema è fornito dalla traccia del progetto e vale $x_{eq} = (425,0286; 1,7; 0; 0)$, il sistema linearizzato deve essere valutato in corrispondenza di x_{eq} .

RAPPRESENTAZIONE MATRICIALE

A partire dalla linearizzazione ottenuta nel paragrafo precedente e dalla relativa valutazione in x_{eq} è possibile estrapolare le seguenti matrici:

$$A_{eq} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -700 & 175.000 & -230 & 7.500 \\ \frac{175}{9} & -\frac{43.750}{9} & \frac{5}{6} & -\frac{625}{3} \end{bmatrix} \quad B_{eq} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1.000 & 0 & 0 & 0 \\ 0 & -\frac{1}{9} & 0 & 0 \end{bmatrix}$$

$$C_{eq} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad D_{eq} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

INGRESSI DI EQUILIBRIO

Per poter effettuare il calcolo degli ingressi di equilibrio u_{eq} , ovvero di quegli ingressi che, se applicati al sistema, permettono a quest'ultimo di raggiungere il punto di equilibrio x_{eq} , si rende necessario effettuare i seguenti passaggi:

1. In corrispondenza di $u(k) = \bar{u}$ bisogna imporre $x(k+1) = x(k) = \bar{x}$, da cui:

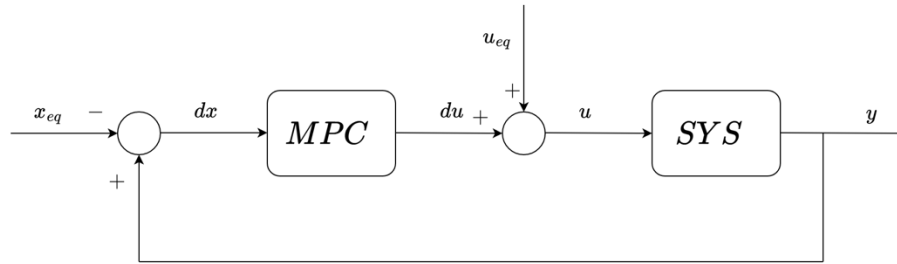
$$\bar{x} = A_{eq}\bar{x} + B_{eq}\bar{u} \Rightarrow (I - A_{eq})\bar{x} = B_{eq}\bar{u}$$

2. Sapendo che $x_{eq} = \bar{x}$ e che $u_{eq} = \bar{u}$, è possibile isolare u_{eq} :

$$u_{eq} = B_{eq}^{-1}(I - A_{eq})x_{eq}$$

DESIGN MPC

Schema di riferimento:



L'obiettivo è quello di progettare un MPC, ovvero un controllore che consente di spostare il sistema *SYS* dalla condizione iniziale al punto di equilibrio x_{eq} . La progettazione di un MPC che funzioni correttamente prevede la risoluzione di un problema di programmazione quadratica del tipo:

$$\min_{\mathbf{u}} \frac{1}{2} \mathbf{u}^T \mathbf{H} \mathbf{u} + \mathbf{f}^T \mathbf{u}$$

$$s. t. \mathbf{A} \mathbf{u} \leq \mathbf{b}$$

Si rende necessario progettare uno ad uno i seguenti componenti: \mathbf{H} , \mathbf{f} , \mathbf{A} , \mathbf{b} .

PROGETTAZIONE DI H E f

Nella progettazione dei componenti H e f bisogna partire dalla definizione della funzione di costo che tipicamente viene scelta per effettuare l'ottimizzazione (minimizzazione):

$$J(x(k), x(\cdot)) = \sum_{j=0}^{N-1} l(x(j), u(j)) + V_f(x(N))$$

$$l(x(j), u(j)) = \|x(j) - x_r\|_Q^2 + \|u(j) - u_r\|_R^2$$

$$V_f(x(N)) = \|x(N) - x_r\|_P^2$$

Le matrici coinvolte nelle precedenti espressioni sono:

- Q = matrice che pesa la priorità dello stato rispetto all'ingresso;
- R = matrice che pesa la priorità dell'ingresso rispetto allo stato;
- P = soluzione dell'equazione di Riccati.

Ad ogni passo viene risolto un problema di ottimizzazione basato su N passi di predizione e, dato che l'espressione della funzione di costo contempla anche un costo aggiuntivo basato sullo stato finale in cui si deve venire a trovare il sistema, bisogna costruire delle matrici \tilde{Q} e \tilde{R} impostandole così:

$$\tilde{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 & 0 \\ 0 & Q & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & Q & 0 \\ 0 & \cdots & \cdots & 0 & P \end{bmatrix} \in \mathcal{M}((N+1) * \#stati, (N+1) * \#stati)$$

$$\tilde{R} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & R \end{bmatrix} \in \mathcal{M}(N * \#ingressi, N * \#ingressi)$$

Dunque, la funzione di costo da analizzare per poter estrarre le espressioni di H e f è:

$$(x_F - x_r)' \tilde{Q} (x_F - x_r) + (u_F - u_r)' \tilde{R} (u_F - u_r)$$

In cui bisogna riconoscere che:

$$x_F = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N) \end{bmatrix} \quad u_F = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix}$$

La precedente espressione deve necessariamente essere espressa in funzione della sola variabile di ottimizzazione u_F , perciò bisogna effettuare la seguente sostituzione:

$$x_F = \mathcal{A}x(0) + \mathcal{B}u_F$$

Per proseguire con la trattazione relativa alla funzione di costo occorre definire \mathcal{A} , \mathcal{B} e $x(0)$:

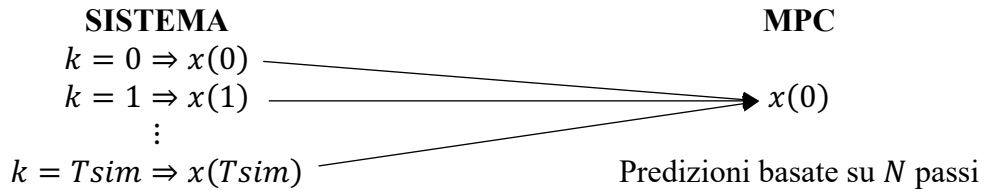
- La matrice \mathcal{A} contiene tutti i termini necessari per il calcolo del movimento libero del sistema. La dimensionalità di \mathcal{A} è determinata dal fatto che vengono risolti N problemi di ottimizzazione. A questo si aggiunge il passo di inizializzazione del primo elemento all'identità.

$$\mathcal{A} = \begin{bmatrix} I_n \\ A \\ \vdots \\ A^N \end{bmatrix} \in \mathcal{M}((N+1) * \#stati, \#stati)$$

- La matrice \mathcal{B} contiene tutti i termini necessari per il calcolo del movimento forzato del sistema. La dimensionalità di \mathcal{B} è determinata dal fatto che vengono risolti N problemi di ottimizzazione e ad ogni passo viene calcolata una sequenza di azioni di controllo (delle quali solo la prima viene applicata per muovere il sistema). Si evince che \mathcal{B} debba avere la seguente struttura:

$$\mathcal{B} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ B & 0 & 0 & \cdots & 0 \\ AB & B & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & A^{N-3}B & \cdots & B \end{bmatrix} \in \mathcal{M}((N+1) * \#ingressi, N * \#uscite)$$

- $x(0)$ è il feedback di stato, ovvero la condizione iniziale da cui parte l'MPC per effettuare le predizioni ad ogni passo del problema di ottimizzazione.



In virtù del fatto che l'MPC ragiona in termini di “variazioni” di ingressi e uscite, alla quantità $x(0)$ deve essere sottratto ad ogni iterazione il valore dello stato di equilibrio. Da ciò ne segue che l'ingresso effettivo dell'MPC ad ogni iterazione è $dx = x(0) - x_{eq}$.

La funzione di costo da studiare, quindi, è:

$$(\mathcal{A}x(0) + B\mathbf{u}_F - \mathbf{x}_r)' \tilde{Q} (\mathcal{A}x(0) + B\mathbf{u}_F - \mathbf{x}_r) + (\mathbf{u}_F - \mathbf{u}_r)' \tilde{R} (\mathbf{u}_F - \mathbf{u}_r)$$

Svolgendo tutti i conti si giunge alla seguente espressione:

$$\mathbf{u}_F' (\tilde{R} + B' \tilde{Q} B) \mathbf{u}_F + (2x(0)' \mathcal{A}' \tilde{Q} B - 2\mathbf{x}_r' \tilde{Q} B - 2\mathbf{u}_r' \tilde{R}) \mathbf{u}_F$$

A questo punto è immediato estrarre le matrici H e \mathbf{f} :

$$H = \tilde{R} + B' \tilde{Q} B$$

$$\mathbf{f} = 2x(0)' \mathcal{A}' \tilde{Q} B - 2\mathbf{x}_r' \tilde{Q} B - 2\mathbf{u}_r' \tilde{R}$$

Nel codice:

$$f1_{cost} = 2\mathcal{A}' \tilde{Q} B \Rightarrow f1 = x(0)' f1_{cost}$$

$$f_2 = -2\mathbf{x}_r' \tilde{Q} B - 2\mathbf{u}_r' \tilde{R}$$

PROGETTAZIONE DI A E b

Per giungere alla trattazione delle matrici \tilde{H}_u e \tilde{H}_x bisogna introdurre le matrici H_u e H_x .

Il punto della situazione è che, nel progettare un controllore MPC che sia efficace e quanto più possibile efficiente, bisogna fare in modo che nell'arco di tutte le N iterazioni siano soddisfatti dei vincoli. I vincoli limitano la libertà delle variabili in gioco di assumere qualunque valore e possono essere definiti sia sugli stati sia sugli ingressi. La matrice H_x raccoglie tutte le informazioni relative ai vincoli imposti sugli stati, mentre H_u tutte quelle relative ai vincoli imposti sugli ingressi.

Fatte le dovute premesse, ricordando che il problema di ottimizzazione viene risolto N volte e che, per quanto riguarda gli stati, bisogna calcolare anche un costo aggiuntivo basato sullo stato finale in cui si deve venire a trovare il sistema, bisogna specificare un ulteriore aspetto: da un punto di vista pratico, in virtù del fatto che il problema di ottimizzazione deve essere formulato sfruttando vincoli della forma $Au \leq b$, la matrici H_x e H_u devono essere realizzate tenendo da conto di questo aspetto.

$$H_u u(i) \leq h_u, i \in [0, N-1]$$

$$\tilde{H}_u = \begin{bmatrix} H_u & 0 & \cdots & 0 \\ 0 & H_u & \ddots & \vdots \\ \vdots & \ddots & H_u & 0 \\ 0 & \cdots & 0 & H_u \end{bmatrix} \quad \tilde{h}_u = \begin{bmatrix} h_u \\ h_u \\ \vdots \\ h_u \end{bmatrix}$$

$$H_x x(i) \leq h_x, i \in [0, N]$$

$$\tilde{H}_x = \begin{bmatrix} H_x & 0 & \cdots & 0 \\ 0 & H_x & \ddots & \vdots \\ \vdots & \ddots & H_x & 0 \\ 0 & \cdots & 0 & H_x \end{bmatrix} \quad \tilde{h}_x = \begin{bmatrix} h_x \\ h_x \\ \vdots \\ h_x \end{bmatrix}$$

I vincoli che devono essere imposti, dunque, sono:

$$\tilde{H}_u \mathbf{u}_F \leq \tilde{h}_u$$

$$\tilde{H}_x \mathbf{x}_F \leq \tilde{h}_x$$

Al pari di quanto fatto nel contesto della progettazione di H e f , anche in questo caso, sfruttando la medesima trasformazione $\mathbf{x}_F = \mathcal{A}x(0) + B\mathbf{u}_F$ devo esprimere tutte le condizioni in funzione di \mathbf{u}_F :

$$\tilde{H}_x(\mathcal{A}x(0) + B\mathbf{u}_F) \leq \tilde{h}_x$$

$$\tilde{H}_x B\mathbf{u}_F \leq \tilde{h}_x - \tilde{H}_x \mathcal{A}x(0)$$

Mettendo tutti insieme si ottiene:

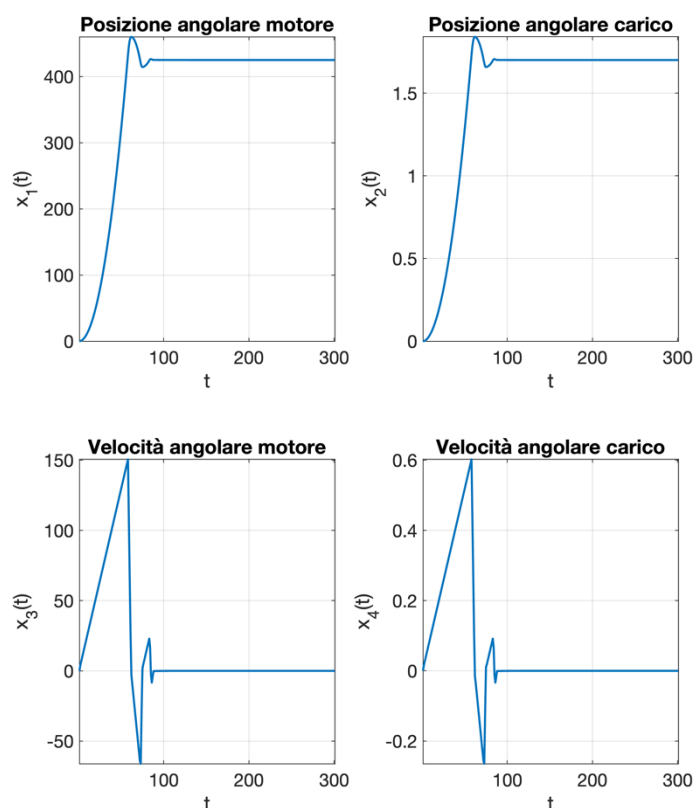
$$\begin{bmatrix} \tilde{H}_u \\ \tilde{H}_x B \end{bmatrix} \mathbf{u}_F \leq \begin{bmatrix} \tilde{h}_u \\ \tilde{h}_x - \tilde{H}_x \mathcal{A}x(0) \end{bmatrix}$$

A questo punto è immediato estrarre le matrici A e b :

$$A = \begin{bmatrix} \tilde{H}_u \\ \tilde{H}_x B \end{bmatrix} \quad b = \begin{bmatrix} \tilde{h}_u \\ \tilde{h}_x - \tilde{H}_x \mathcal{A}x(0) \end{bmatrix}$$

SIMULAZIONE

COMPORTAMENTO DEGLI STATI CON $Q = R = I$, $T_s = 0,1$ e $N = 10$



Analizzando sia i grafici relativi agli stati $x_1(t)$ e $x_2(t)$ sia quelli relativi agli stati $x_3(t)$ e $x_4(t)$ salta all'occhio un aspetto particolare: i grafici presentano le medesime forme, con l'unica differenza che risiede nel fattore di scala. Commento entrambi gli aspetti.

FATTORE DI SCALA

Il servomeccanismo in questione prevede che il motore e il carico siano collegati da una trasmissione non ideale, ovvero non rigida.

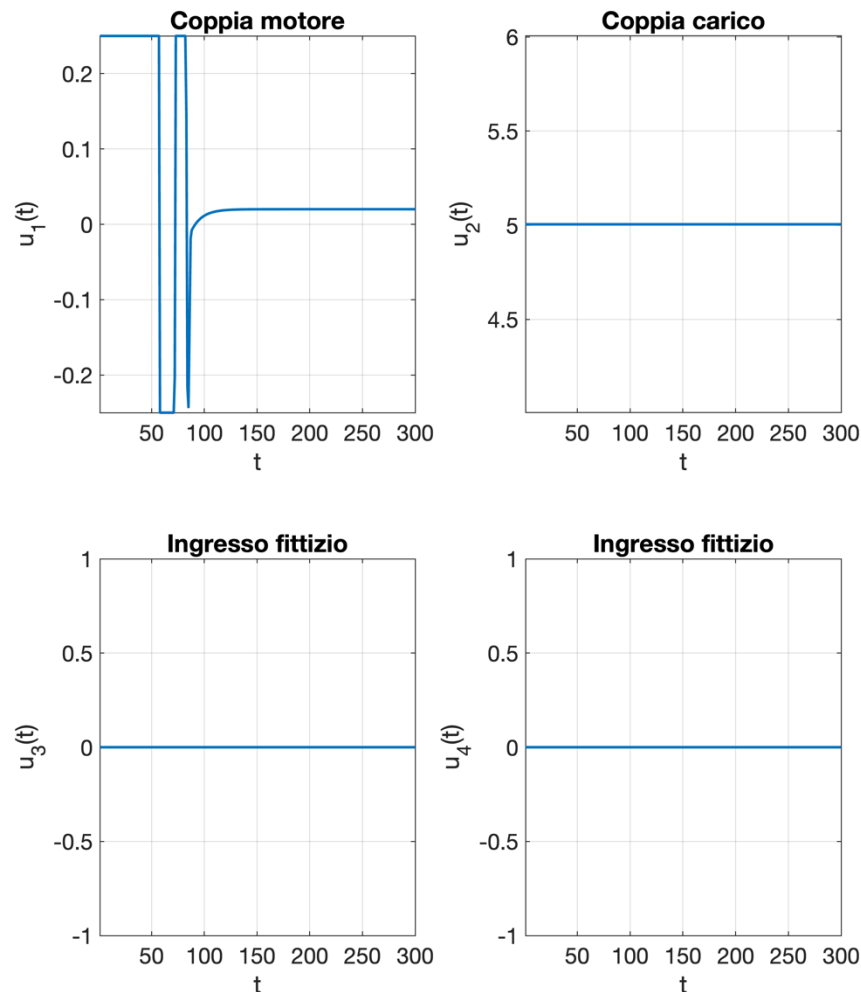
Se la trasmissione fosse rigida, la coppia generata dal motore verrebbe interamente trasmessa al carico (in tal caso i grafici, oltre che avere le stesse forme, sarebbero perfettamente sovrapponibili). Il fatto che la trasmissione sia elastica fa sì che questa vada ad introdurre nel sistema una dinamica nel momento in cui il motore genera una coppia. Tale dinamica è tale da far sì che la coppia generata dal motore venga trasmessa al carico diminuita di n volte, dove n = rapporto di trasmissione.

MEDESIME FORME

La trasmissione, pur essendo elastica, non si comporta in modo anomalo: il motore gira in una certa direzione per generare la coppia e la trasmissione riporta al carico la coppia ridotta girando nella stessa direzione in cui gira il motore.

Dall'analisi dei grafici relativi a $x_3(t)$ e $x_4(t)$ emerge un altro aspetto apparentemente controverso: i picchi. Trattandosi di un controllo di posizione, il motore, girando in una certa direzione, provoca uno spostamento del carico nella medesima direzione. Dato che il sistema non è ideale si verificano sia delle sovraelongazioni sia delle sottoelongazioni (il motore porta il sistema un po' più in là rispetto al suo obiettivo reale). In tal caso il motore si ferma istantaneamente e inverte la direzione di marcia.

COMPORTAMENTO DEGLI INGRESSI CON $Q = R = I$, $T_s = 0,1$ e $N = 10$

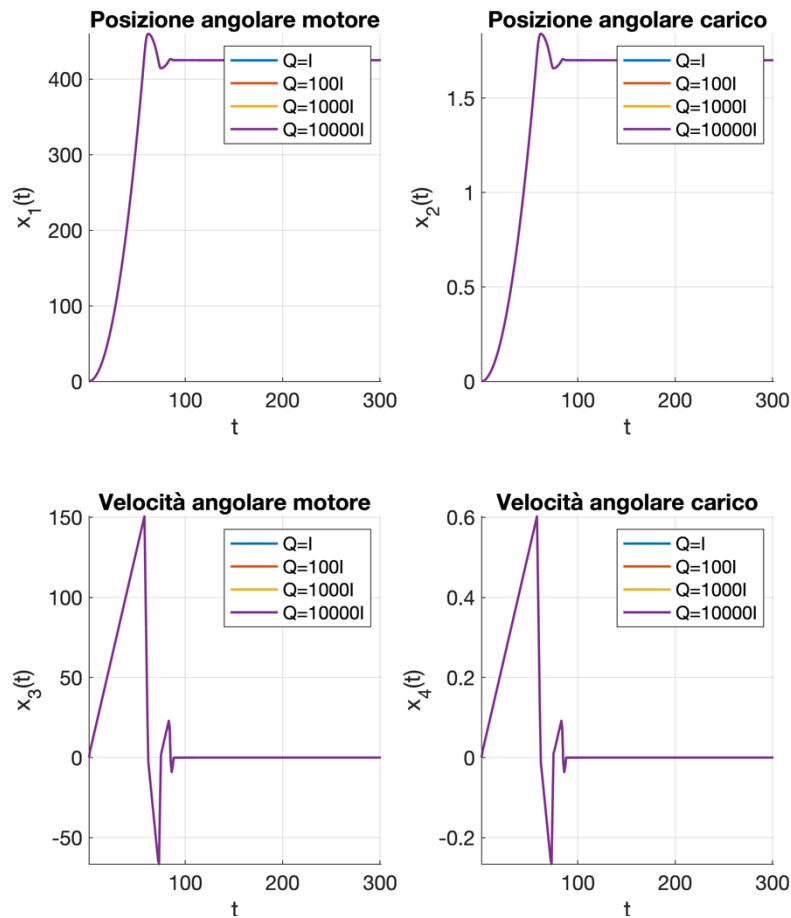


Nota sugli ingressi $u_3(t)$ e $u_4(t)$: il testo del progetto fornisce informazioni solo relativamente agli ingressi $u_1(t)$ e $u_2(t)$, ma si è reso necessario introdurre gli ingressi fittizi $u_3(t)$ e $u_4(t)$ per far sì che l'ipotesi di accessibilità dello stato possa risultare vera. Di fatto, come evince dai grafici, $u_3(t)$ e $u_4(t)$ non hanno alcun impatto per il sistema.

Il comportamento degli ingressi $u_1(t)$ e $u_2(t)$ è perfettamente in linea con le specifiche sul progetto. Dal grafico di $u_1(t)$ si possono notare le saturazioni a 0,25 e $-0,25$. Questo comportamento lascia intendere che, per giungere a convergenza, il sistema vorrebbe applicare delle azioni di controllo più aggressive, ma non può per via dei vincoli imposti.

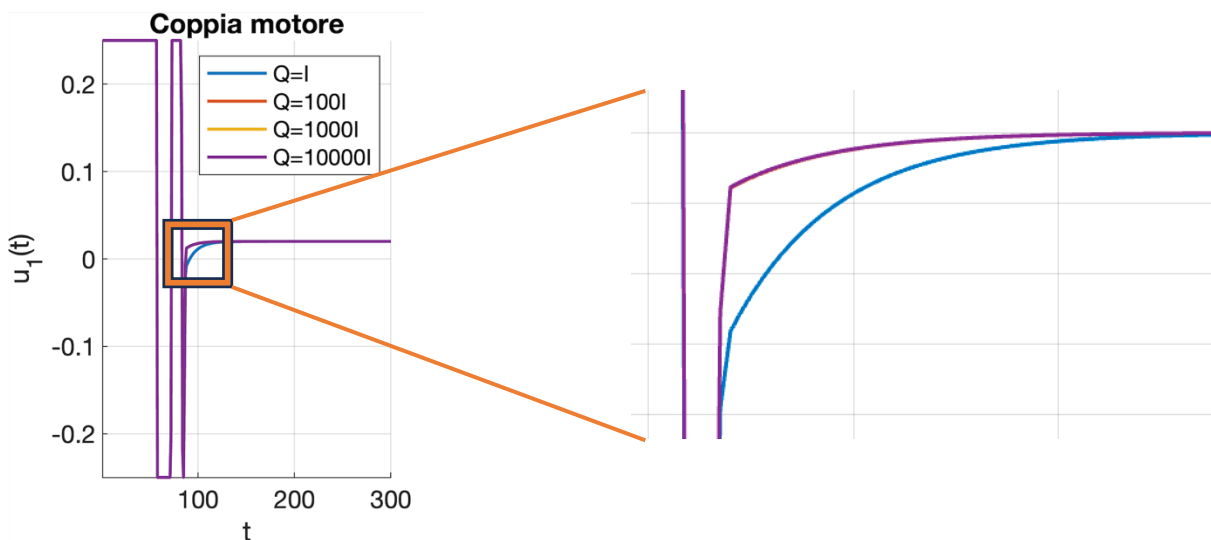
Per il proseguo farò riferimento al solo grafico relativo all'ingresso $u_1(t)$.

COMPORTAMENTO DEGLI STATI AL VARIARE DI Q , $R = I$, $T_s = 0,1$ E $N = 10$



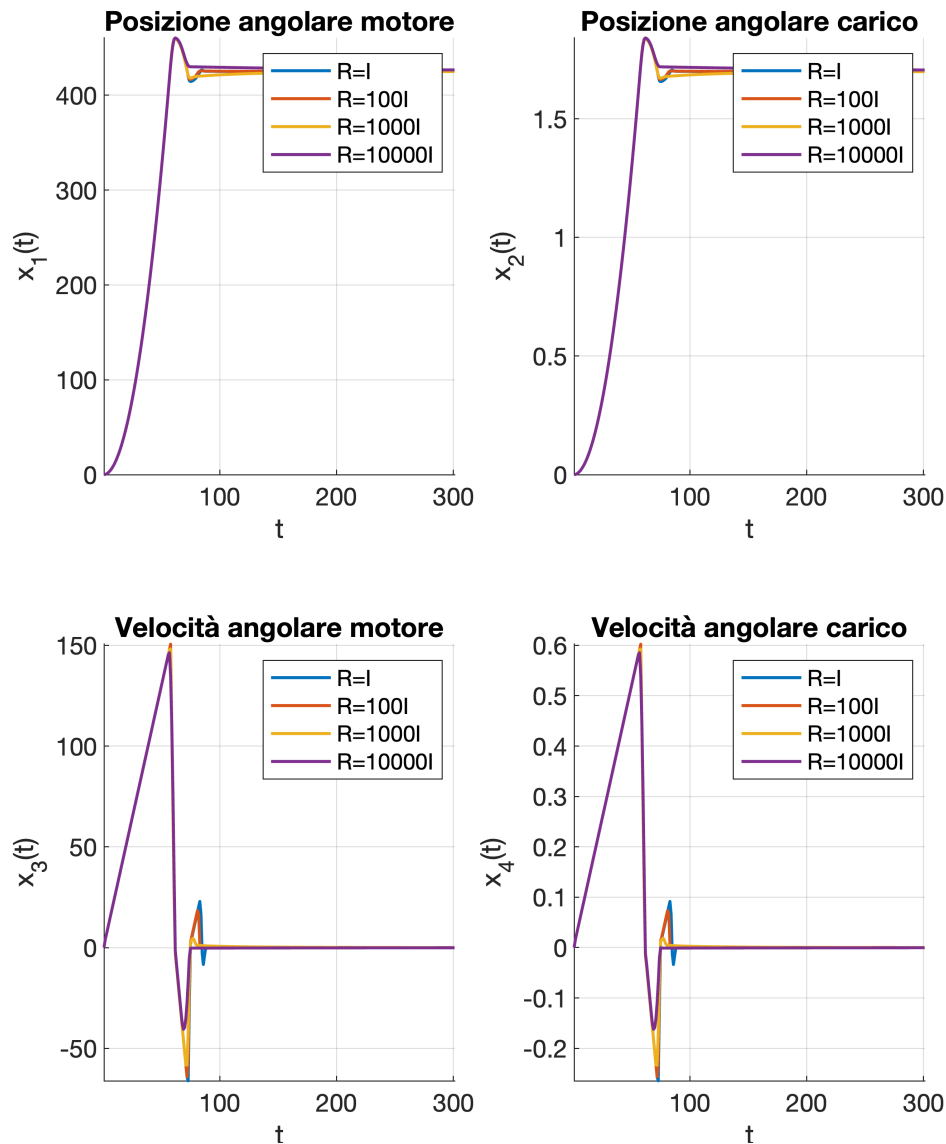
Priorizzare sempre di più la minimizzazione dello stato lasciando inalterata la matrice R non porta alcun beneficio dal punto di vista dell'analisi degli stati.

COMPORTAMENTO DEGLI INGRESSI AL VARIARE DI Q , $R = I$, $T_s = 0,1$ E $N = 10$



Priorizzare sempre di più la minimizzazione dello stato lasciando inalterata la matrice R non porta alcun beneficio nemmeno dal punto di vista dell'analisi degli ingressi, se non nel frame evidenziato.

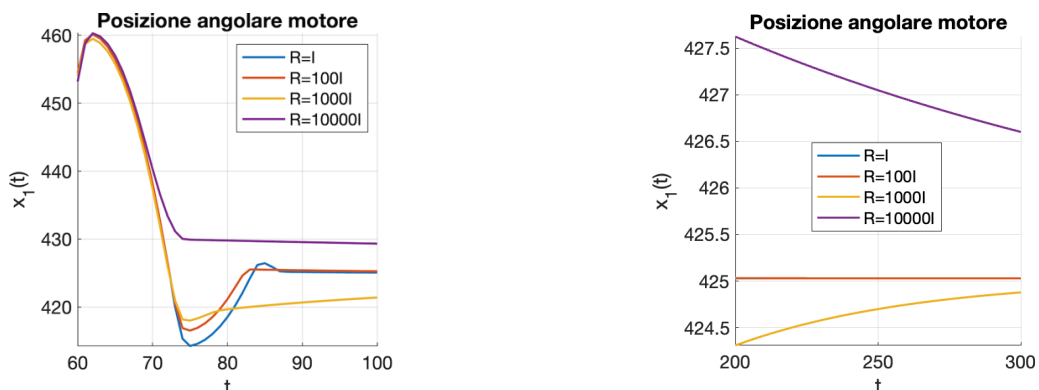
COMPORTAMENTO DEGLI STATI AL VARIARE DI R , $Q = I$, $T_s = 0,1$ E $N = 10$



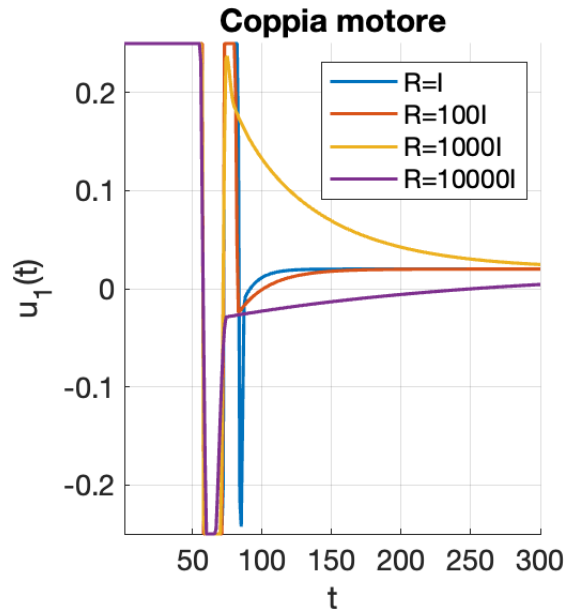
Priorizzare sempre di più la minimizzazione dell'azione di controllo porta a far sì che:

- Il sistema si comporti in modo più lineare (per valori di R “bassi” il sistema è soggetto a sovraelongazioni più accentuate nel transitorio rispetto a quelle che presenta per R “alti”);
- Il sistema giunga più lentamente a convergenza.

Queste osservazioni sono verificabili zoomando, ad esempio, il grafico dello stato $x_1(t)$ negli intervalli $60 \leq t \leq 100$ e $200 \leq t \leq 300$:



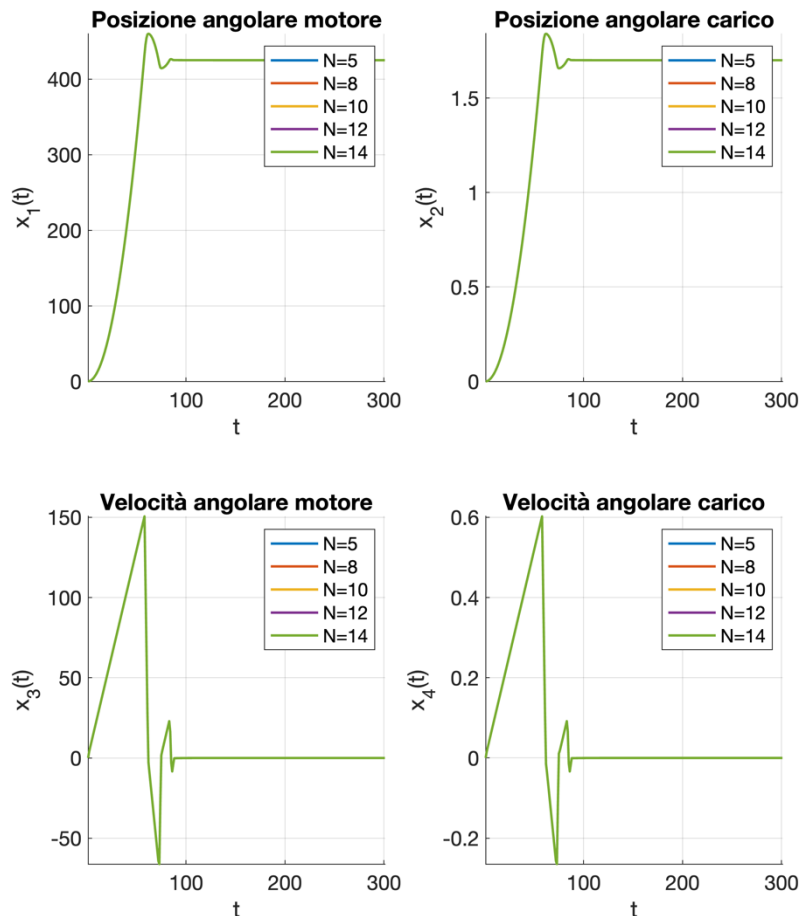
COMPORTAMENTO DEGLI INGRESSI AL VARIARE DI R , $Q = I$, $T_s = 0,1$ E $N = 10$



Come si evince dal grafico, l'azione di controllo $u_1(t)$ ha circa lo stesso comportamento al variare di R nell'intervallo di tempo $0 \leq t \leq 65$. Le differenze causate da R emergono per $65 \leq t \leq 300$. In particolare, più R aumenta, più $u_1(t)$ tende ad essere cauta:

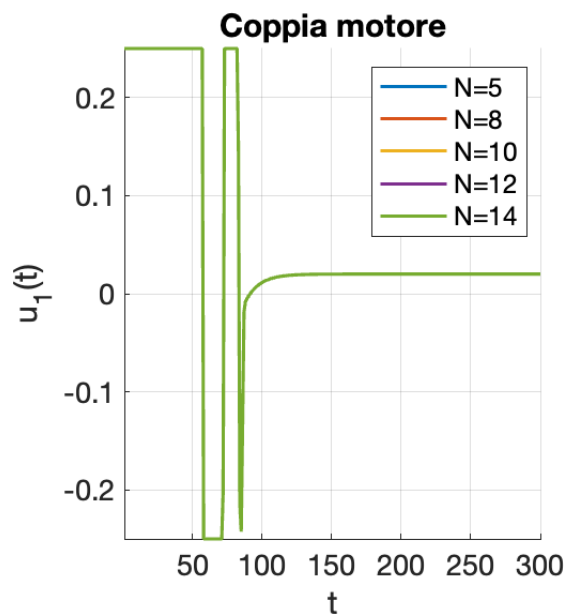
- $R = 100I$ fa sì che $u_1(t)$ saturi a 0,25, ma per un tempo inferiore rispetto a quello per $R = I$;
- $R = 1000I$ non fa saturare mai $u_1(t)$ in $65 \leq t \leq 300$;
- $R = 10000I$ fa sì che $u_1(t)$ mantenga un comportamento più lineare rispetto a quello che si ottiene per $R = 1000I$ nell'intervallo $65 \leq t \leq 120$.

COMPORTAMENTO DEGLI STATI AL VARIARE DI N , $Q = R = I$, $T_s = 0,1$



Come si evince dai grafici, far variare il numero di passi di predizione N fissando gli altri parametri ai valori $Q = R = I$ e $T_s = 0,1$, non produce alcun effetto degno di nota.

COMPORTAMENTO DEGLI INGRESSI AL VARIARE DI N , $Q = R = I$, $T_s = 0,1$



Per gli ingressi del sistema vale lo stesso discorso fatto per gli stati.