



UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

DIPARTIMENTO DI INGEGNERIA
MECCANICA E INDUSTRIALE

Corso di Laurea
in Ingegneria dell'Automazione Industriale

Relazione Finale ServoSystem & Robotics

Cinematica, Dinamica e Controllo
di un Manipolatore a tre assi

Relatore: Prof. Giovanni Legnani

A cura di:
Msaad Abdelghani
Matricola n. 723570
Zanetti Andrea
Matricola n. 723619

Indice

Introduzione	v
1 Presentazione del Manipolatore	1
1.1 Le specifiche di progetto	1
1.1.1 Il robot assegnato	1
1.1.2 Il compito da svolgere	2
1.2 Dati	2
1.2.1 Link	2
1.2.2 Giunti	4
1.3 Spazio di lavoro del robot	5
1.3.1 Singularità	6
2 Cinematica	7
2.1 Fasi del movimento	7
2.2 Metodologie di calcolo per la cinematica	8
2.2.1 Cinematica mediante Jacobiano	8
2.2.2 Cinematica mediante Matrici 4x4	11
2.3 Risultati della cinematica	15
2.3.1 Fase 1	16
2.3.2 Fase 2	17
2.3.3 Fase 3	19
2.3.5 Tutte le fasi	21
3 Dinamica	23
3.1 Metodologie di calcolo per la dinamica	23
3.1.1 Dinamica mediante Matrici 4x4	23
3.2 Risultati della dinamica	26
3.2.1 Fasi 1 e 3	26
3.2.2 Fase 2	31
4 Sviluppo In SimScape	34
4.1 Modello	34
4.2 Verifica dei calcoli cinematici e dinamici e modello sviluppato	36

5 Controllo	39
5.1 Il controllo in cascata per posizione e velocità	39
Conclusioni	41

Elenco delle figure

1.1	Robot assegnato	1
1.2	Simbolo da lavorare	2
1.3	Dimensioni geometriche del manipolatore a tre giunti	3
1.4	Workspace secondo la norma UNI ISO 29946	5
1.5	Workspace secondo la norma UNI ISO 29946	5
1.6	Workspace 3D	6
1.7	Singularità giunti	6
1.8	Singularità spazio	6
2.1	Traiettoria, fase 1	15
2.2	Spazio cartesiano fase 1	16
2.3	Traiettoria, fase 2	17
2.4	Spazio cartesiano fase 2	18
2.5	Traiettoria, fase 3	19
2.6	Spazio cartesiano fase 3	20
2.7	Traiettoria, movimento completo	21
2.8	Spazio cartesiano fase totale	22
3.1	Coppie e forze degli attuatori, fase 1	27
3.2	Coppie e forze degli attuatori, fase 3	27
3.3	Evoluzione temporale delle energie, fase 1	28
3.4	Evoluzione temporale delle energie, fase 1	28
3.5	Derivata dell'energia totale e potenza totale, fase 1	29
3.6	Derivata dell'energia totale e potenza totale, fase 3	29
3.7	Confronto potenze, fase 1	30
3.8	Confronto potenze, fase 3	30
3.9	Coppie e forze degli attuatori, fase 2	31
3.10	Evoluzione temporale delle energie, fase 2	32
3.11	Derivata dell'energia totale e potenza totale, fase 2	33
3.12	Confronto potenze, fase 2	33
4.1	Schema del manipolatore in SimScape, giunto 1	34
4.2	Blocco per leggere ed esportare dati cinematici	35

4.3	Confronto giunto 1 MatLab - SimScape	36
4.4	Confronto giunto 3 MatLab - SimScape	36
4.5	Confronto giunto 2 MatLab - SimScape	37
4.6	Confronto coppia 1 MatLab - SimScape	37
4.7	Confronto coppia 2 MatLab - SimScape	38
4.8	Confronto coppia 3 MatLab - SimScape	38
5.1	Controllori utilizzati nello schema SimScape	40

Introduzione

Il presente documento illustra lo svolgimento del progetto finale del corso Servosystems & Robotics, tenuto dal professor Giovanni Legnani, frequentato durante il secondo semestre dell'anno accademico 2021-2022.

Nel capitolo 1, viene spiegato come, assegnati la forma del manipolatore ed il compito che esso deve svolgere, tramite un confronto con i dati di robot reali, sono state definite le dimensioni geometriche, le masse e le inerzie dei vari link ed i limiti di posizione, velocità, accelerazione e di coppia dei vari giunti.

Successivamente è stato sviluppato lo studio della cinematica e della dinamica tramite il software MatLab.

Nel capitolo 2 si illustra il modo in cui, in accordo con il compito che il robot deve svolgere, è stata definita la cinematica del movimento, sviluppata sia tramite l'uso dello Jacobiano che tramite le matrici tridimensionali.

All'interno del capitolo 3 si trova invece l'analisi della dinamica, ovvero il calcolo delle forze e delle coppie che ogni giunto deve esprimere per poter assicurare il moto richiesto. In questo caso i calcoli sono stati svolti tramite il metodo delle matrici tridimensionali.

Successivamente, come illustrato nel capitolo 4, mediante il software di simulazione SimScape, il robot è stato modellizzato e ciò ha reso possibile la verifica della correttezza dei calcoli cinematici e dinamici precedentemente svolti.

In ultimo luogo, il capitolo 5 mostra come, sempre in ambiente SimScape, è stato possibile implementare e testare una strategia di controllo del movimento del robot.

Capitolo 1

Presentazione del Manipolatore

1.1 Le specifiche di progetto

1.1.1 Il robot assegnato

Il robot assegnato, come si può osservare nella figura 1.1, è di tipo seriale e presenta tre giunti:

- Il primo giunto, di tipo rotoidale, consente la rotazione degli elementi successivi rispetto all'asse verticale, ortogonale al piano XY;
- Il secondo, di tipo rotoidale, consente la rotazione degli elementi successivi rispetto all'asse orizzontale, parallelo al piano XY;
- Il terzo, ancora di tipo rotoidale, consente la rotazione dell'ultimo link.

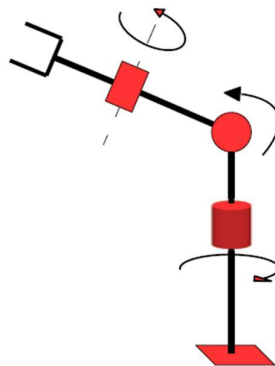


Figura 1.1: Robot assegnato

1.1.2 Il compito da svolgere

Il compito che viene richiesto di svolgere al manipolatore è il seguente:

- dalla posizione di riposo, raggiungere il luogo in cui si trova il simbolo da percorrere, simile a quello rappresentato in figura 1.2, nel minor tempo possibile;
- percorrere la forma del numero;
- tornare nella posizione di riposo, come all'inizio, nel minor tempo

possibile. Utilizzando l'algoritmo di Look-Ahead nel tratto di ripasso del simbolo.



Figura 1.2: Simbolo da percorrere

1.2 Dati

Per assegnare i parametri del manipolatore, si è fatto riferimento a un modello creato con SolidWorks e dati ricavanti da esempi simili.

1.2.1 Link

Nella tabella 1.1 vengono illustrate le grandezze caratterizzanti i differenti link: dimensione, massa e inerzia alla rotazione rispetto all'asse.

Link	Limiti				
	Dimensioni [m]	Massa [kg]	Inerzia X [kg*m]	Inerzia Y [kg*m]	Inerzia Z [kg*m]
L1	1.2	101.75	12.46	12.46	0.51
L2	0.7	33.34	0.09	1.4	1.4
L3	0.4	8.48	0.01	0.12	0.12

Tabella 1.1: Dimensioni, masse ed inerzie dei link

Per poter comprendere meglio la notazione utilizzata si può fare riferimento alla figura 1.3. Con L1 si indica la dimensione del link verticale, ovvero la base che permette la rotazione attorno all'asse Z. Le dimensioni dei due ulteriori link del robot sono invece indicate con L2 ed L3 che consentono i movimenti nello spazio.

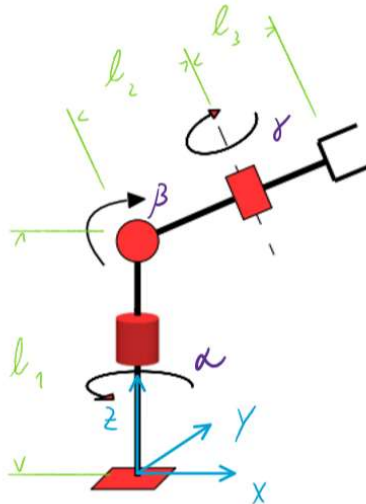


Figura 1.3: Dimensioni geometriche del manipolatore a tre giunti

Le masse e le inerzie sono espresse prendendo in considerazione un sistema di riferimento baricentrale e ricavate sperimentalmente mediante una simulazione paritetica con l'ausilio del software SolidWorks.

1.2.2 Giunti

Per quanto riguarda i tre giunti presenti nel manipolatore, introdotti nel paragrafo 1.1.1, i dati cinematici sono stati scelti arbitrariamente vista l'assenza di linee di progetto, sono comunque stati assegnati valori verosimili congruenti con la tipologia di robot utilizzato.

Per quanto riguarda i limiti sulle accelerazioni, non avendo alcuno dato disponibile, si è deciso di renderli poco più alti rispetto ai valori limite delle velocità. La tabella 1.2 riporta i valori di accelerazioni e velocità massime per ogni giunto rotativo, q_1 , q_2 e q_3 .

Dalla figura 1.2 mostrata in precedenza si può notare come le variabili di giunto (alfa, beta e gamma) influenzino la posa del manipolatore.

Giunto	Limiti			
	posizione [°]	velocità [rad/s]	accelerazione [rad/s ²]	coppia [Nm]
$q_1=$	0 - 360	2	3	-200 - 200
$q_2=$	-90 - 90	3	4	-200 - 200
$q_3=$	0 - 360	4	5	-100 - 100

1.3 Spazio di lavoro del robot

Le figure sottostanti, rappresentano, con proporzioni reali, lo spazio di lavoro del robot e la posa di quest'ultimo quando tutti i giunti si trovano in posizione minima o massima.

La figura 1.4 mostra la vista dall'alto, piano XY. Si può notare che, a causa della limitazione geometrica, esso non può raggiungere posizioni vicine al secondo giunto.

La figura 1.5 rappresenta invece la vista laterale, piano XZ. Anche in questa vi è presente la rappresentazione delle limitazioni geometriche.

In rosso rosso viene mostrato il primo link, in blu il link e in verde il terzo.

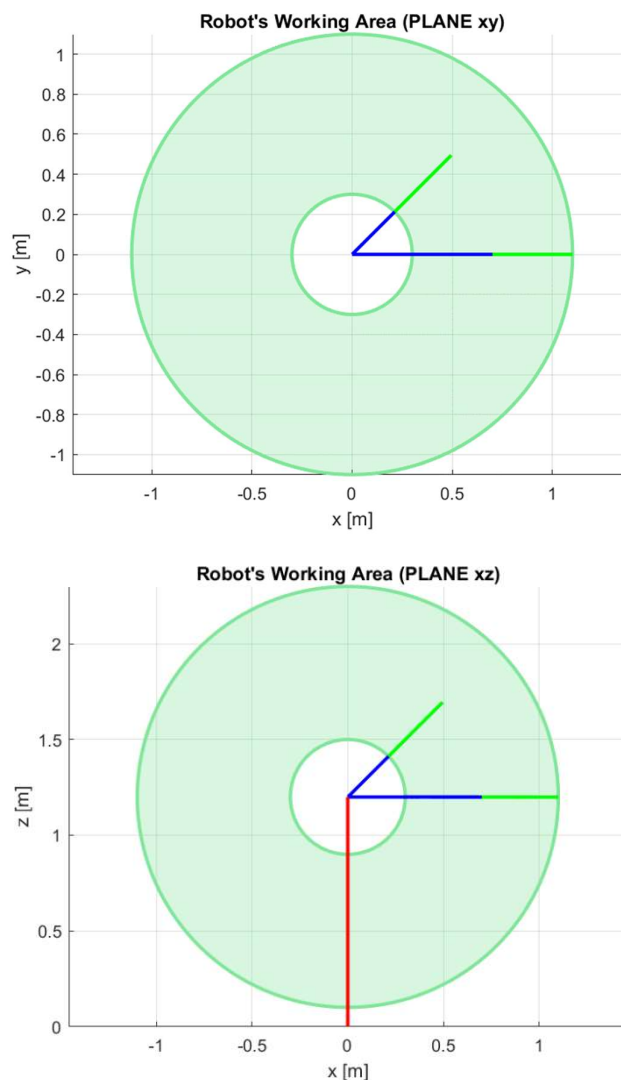


Figura 1.4 – 1.5: Workspace secondo la norma UNI ISO 2994

Nel programma è inoltre proposto il Workspace 3D così da poter osservare l'area di lavoro da ogni sua angolazione, figura 1.6.

Si può notare infatti come il reale spazio di lavoro tridimensionale sia il luogo compreso tra le due sfere.

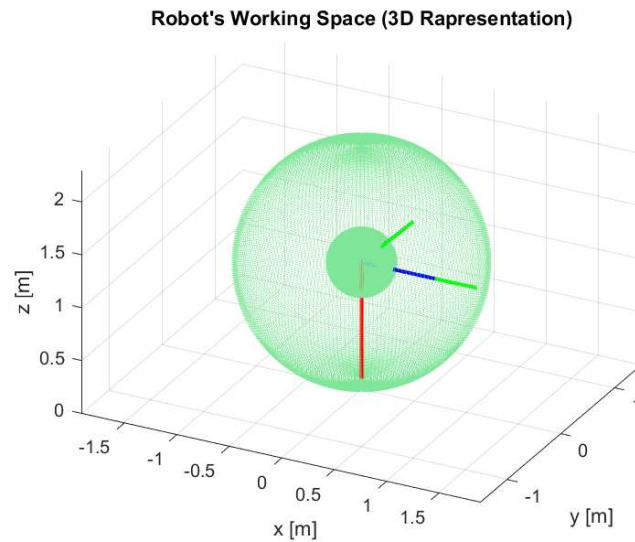


Figura 1.6: Workspace 3D

1.3.1 Singolarità

Oltre allo studio del Workspace è stato svolto uno studio relativo alle posizioni di singolarità del robot, creando così una rappresentazione grafica sia in funzione dei giunti che in funzione degli assi cartesiani, rispettivamente figura 1.7 e figura 1.8.

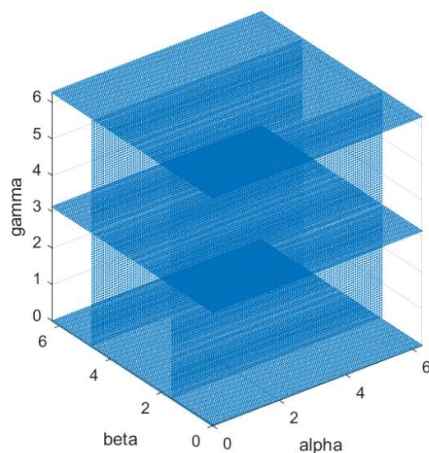


Figura 1.7: Singolarità giunti

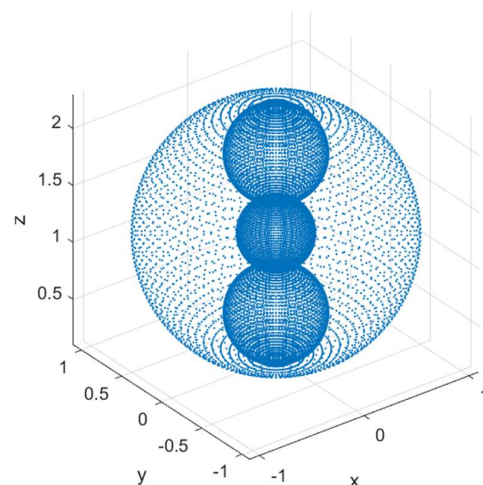


Figura 1.8: Singolarità spazio

Le configurazioni di singolarità sono state ottenute ponendo $\det(J) = 0$. Quindi trovando valori di α, β e γ per cui il determinante della matrice jacobiana si annulla. Di seguito il risultato ottenuto mentre per un più concisa analisi sulla matrice dello jacobiano si può fare riferimento al capitolo successivo.

$$\begin{aligned}
0 = & l_3^3 \cos^2(\alpha) \cos^3(\beta) \cos^2(\gamma) \sin(\gamma) - \\
& l_3^3 \cos(\beta) \cos^2(\gamma) \sin^2(\alpha) \sin(\gamma) - l_3^3 \cos^2(\alpha) \cos(\beta) \cos^2(\gamma) \sin(\gamma) + \\
& l_3^3 \cos^3(\beta) \cos^2(\gamma) \sin^2(\alpha) \sin(\gamma) + l_2^2 l_3 \cos^2(\alpha) \cos^3(\beta) \sin(\gamma) + \\
& l_2^2 l_3 \cos^3(\beta) \sin^2(\alpha) \sin(\gamma) - \\
& l_2 l_3^2 \cos^2(\alpha) \cos(\beta) \cos(\gamma) \sin(\gamma) - l_2 l_3^2 \cos(\beta) \cos(\gamma) \sin^2(\alpha) \sin(\gamma) + \\
& l_3^3 \cos^2(\alpha) \cos(\beta) \cos(\gamma)^2 \sin^2(\beta) \sin(\gamma) + l_3^3 \cos(\beta) \cos^2(\gamma) \sin^2(\alpha) \sin^2(\beta) \sin(\gamma) + \\
& 2l_2 l_3^2 \cos^2(\alpha) \cos^3(\beta) \cos(\gamma) \sin(\gamma) + l_2^2 l_3 \cos^2(\alpha) \cos(\beta) \sin^2(\beta) \sin(\gamma) + \\
& 2l_2 l_3^2 \cos^3(\beta) \cos(\gamma) \sin^2(\alpha) \sin(\gamma) + l_2^2 l_3 \cos(\beta) \sin^2(\alpha) \sin^2(\beta) \sin(\gamma) + \\
& 2l_2 l_3^2 \cos^2(\alpha) \cos(\beta) \cos(\gamma) \sin^2(\beta) \sin(\gamma) + \\
& 2l_2 l_3^2 \cos(\beta) \cos(\gamma) \sin^2(\alpha) \sin^2(\beta) \sin(\gamma)
\end{aligned}$$

Capitolo 2

Cinematica

Nel paragrafo 2.1 viene illustrato come è stato diviso il movimento, nel paragrafo 2.2 vengono illustrati sinteticamente i due metodi di calcolo della cinematica e nel paragrafo 2.3 vengono invece commentati i risultati ottenuti.

2.1 Fasi del movimento

In accordo con il compito che il manipolatore deve svolgere, illustrato nel paragrafo 1.1.2, si è diviso il movimento in tre differenti fasi:

- Dalla posizione di riposo, il robot raggiunge il punto di partenza del simbolo nel minimo tempo di attuazione;
- Il robot ricalca il simbolo fornito mediante l'algoritmo Look-Ahead;
- Il robot torna nella posizione di riposo con il minimo tempo di attuazione.

La tabella 2.1 mostra, per ogni fase, la legge di moto che si è scelto di utilizzare ed il tempo impiegato.

Fase	Legge di moto usata	Spazio	Tempo richiesto [s]
1	tre tratti	dei giunti	1.4
2	Tre tratti + L.A.	cartesiano	2.5
3	tre tratti	dei giunti	1.4

Tabella 2.1: Dati di moto delle differenti fasi di movimento del robot

2.2 Metodologie di calcolo per la cinematica

La tabella 2.2 introduce e riassume quanto viene spiegato in questo paragrafo. I calcoli riguardanti la cinematica sono stati sviluppati con due metodi differenti: quello dello Jacobiano, illustrato nel paragrafo 2.2.1, e quello delle matrici, trattato nel paragrafo 2.2.2.

Questi due metodi usano formalismi di calcolo differenti ma condividono alcune formule. Tra queste, quelle per passare dalla velocità o dall'accelerazione nello spazio cartesiano (in coordinate $\dot{x}, \dot{y}, \ddot{x}, \ddot{y} \dots$) alla velocità o all'accelerazione nello spazio dei giunti (in coordinate $\dot{\alpha}, \dot{\beta}, \ddot{\alpha}, \ddot{\beta} \dots$).

Trasformazione	Jacobiano	Matrici
Posizione		
giunti \rightarrow cartesiano	cinematica diretta	matrici
cartesiano \rightarrow giunti	cinematica inversa	Jacobiano
Velocità		
giunti \rightarrow cartesiano	$\dot{S} = J * \dot{Q}$	matrici
cartesiano \rightarrow giunti	$\dot{Q} = J^{-1} * \dot{S}$	Jacobiano
Accelerazione		
giunti \rightarrow cartesiano	$\ddot{S} = \dot{J} * \dot{Q} + J * \ddot{Q}$	matrici
cartesiano \rightarrow giunti	$\ddot{Q} = J^{-1} * (\ddot{S} - \dot{J} * \dot{Q})$	Jacobiano

Tabella 2.2: Riassunto delle metodologie di calcolo per la cinematica

2.2.1 Cinematica mediante Jacobiano

Per avere informazioni complete sul moto, una volta definita la legge di moto, che sia nello spazio dei giunti o cartesiano, mediante lo Jacobiano è possibile convertire le informazioni sull'andamento di posizione, velocità ed accelerazione nello spazio dei giunti a quello cartesiano e viceversa.

Di seguito le equazioni dello Jacobiano:

$$J_{11} = -l_2 \sin(\alpha) \cos(\beta) - l_3 \cos(\alpha) \sin(\gamma) - l_3 \sin(\alpha) \cos(\beta) \cos(\gamma)$$

$$J_{12} = -l_2 \cos(\alpha) \sin(\beta) - l_3 \cos(\alpha) \sin(\beta) \cos(\gamma)$$

$$J_{13} = -l_2 \sin(\alpha) \cos(\gamma) - l_3 \cos(\alpha) \cos(\beta) \sin(\gamma)$$

$$J_{21} = l_2 \cos(\alpha) \cos(\beta) - l_3 \sin(\alpha) \sin(\gamma) + l_3 \cos(\alpha) \cos(\beta) \cos(\gamma)$$

$$J_{22} = -l_2 \sin(\alpha) \sin(\beta) - l_3 \sin(\alpha) \sin(\beta) \cos(\gamma)$$

$$J_{23} = l_3 \cos(\alpha) \cos(\gamma) - l_3 \sin(\alpha) \cos(\beta) \sin(\gamma)$$

$$J_{31} = 0$$

$$J_{32} = -l_2 \cos(\beta) - l_3 \cos(\beta) \cos(\gamma)$$

$$J_{33} = l_3 \sin(\beta) \sin(\gamma)$$

2.2.1.1 Posizione

Per passare dallo spazio dei giunti allo spazio cartesiano, si usa la cinematica diretta.

In questo caso viene esplicitata mediante le seguenti formule:

$$x = l2 * \cos(\alpha) * \cos(\beta) - l3 * \sin(\alpha) * \sin(\gamma) + l3 * \cos(\alpha) * \cos(\beta) * \cos(\gamma)$$

$$y = l2 * \sin(\alpha) * \cos(\beta) - l3 * \cos(\alpha) * \sin(\gamma) + l3 * \sin(\alpha) * \cos(\beta) * \cos(\gamma)$$

$$z = l1 - l2 * \sin(\beta) - l3 * \sin(\beta) * \cos(\gamma)$$

Queste coordinate vengono poi salvate all'interno del vettore S:

$$S = [x; y; z]$$

In quanto alla cinematica inversa, per conoscere le rotazioni dei giunti q1, q2 e q3 che assicurano il raggiungimento di una data posizione, si utilizzano le formule riportate:

$$\alpha = \pm \arccos\left(\frac{x * (l2 * \cos(\beta) + l3 * \cos(\beta) * \cos(\gamma)) + y * l3 * \sin(\gamma)}{l3 * \sin(\gamma)^2 + (l2 * \cos(\beta) + l3 * \cos(\beta) * \cos(\gamma))^2}\right)$$

$$\beta = \arcsin\left(\frac{l1 - z}{l2 + l3 * \cos(\gamma)}\right)$$

$$\gamma = \pm \arccos\left(\frac{x^2 + y^2 + (z - l1)^2 - l3^2 - l2^2}{2 * l2 * l3}\right)$$

Il valore di α e γ può assumere sia valore positivo che negativo, questo in base al quadrante in cui stiamo lavorando, la scelta viene elaborata tramite software.

Queste coordinate vengono poi salvate all'interno del vettore Q:

$$Q = [\alpha, \beta, \gamma]$$

2.2.1.2 Velocità

Per poter convertire le velocità dei giunti in velocità nello spazio cartesiano, si usa lo Jacobiano:

$$\dot{S} = J * \dot{Q}$$

La formula inversa, per passare da velocità nello spazio cartesiano a velocità dei giunti è invece:

$$\dot{Q} = J^{-1} * \dot{S}$$

Lo Jacobiano, definito come derivata delle coordinate nello spazio cartesiano rispetto alle coordinate di giunto:

$$J = \frac{\partial S}{\partial Q}$$

2.2.1.3 Accelerazione

Per quanto riguarda le accelerazioni, la formula diretta utilizzata è:

$$\ddot{S} = \dot{J} * \dot{Q} + J * \ddot{Q}$$

La sua formula inversa:

$$\ddot{Q} = J^{-1} * (\ddot{S} - \dot{J} * \dot{Q})$$

2.2.2 Cinematica mediante Matrici 4x4

Non sempre la cinematica diretta ed inversa, illustrate nel paragrafo 2.2.1.1, possono essere calcolate in maniera semplice come per questo manipolatore. Nel caso di robot più complessi, un metodo alternativo allo studio è quello dell'uso delle matrici tridimensionali.

Queste forniscono un utile strumento per lo studio della cinematica diretta ma, per la cinematica inversa, è ancora necessario sfruttare lo Jacobiano. Se non è possibile scrivere le formule della cinematica inversa, nel caso non si abbiano relazioni analitiche è possibile applicare il metodo di Newton Raphson, un metodo per il calcolo numerico di una soluzione che però non sempre può fornire un risultato. Nel nostro caso le relazioni analitiche erano disponibili quindi non è stato necessario procedere con il calcolo della soluzione numerica.

Allo stesso modo, per conoscere velocità ed accelerazioni nello spazio cartesiano date quelle espresse nello spazio dei giunti, le matrici offrono uno strumento di calcolo potente, illustrato rispettivamente in 2.2.2.2 e 2.2.2.3. Per compiere invece il calcolo inverso si è utilizzato il metodo dello Jacobiano, illustrato in 2.2.1.2 e 2.2.1.3.

2.2.2.1 Posizione

Mediante il metodo delle matrici, la posa di ogni sistema di riferimento rispetto al precedente viene definita tramite una matrice di rototraslazione.

La prima matrice è quella che va dalla base di appoggio del robot alla parte terminale del primo link, di dimensione l1.

$$M_{01} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & l1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La seconda matrice è definita dalla parte terminale del primo link alla parte terminale del secondo link, di dimensione l2.

$$M_{12} = \begin{bmatrix} \cos(\beta) & 0 & -\sin(\beta) & l2 * \cos(\beta) \\ 0 & 1 & 0 & 0 \\ \sin(\beta) & 0 & \cos(\beta) & l2 * \sin(\beta) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La terza matrice è quella che va dalla parte terminale del secondo link alla parte terminale del terzo link, di dimensione l3.

$$M_{23} = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & l3 * \cos(\gamma) \\ \sin(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & l3 * \sin(\gamma) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

É possibile portare qualunque matrice al sistema di riferimento di base mediante la seguente formula:

$$M_{0,k/0} = M_{0,1/0} * M_{1,2/1} * M_{2,3/2} * \dots * M_{k-2,k-1/k-2} * M_{k-1,k/k-1}$$

Si può procedere con il calcolo di velocità ed accelerazioni, in cui entrano in gioco anche le matrici L, così definite:

- per le matrici che esprimono rotazione rispetto all'asse Y locale:

$$L_{ry} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- per le matrici che esprimono rotazione rispetto all'asse Z locale:

$$L_{rz} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

2.2.2.2 Velocità

Per il calcolo della velocità, si parte dalla velocità del sistema di riferimento corrente per la matrice L corrente.

$$W_{rel,k/k-1} = L * \dot{Q}$$

Si porta questa velocità relativa nel sistema di riferimento di base:

$$W_{rel,k/0} = M_{0,k/0} * W_{rel,k/k-1} * M_{0,k/0}^{-1}$$

Per trovare la velocità assoluta nel sistema di riferimento di base si somma la velocità del frame precedente a quella del frame corrente rispetto al precedente, entrambi espressi rispetto al sistema di riferimento di base:

$$W_{ass,k/0} = W_{ass,k-1/0} + W_{rel,k/0}$$

La formula inversa, per passare da velocità nello spazio cartesiano a velocità dei giunti è la stessa usata nel metodo dello Jacobiano:

$$\dot{Q} = J^{-1} * \dot{S}$$

2.2.2.3 Accelerazione

Per quanto riguarda il calcolo delle accelerazioni, si applica un ragionamento simile:

$$H_{rel,k/k-1} = L * \ddot{Q} + L^2 * \dot{Q}^2$$

Si porta quest'accelerazione relativa nel sistema di riferimento di base:

$$H_{rel,k/0} = M_{0,k/0} * H_{rel,k/k-1} * M_{0,k/0}^{-1}$$

Per trovare l'accelerazione assoluta nel sistema di riferimento di base si utilizza il teorema di Coriolis:

$$H_{ass,k/0} = H_{ass,k-1/0} + H_{rel,k/0} + 2 * W_{ass,k-1/0} * W_{rel,k/0}$$

La formula inversa, per passare da accelerazione nello spazio cartesiano ad accelerazione dei giunti è la stessa usata nel metodo dello Jacobiano:

$$\ddot{Q} = J^{-1} * (\ddot{S} - \dot{J} * \dot{Q})$$

2.3 Risultati della cinematica

Di seguito vengono mostrate le traiettorie ed i profili di posizione, velocità ed accelerazione nello spazio dei giunti e nello spazio cartesiano delle tre fasi di movimento.

Le fasi 1 e 3 sono dedicate rispettivamente all'avvicinamento e allontanamento dall'area di lavorazione.

Mentre la fase 2 è dedicata interamente alla lavorazione assegnata, rispettando le specifiche di progetto fornite.

2.3.1 Fase 1

Nella prima fase il robot si muove dalla posizione di riposo alla posizione di lavorazione, come si può vedere nella figura 2.1.

Per fare in modo che questa fase avvenga nel minimo tempo possibile, si è deciso di attuare il robot definendo la legge di moto di tipo tre tratti nello spazio dei giunti al fine di poter accelerare al massimo fino al raggiungimento della velocità limite e mantenerla fino al momento di frenare per giungere in posizione.

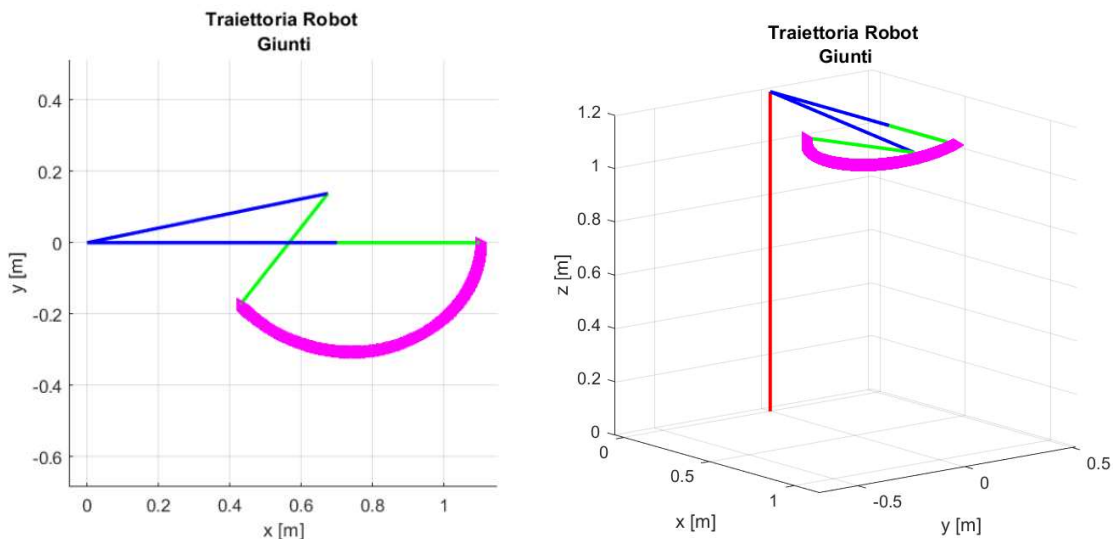


Figura 2.1: Traiettorie, fase 1

Per quanto riguarda il movimento nello spazio cartesiano, la figura 2.2.1 mostra l'evoluzione della posizione, velocità e accelerazione nella variabile x. Gli stessi grafici sono mostrati per la variabile y e z rispettivamente nelle figure 2.2.2 e 2.2.3.

Si può notare come le curve di velocità e accelerazione sono perfettamente concordi con i valori teorici stimati (in figura rappresentati con il simbolo “#”), utilizzati dunque come modalità di verifica, ottenuti mediante la derivata numerica della posizione e poi della velocità.

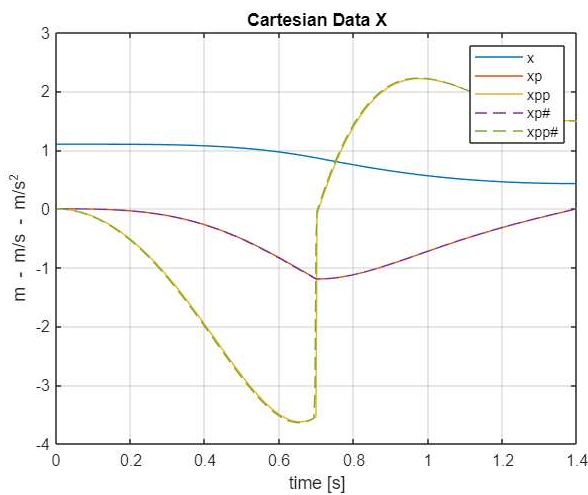


Figura 2.2.1: Spazio cartesiano fase 1 – asse x

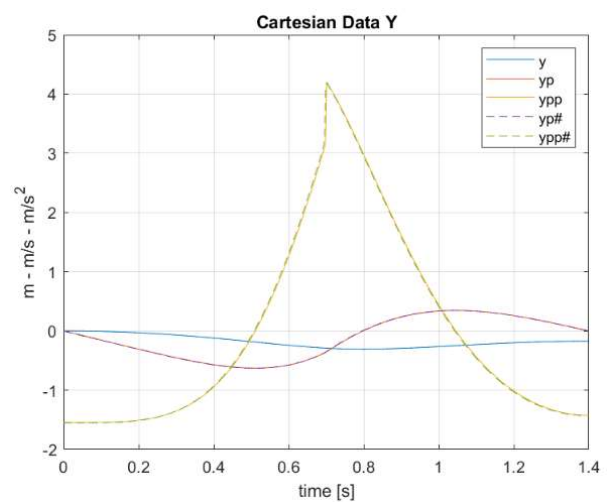


Figura 2.2.2: Spazio cartesiano fase 1 – asse y

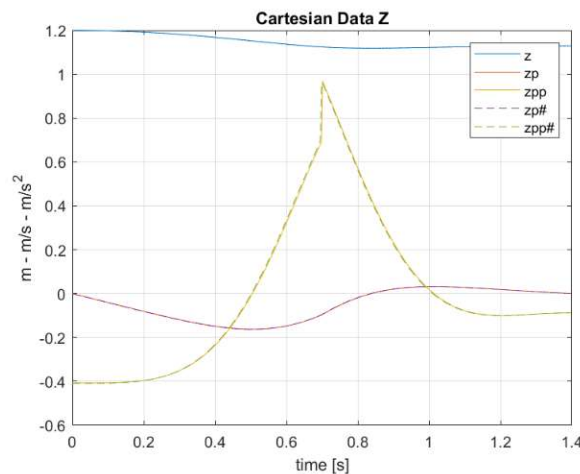


Figura 2.2.3: Moto spazio cartesiano fase 1 – asse z

2.3.2 Fase 2

Una volta arrivati al punto di partenza della lavorazione, il robot inizierà a ripassare la figura fornita. Questa lavorazione si basa sull'algoritmo Look-Ahead, questo implica uno studio predittivo della traiettoria e il robot accelererà ove possibile rallentando in prossimità di curve, così da essere il più preciso possibile nel ripassare il simbolo.

Tramite questo algoritmo siamo riusciti a creare un movimento fluido e costante, senza dover fermare il robot a causa di un cambio di direzione ma semplicemente rallentandolo, come accade a metà della traiettoria.

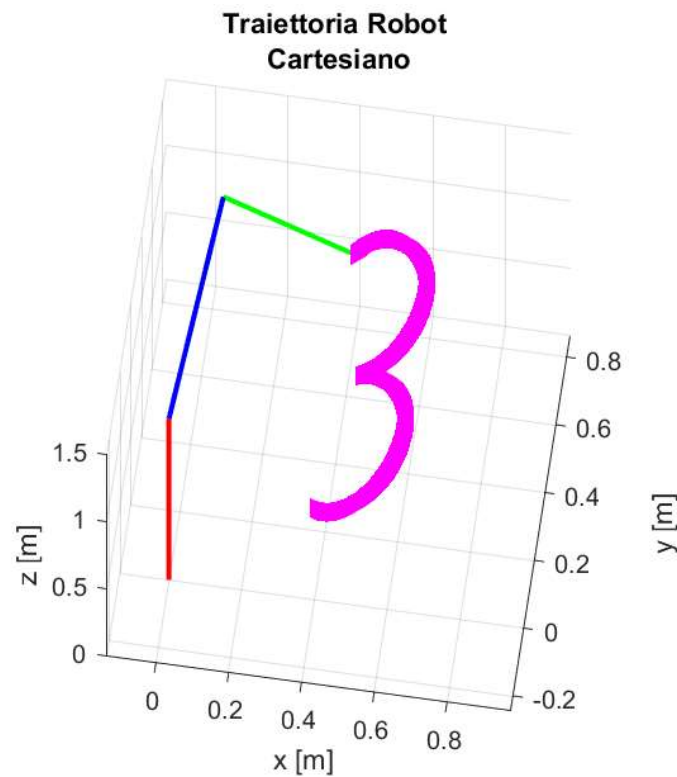


Figura 2.3: Traiettoria, fase 2

Nello spazio cartesiano vengono riportate sottostante le evoluzioni della posizione, velocità e accelerazione nella variabile x, y, z.

In questa seconda fase possiamo vedere esattamente come il robot segua la forma del 3 con variazioni speculari negli assi.

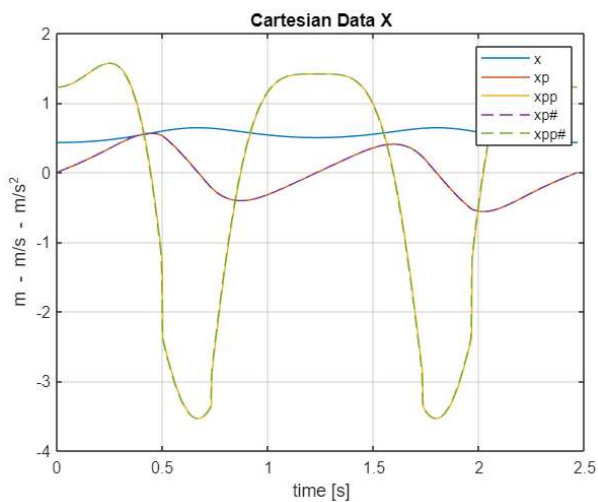


Figura 2.4.1: Spazio cartesiano fase 2 – asse x

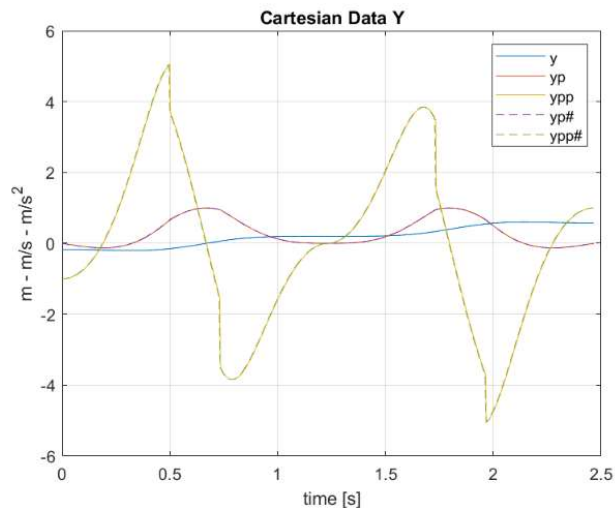


Figura 2.4.2: Spazio cartesiano fase 2 – asse y

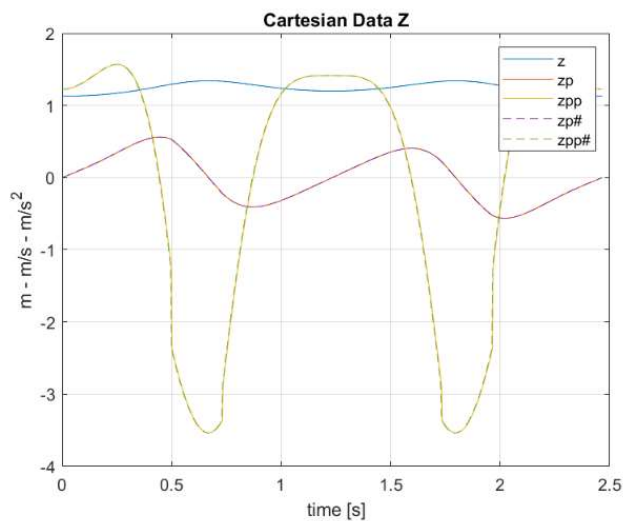


Figura 2.4.3: Moto spazio cartesiano fase 2 – asse z

2.3.3 Fase 3

Nell'ultima fase il robot deve tornare alla posizione di riposo dalla posizione finale della lavorazione svolta, come si può vedere nella figura 2.5.

Per fare in modo che questa fase avvenga nel minimo tempo possibile, si è deciso di muovere il robot mediante la legge di moto di tipo tre tratti nello spazio dei giunti, così da ottenere le massime performance nella movimentazione da parte del robot.

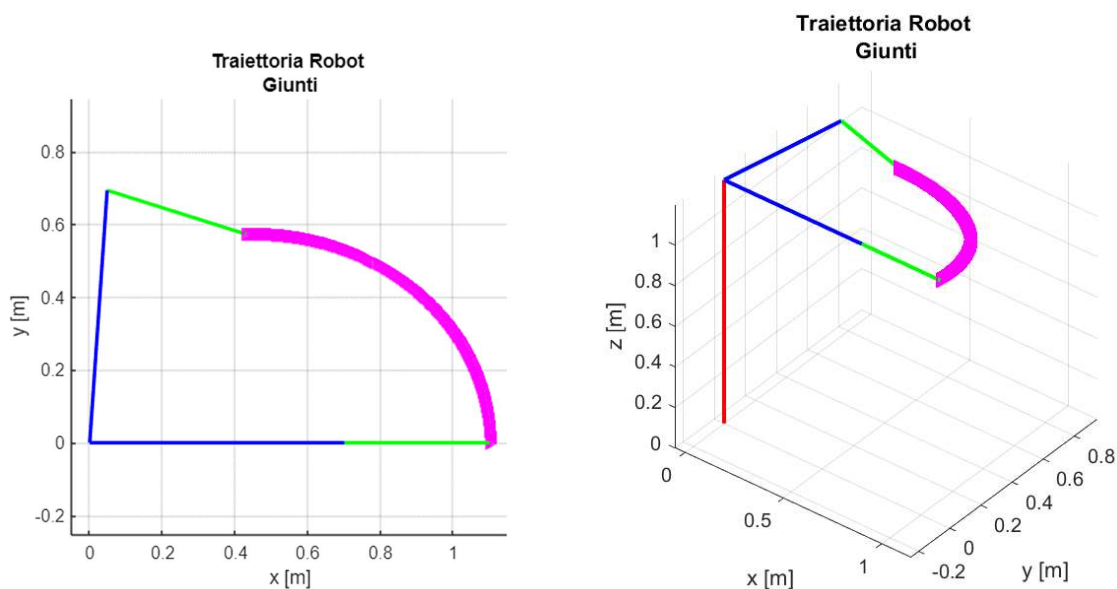


Figura 2.5: Traiettorie, fase 3

Per quanto riguarda il movimento nello spazio cartesiano, la figura 2.6.1 mostra l'evoluzione della posizione, velocità e accelerazione nella variabile x . Gli stessi grafici sono mostrati per la variabile y e z rispettivamente nelle figure 2.6.2 e 2.6.3.

Si può notare come le curve di velocità e accelerazione sono perfettamente concordi con i valori teorici stimati.

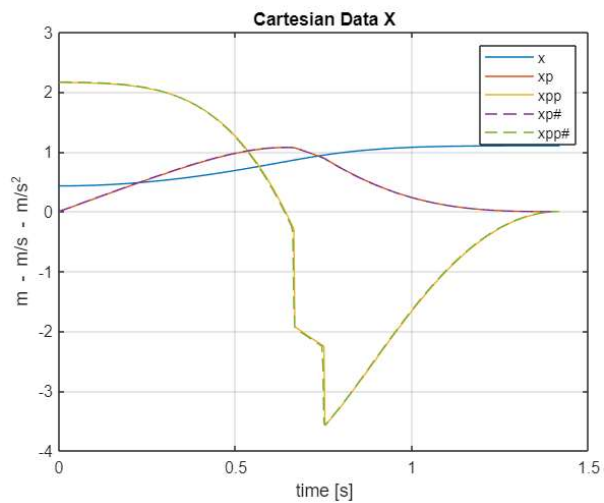


Figura 2.6.1: Spazio cartesiano
fase 3 – asse x

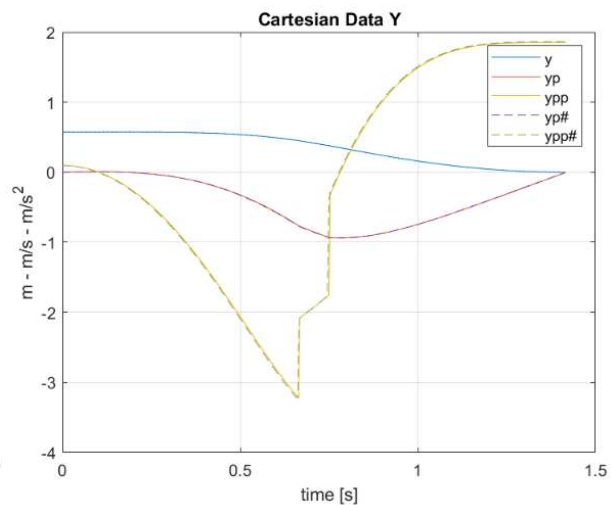


Figura 2.6.2: Spazio cartesiano
fase 3 – asse y

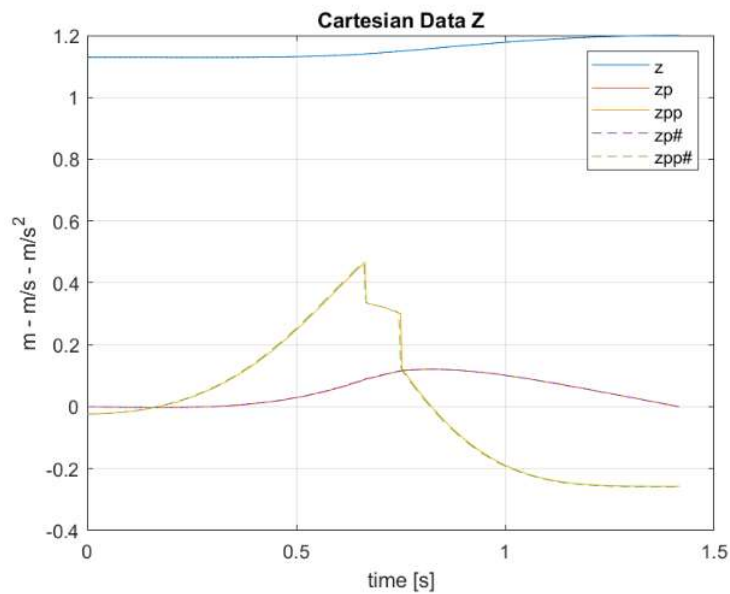


Figura 2.6.3: Moto spazio cartesiano fase 3 – asse z

2.3.5 Tutte le fasi

La figura 2.7 mostra la traiettoria completa del robot durante tutta la fase operativa.

Nel software si trova inoltre una simulazione in tempo reale dei movimenti del robot.

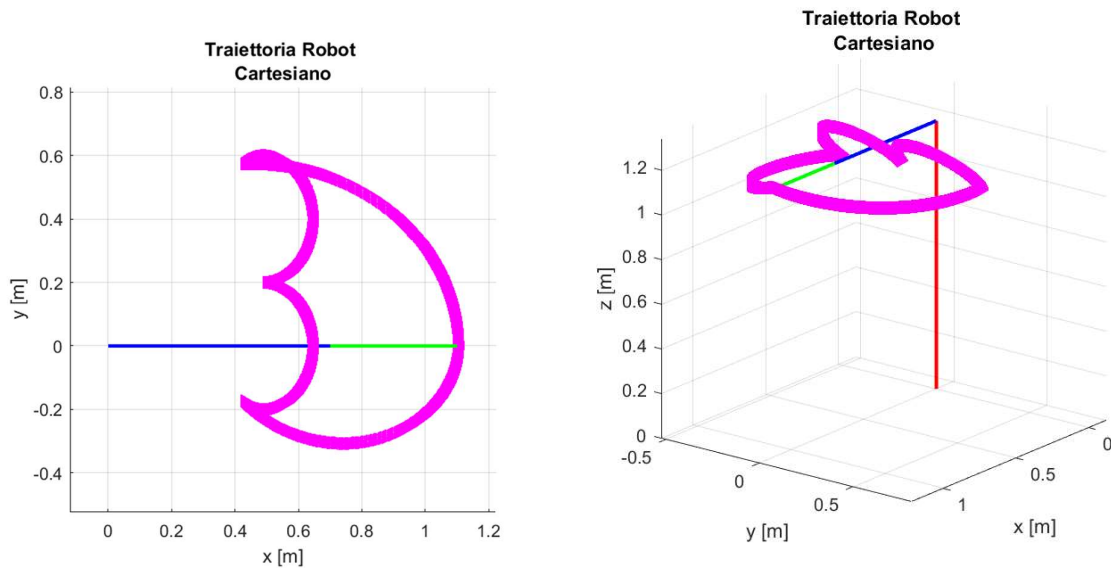


Figura 2.7: Traiettoria, movimento completo

In ultima analisi, nella figura 2.8 vengono mostrati i grafici relativi alla posizione, velocità e accelerazione nello spazio cartesiano di tutte le fasi. Osservando i grafici si può notare in modo chiaro la divisione delle tre fasi col relativo cambio di legge di moto, come ad esempio nel caso della coordinata z , dove la prima e terza fase sono caratterizzate da accelerazioni brusche, mentre la seconda le varia in modo costante e non netto.

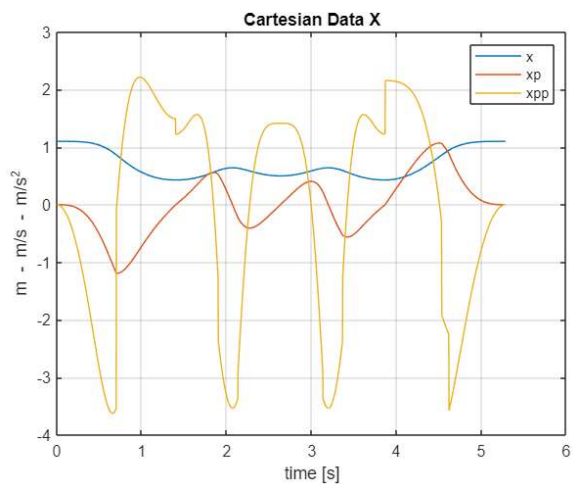


Figura 2.8.1: Spazio cartesiano
fase totale – asse x

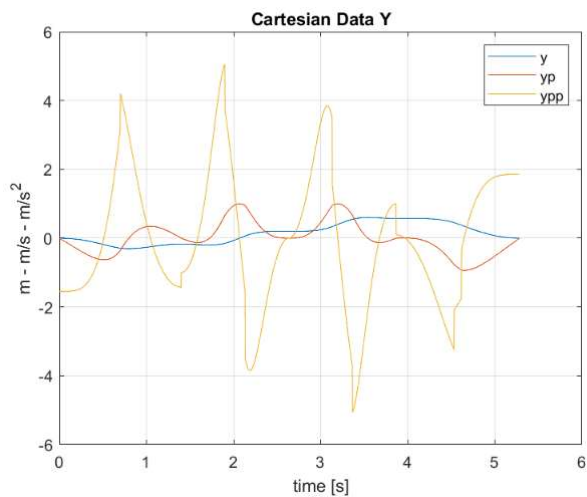


Figura 2.8.2: Spazio cartesiano
fase totale – asse y

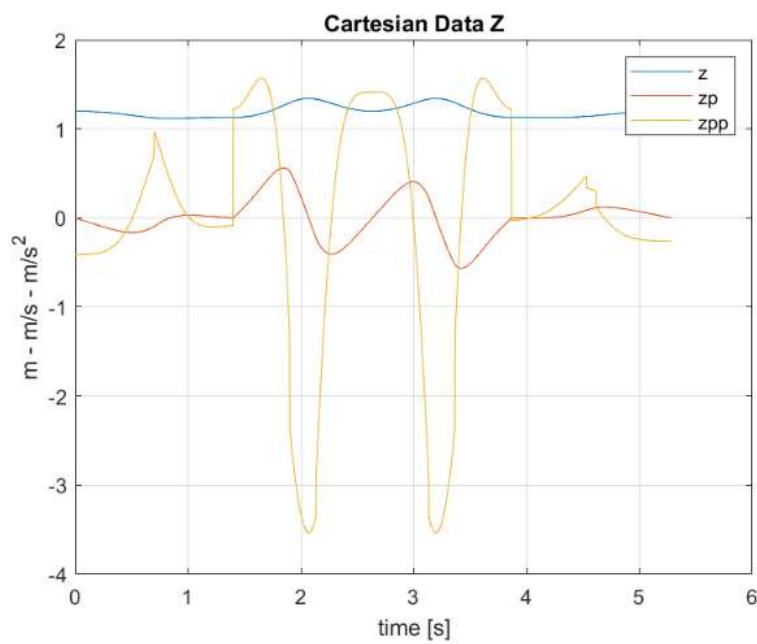


Figura 2.8.3: Moto spazio cartesiano fase totale – asse z

Capitolo 3

Dinamica

Una volta terminato il calcolo della cinematica e verificato che il robot si comportasse come desiderato, si è passati a sviluppare il calcolo delle forze e delle coppie che i giunti devono esprimere per assicurare il movimento.

In questo caso i calcoli sono stati svolti secondo il metodo delle matrici.

3.1 Metodologie di calcolo per la dinamica

3.1.1 Dinamica mediante Matrici 4x4

Per quanto riguarda lo studio della dinamica tramite le matrici tridimensionali, si parte dal link più esterno e, tenendo conto delle forze agenti su di esso, si calcolano quelle scambiate con il precedente. Con una procedura iterativa, si arriva fino alla base.

Le inerzie vengono convertite dalla notazione in J alla notazione in I per poter scrivere la matrice delle pseudo-inerzie:

$$I = \begin{bmatrix} \frac{-J_x + J_y + J_z}{2} & -J_{xy} & -J_{xz} & 0 \\ -J_{xy} & \frac{-J_y + J_x + J_z}{2} & -J_{yz} & 0 \\ -J_{xz} & -J_{yz} & \frac{-J_z + J_x + J_y}{2} & 0 \\ 0 & 0 & 0 & m \end{bmatrix}$$

Le matrici di pseudo-inerzia vengono calcolate rispetto baricentro del link, in seguito devono essere convertite mediante le shift matrixes per ogni link, che spostano l'inerzia vista dal sistema di riferimento baricentrale a uno solidale con il link all'estremità di quest'ultimo:

$$Inn = Mnn * Inn * Mnn'$$

A questo punto ogni matrice deve essere traslata così da esser riferita alla base del robot:

$$Inn_0 = M0n * Inn * M0n'$$

Le azioni esterne che attuano sull'ultimo giunto devono essere anch'esse traslate mediante le shift matrixes per essere considerate correttamente nei calcoli della dinamica, la matrice utilizzata sarà una matrice identità ma con l'ultima colonna pari a quella della matrice di collegamento primo e ultimo link.

$$Fext_0 = M0a * Fext * M0a'$$

Per calcolare il contributo dato dalla forza peso si sfrutta il seguente calcolo tenendo conto del fatto che la matrice di pseudo-inerzia I è espressa rispetto alla base:

$$\Phi_{g/0} = skew(Hg * I_0)$$

Per calcolare invece il contributo dato dalle forze d'inerzia si moltiplicano le accelerazioni di ogni giunto nel sistema di riferimento di base per l'inerzia del link rispetto alla base:

$$\Phi_{in/0} = skew(HH_0 * I_0)$$

Per calcolare la forza scambiata tra ogni link ed il suo precedente si usano le seguenti formule:

- Se è il link più esterno non si tiene conto delle forze scambiate con quello distale (in quanto non esiste) ma di quelle esterne:

$$\Phi_{vinc,k/0} = -\Phi_{in,k} + \Phi_{g,k} + \Phi_{ext/0}$$

- Per tutti gli altri, le forze esterne vengono sostituite dalla forza scambiata con il link distale:

$$\Phi_{vinc,k/0} = -\Phi_{in,k} + \Phi_{g,k} + \Phi_{vinc,k+1/0}$$

Per quanto riguarda il calcolo dell'evoluzione dell'energia cinetica e potenziale durante il moto:

$$E_K = trace(\frac{1}{2} * WW_0 * I_0 * WW_0^T)$$

$$E_P = -trace(H_g * I_0)$$

Con H_g che indica l'accelerazione di gravità in Z.

In ultima analisi si può calcolare la potenza espressa dai motori, dalle forze peso e dalle forze esterne:

$$W_q = -pseudoScalar(\Phi_{vinc,k/0}, WW_{rel/0})$$

$$W_{ext} = pseudoScalar(\Phi_{ext/a}, WW_0)$$

Come verifica del risultato antecedente si è fatto un calcolo considerando la coppia e la velocità angolare del singolo giunto anziché forza e velocità lineare.

Un metodo alternativo di verifica usato per calcolare la potenza è quello di sommare le potenze dei singoli giunti calcolate mediante la seguente formula:

$$wwn = CCn * qp(n)$$

$$WW_{tot} = \sum wwn$$

E successivamente paragonarlo alla derivata temporale numerica dell'energia totale come si potrà vedere nei grafici del prossimo capitolo, ad esempio, la figura 3.5.

3.2 Risultati della dinamica

I risultati vengono mostrati divisi in due parti:

- le fasi 1 e 3 dove il robot si sposta mediante il minimo tempo di attuazione;
- la fase 2 in cui il manipolatore esegue la lavorazione richiesta.

Nei grafici vengono mostrate coppie e le forze che i tre attuatori devono esprimere, l'evoluzione delle energie cinetica, potenziale e totale.

Vengono aggiunti anche i grafici utilizzati per verificare la correttezza di tutti i calcoli: quelli che confrontano la derivata dell'energia cinetica con la potenza totale espressa dagli attuatori, dalle forze esterne e dalle forze peso e quelli che confrontano la derivata dell'energia totale con la potenza degli attuatori e delle forze esterne, non considerando la potenza delle forze peso.

3.2.1 Fasi 1 e 3

Le fasi 1 e 3 sono quelle in cui si richiede che il movimento venga svolto nel minimo tempo di attuazione, le accelerazioni sono conseguentemente molto alte e questo si riflette nei valori delle coppie richieste ai giunti, come si può vedere nella figura 3.1 e nella figura 3.2.

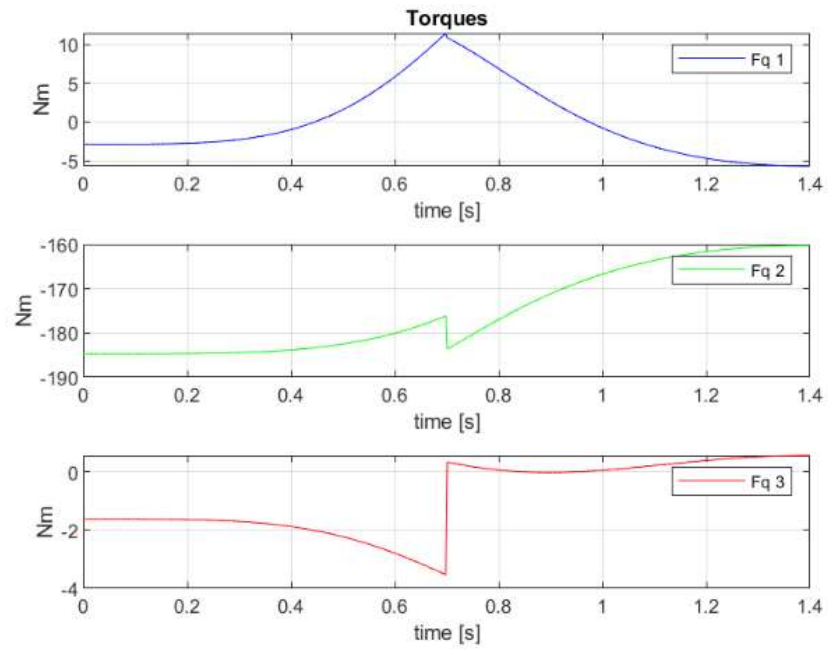


Figura 3.1: Coppie e forze degli attuatori, fase 1

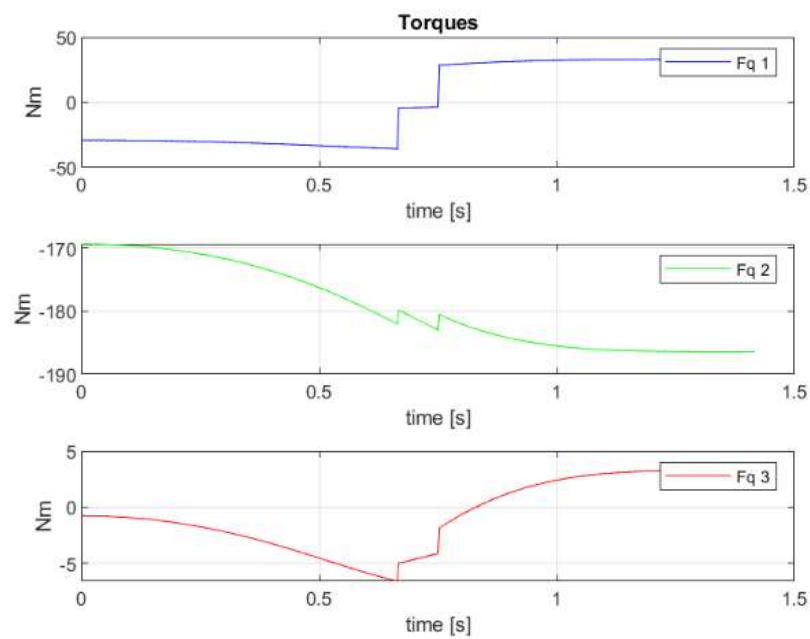


Figura 3.2: Coppie e forze degli attuatori, fase 3

Come si può notare dalla figura 2.7 relativa alla traiettoria totale, per passare dalla posizione di riposo alla posizione di inizio lavorazione, l'end effector del robot dovrà abbassarsi e questo comporterà la diminuzione dell'energia potenziale, osservabile nella figura 3.3. Viceversa si avrà un aumento dell'energia potenziale durante la fase 3 di ritorno alla posizione di riposo.

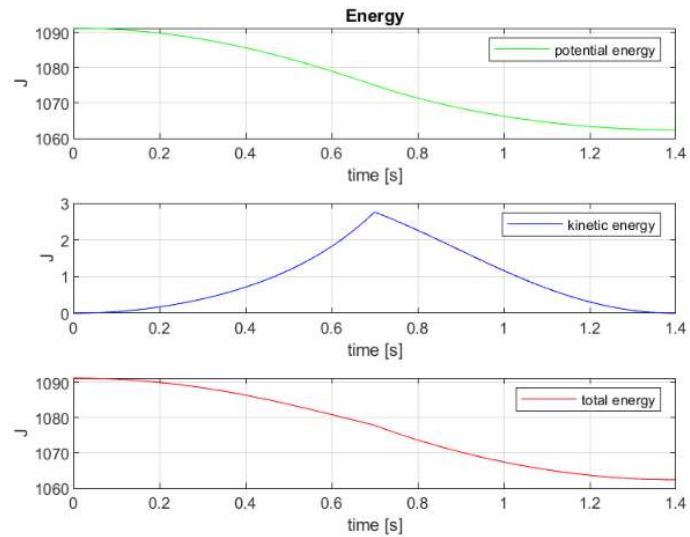


Figura 3.3: Evoluzione temporale delle energie, fase 1

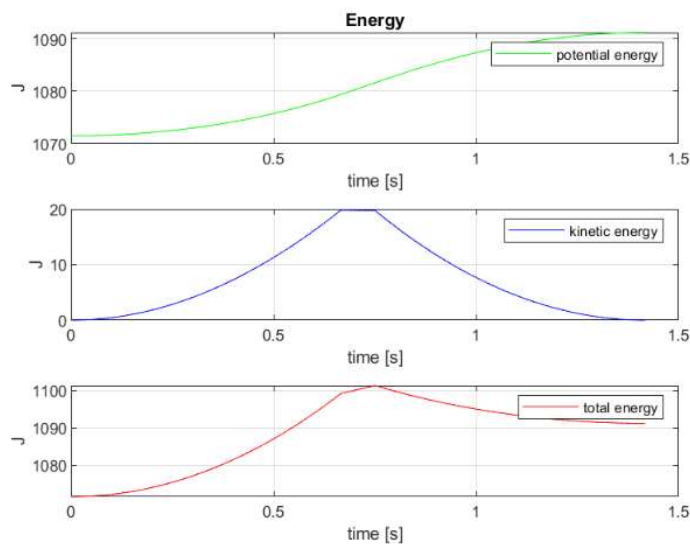


Figura 3.4: Evoluzione temporale delle energie, fase 3

Le due figure 3.5 e 3.6 confrontano la derivata dell'energia totale rispetto al tempo indicata come "Theoric power", potenza calcolata usando forze e velocità lineari indicata come "Real power", e potenza calcolata usando coppie e velocità angolari dei giunti indicata come "Real power 2".

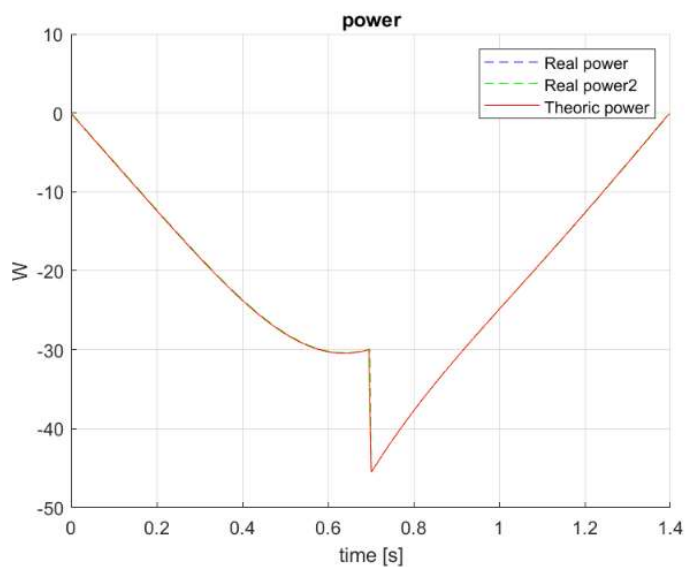


Figura 3.5: Derivata dell'energia totale e potenza totale, fase 1

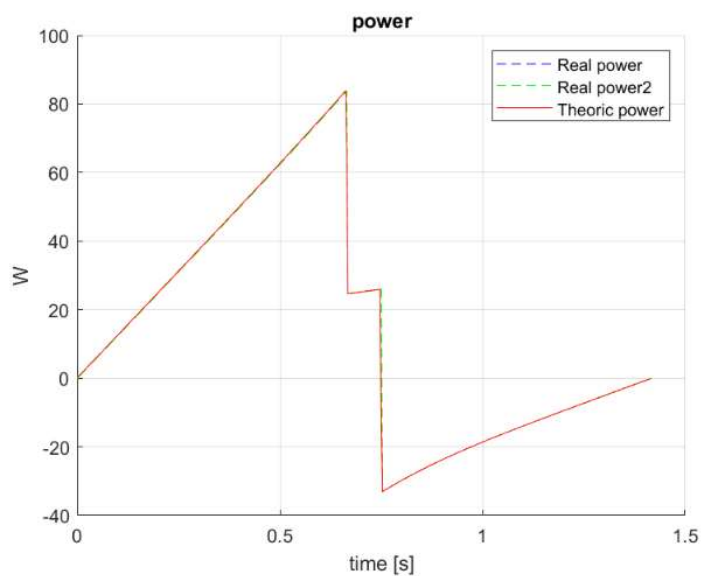


Figura 3.6: Derivata dell'energia totale e potenza totale, fase 3

Le seguenti figure 3.7 e 3.8 confrontano invece tutte le potenze calcolate per ogni giunto, così da poter valutare quali giunti vengono più sollecitati durante il movimento.

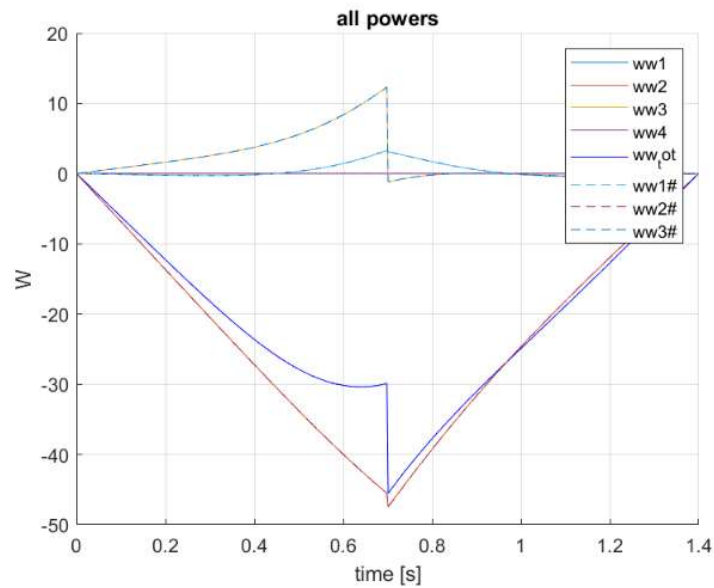


Figura 3.7: Confronto potenze, fase 1

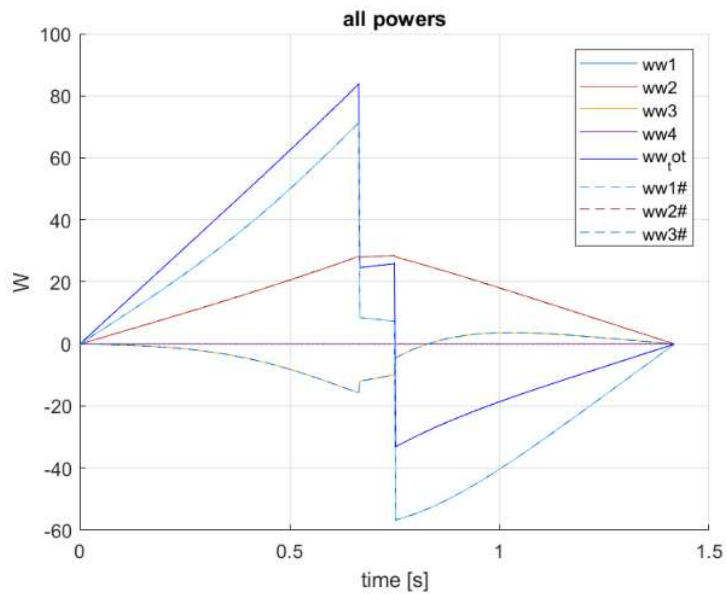


Figura 3.8: Confronto potenze, fase 3

3.2.2 Fase 2

Confrontando le forze e le coppie espresse dai giunti per le fasi in cui non viene richiesto il minimo tempo di attuazione, mostrate nella figura 3.9, con quelle precedenti, figure 3.1 e 3.2, si nota che nonostante il movimento sia più lento e fluido per la differente legge di moto, a causa della forma del simbolo abbiamo un maggiore range in cui variano le coppie, come ad esempio per la coppia f_{q2} .

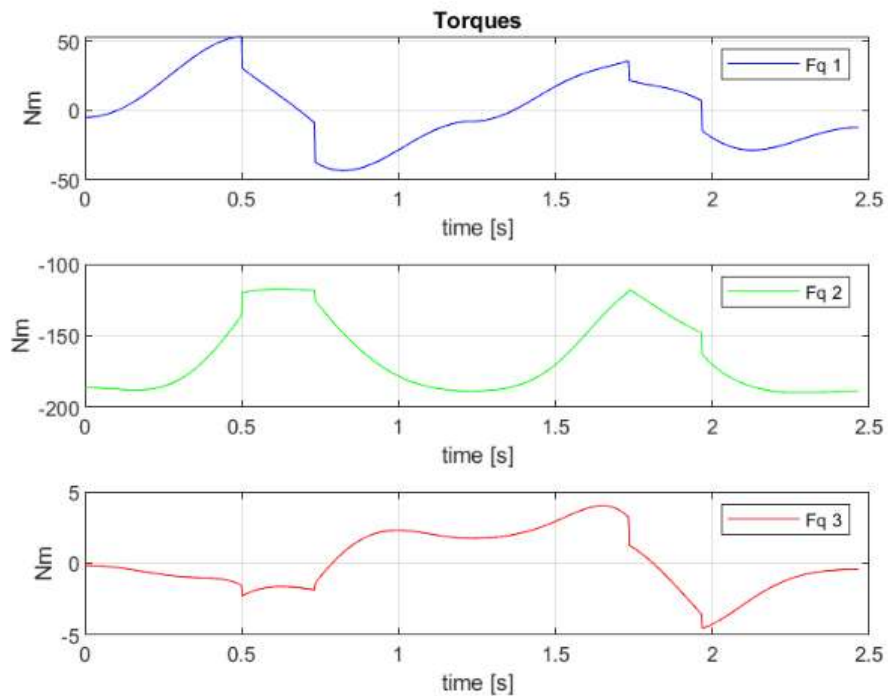


Figura 3.9: Coppie e forze degli attuatori, fase 2

Per quanto riguarda l'evoluzione dell'energia potenziale, osservando la figura 3.10 si può notare come il robot si alzi e abbassi più volte a causa dell'orientamento del simbolo nello spazio, variando così la sua quota.

Guardando invece il grafico dell'energia cinetica, questa è maggiore nelle fasi paraboliche, dove il manipolatore riesce a raggiungere le massime velocità per questa lavorazione, mentre possiamo apprezzare la riduzione, ma non annullamento, della velocità nella fase critica di inversione del moto, dove si congiunge la parte alta e bassa del simbolo.

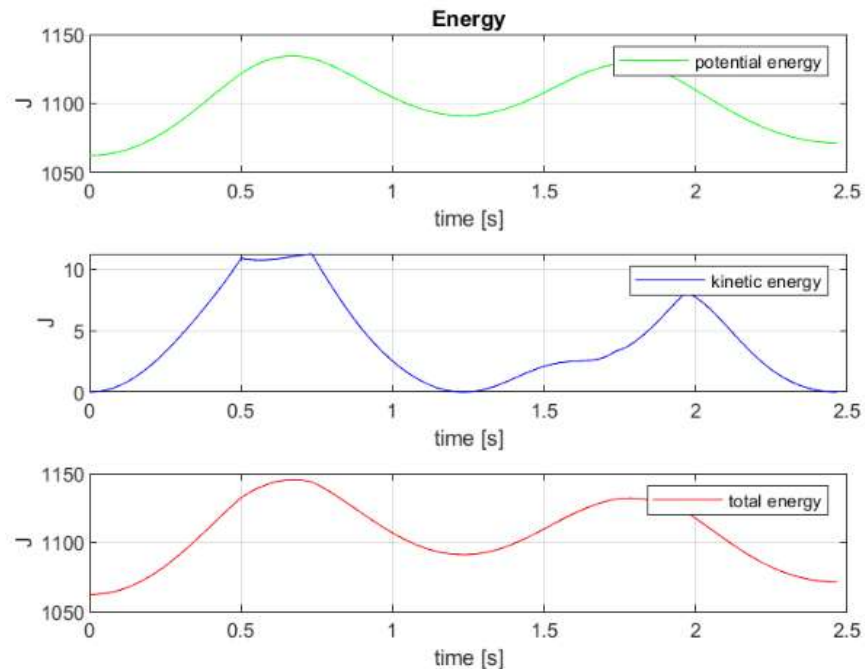


Figura 3.10: Evoluzione temporale delle energie, fase 2

Anche in questo caso la figura 3.11, sotto riportate, confronta la derivata dell'energia totale rispetto al tempo, con la potenza totale reale fornita dagli attuatori.

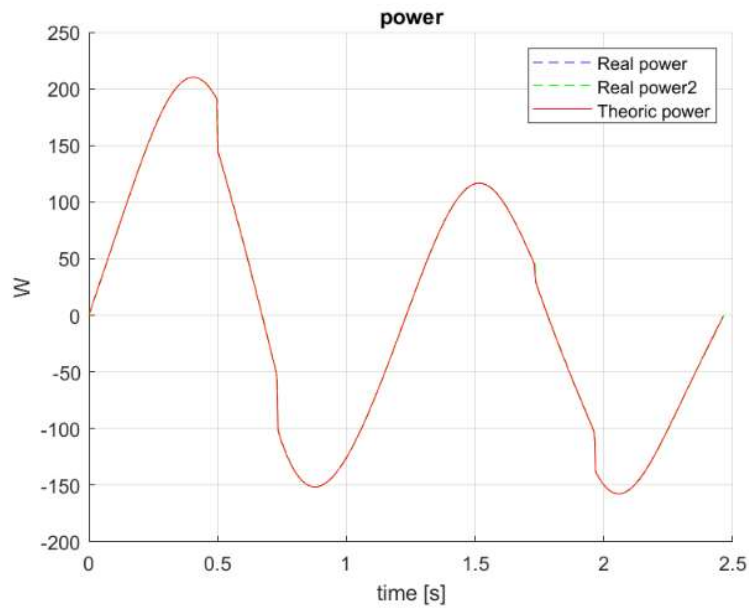


Figura 3.11: Derivata dell'energia totale e potenza totale, fase 2

L'ultima figura proposta confronta invece tutte le potenze calcolate per ogni giunto, così da poter valutare quali giunti vengono più sollecitati durante il movimento.

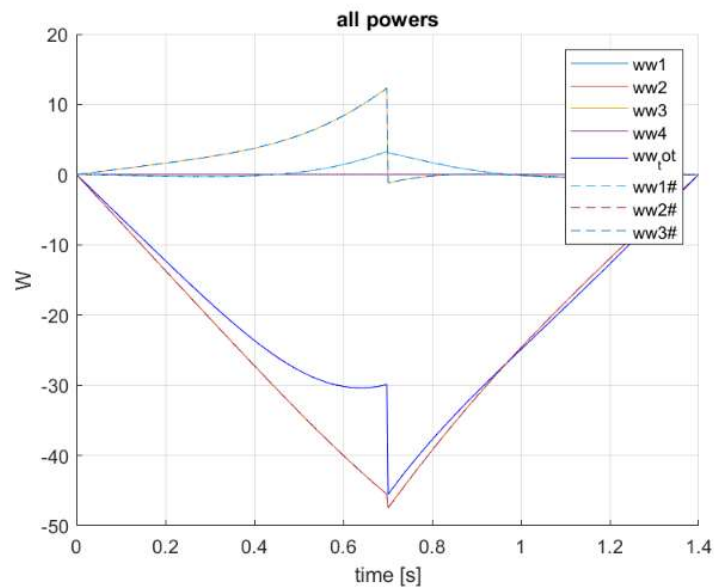


Figura 3.12: Confronto potenze, fase 2

Capitolo 4

Sviluppo In SimScape

Terminata la parte di sviluppo dei calcoli dinamici e cinematici, si è passati alla modellizzazione del robot in ambiente SimScape.

4.1 Modello

La figura 4.1 mostra lo schema utilizzato per realizzare il modello del primo giunto.

È stato realizzato un modello per ogni giunto, partendo dalla stessa base logica e modificando solo i relativi dati per link.

Come possiamo notare dalla figura sotto riportata, possiamo dividere lo schema in più zone:

- Giunto
- Controller

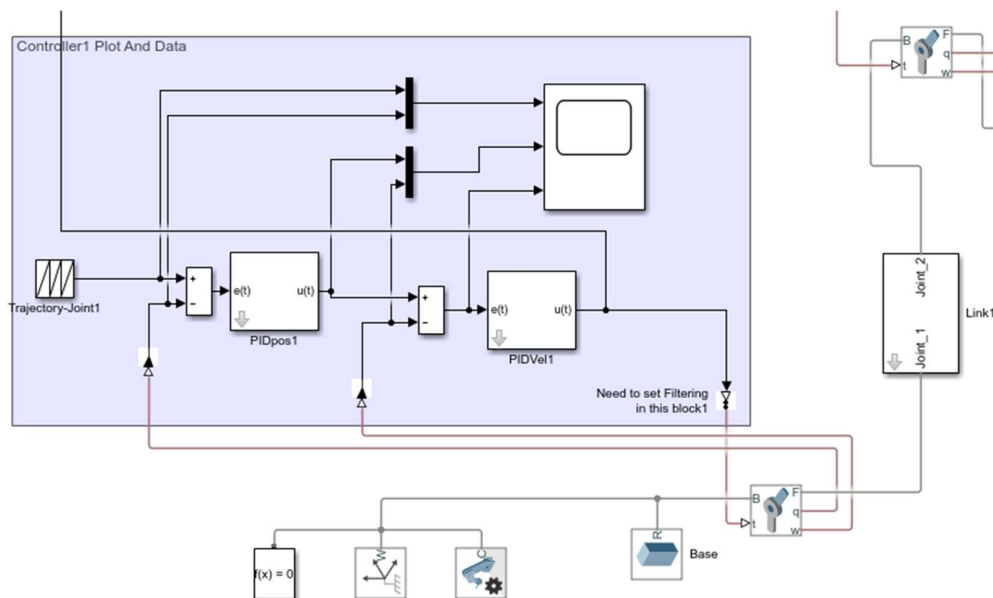


Figura 4.1: Schema del manipolatore in SimScape, giunto 1

Il controller è composto da più parti, si incomincia dal blocco denominato "Trajectory joint 1", questo blocco contiene i dati della traiettoria in relazione al tempo nello spazio dei giunti.

Si passa in seguito ai PID dedicati al controllo di posizione e velocità, "PIDPos1" e "PIDVel1".

E in fine abbiamo uno scope per monitorare i dati durante il funzionamento.

Oltre a questa zona, troviamo il blocco "link" che serve a definire i dati meccanici per ogni link, come la lunghezza, densità e raggio, avendo ipotizzato link cilindrici.

Il controller successivo partirà proprio da quest'ultimo blocco come base di partenza, questo processo si ripete per ogni link costituente il manipolatore.

Per paragonare le performance del controllo elaborato mediante SimScape col software Matlab, esportiamo i dati relativi alle coppie dei giunti e quelli relativi alla posizione, velocità e accelerazione.

In figura 4.2 viene mostrato il blocco "GripperMotionPlotSave" che serve per poter leggere informazioni di posizione, velocità, accelerazioni nelle direzioni X, Y, Z della parte terminale dell'ultimo giunto.

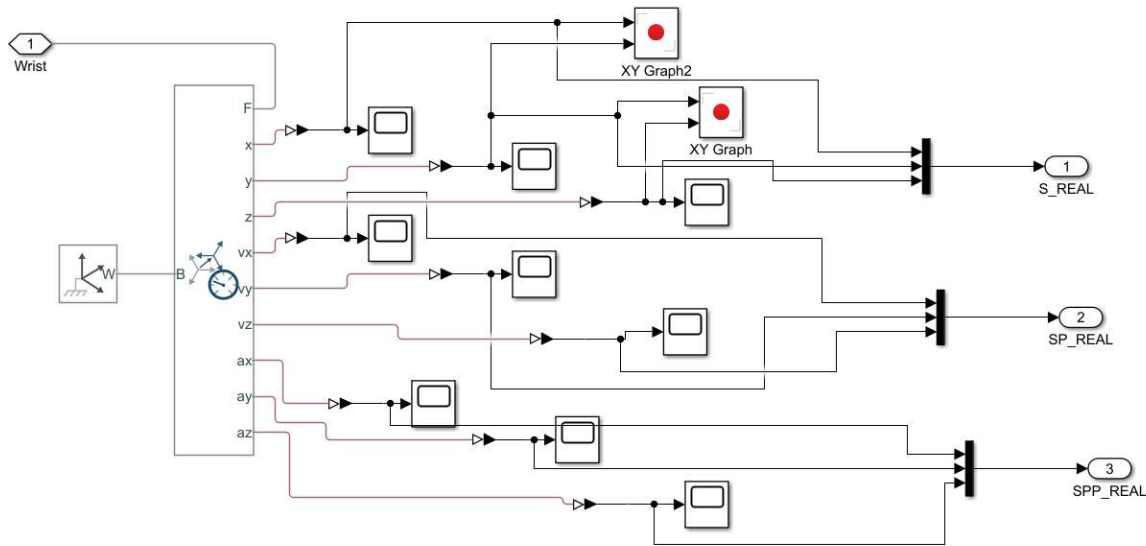


Figura 4.2: Blocco per leggere ed esportare dati cinematici

4.2 Verifica dei calcoli cinematici e dinamici e modello sviluppato

Per poter verificare la correttezza del modello in SimScape e la coerenza tra i calcoli offerti da questo software e quelli cinematici e dinamici sviluppati in MatLab e descritti nei capitoli 2 e 3, si è provato ad attuare i giunti fornendo l'evoluzione temporale di α , β e γ verificando che le coppie e le forze espresse dai giunti per assicurare il moto richiesto coincidessero con quelle precedentemente calcolate ed illustrate nel paragrafo 3.2.

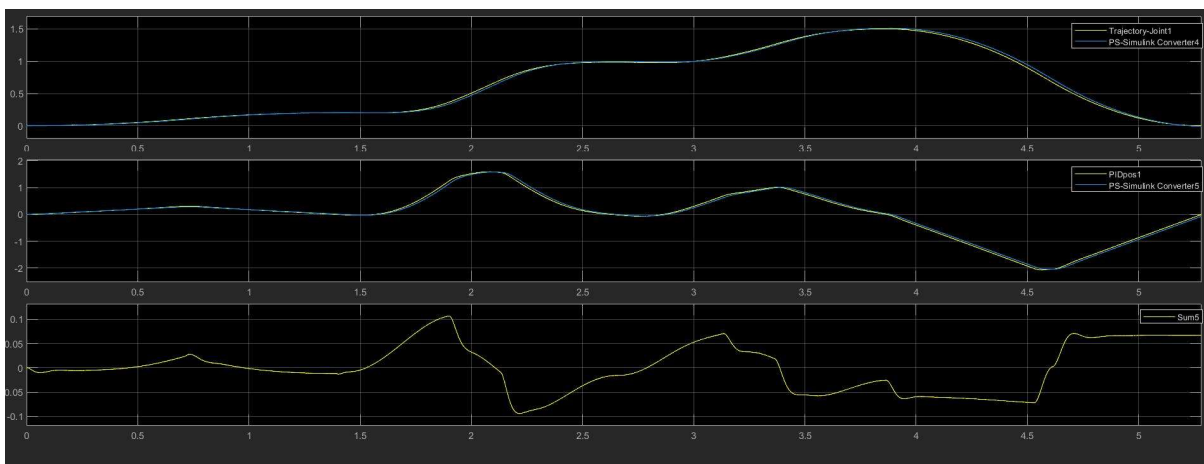


Figura 4.3: Confronto giunto 1 MatLab - SimScape

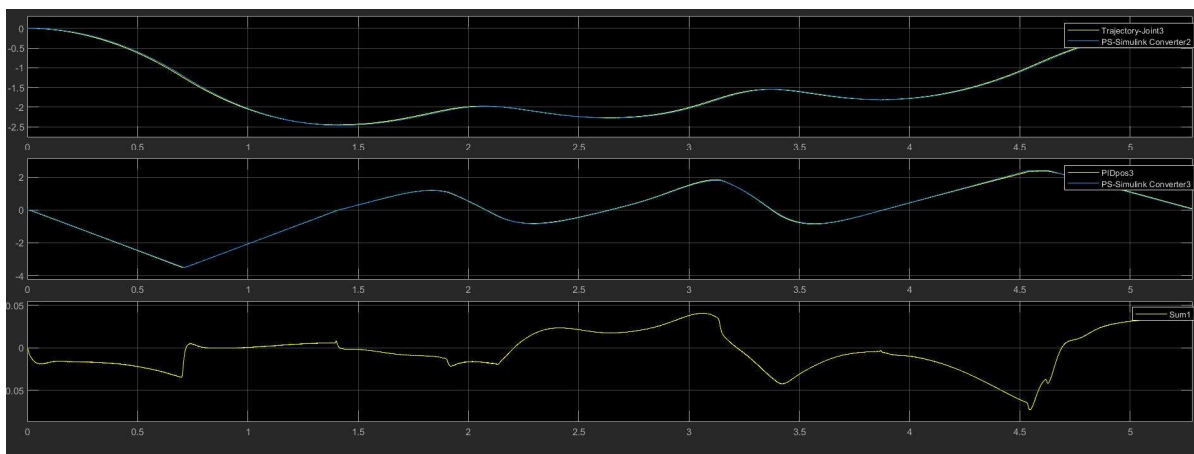


Figura 4.4: Confronto giunto 3 MatLab - SimScape

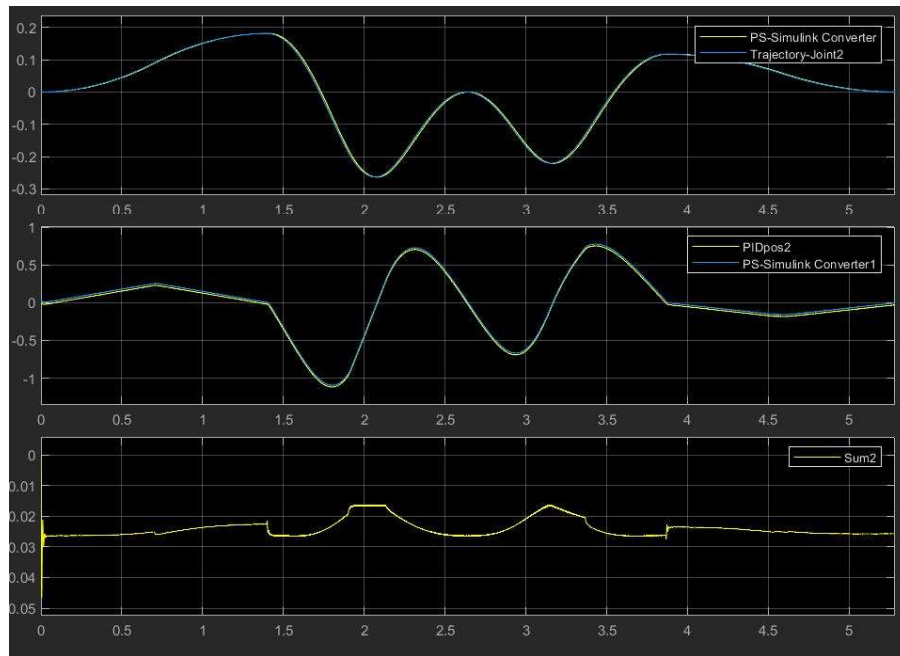


Figura 4.5: Confronto giunto 2 MatLab - SimScape

Le figure 4.6, 4.7 e 4.8 dimostrano la correttezza dei calcoli e del processo di modellizzazione, con una leggera traslazione dei dati forniti dal controllore SimScape, dovuta al diverso tempo di computazione e a un maggior numero di dati ricavati.

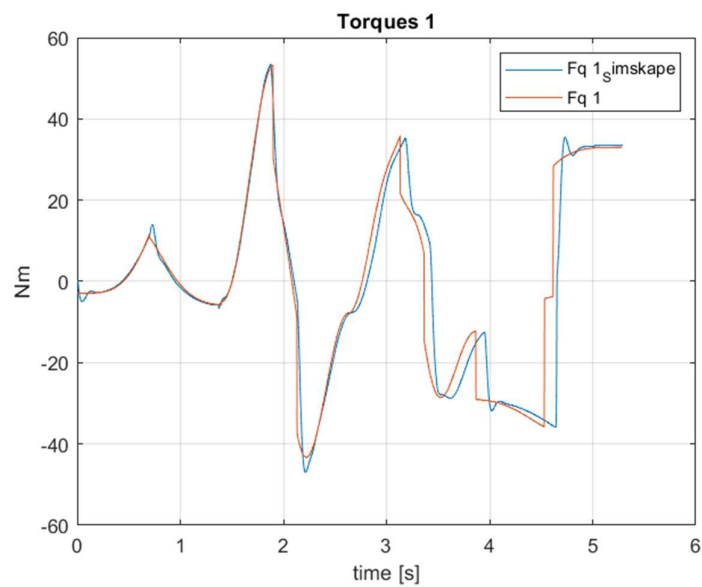


Figura 4.6: Confronto coppia 1 MatLab - SimScape

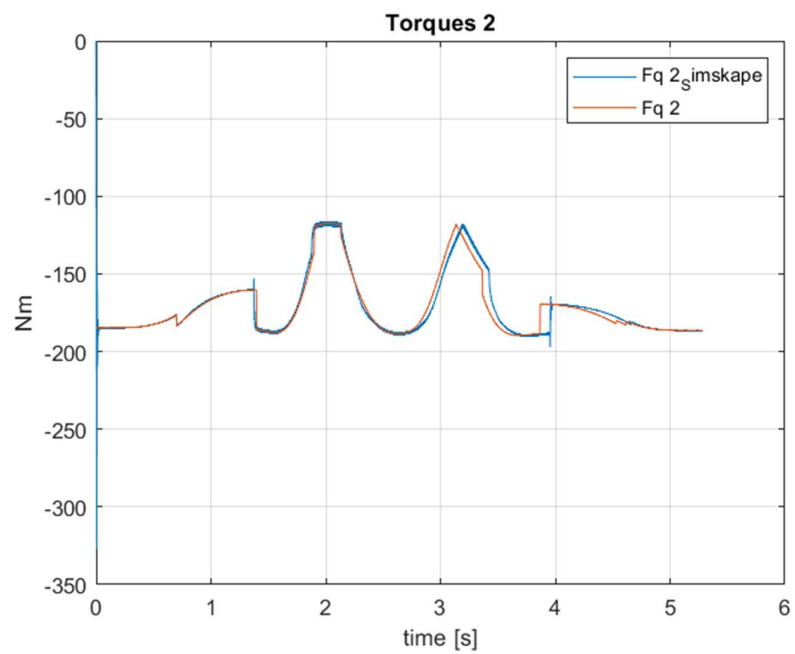


Figura 4.7: Confronto coppia 2 MatLab - SimScape

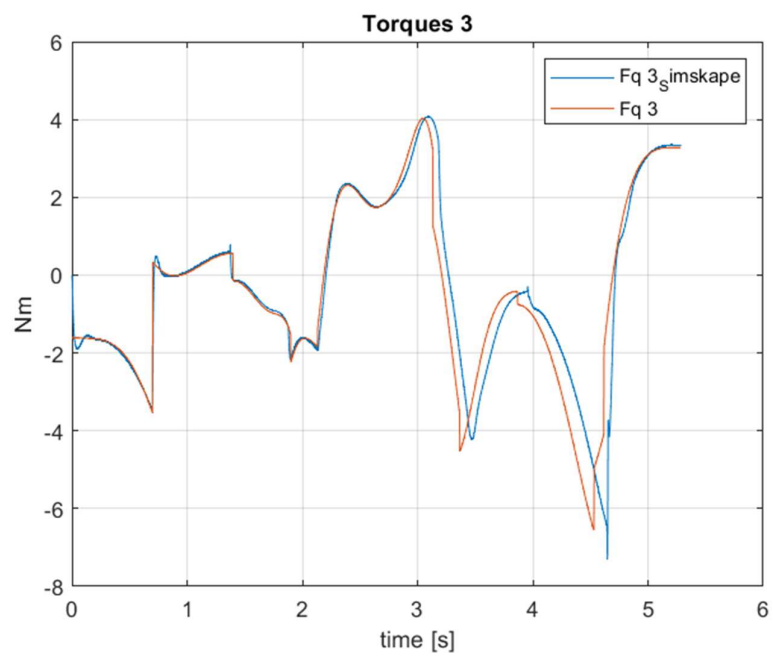


Figura 4.8: Confronto coppia 3 MatLab - SimScape

Capitolo 5

Controllo

Una volta verificata la correttezza del modello in SimScape si è creata una strategia di controllo per comandare i giunti.

Ogni giunto ha il suo controllore, si è adottato quindi uno schema di controllo di tipo decentralizzato.

5.1 Il controllo in cascata per posizione e velocità

Il collegamento del controllore si può osservare dalla figura 4.1. Il controllore riceve in input la traiettoria di posizione, per q_1 è quindi il valore di α e due variabili di processo: la posizione (angolare) e la velocità (angolare) del giunto.

Tramite queste due variabili, come si vede nella figura 5.1, si crea un controllo in cascata: un primo PID genera una variabile di controllo calcolato sull'errore di posizione. Il secondo PID riceve la variabile di controllo appena calcolata e, tramite un confronto con la velocità del giunto, calcola la coppia che deve essere generata dall'attuatore. I vantaggi di un controllo in cascata sono molteplici, tra questi: un aumento delle prestazioni, la possibilità di compiere il tuning del loop interno in maniera aggressiva per eliminare i disturbi ed infine la possibilità di assicurarsi un errore di posizione a regime permanente nullo usando la costante di tempo integrale nel loop esterno.

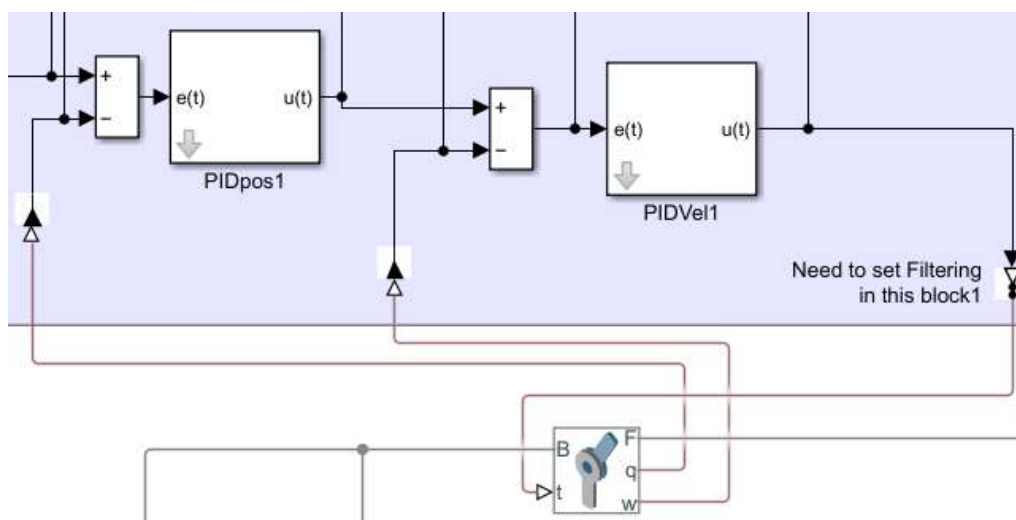


Figura 5.1: Controllori utilizzati nello schema SimScape

La tabella 5.1 mostra i parametri di ogni PID. È interessante notare che i controllori per i loop interni, quelli segnati con "vel" sono in linea generale più aggressivi di quelli per i loop esterni, questo perché i loop interni sono ottimizzati per la reiezione dei disturbi sul carico, quindi, richiedono una maggiore aggressività per compensare questi disturbi nel minor tempo possibile.

I loop esterni invece sono meno rigidi dato che hanno il compito di garantire il corretto trajectory following riducendo al minimo vibrazioni e oscillazioni.

Si è deciso quindi di spegnere completamente l'azione derivativa in ogni controllore così per ridurre i rumori presenti.

Parametri PID			
Controllore	Kp	Ki	Kd
q1 - pos	40	20	0
q1 - vel	500	0	0
q2 - pos	100	25	0
q2 - vel	7000	50	0
q3 - pos	80	40	0
q3 - vel	100	0	0

Tabella 5.1: Parametri dei controllori

Conclusioni

Il lavoro svolto è iniziato con l'analisi del robot e del compito assegnato. Sono stati definiti i differenti parametri geometrici e fisici e i vari limiti. Alcuni di essi sono stati tratti dalla documentazione fornita, altri sono stati scelti in maniera verosimile.

Si è quindi passati allo svolgimento dei calcoli per la cinematica e la dinamica, verificandone poi la coerenza con il modello SimScape.

Per la verifica delle coordinate cartesiane di posizione, velocità e accelerazione, si è optato per un ricalcolo mediante la cinematica diretta partendo dai dati forniti da SimScape.

Possibili sviluppi futuri che potrebbero essere considerati sono i seguenti:

- Si potrebbe optare per una differente tecnica di tuning, concentrandosi sull'ottimizzazione nella prima e nell'ultima fase, migliorando così la risposta del robot.
- Le oscillazioni delle coppie e delle forze degli attuatori, per la fase di avvicinamento alla posizione di inizio lavorazione nel tempo minimo (fase 1), potrebbero non sempre essere accettabili. Si potrebbe quindi riconsiderare il concetto di minimo tempo di attuazione per quella fase: aumentando il tempo di attuazione, i controllori potrebbero essere settati in maniera meno aggressiva.
- Il robot, come illustrato nella sezione 1.2.2, presenta limiti di posizione negli attuatori. Tutte le traiettorie considerate fino ad ora fanno in modo da evitare situazioni in cui i limiti possano dare problemi ma nessun algoritmo software è stato implementato per evitare che ciò accada. Se si dovesse scegliere di muovere il robot facendolo passare per un punto non raggiungibile, i giunti si bloccherebbero.

Una possibile soluzione potrebbe essere una procedura di verifica della traiettoria dal punto iniziale (a) al punto finale (b), se la traiettoria dovesse passare per una zona non raggiungibile, verrebbe aggiunto un punto intermedio (c) all'interno del workspace. Si verificherebbe poi la fattibilità della traiettoria da (a) a (c) e poi da (c) a (b), aggiungendo eventualmente altri punti intermedi.

- Si potrebbe approfondire lo studio relativo allo sfasamento temporale fra i dati forniti da MatLab e quelli di SimScape.