

Controllo di un manipolatore seriale RRRRRR completamente attuato



Livio Bisogni

Anno Accademico 2021-2022



UNIVERSITÀ DI PISA

SCUOLA DI INGEGNERIA

DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

Corso di Laurea Magistrale
in Ingegneria Robotica e
dell'Automazione

CONTROLLO DEI ROBOT

Overview

- Descrizione del manipolatore
- Controllo cinematico
 - Sintesi
 - Simulazione (Metodo del gradiente — Controllo di postura)
 - Simulazione (Metodo di Levenberg-Marquardt — Controllo di postura)
 - Simulazione (Metodo di Levenberg-Marquardt — Inseguimento di traiettoria circolare)
- Controllo dinamico
- Controllore PD
 - Sintesi
 - Simulazione (Pick and place)
- Controllore computed torque
 - Sintesi
 - Simulazione (Pick and place)
 - Simulazione (Inseguimento di traiettoria circolare)

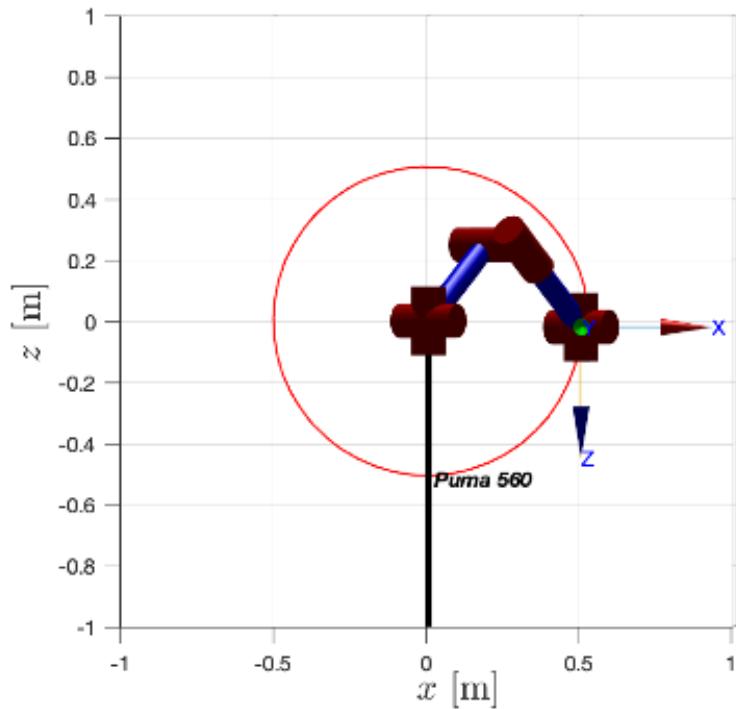
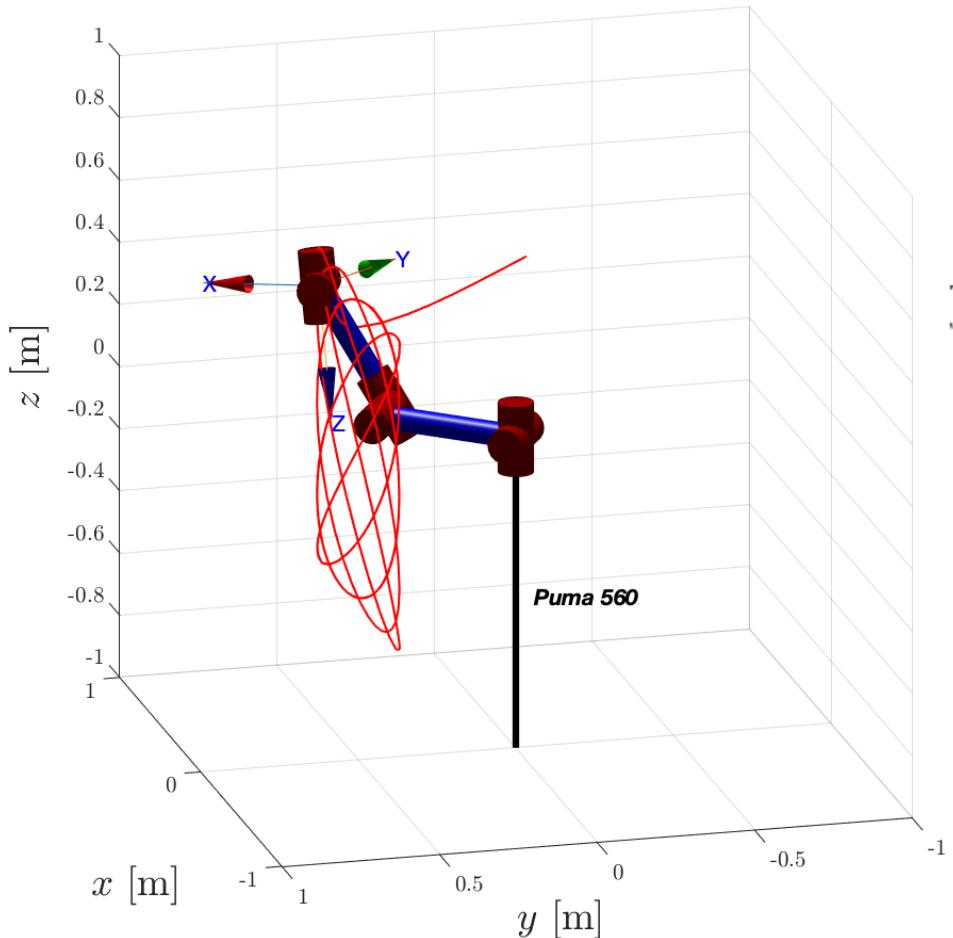
Overview

- Controllore backstepping
 - Sintesi
 - Simulazione (Pick and place)
 - Simulazione (Inseguimento di traiettoria circolare)
- Controllo adattivo
- Controllore computed torque adattivo
 - Sintesi
 - Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)
 - Stima dei parametri
- Controllore backstepping adattivo
 - Sintesi
 - Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)
 - Stima dei parametri

Descrizione del manipolatore

- Manipolatore a 6 giunti rotoidali Puma 560

P. I. Corke and B. Armstrong-Helouvry, "A search for consensus among model parameters reported for the PUMA 560 robot" Proceedings of the 1994 IEEE International Conference on Robotics and Automation, 1994, pp. 1608-1613 vol.2, doi: 10.1109/ROBOT.1994.351360.



Descrizione del manipolatore

- Tabella di Denavit-Hartenberg

Link_i	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	0	q_1
2	0.4318	0	0	q_2
3	0.0203	$-\pi/2$	0.15005	q_3
4	0	$\pi/2$	0.4318	q_4
5	0	$-\pi/2$	0	q_5
6	0	0	0	q_6

Controllo cinematico

Sintesi

- I metodi di controllo cinematico sfruttano il fatto di poter scegliere il valore della variabile u in:

$$\dot{q} = u$$

- Il task è di posizionare l'*end effector* nella posizione e orientazione desiderata:
 - ξ_d (controllo di postura)
 - $\xi_d(t)$ (inseguimento di traiettoria)
- La dimensione del task è pari al numero di giunti del manipolatore ($m = 6 = n$)
- Lo scopo è annullare, rispettivamente:
 - l'errore di posizionamento: $e(t) = \xi_d - \xi(t)$ (controllo di postura)
 - l'errore di inseguimento: $e(t) = \xi_d(t) - \xi(t)$ (inseguimento di traiettoria)
- L'approccio del metodo di Newton-Raphson ha lo svantaggio di essere instabile nelle configurazioni che rendono singolare il jacobiano J , per cui non verrà usato:

$$u = J^{-1}e$$

Controllo cinematico

Sintesi

Approcci alternativi (numericamente stabili) qui usati sono:

- Il metodo di discesa del gradiente (*steepest descent*), per il controllo di postura:

$$u = KJ^T e$$

dove K è una matrice definita positiva.

- Il metodo di Levenberg-Marquadt (pseudoinversa smorzata), per
 - il controllo di postura:

$$u = (J^T J + \mu_0 \text{diag}(J^T J))^{-1} J^T e$$

- e l'inseguimento di movimento:

$$u = (J^T J + \mu_0 \text{diag}(J^T J))^{-1} J^T (\dot{\xi}_d + K e)$$

dove K è una matrice definita positiva (ad es. diagonale, per semplicità) e $\mu_0 > 0$ (il caso $\mu_0 = 0$ degenera nel metodo della pseudoinversa).

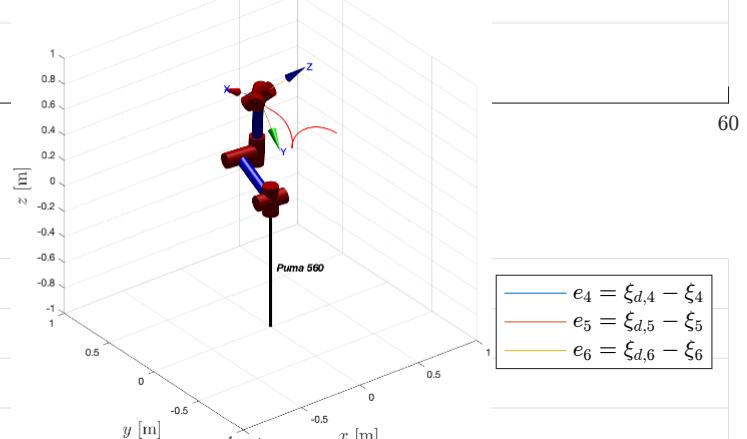
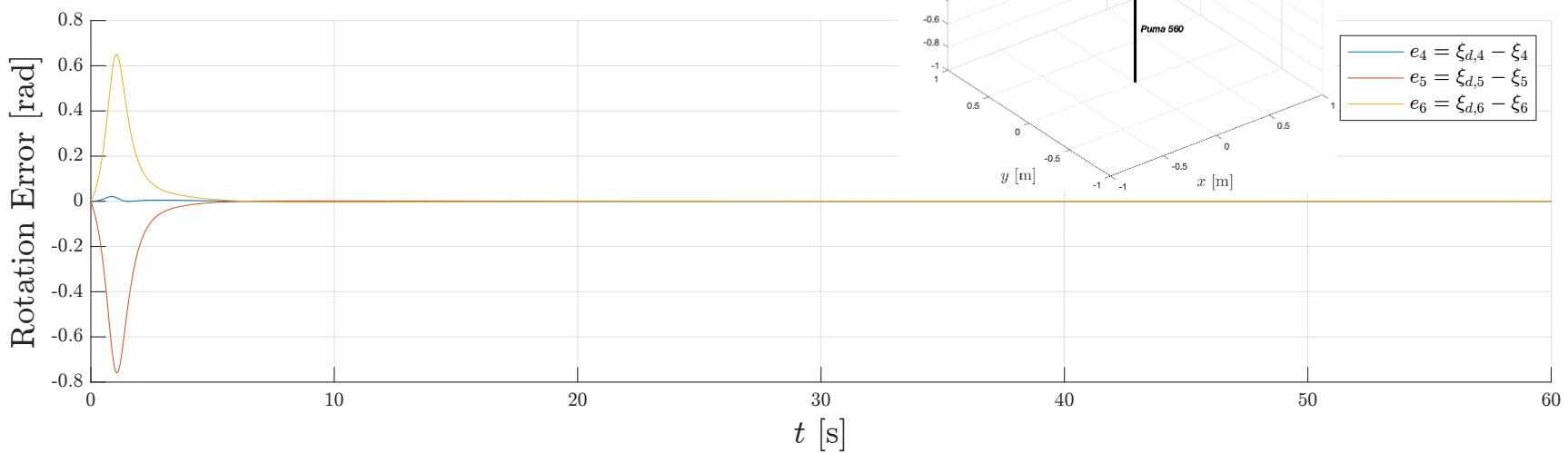
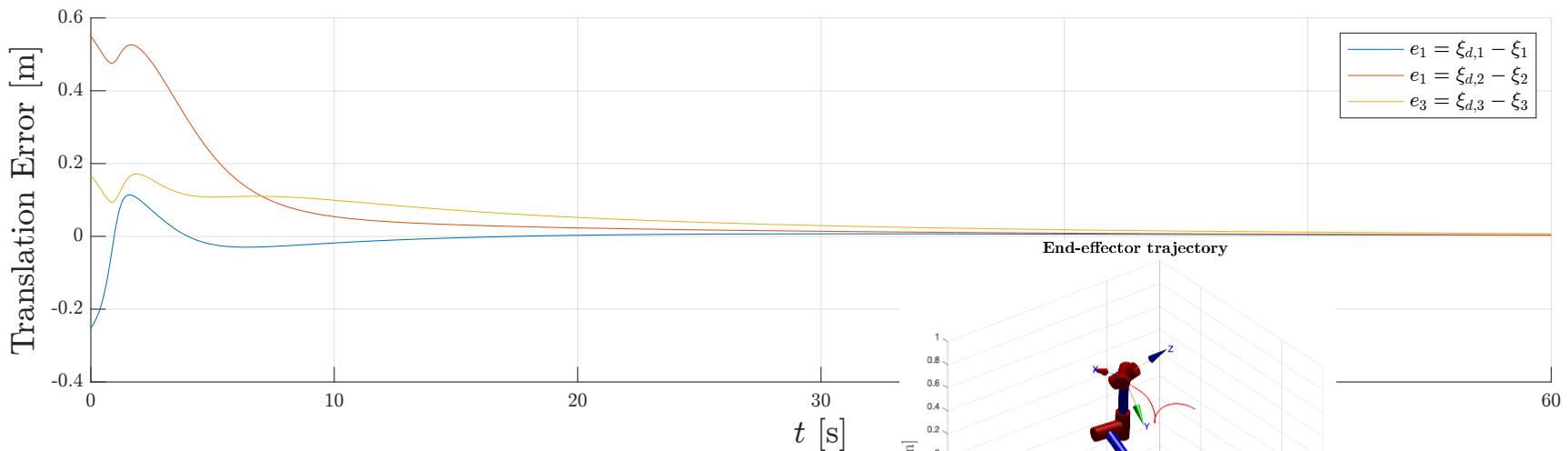
Controllo cinematico — Metodo del gradiente

Simulazione (*Controllo di postura*)

Configurazione desiderata:

$$\xi_d = [0.2 \quad 0.4 \quad 0.6 \quad 0 \quad 0 \quad 0]^T$$

End-effector error



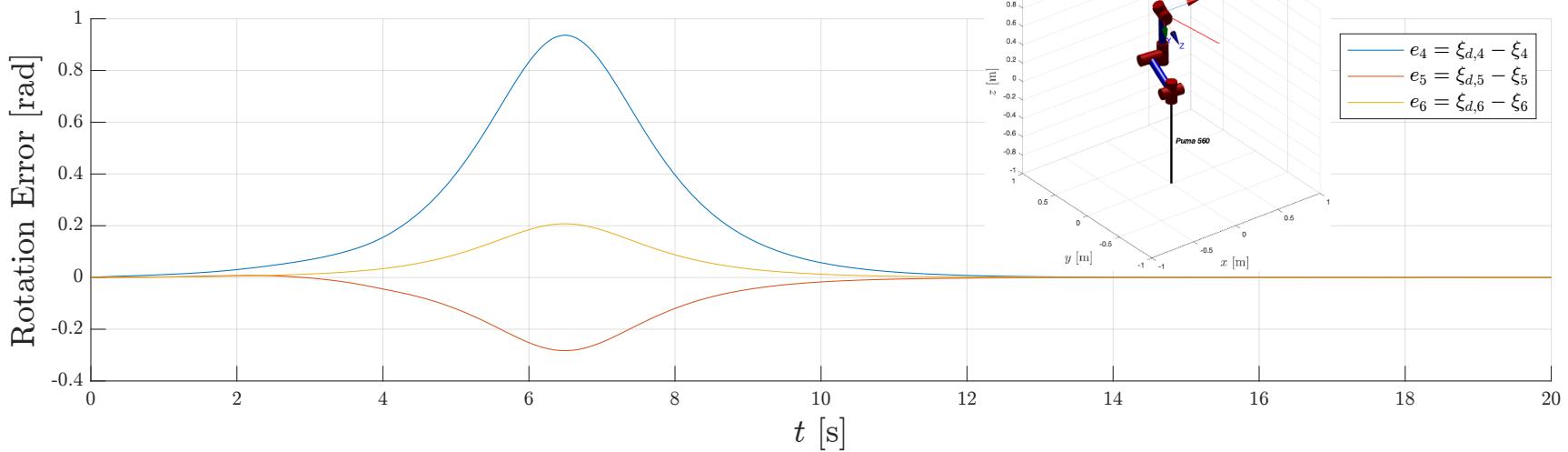
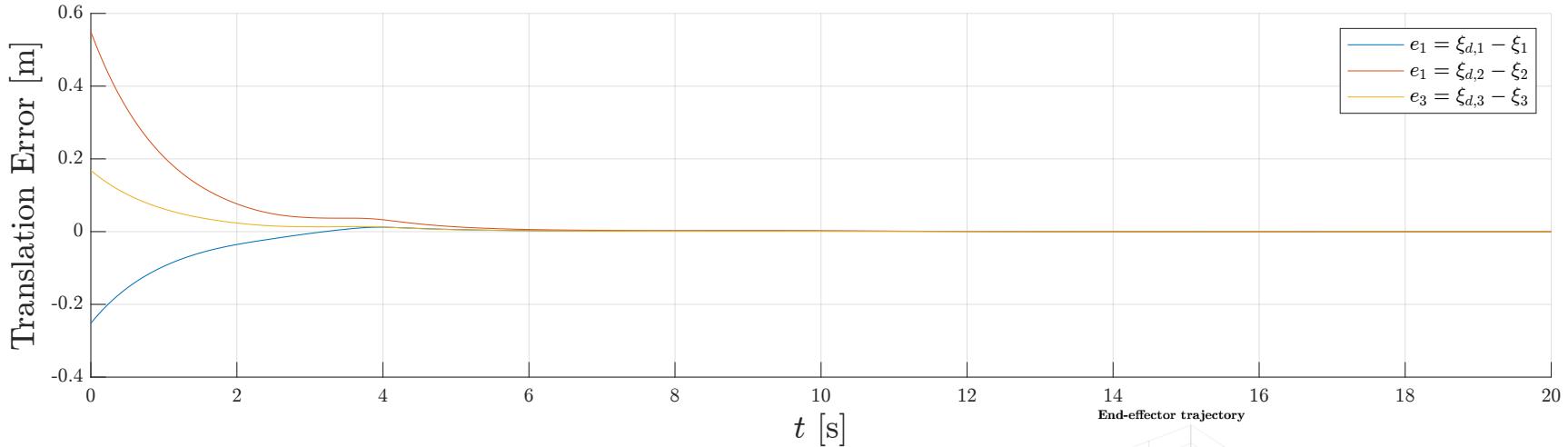
Controllo cinematico — Metodo di Levenberg-Marquardt

Simulazione (Controllo di postura)

Configurazione desiderata:

$$\xi_d = [0.2 \quad 0.4 \quad 0.6 \quad 0 \quad 0 \quad 0]^T$$

End-effector error



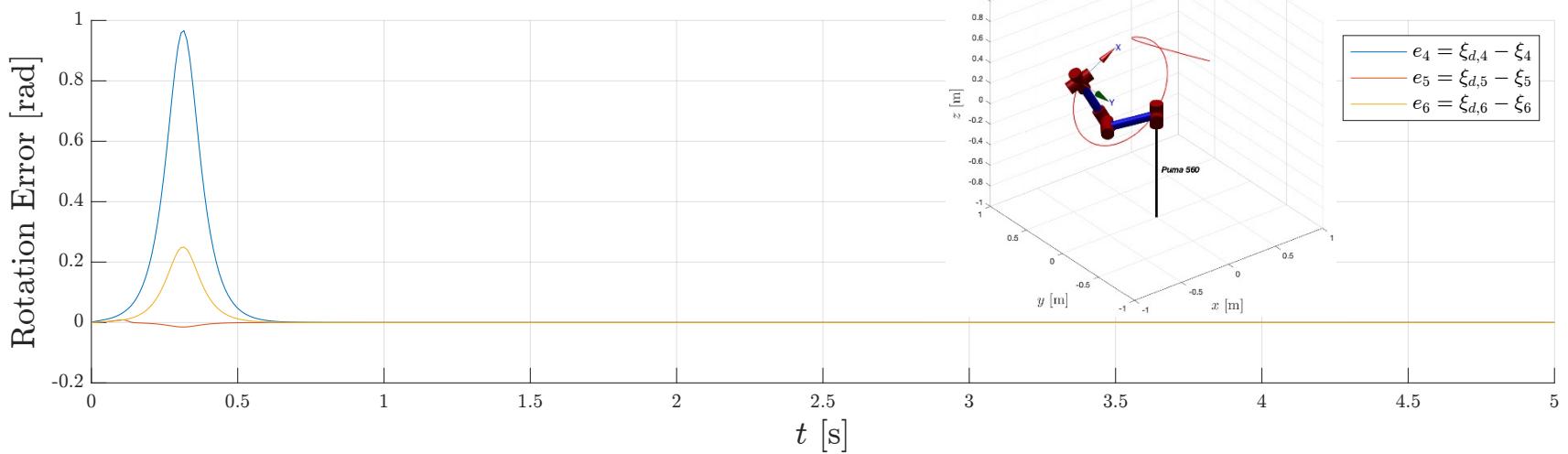
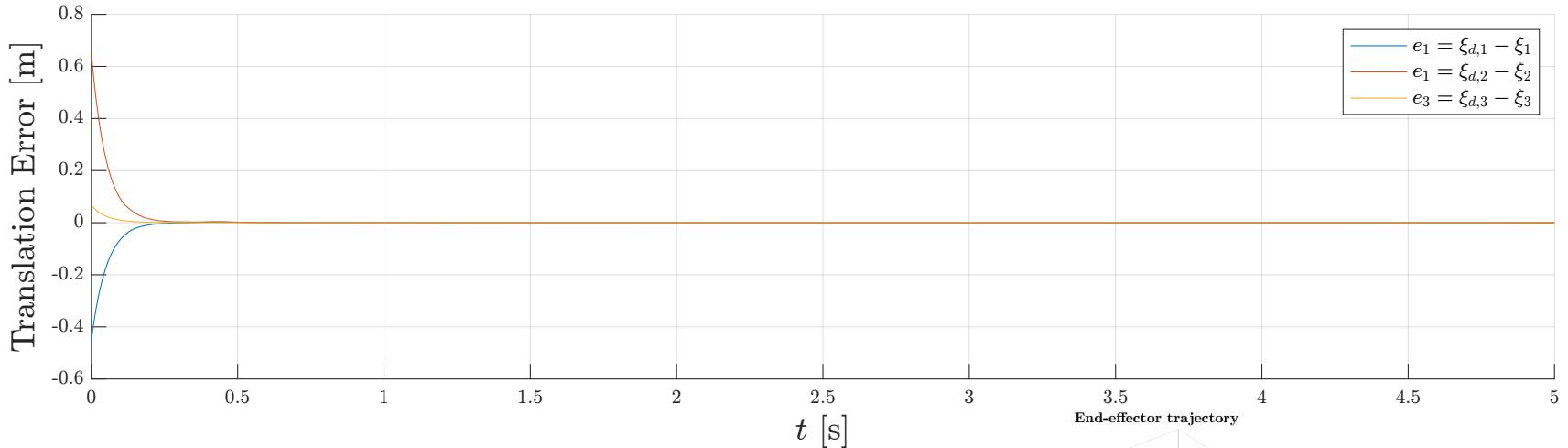
Controllo cinematico — Metodo di Levenberg-Marquardt

Simulazione (Inseguimento di traiettoria circolare)

Configurazione desiderata:

$$\xi_d(t) = \begin{bmatrix} \frac{1}{2} \sin\left(\frac{\pi}{3}t\right) & \frac{1}{2} & \frac{1}{2} \cos\left(\frac{\pi}{3}t\right) & 0 & 0 & 0 \end{bmatrix}^T$$

End-effector error



Controllo dinamico

- Si basa sull'equazione dinamica:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$$

Dove:

- q : vettore degli angoli di giunto del seriale
 - $M(q)$: matrice di inerzia
 - $C(q, \dot{q})$: matrice delle forze centrifughe e di Coriolis
 - $G(q)$: matrice dei termini gravitazionali
 - τ : vettore delle forze generalizzate
-
- Le forze generalizzate τ sono legate alle forze sugli attuatori f dalla seguente equazione lineare:

$$\tau = F(q)f$$

- Il numero di attuatori è pari al numero di giunti ($l = 6 = n$) per cui la matrice $F(q)$ è quadrata e, supponendola invertibile nelle configurazioni q di interesse, ($\text{rank}(F(q)) = n$), il sistema è completamente attuato.

Controllore PD

Sintesi

- Si basa sul far coincidere le forze generalizzate con termini proporzionali all'errore di posizione e alla sua derivata:

$$\tau = K_p e + K_v \dot{e}$$

con K_p e K_v matrici diagonali definite positive.

- Per compensare l'azione di gravità agente sui giunti si preferisce aggiungere il termine G , ottenendo un controllore PD indipendente con compensazione di gravità (legge di Arimoto):

$$\tau = K_p e + K_v \dot{e} + G(q)$$

- Se la configurazione desiderata è costante ($\dot{q}_d = \ddot{q}_d = 0$) allora il punto $e(t) = \dot{e}(t) = 0$ è di equilibrio poiché si ha:

$$M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d)\dot{q}_d = 0$$

altrimenti ciò non è più garantito.

- Pertanto, va bene solo per controlli punto a punto.

Controllore PD

Simulazione (pick and place)

Configurazione pick:

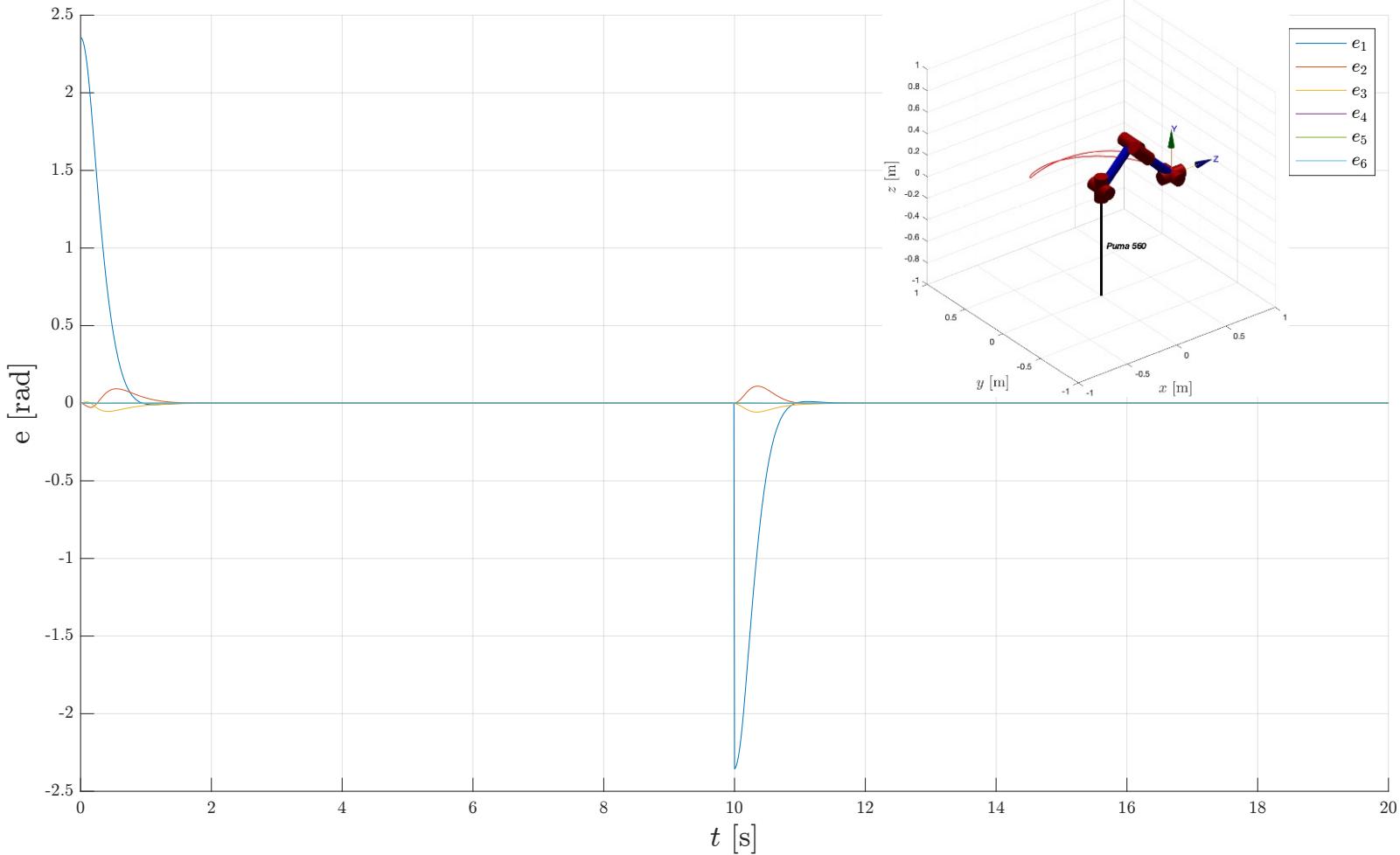
$$q_{d,1} = \begin{bmatrix} \frac{3}{4}\pi & \frac{\pi}{4} & \pi & 0 & \frac{\pi}{4} & \frac{\pi}{2} \end{bmatrix}^T$$

Configurazione start and place:

$$q_{d,2} = \begin{bmatrix} 0 & \frac{\pi}{4} & \pi & 0 & \frac{\pi}{4} & \frac{\pi}{2} \end{bmatrix}^T$$

Joint errors

End-effector trajectory



Controllore *computed torque*

Sintesi

- Nell'ipotesi di:
 - poter calcolare un modello esatto per la dinamica del manipolatore
 - poter misurare $q(t)$ e $\dot{q}(t)$

la coppia di controllo da applicare è:

$$\tau(q, \dot{q}, e, \dot{e}) = M(q)\ddot{q}_d + C(q, \dot{q})\dot{q} + G(q) + K'_p e + K'_v \dot{e}$$

- Scegliendo i guadagni come matrici diagonali definite positive $K'_p = M(q)K_p$ e $K'_v = M(q)K_v$, si ha la convergenza a zero per la dinamica dell'errore di inseguimento:

$$\ddot{e} + K_v \dot{e} + K_p e = 0$$

- Se K_v e K_p sono matrici diagonali, si hanno $n = 6$ sistemi del secondo ordine disaccoppiati ed indipendenti.
- Inoltre, se definite positive si ha che il punto $\dot{e} = e = 0$ è di equilibrio asintoticamente stabile (anche nel caso di inseguimento).

Controllore *computed torque*

Simulazione (Pick and place)

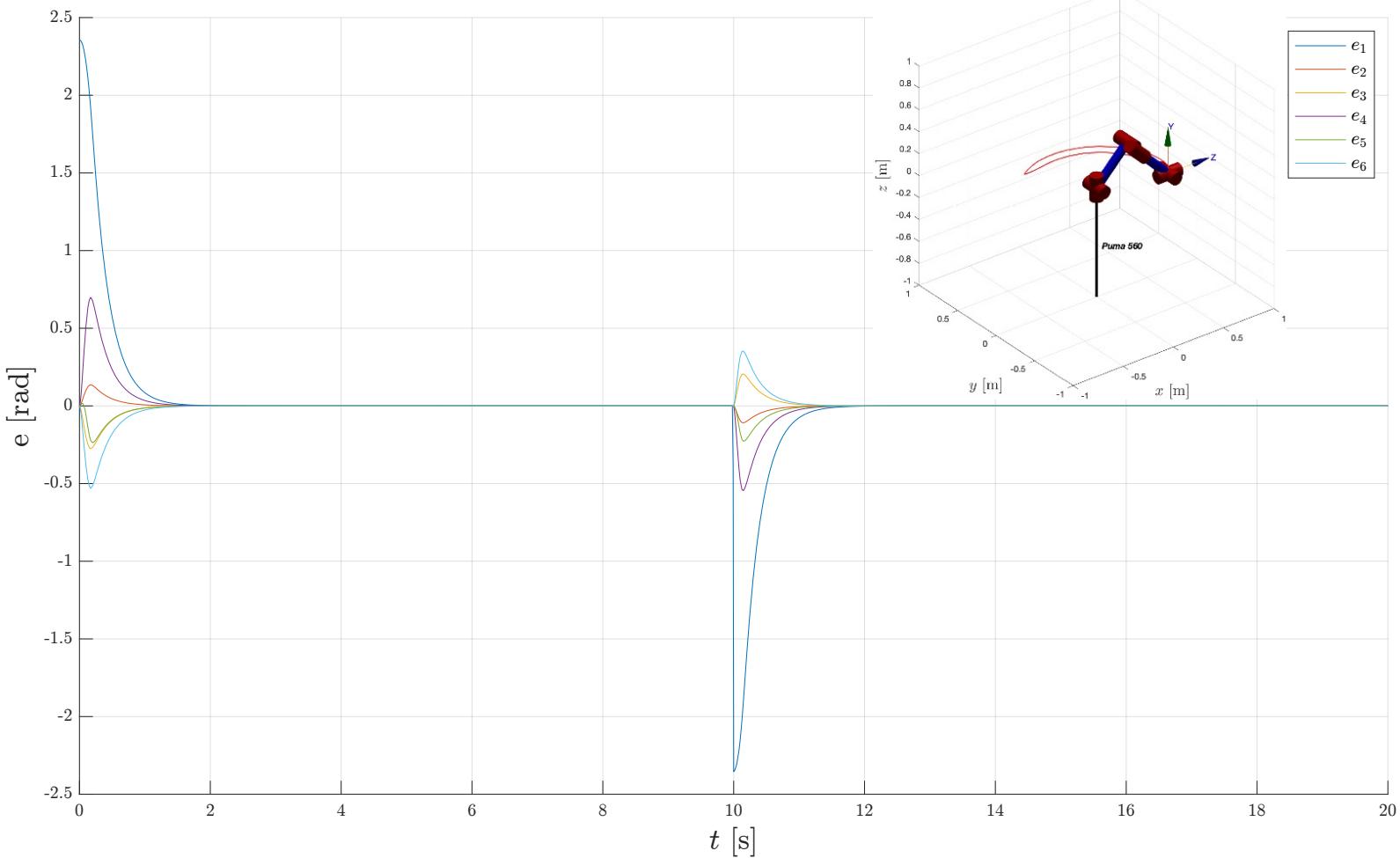
Configurazione *pick*:

$$q_{d,1} = \left[\frac{3}{4}\pi \quad \frac{\pi}{4} \quad \pi \quad 0 \quad \frac{\pi}{4} \quad \frac{\pi}{2} \right]^T$$

Configurazione *start and place*:

$$q_{d,2} = [0 \quad \frac{\pi}{4} \quad \pi \quad 0 \quad \frac{\pi}{4} \quad \frac{\pi}{2}]^T$$

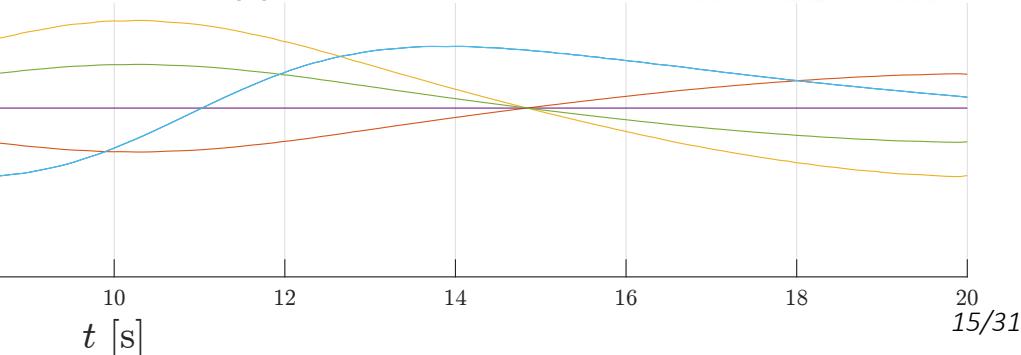
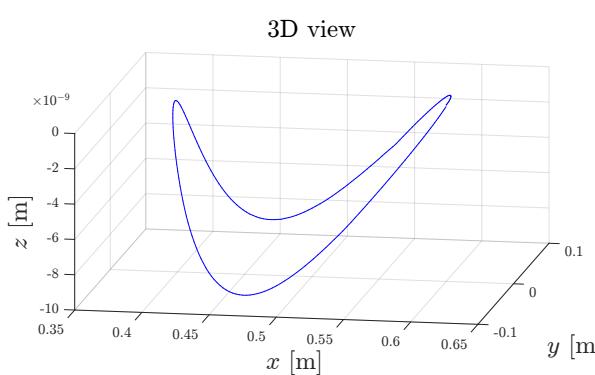
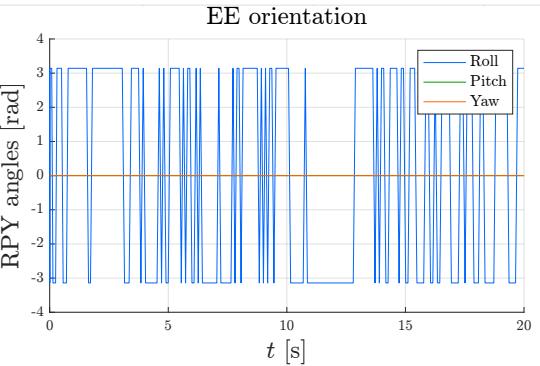
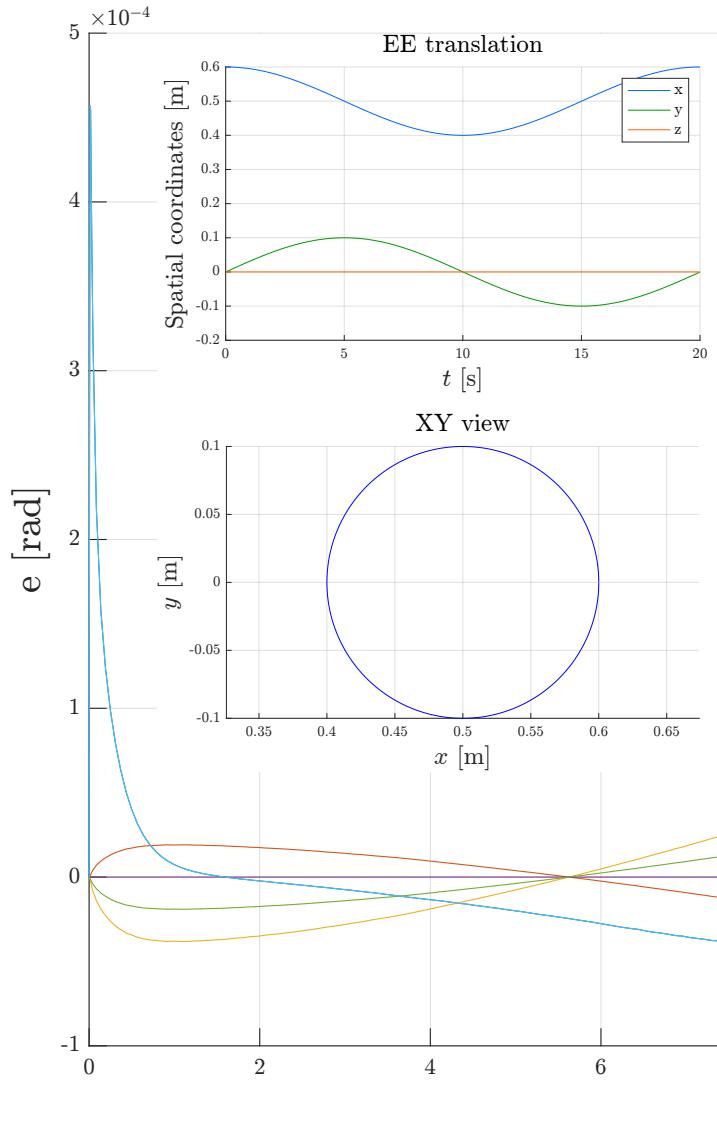
Joint errors



Controllore *computed torque*

Simulazione (Inseguimento di traiettoria circolare)

Joint errors



Controllore *backstepping*

Sintesi

- Si suddivide il problema di controllo in due fasi:
 1. controllare il modello cinematico usando le velocità come ingressi nominali
 2. design di un controllore tale che le velocità reali del manipolatore (\dot{q}) tendano a quelle di riferimento (\dot{q}_r) usate per il controllo in velocità
- Definendo:
 - $e = \xi_d - \xi$, l'errore di inseguimento nello spazio operativo
 - $s = \dot{q}_r - \dot{q} = J^{-1}(q)(\dot{e} + \Lambda e)$, l'errore in velocità (Λ matrice di guadagni simmetrica e definita positiva)

si applica la seguente legge di controllo:

$$\tau = M(q)\ddot{q}_r + C(q, \dot{q})\dot{q}_r + G(q) + K_d s + J^T e$$

- Si dimostra via Lyapunov che la legge così definita rende l'equilibrio $(e, s) = (0, 0)$ g.a.s. e pertanto la traiettoria dell'*end-effector* convergerà asintoticamente a quella desiderata.

Controllore *backstepping*

Simulazione (Pick and place)

Configurazione *pick*:

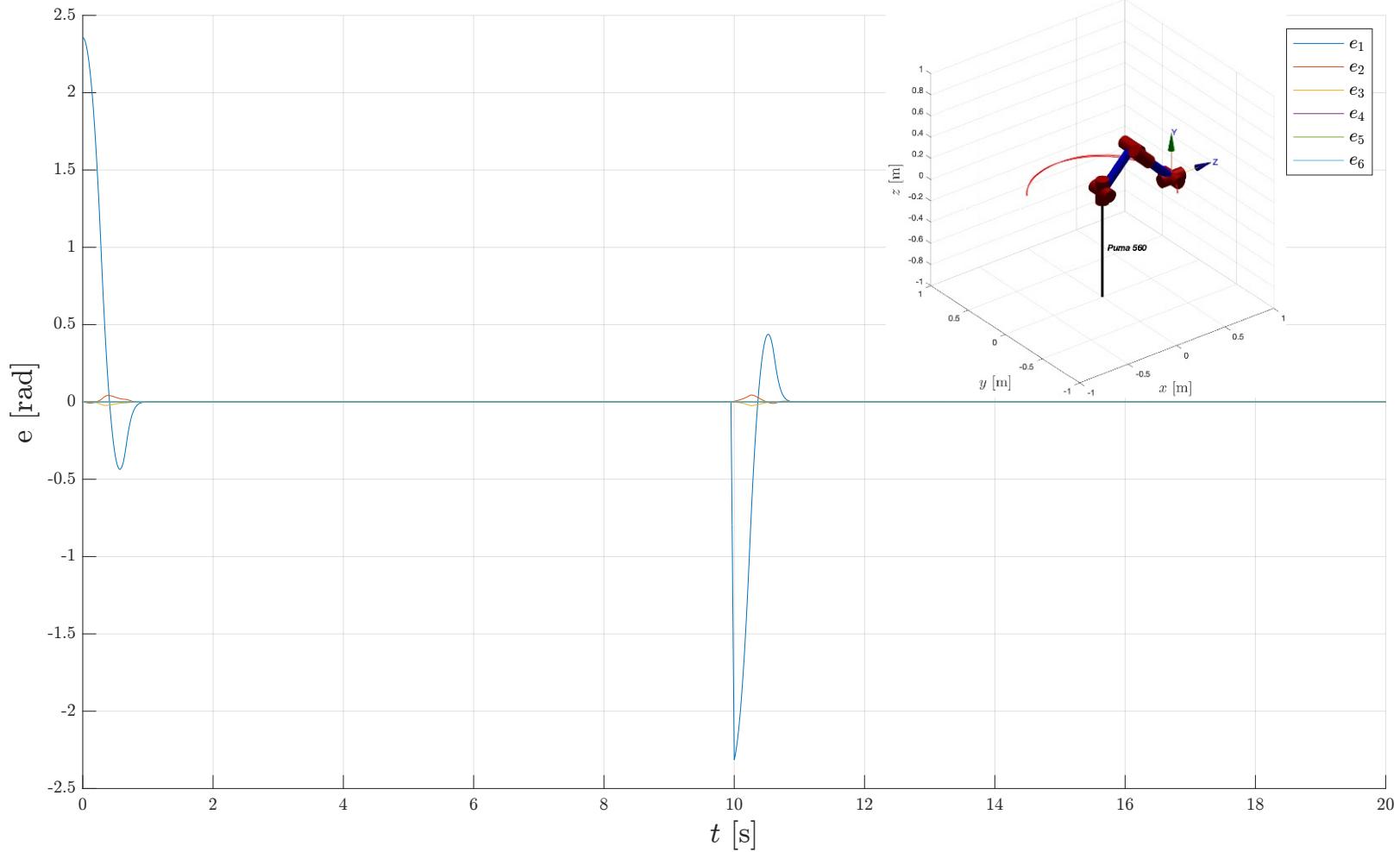
$$q_{d,1} = \begin{bmatrix} \frac{3}{4}\pi & \frac{\pi}{4} & \pi & 0 & \frac{\pi}{4} & \frac{\pi}{2} \end{bmatrix}^T$$

Joint errors

Configurazione *start and place*:

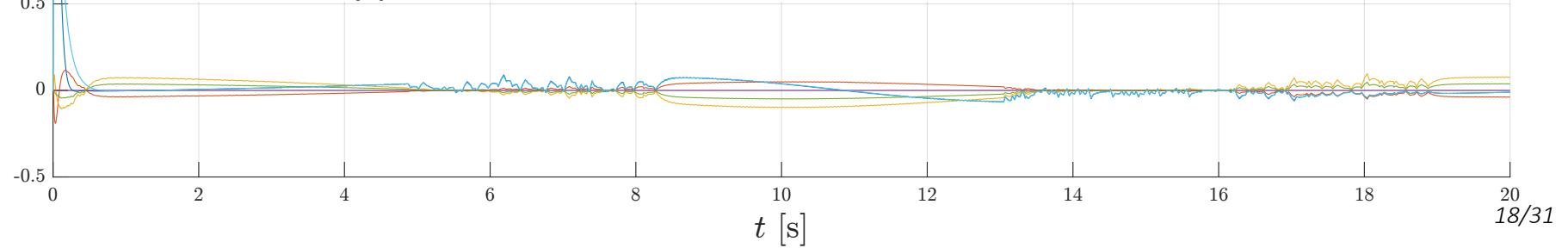
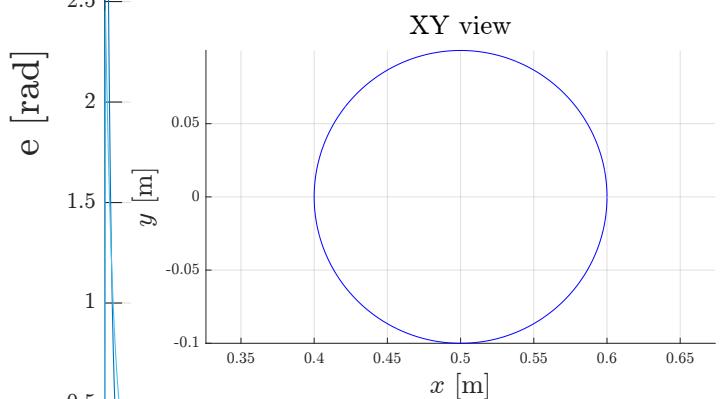
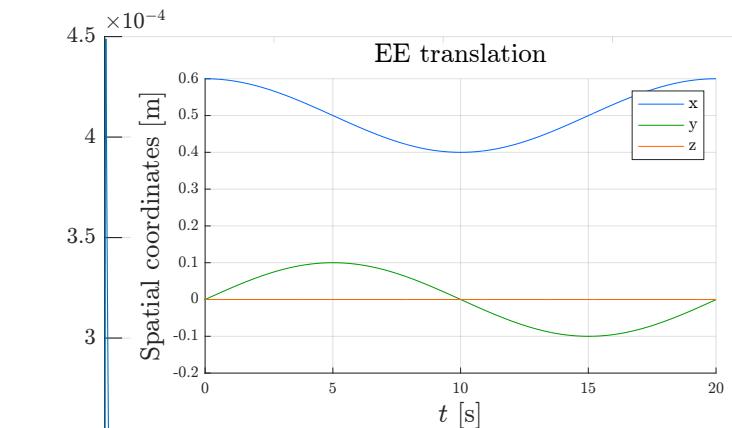
$$q_{d,2} = \begin{bmatrix} 0 & \frac{\pi}{4} & \pi & 0 & \frac{\pi}{4} & \frac{\pi}{2} \end{bmatrix}^T$$

End-effector trajectory



Controllore *backstepping*

Simulazione (Inseguimento di traiettoria circolare)



Controllo adattivo

Sintesi

- Nella realtà, (purtroppo) non conosciamo con esattezza il modello dinamico del sistema (le matrici dinamiche): $M(q), C(q, \dot{q}), G(q)$
- Conosciamo soltanto sue approssimazioni: $\hat{M}(q), \hat{C}(q, \dot{q}), \hat{G}(q)$
- Le loro differenze sono le matrici di errore di modello:

$$\begin{cases} \tilde{M}(q) = \hat{M}(q) - M(q) \\ \tilde{C}(q, \dot{q}) = \hat{C}(q, \dot{q}) - C(q, \dot{q}) \\ \tilde{G}(q) = \hat{G}(q) - G(q) \end{cases}$$

- Il controllo è basato sulle matrici approssimate

Controllore *computed torque* adattivo

Sintesi

C. D. Sousa, *SymPyBotics software* (available at <https://github.com/cdsousa/SymPyBotics>), an opensource Python package for symbolic robot dynamics, based on SageRobotics (Sousa and Cortesão, 2012), doi: 10.5281/zenodo.11365

- È ancora possibile controllare il sistema mediante la legge di controllo:

$$\tau(q, \dot{q}, e, \dot{e}) = \hat{M}(q)[\ddot{q}_d + K_v \dot{e} + K_p e] + \hat{C}(q, \dot{q})\dot{q} + \hat{G}(q)$$

con $\hat{M}(q)$, $\hat{C}(q, \dot{q})$, e $\hat{G}(q)$ stime delle matrici del sistema.

- Esprimendo la dinamica del sistema via regressore, la dinamica dell'errore d'inseguimento diviene (supponendo $\hat{M}(q)$ invertibile):

$$\ddot{e} + K_v \dot{e} + K_p e = \hat{M}^{-1}(q)Y(q, \dot{q}, \ddot{q})\tilde{\pi} = \eta$$

dove $\tilde{\pi} = \pi - \hat{\pi}$ rappresenta l'errore di stima dei parametri non noti.



A causa di η non è garantita la convergenza a zero dell'errore d'inseguimento $e(t)$

Controllore *computed torque* adattivo

Sintesi

- Si fa in modo che la stima dei parametri dinamici $\hat{\pi}$ venga aggiornata durante l'esecuzione del controllo stesso, con legge di aggiornamento:

$$\dot{\hat{\pi}} = u_\pi = R^{-1}Y^T(q, \dot{q}, \ddot{q})\hat{M}^{-T}(q)B^TP\xi$$

con $B = [0 \quad I]^T$, $\xi = (e, \dot{e})^T$, R matrice quadrata definita positiva, e P soluzione dell'equazione di Lyapunov:

$$A^T P + P A = -Q, \quad Q > 0$$



Il sistema in ciclo chiuso tende ad azzerare l'errore di inseguimento $e(t)$, ma nulla è garantito sulla convergenza dell'errore di stima $\tilde{\pi}$

Controllore *computed torque* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

Traiettoria desiderata:

$$\begin{cases} q_{i,d}(t) = \sum_{l=1}^L \frac{a_{i,l}}{\omega_f l} \sin(\omega_f lt) - \frac{b_{i,l}}{\omega_f l} \cos(\omega_f lt) \\ \dot{q}_{i,d}(t) = \sum_{l=1}^L a_{i,l} \cos(\omega_f lt) + b_{i,l} \sin(\omega_f lt) \quad i = 1, \dots, n \\ \ddot{q}_{i,d}(t) = \omega_f l \sum_{l=1}^L -a_{i,l} \sin(\omega_f lt) + b_{i,l} \cos(\omega_f lt) \end{cases}$$

dove:

- ω_f è la frequenza fondamentale (comune a tutti i giunti, per rendere periodica la traiettoria eccitante)
- $a_{i,l}$ e $b_{i,l}$ sono i coefficienti della serie di Fourier
- L è l'ordine della serie di Fourier

Controllore *computed torque* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

S. Krishnamoorthi, "Model-Based Compensation for Serial Manipulators through Semi-Parametric Gaussian Process Regression" Master of Science Thesis, 2018, Faculty of Mechanical, Maritime and Materials Engineering (3mE), Delft University of Technology, Delft, Netherlands

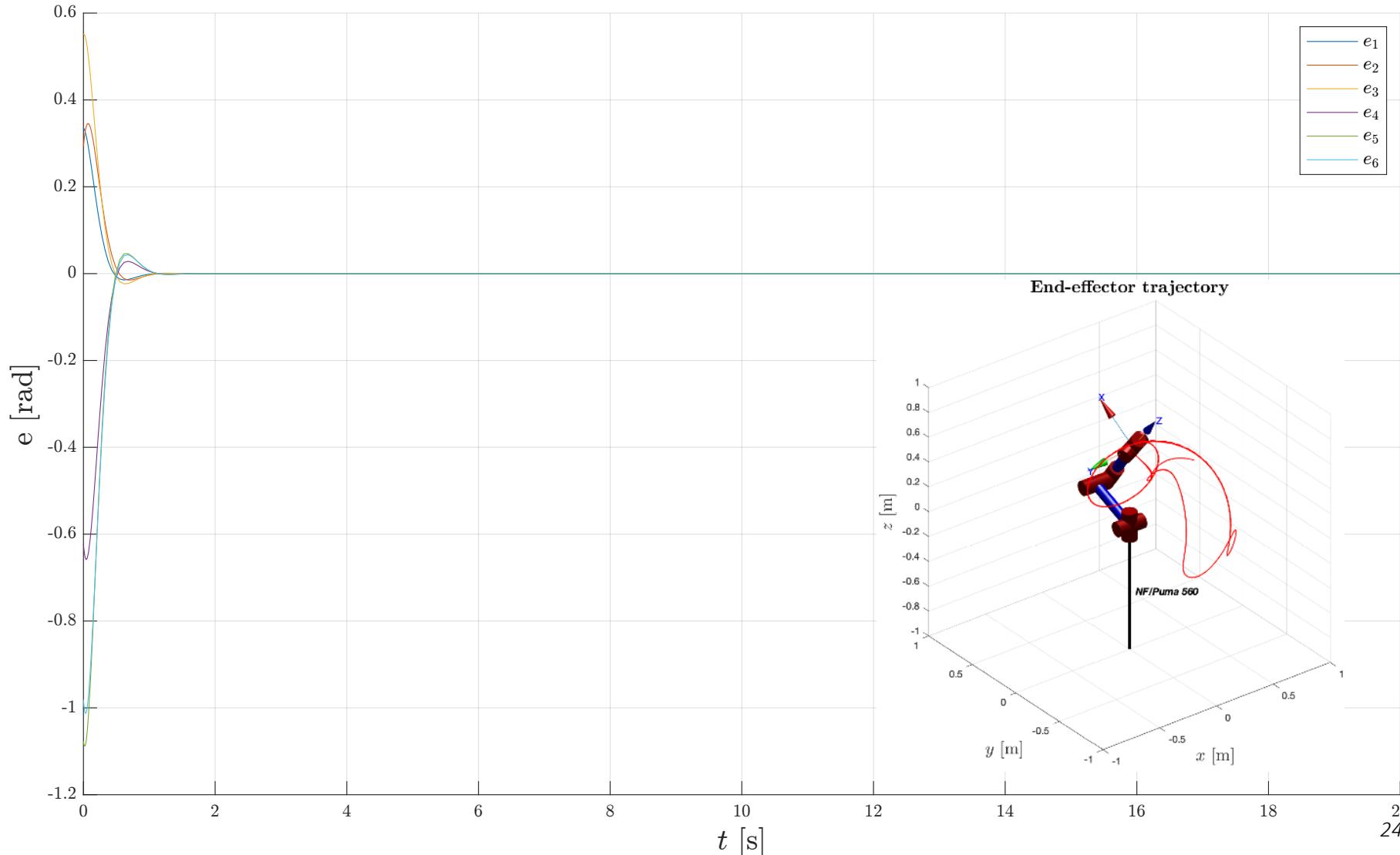
$$\begin{aligned}\omega_f &= 1 \text{ Hz} \\ L &= 4\end{aligned}$$

Variable	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
a_1	0.88	1.20	0.0008	-0.28	-0.09	-1.44
a_2	0.26	-0.13	0.0006	0.41	-0.026	0.19
a_3	-0.54	0.41	0.066	-0.88	-0.86	0.26
a_4	-0.76	0.21	0.11	-0.99	-0.22	-1
b_1	-0.57	0.27	-0.63	0.24	0.75	0.091
b_2	0.26	0.28	-0.30	-0.067	0.74	1.33
b_3	0.23	-1.89	0.56	0.92	0.20	1.08
b_4	0.11	-0.29	0.17	0.44	-0.44	-0.54

Controllore *computed torque* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

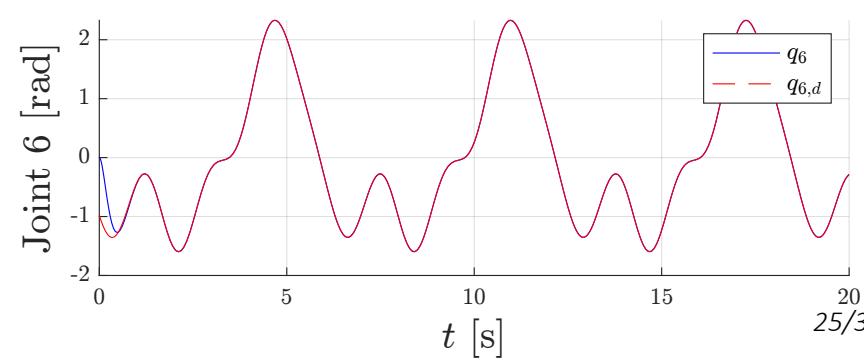
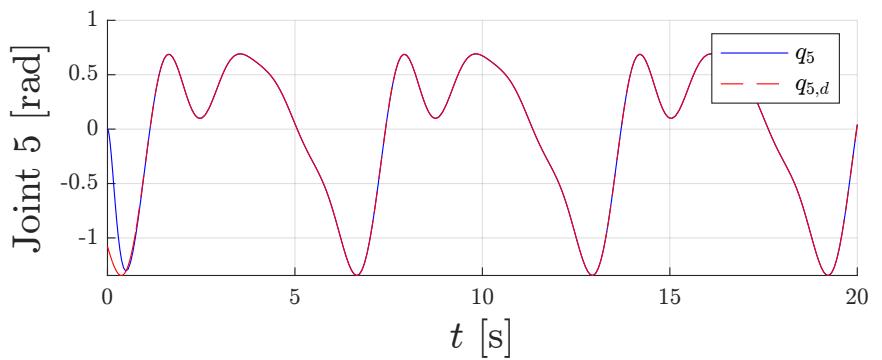
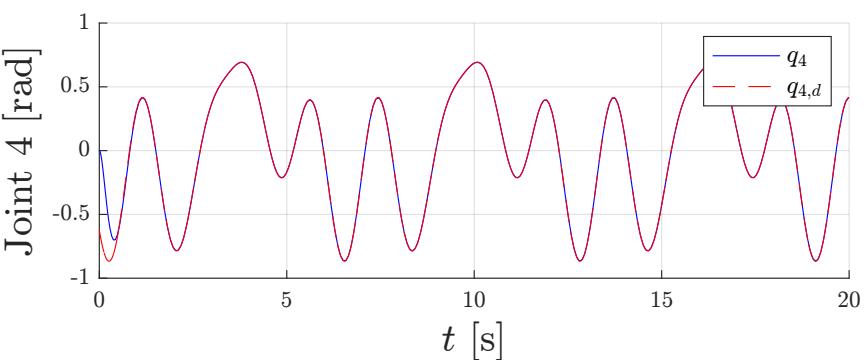
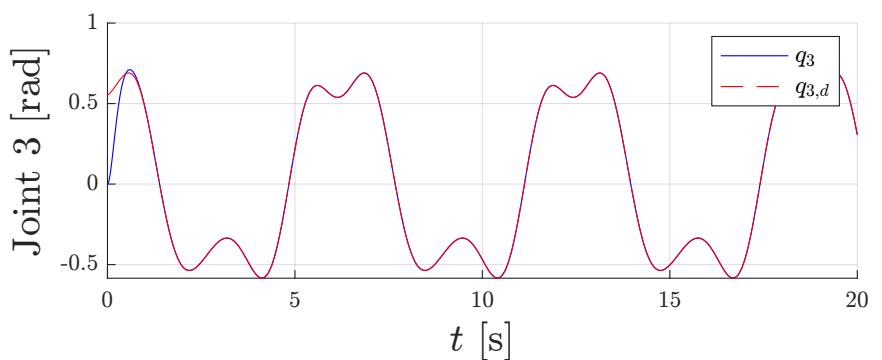
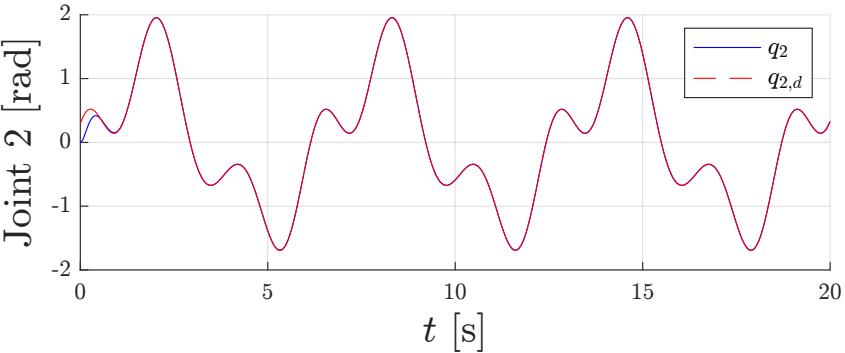
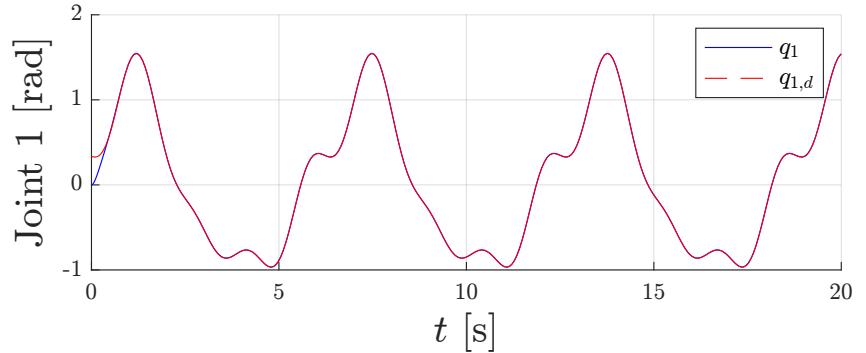
Joint errors



Controllore *computed torque* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

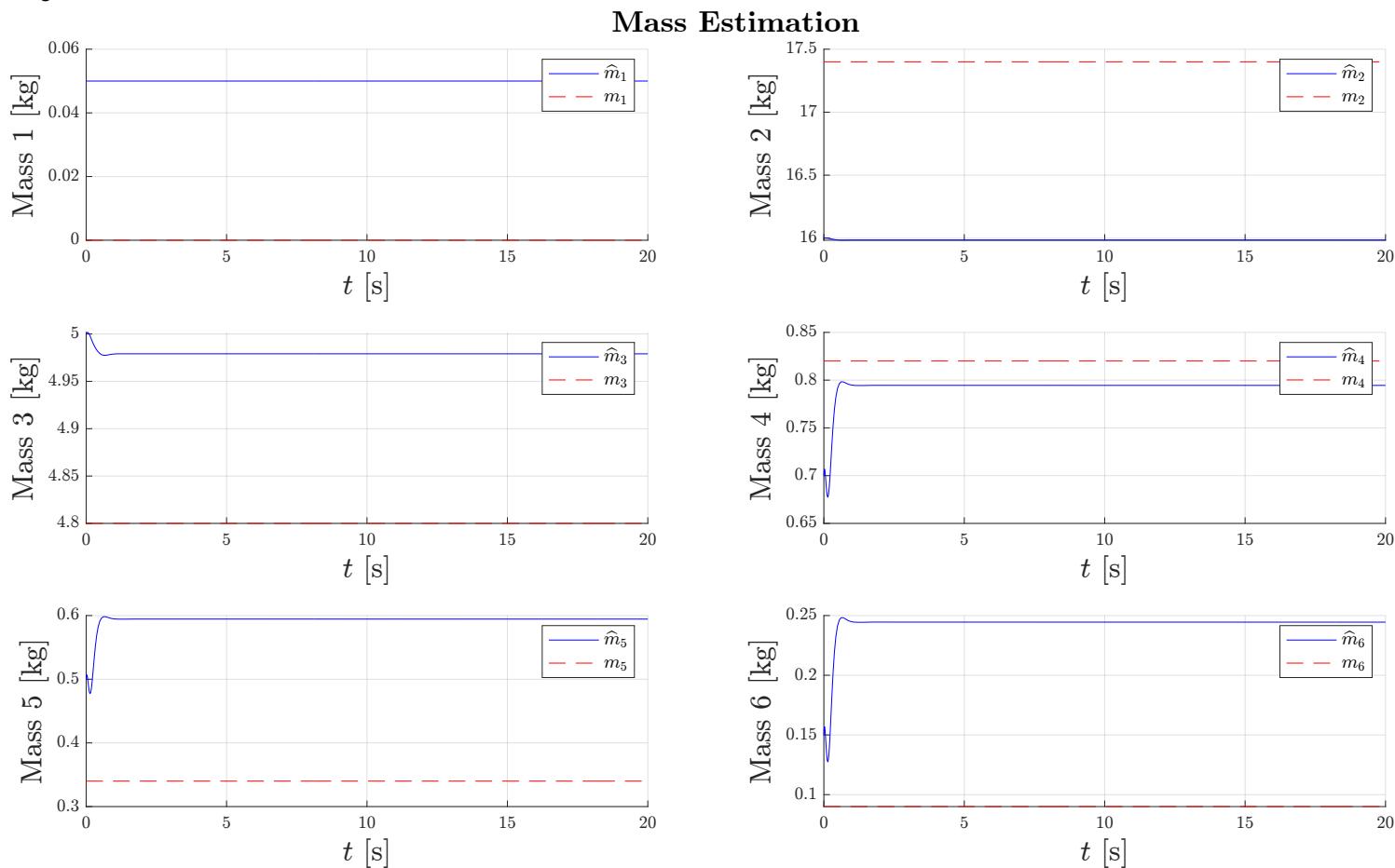
Joint trajectories



Controllore *computed torque* adattivo

Stima dei parametri

	m_1 [kg]	m_2 [kg]	m_3 [kg]	m_4 [kg]	m_5 [kg]	m_6 [kg]
π	0	17.4	4.8	0.82	0.34	0.09
$\hat{\pi}_0$	0.05	16	5	0.7	0.5	0.15



Controllore *backstepping* adattivo

Sintesi

- È ancora possibile controllare il sistema mediante la legge di controllo:

$$\tau = Y(q, \dot{q}_r, \ddot{q}_r)\hat{\pi} + K_d s + e$$

con $e = q_r - q$, $s = \dot{e} + \Lambda e = \dot{q}_r - \dot{q}$.

- Si aggiornano i parametri dinamici $\hat{\pi}$ con legge di aggiornamento:

$$\dot{\hat{\pi}} = u_\pi = R^{-1}Y^T(q, \dot{q}_r, \ddot{q}_r)s$$

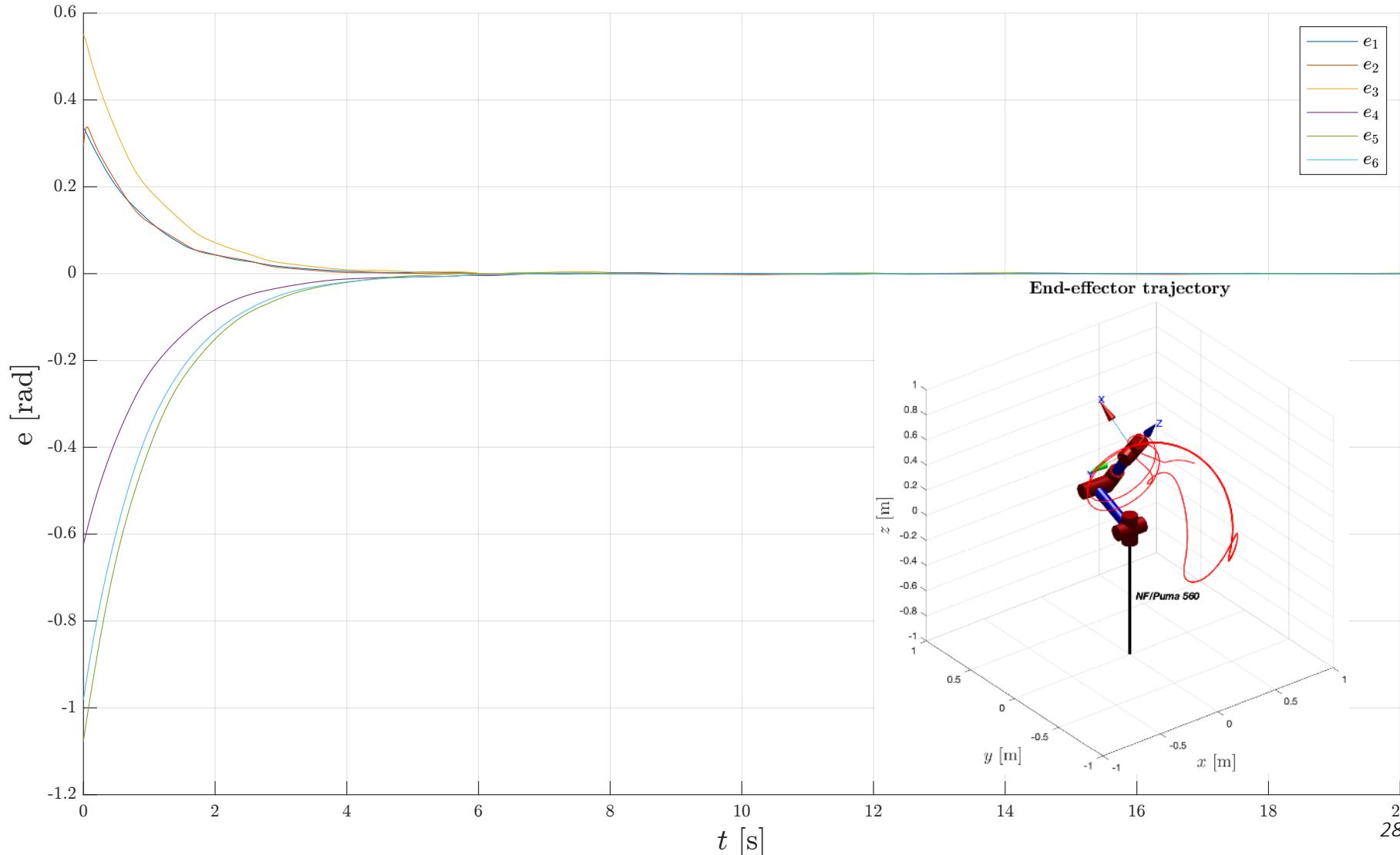
dove R è una matrice simmetrica definita positiva di design.

- Si ha convergenza dell'errore di inseguimento $e(t)$, tuttavia anche in questo caso non è garantita circa la convergenza dei parametri.
- Non è richiesto il computo di $\hat{M}^{-1}(q)$, il che rende il metodo numericamente più semplice (oltre al fatto che possa non esistere).

Controllore *backstepping* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

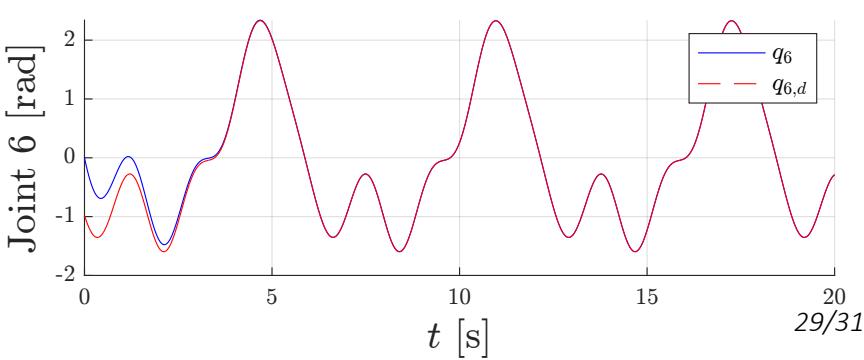
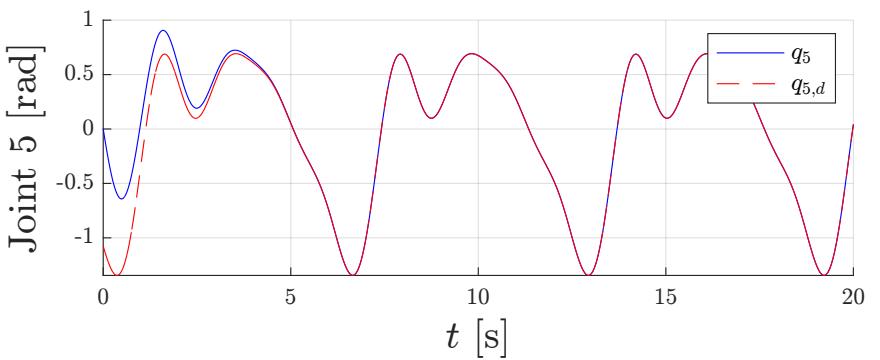
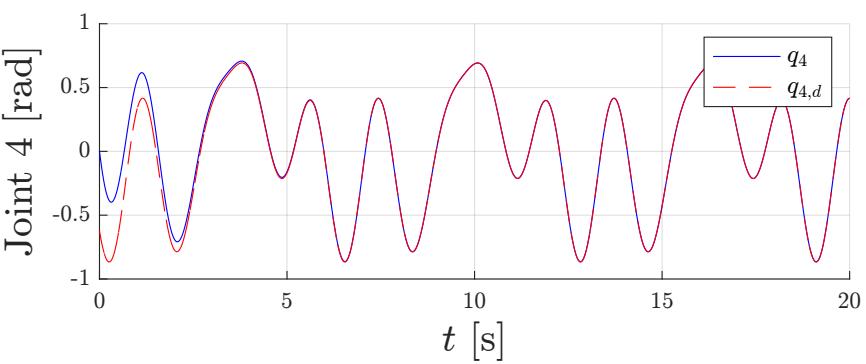
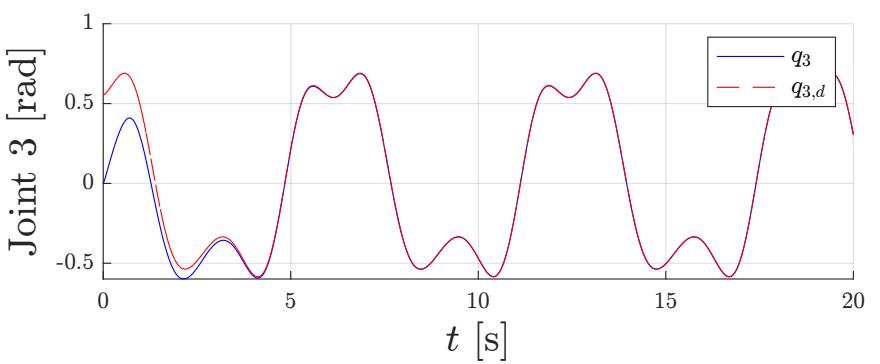
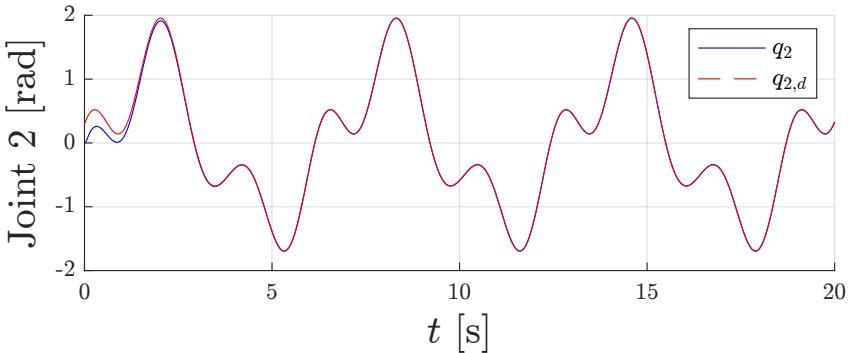
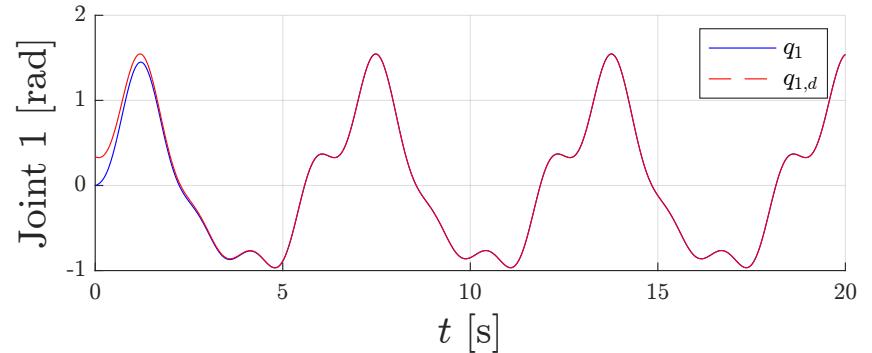
Joint errors



Controllore *backstepping* adattivo

Simulazione (Inseguimento di traiettoria da FFS - Finite Fourier Series)

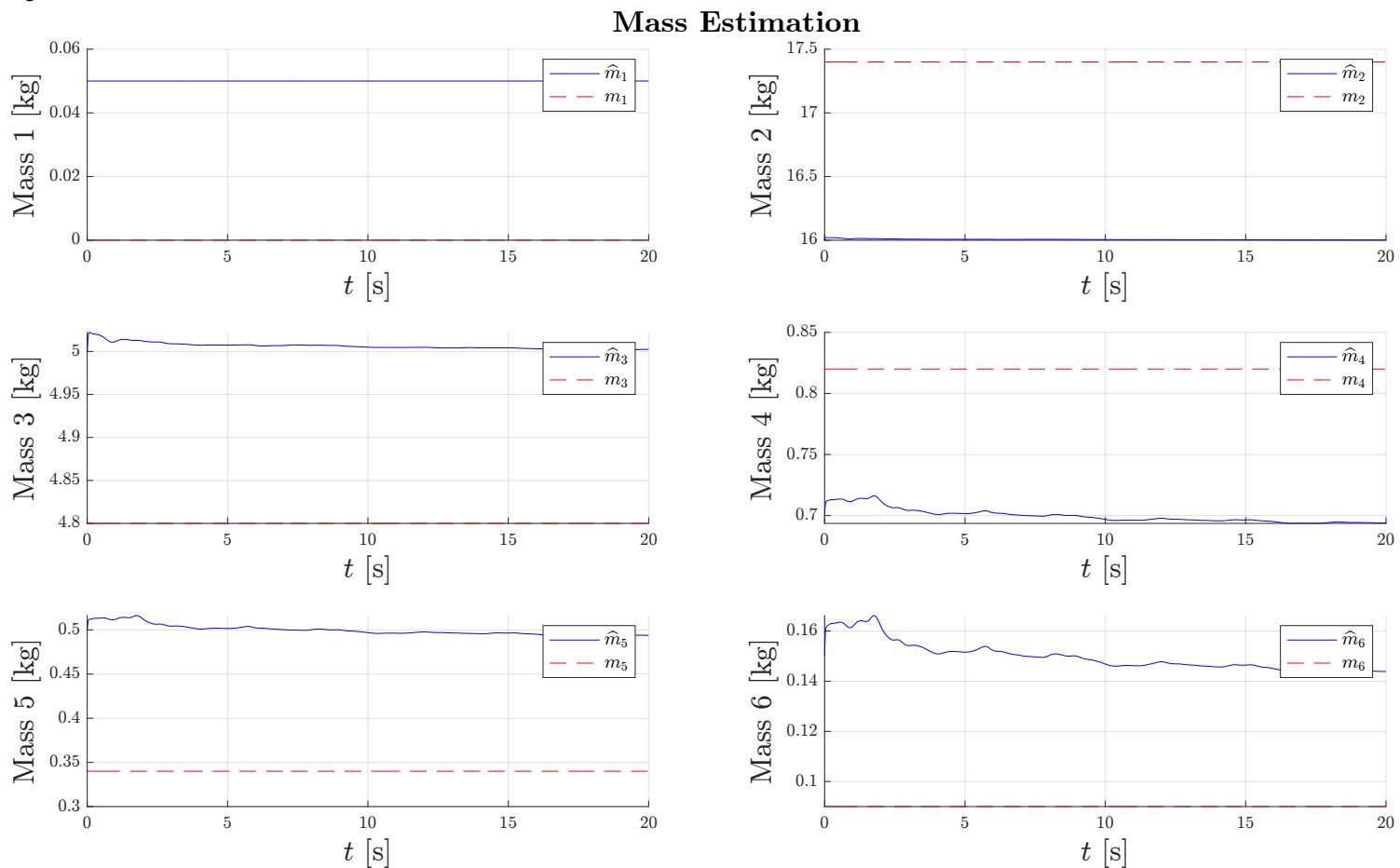
Joint trajectories



Controllore *backstepping* adattivo

Stima dei parametri

	m_1 [kg]	m_2 [kg]	m_3 [kg]	m_4 [kg]	m_5 [kg]	m_6 [kg]
π	0	17.4	4.8	0.82	0.34	0.09
$\hat{\pi}_0$	0.05	16	5	0.7	0.5	0.15



Controllo adattivo

Stima dei parametri

- Né col controllo a coppia calcolata né col controllo backstepping adattivi si è ottenuta la convergenza dei parametri ai valori reali – cosa in effetti non garantita
- Pertanto, i riferimenti usati nelle simulazioni non eccitano sufficientemente la dinamica dell'errore di stima tanto da permettere il raggiungimento di stima esatta nell'intervallo di tempo considerato
- In generale, l'errore di stima $\tilde{\pi}$ tende a un valore costante $\bar{\tilde{\pi}}$ tale per cui, ad ogni istante t considerato:

$$Y(q, \dot{q}, \ddot{q})\bar{\tilde{\pi}} = 0$$

- La matrice di regressione varia nel tempo, per cui $\bar{\tilde{\pi}}$ deve stare nell'intersezione di tutti i nulli delle varie matrici Y
- Inoltre, poiché la colonna di Y corrispondente alla stima della massa del primo link è sempre nulla, la sua stima \hat{m}_1 rimarrà sempre invariata rispetto al valore di stima iniziale $\hat{m}_{1,0}$
- Ciò con qualsiasi riferimento, anche qualora fosse sufficientemente eccitante

A large, vibrant green banana leaf with a prominent central vein is floating on the surface of clear, turquoise-blue ocean water. The water is filled with fine, radiating ripples and light reflections, creating a sense of depth and tranquility. The horizon line is visible in the distance under a clear blue sky.

- Grazie per l'attenzione