



SAPIENZA
UNIVERSITÀ DI ROMA

Controllo di un robot self-balancing

Facoltà di Ingegneria dell'informazione, informatica e statistica

Corso di Laurea in Ingegneria Informatica e Automatica

Candidato

Matteo Leonforte

Matricola 1885269

Relatore

Prof. Giuseppe Oriolo

Correlatore

Tommaso Belvedere

Anno Accademico 2021/2022

Tesi discussa il 08/07/2022
di fronte a una commissione esaminatrice composta da:
Prof. Stefano Battilotti (presidente)
Prof. Claudia Califano

Controllo di un robot self-balancing

Tesi di Laurea. Sapienza – Università di Roma

© 2022 Matteo Leonforte. Tutti i diritti riservati

Questa tesi è stata composta con L^AT_EX e la classe Sapthesis.

Email dell'autore: leonforte.1885269@studenti.uniroma1.it

Sommario

Questa tesi prenderà in esame il controllo di una tipologia di Wheeled Inverted Pendulum (WIP): una tipologia di self-balancing robot intrinsecamente instabile, che utilizza degli attuatori applicati alle ruote per il bilanciamento e lo spostamento.

Verranno analizzate due tecniche di controllo chiamate Linear Quadratic Regulator e Intrinsically-Stable Model Predictive Control. Esse verranno impiegate prima per stabilizzare il robot intorno ad una posizione di equilibrio e poi per ottenere l'inseguimento di una traiettoria desiderata. Il sistema, composto da processo e controllore, verrà poi simulato in Matlab e i risultati verranno discussi.

Indice

1	Introduzione	1
1.1	Struttura della tesi	2
2	Modello dinamico	3
2.1	Calcolo del modello	4
3	Linear Quadratic Regulator	7
3.1	Linearizzazione del sistema	8
3.2	Stabilizzazione nell'origine	9
3.3	Traiettoria costante	11
3.4	Traiettoria a velocità costante	13
3.5	Traiettoria sinusoidale	14
3.6	Considerazioni sull'LQR	15
4	Intrinsically-Stable Model Predictive Control	16
4.1	Trasformazione di coordinate e linearizzazione	17
4.2	Trajectory tracking	18
4.3	Stabilità della Zero Dynamics	19
4.4	Discretizzazione del sistema	20
4.5	Funzione di costo	20
4.6	Vincolo di stabilità	21
4.7	Traiettoria costante	21
4.8	Traiettoria sinusoidale	22
5	Conclusioni	24
	Bibliografia	25

Capitolo 1

Introduzione

Con l'avanzamento tecnologico, i robot stanno diventando sempre più presenti nelle nostre vite. Nell'immaginario collettivo, la parola robot è solitamente associata ad un automa dalle sembianze umanoidi in grado di mostrare una qualche forma di intelligenza. In realtà il termine "robot" è molto più ampio ed include una grandissima varietà di dispositivi dalle caratteristiche ed applicazioni molto eterogenei.

I robot mobili, in grado di orientarsi e navigare nello spazio, hanno acquisito grande popolarità in campi come il trasporto, la pulizia e la logistica. Alcuni di essi, definiti staticamente stabili, necessitano di un centro di massa posizionato in basso e di un ampio poligono di supporto. Queste caratteristiche consentono loro di restare in equilibrio, senza che nessun controllo sia applicato dagli attuatori. In contrasto, sistemi intrinsecamente instabili come il Wheeled Inverted Pendulum (WIP) necessitano una continua azione di controllo per mantenere l'equilibrio, ma in compenso sono in grado di accelerare e decelerare rapidamente e presentano un poligono di supporto molto ridotto. Queste caratteristiche consentono loro di operare facilmente nell'ambiente umano.



(a) Ascento - ETH Zürich



(b) Handle - Boston Dynamics

Il WIP rientra nei cosiddetti sistemi *sottoattuatori*, nei quali cioè il numero di input di controllo è inferiore rispetto ai gradi di libertà. I motori delle ruote rappresentano gli attuatori attraverso cui è applicata la legge di controllo al sistema.

In questa tipologia di robot possiamo distinguere due dinamiche:

1. I/O dynamics: rappresenta come l'input di controllo influenza la posizione orizzontale del robot (variabile controllata);
2. Zero dynamics: rappresenta l'evoluzione delle dinamiche interne (come ad esempio l'inclinazione)

La zero dynamics è intrinsecamente instabile: sistemi di questo tipo prendono il nome di sistemi a *fase non minima*. Per un sistema lineare questo equivale ad avere una funzione di trasferimento ingresso-uscita con degli zeri a parte reale positiva. La difficoltà nel controllo di sistemi a fase non minima risiede nel dover controllare la dinamica input-output, assicurandosi che la zero dynamics non diverga nel tempo.

In questa tesi verranno analizzate delle tecniche di controllo per la stabilizzazione e l'inseguimento di una traiettoria di un WIP.

Verranno messe a confronto due diverse metodologie:

- La prima prende il nome di Linear Quadratic regulator (LQR). Prevede un controllo basato su un *feedback dallo stato* di tipo proporzionale.
- La seconda metodologia è chiamata Intrinsically-Stable MPC (IS-MPC) e consiste nell'unione di due tecniche: *linear stable inversion* e *Model Predictive Control*. Si basa sulla possibilità di trasformare il problema di controllo in un problema di ottimizzazione, a cui è possibile applicare vincoli e per il quale esistono algoritmi efficienti per la ricerca della soluzione ottima. In questo contesto, utilizzeremo un'ulteriore tecnica chiamata *input-output feedback linearization*.

1.1 Struttura della tesi

Nel **Capitolo 2** viene ricavato il modello dinamico di un WIP planare; nel **Capitolo 3** e nel **Capitolo 4** vengono formulati dei controllori per il sistema mediante le due metodologie accennate (LQR e IS-MPC). In un primo momento il controllo è finalizzato alla sola stabilizzazione del robot intorno ad una posizione di equilibrio, dopodiché si procede con l'inseguimento di traiettorie progressivamente più complesse. Le performance del controllore vengono simulate di volta in volta in Matlab. Nel **Capitolo 5** infine vengono riassunte le differenze tra i due metodi e vengono riportate le conclusioni.

Capitolo 2

Modello dinamico

Nella maggior parte delle applicazioni reali, il WIP si muove tramite due ruote, ma in alcuni casi, il robot potrebbe disporre solamente di una ruota. Mentre in quest'ultimo caso il robot è instabile in tutte le direzioni del piano, nel caso a due ruote esso è stabile lungo il piano frontale (in quanto la proiezione del centro di massa è sempre compresa tra i due punti di contatti con il suolo), ma instabile lungo il piano sagittale.

La dinamica lungo il piano sagittale rappresenta l'aspetto più complesso da controllare, ed è anche quello su cui si focalizzerà questa tesi. A tale scopo, indipendentemente dal fatto che il robot abbia una o due ruote, il problema verrà modellato in due dimensioni come la composizione di due corpi rigidi, una ruota e un corpo, come mostrato nella Figura 2.1.

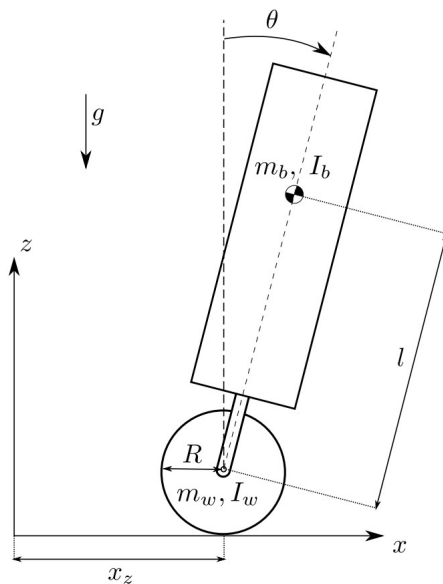


Figura 2.1. Parametri e coordinate del wheeled inverted pendulum

Nomenclatura:

- massa del corpo: m_b
- momento di inerzia del corpo: I_b

- massa della ruota: m_w
- momento di inerzia della ruota: I_w
- raggio della ruota: R
- distanza tra asse della ruota e CdM del corpo: l

Il sistema ha due gradi di libertà: sono sufficienti cioè due variabili per definirne in modo univoco la posizione nello spazio. Esse sono l'ascissa (x_z) e l'angolo di inclinazione (θ). La configurazione del sistema è definita quindi come $q = (x_z, \theta)^T$ e lo stato come $x = (q, \dot{q})^T$

2.1 Calcolo del modello

Per ricavare il modello dinamico del sistema possono essere utilizzate le equazioni di Eulero-Lagrange [1]. In forma canonica sono scritte come:

$$\frac{d}{dt} \frac{dL}{dq_i} - \frac{dL}{dq_i} = Q_i \quad i = 1, 2, \dots, n \quad (2.1)$$

dove:

- L è definito come la differenza tra energia cinetica ed energia potenziale.

$$L = T - U \quad (2.2)$$

- q_i è l'i-esima coordinata generalizzata del sistema
- n è il numero di gradi di libertà
- Q_i è la risultante delle forze **non conservative** che agiscono sulla coordinata generalizzata q_i

Energia cinetica L'energia cinetica può essere calcolata genericamente come la somma di una componente traslazionale ed una rotazionale:

$$T = T_T + T_R = \frac{1}{2} m v_{CoM}^2 + \frac{1}{2} I \omega^2 \quad (2.3)$$

È necessario calcolare la velocità del centro di massa e la velocità angolare della ruota e del corpo a partire dalle coordinate.

Posizione e velocità del centro di massa (CdM) della ruota:

$$\begin{cases} x_{CdM}^w(t) = x_z(t) \\ z_{CdM}^w(t) = R \end{cases} \Rightarrow \begin{cases} \dot{x}_{CdM}^w(t) = \dot{x}_z(t) \\ \dot{z}_{CdM}^w(t) = 0 \end{cases} \quad (2.4)$$

Posizione e velocità del centro di massa (CdM) del corpo:

$$\begin{cases} x_{CdM}^b(t) = x_z(t) + l \sin(\theta(t)) \\ z_{CdM}^b(t) = R + l \cos(\theta(t)) \end{cases} \Rightarrow \begin{cases} \dot{x}_{CdM}^b(t) = \dot{x}_z(t) + l \cos(\theta(t)) \dot{\theta}(t) \\ \dot{z}_{CdM}^b(t) = -l \sin(\theta(t)) \dot{\theta}(t) \end{cases} \quad (2.5)$$

Velocità angolare della ruota:

$$\dot{x}_{CdM}^w(t) + R w^w(t) = 0 \quad \Rightarrow \quad w^w(t) = -\dot{x}_{CoM}^w(t)/R \quad (2.6)$$

Velocità angolare del corpo:

$$w^b(t) = -\dot{\theta}(t) \quad (2.7)$$

Possiamo ora calcolare l'energia cinetica della ruota (T_w), del corpo (T_b) e l'energia cinetica complessiva.

$$\begin{aligned} T_w &= \frac{1}{2} m_w [(\dot{x}_{CdM}^w)^2 + (\dot{z}_{CdM}^w)^2] + \frac{1}{2} I_w [w^w]^2 = \\ &= \frac{1}{2} \underbrace{[m_w + I_w/R^2]}_{m_w^{eq}} \dot{x}_z^2 = \frac{1}{2} m_w^{eq} \dot{x}_z^2 \\ T_b &= \frac{1}{2} m_b [(\dot{x}_{CdM}^b)^2 + (\dot{z}_{CdM}^b)^2] + \frac{1}{2} I_b [w^b]^2 = \\ &= \frac{1}{2} m_b \dot{x}_z^2 + \frac{1}{2} \underbrace{(m_b l^2 + I_b)}_{I_b^{eq}} \dot{\theta}^2 + m_b l \cos(\theta) \dot{x}_z \dot{\theta} = \\ &= \frac{1}{2} m_b \dot{x}_z^2 + \frac{1}{2} I_b^{eq} \dot{\theta}^2 + m_b l \cos(\theta) \dot{x}_z \dot{\theta} \\ T &= T_w + T_b \end{aligned} \quad (2.8)$$

Abbiamo introdotto due nuove variabili chiamate m_w^{eq} e I_b^{eq} : la prima equivale alla massa di un corpo in moto di sola traslazione con stessa velocità e energia cinetica della ruota; la seconda corrisponde al momento di inerzia del corpo ma calcolato rispetto all'asse di rotazione della ruota.

Energia potenziale L'energia potenziale coincide con l'energia potenziale gravitazionale del corpo.

$$V = V_b = m_b l g \cos(\theta) \quad (2.9)$$

Scelte come coordinate generalizzate le variabili di stato x_z e θ , attraverso le equazioni lagrangiane possiamo ricavare le equazioni del sistema:

$$\begin{aligned} \frac{d}{dt} \frac{dL}{d\dot{x}_z} - \frac{dL}{dx_z} &= Q_x \\ \frac{d}{dt} \frac{dL}{d\dot{\theta}} - \frac{dL}{d\theta} &= Q_\theta \end{aligned} \quad (2.10)$$

Da esse ricaviamo il seguente sistema di equazioni differenziali di secondo ordine:

$$\begin{aligned} M(q)\ddot{q} &= C(q, \dot{q}) + Q \\ \underbrace{\begin{pmatrix} m_b + m_w^{eq} & m_b l \cos(\theta) \\ m_b l \cos(\theta) & I_b^{eq} \end{pmatrix}}_{M(q)} \underbrace{\begin{pmatrix} \ddot{x}_z \\ \ddot{\theta} \end{pmatrix}}_{\ddot{q}} &= \underbrace{\begin{pmatrix} m_b l \sin(\theta) \dot{\theta}^2 \\ m_b l g \sin(\theta) \end{pmatrix}}_{C(q, \dot{q})} + \begin{pmatrix} Q_x \\ Q_\theta \end{pmatrix} \end{aligned} \quad (2.11)$$

La matrice $M(q)$ prende il nome di matrice delle inerzie ed è una matrice simmetrica definita positiva invertibile. $C(q, \dot{q})$ rappresenta il vettore di Coriolis e le forze gravitazionali, mentre Q è il vettore delle forze non conservative che agiscono sul sistema. In questa tesi il vettore Q rappresenterà solamente gli input provenienti dagli attuatori e non gli attriti, che verranno invece modellati come disturbi.

Nel modello considerato i motori producono una coppia di forze direttamente sull'asse della ruota. Trattandosi di un'azione interna, il corpo e la ruota sono soggetti a coppie della stessa intensità ma versi opposti.

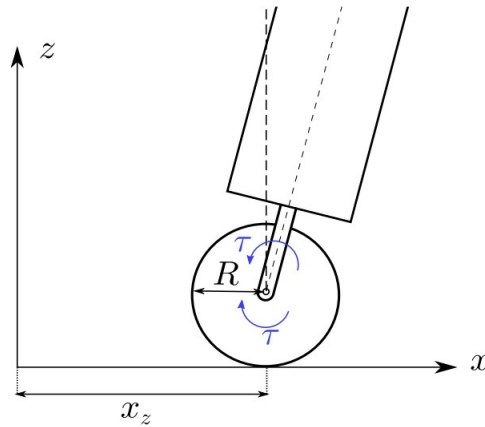


Figura 2.2. Coppia sull'asse delle ruote

Tra forza e coppia sussiste la relazione $\tau = F \cdot R$ dunque posso scrivere Q come:

$$Q = \begin{pmatrix} \tau/R \\ -\tau \end{pmatrix} \quad (2.12)$$

Capitolo 3

Linear Quadratic Regulator

In questo capitolo viene studiato come può essere utilizzata la tecnica dell'Linear Quadratic Regulatori (LQR) per il bilanciamento il robot e per ottenere l'inseguimento di una traiettoria desiderata. Questo obiettivi vengono realizzati attraverso un'azione di *feedback*.^[1]

Feedback Capacità di un sistema dinamico di tenere conto dei risultati del sistema per modificare le caratteristiche del sistema stesso.

Prima di ricavare la legge di controllo è possibile riformulare il problema e passare dall'equazione 2.11 ad un sistema di equazioni differenziali ordinarie (ODE) del primo ordine, definendo lo stato del sistema come:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_z \\ \theta \\ \dot{x}_z \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} q \\ \dot{q} \end{pmatrix} \quad (3.1)$$

È possibile riscrivere il sistema come:

$$\dot{x} = f(x) + g(x)u \quad (3.2)$$

dove

$$\begin{aligned} f(x) &= \begin{pmatrix} x_3 \\ x_4 \\ \frac{m_b l I_b^{eq} \sin(x_2) x_4^2 - m_b^2 l^2 g \cos(x_2) \sin(x_2)}{\det(M)} \\ \frac{(m_b + m_w^{eq}) m_b l g \sin(x_2) - m_b^2 l^2 \cos(x_2) \sin(x_2) x_4^2}{\det(M)} \end{pmatrix} \\ g(x) &= \begin{pmatrix} 0 \\ 0 \\ \frac{I_b^{eq} + m_b l R \cos(x_2)}{\det(M)} \\ \frac{-((m_b + m_w^{eq}) R + m_b l \cos(x_2))}{\det(M)} \end{pmatrix} \\ u &= F = \frac{\tau}{R} \end{aligned} \quad (3.3)$$

$$\det(M) = (I_b + m_b l^2)(m_w + I_w/R^2) + m_b I_b + m_b^2 l^2 \sin^2(x_2)$$

In questa tesi si assume che l'intero stato sia disponibile. Esso può essere misurato direttamente o ricostruito mediante un osservatore.

3.1 Linearizzazione del sistema

In questa sezione si vuole determinare il modello lineare del sistema intorno ad un generico punto di equilibrio.

Un punto di equilibrio x_e è definito come $x_e \in \mathbb{R}^4$ tale che $f(x_e) = 0$, che implica $\dot{x}_e = 0$ quando l'ingresso è nullo (Formula 3.2). Può essere banalmente verificato che la configurazione verticale con velocità nulla rappresenta un punto di equilibrio instabile, attorno al quale è possibile linearizzare il problema attraverso un'approssimazione di Taylor al primo ordine.

$$x_e = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_{ze} \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.4)$$

Definendo la variabile $\tilde{x} = x - x_e$, il sistema linearizzato può essere scritto come:

$$\dot{\tilde{x}} = A\tilde{x} + B u \quad (3.5)$$

dove

$$A = \left. \frac{\partial f}{\partial t} \right|_{x=x_e} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{m_b^2 l^2 g}{\det(M_0)} & 0 & 0 \\ 0 & \frac{m_b l g (m_b + m_w^{eq})}{\det(M_0)} & 0 & 0 \end{pmatrix}$$

$$B = g(x_e) = \begin{pmatrix} 0 \\ 0 \\ \frac{I_b + m_b l(l+R)}{\det(M_0)} \\ \frac{-[m_b(l+R) + m_w^{eq}]}{\det(M_0)} \end{pmatrix} \quad (3.6)$$

$$\det(M_0) = (I_b + m_b l^2)(m_w + I_w/R^2) + m_b I_b$$

Scegliendo x_z come uscita possiamo infine definire

$$C = (1 \quad 0 \quad 0 \quad 0) \quad (3.7)$$

Il sistema è intuitivamente instabile. Possiamo verificarlo calcolando gli autovalori della matrice A e osservando che uno di essi ha parte reale positiva.

$$\lambda_1 = 0 \quad \lambda_2 = 0 \quad \lambda_3 = \sqrt{\frac{m_b l g (m_b + m_w^{eq})}{\det(M_0)}} \quad \lambda_4 = -\sqrt{\frac{m_b l g (m_b + m_w^{eq})}{\det(M_0)}} \quad (3.8)$$

3.2 Stabilizzazione nell'origine

Un approccio molto comune per la stabilizzazione di un sistema lineare consiste nell'applicare un feedback dallo stato. A tal fine è però necessario che il sistema sia controllabile: condizione che è possibile esaminare banalmente, verificando che la matrice di controllabilità abbia rango pieno.

Per un sistema controllabile, è possibile definire un feedback dallo stato della forma $u = -Kx$, con $x \in \mathbb{R}^n$ e $K \in \mathbb{R}^{1 \times n}$. La matrice dinamica del sistema diventa $A + BK$. Stabilizzare il sistema significa dunque scegliere un K tale per cui gli autovalori del sistema ad anello chiuso siano tutti a parte reale negativa.

La scelta di K non è univoca e il suo valore può essere determinato attraverso svariati approcci. In questa sede utilizzeremo la tecnica del Linear Quadratic Regulator, che si compone dei seguenti step.[\[3\]](#)

1. Definizione di una funzione di costo della forma

$$J = \int_0^\infty (x^T Q x + R u^2) dt \quad (3.9)$$

$Q = \text{diag}(Q_1, Q_2, Q_3, Q_4)$ rappresenta una matrice diagonale in cui ciascun elemento sulla diagonale determina la velocità di convergenza della corrispondente componente dello stato. R è uno scalare che penalizza l'uso di elevati input di controllo.

2. Risoluzione dell'equazione di Riccati nell'incognita P

$$A^T P + P A - P B R^{-1} B^T P + Q = 0 \quad (3.10)$$

3. Il guadagno che minimizza la funzione di costo J sarà dato da

$$K = -R^{-1} B^T P \quad (3.11)$$

Il sistema di controllo ha la seguente struttura:

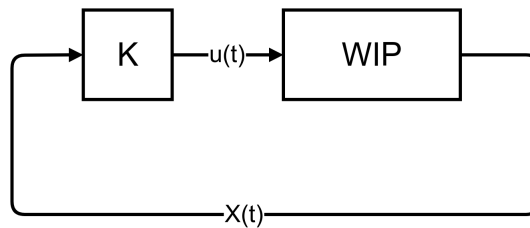
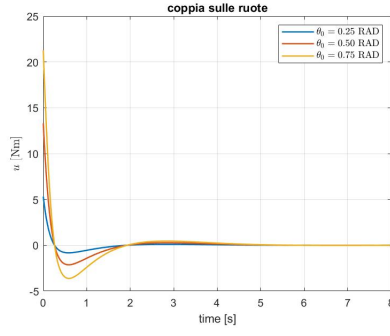
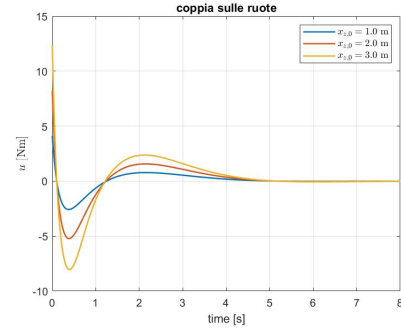
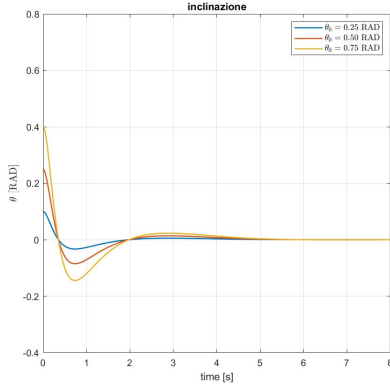
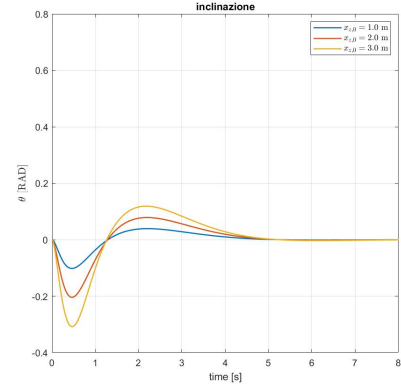
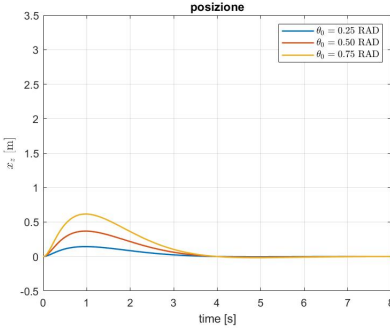
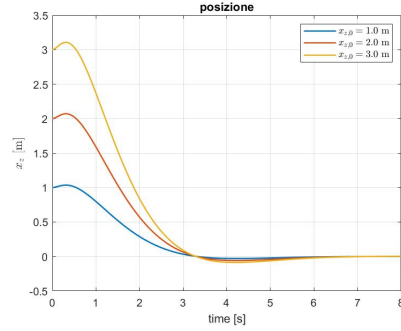


Figura 3.1. Schema di controllo per la stabilizzazione nell'origine

Il controllo è finalizzato a riportare il sistema nella posizione di equilibrio, che coincide con ascissa nulla e orientamento verticale. Le performance del controllore possono essere verificate osservando l'evoluzione del sistema al variare delle condizioni iniziali in [Figura 3.2](#).

(a) Coppia in ingresso al variare di θ_0 (b) Coppia in ingresso al variare di $x_{z,0}$ (c) Inclinazione al variare di θ_0 (d) Inclinazione al variare di $x_{z,0}$ (e) Posizione al variare di θ_0 (f) Posizione al variare di $x_{z,0}$ **Figura 3.2.** Stabilizzazione nell'origine con LQR

La Figura 3.2 mostra l'evoluzione di u , θ e x_z al variare delle condizioni iniziali. La colonna sinistra (3.2a, 3.2c, 3.2e) prende in considerazione posizione e velocità nulla, ma inclinazione iniziale del robot diversa da zero.

La colonna destra invece (3.2b, 3.2d, 3.2f) assume come stato iniziale una configurazione verticale con velocità nulla, a partire da un'ascissa diversa da zero.

Come è possibile notare dai grafici 3.2a e 3.2b, in entrambi i casi, l'azione di controllo inizia con un'elevata coppia applicata alle ruote. In 3.2a l'intensità della coppia iniziale è più elevata, perché ha lo scopo di riportare il sistema in posizione verticale, ed evitare che l'inclinazione diverga. Dal grafico 3.2c osserviamo che il sistema

raggiunge la posizione verticale dopo circa 2 secondi. In questo istante di tempo però l'ascissa è ancora lontana dall'origine (3.2e), che invece viene raggiunta lentamente, mantenendo l'inclinazione intorno a zero.

Nella colonna di destra, la dinamica risulta più omogenea: i valori massimi di u e θ sono inferiori rispetto ai corrispondenti grafici della colonna di sinistra. In 3.2b osserviamo le seguenti fasi: la coppia iniziale ha lo scopo di fornire un'inclinazione non nulla al corpo del robot, in modo da poter accelerare successivamente senza che θ diverga; nella seconda fase il robot accelera e la ruota supera il centro di massa del corpo nella direzione di moto del robot; infine il robot decelera riportando il corpo ad un assetto verticale.

Nei grafici 3.2b e 3.2d dopo circa 1.3 secondi sia u che θ sono nulli. In questo istante contemporaneamente entrambe l'accelerazione e l'inclinazione del robot passano da valori negativi a valori positivi.

È interessante osservare che al variare del discostamento iniziale dall'assetto di equilibrio desiderato, il tempo di convergenza resti pressoché invariato, mentre vari considerevolmente lo sforzo di controllo.

3.3 Traiettoria costante

Immaginiamo di voler formulare una legge di controllo che consenta al robot, a partire dalla condizione iniziale di arrivare ad una nuova ascissa di equilibrio $x_{z,d}$, in cui fermarsi in assetto verticale. A partire dall'ascissa finale desiderata definiamo lo stato finale desiderato: $x_d = (x_{z,d}, 0, 0, 0)$.

Il WIP è un sistema non lineare, che presenta infiniti punti di equilibrio, corrispondenti ad una qualunque configurazione con velocità nulla e orientamento verticale, lungo l'asse delle ascisse. Stabilizzare il sistema nell'origine dunque è analogo a farlo in qualsiasi altro punto, con l'unica differenza che invece di minimizzare lo stato, il feedback dovrà minimizzare l'errore $\tilde{x} = x(t) - x_d$. Fintanto che x_d è costante nel tempo, la dinamica dell'errore e dello stato coincidono ($\dot{\tilde{x}} = \dot{x}$): i due sistemi saranno caratterizzati dalle stesse matrici A e B . Anche i rispettivi guadagni K calcolati dall'LQR (a parità di Q ed R) saranno identici.

Usando un feedback del tipo $u = -K\tilde{x} = -K(x(t) - x_d)$ otterrò un controllore in grado di far convergere l'errore a 0.

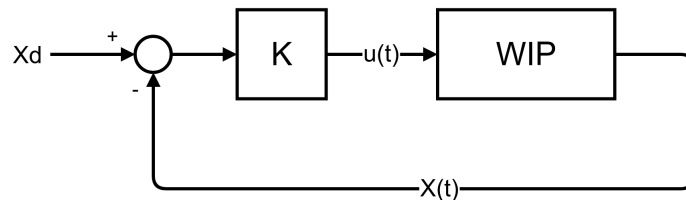


Figura 3.3. Schema di controllo per la stabilizzazione su una traiettoria costante

In Figura 3.4 è possibile osservare come i valori assegnati ai coefficienti della funzione di costo dell'LQR influenzino la velocità di convergenza della dinamica

da uno stato iniziale $x_0 = (0, 0, 0, 0)$ ad uno stato desiderato $x_d = (2, 0, 0, 0)$. Consideriamo un set di parametri di riferimento $\{1\}$ per il quale le performance del controllore siano accettabili. Dopodiché consideriamo 3 ulteriori set di parametri in cui incrementiamo il peso rispettivamente di posizione $\{2\}$, angolo rispetto alla verticale $\{3\}$, sforzo di controllo $\{4\}$. Aumentando il valore di uno di questi parametri stiamo modificando il peso che l'errore su quello specifico parametro avrà sulla funzione di costo complessiva.

$$\begin{aligned} \{1\} &= \begin{cases} Q = \text{diag}([10, 1, 1, 1]) \\ R = 0.01 \end{cases} \\ \{2\} &= \begin{cases} Q = \text{diag}([50, 1, 1, 1]) \\ R = 0.01 \end{cases} \\ \{3\} &= \begin{cases} Q = \text{diag}([1, 5, 1, 1]) \\ R = 0.01 \end{cases} \\ \{4\} &= \begin{cases} Q = \text{diag}([10, 1, 1, 1]) \\ R = 0.05 \end{cases} \end{aligned} \quad (3.12)$$

In Figura 3.4 viene mostrata la dinamica di u , θ e x_z applicando i set di parametri indicati in 3.12.

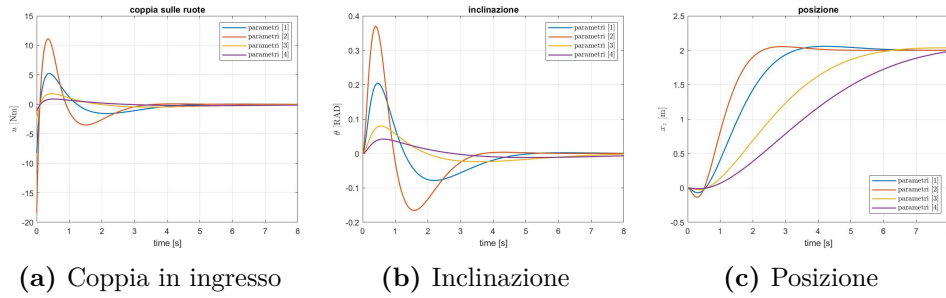


Figura 3.4. Stabilizzazione ad una traiettoria costante al variare dei parametri LQR

Le performance migliori sono ottenute dalla configurazione $\{2\}$, seguita dalle configurazioni $\{1\}$, $\{3\}$ e $\{4\}$, come mostrato dalla velocità di convergenza nel grafico 3.4c. A performance migliori corrisponde però uno sforzo di controllo maggiore, che si traduce in coppie più elevate (grafico 3.4a) e inclinazioni maggiori in modulo (grafico 3.4b).

Coerentemente con la scelta dei parametri effettuata, il grafico corrispondente al set $\{2\}$ garantisce la maggiore velocità di convergenza della posizione e il set $\{4\}$ alla minor coppia in ingresso. Potrebbe tuttavia sorprendere il fatto che nel grafico 3.4b il picco minimo non sia generato dal set $\{3\}$ ma piuttosto dal $\{4\}$. Questo fenomeno è dovuto alla particolare scelta dei parametri: sebbene sia possibile aumentare il peso di uno specifico parametro, l'LQR non ignora gli altri, che quindi hanno un contributo, anche se minore, nel calcolo di K . Aumentando a sufficienza il peso di θ e riducendo quello di u , sarebbe stato possibile ottenere un picco del set $\{3\}$ minore rispetto al set $\{4\}$ nel grafico 3.4b. È importante però osservare che l'inclinazione e l'ingresso sono fortemente legati tra loro. Una maggiore inclinazione, per essere riportata a zero, necessita di una maggiore accelerazione, e dunque una maggiore coppia. Per questo motivo minimizzare la coppia o l'inclinazione producono effetti simili.

3.4 Traiettoria a velocità costante

In questa sezione vogliamo capire come realizzare un controllo in grado di produrre un moto a velocità costante del robot. Lo stato desiderato non è costante, a differenza di quanto detto nella Sezione 3.3. Poiché però la stabilità di un punto di equilibrio è invariante rispetto all'ascissa del robot, posso usare lo stesso schema di controllo del caso 3.1.3 aggiornando ad ogni passo l'ascissa desiderata. Al suo posto posso mettere il valore corrente di x_z o addirittura il valore che avrebbe se il robot si muovesse di moto rettilineo uniforme ($x_{z,d}(t) = v_d \cdot t$). In quest'ultimo caso, la funzione $x_d(t)$ è nota a priori e l'errore è calcolato come $x(t) - x_d(t)$, dove $x_d(t) = (v_d \cdot t, 0, v, 0)$. Lo schema di controllo è il seguente:

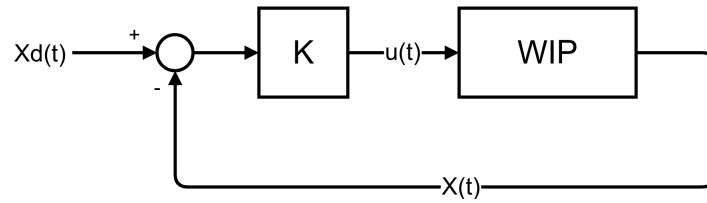


Figura 3.5. Schema di controllo per la stabilizzazione su una traiettoria non costante

In Figura 3.6 sono riportate le dinamiche della coppia in ingresso, della posizione e dell'errore ($x(t) - x_d(t)$).

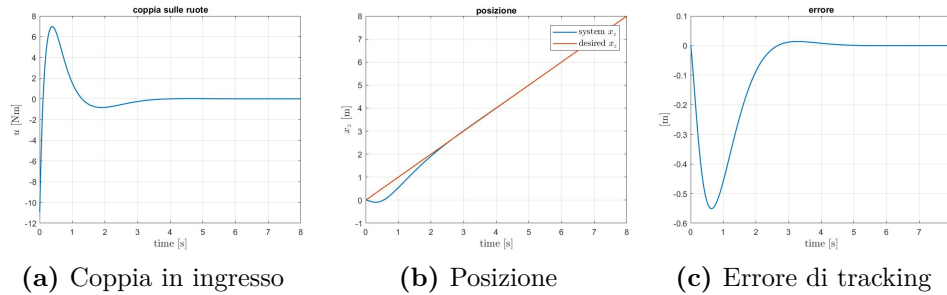


Figura 3.6. Stabilizzazione su una traiettoria a velocità costante

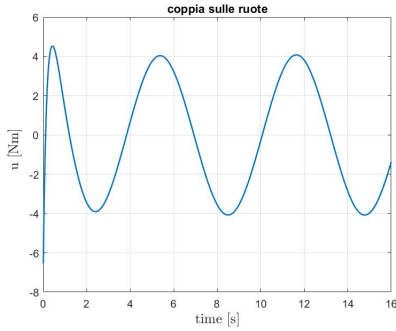
In questa simulazione, la traiettoria desiderata ha velocità costante. La velocità iniziale del robot però è nulla, dunque dovrà essere applicata una coppia iniziale per accelerare il robot fino al raggiungimento della velocità desiderata. Il valore iniziale di tale coppia è negativo poiché la ruota del robot deve accelerare leggermente a sinistra per inclinare il corpo verso destra e accelerare successivamente verso destra. Possiamo notare che l'ingresso (grafico 3.6a) e l'errore (grafico 3.6c) presentano delle curve qualitativamente opposte. In questi due grafici è presente un leggero "overshoot", dovuto al fatto che per recuperare l'errore che si genera nei primi istanti, il robot deve raggiungere una velocità superiore a quella di regime, e dunque è costretto in un secondo momento a decelerare.

3.5 Traiettoria sinusoidale

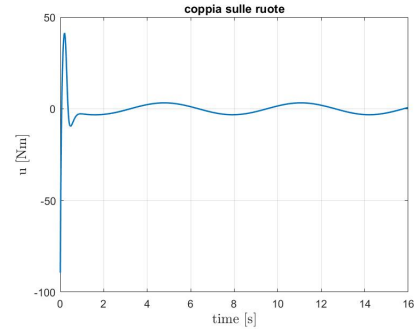
Consideriamo il caso in cui la traiettoria desiderata sia una senoide $x_{z,d}(t) = \sin(t)$. Proviamo ad applicare un controllo simile a quello applicato nella sezione 3.4, ovvero un feedback del tipo $u(t) = -K(x(t) - x_d(t))$ dove $x_d(t) = (\sin(t), 0, \cos(t), 0)$.

La simulazione è stata svolta con dei valori di riferimento dei parametri LQR ($\{1\}$). Per ottenere performance migliori, la simulazione è poi stata ripetuta con i parametri $\{2\}$, in grado di realizzare un inseguimento di traiettoria più "aggressivo".

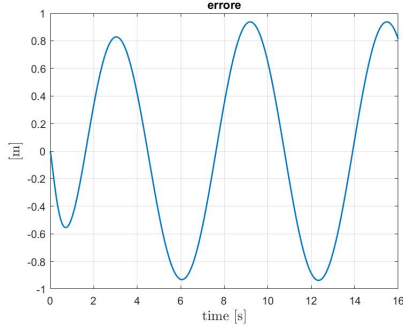
$$\{1\} = \begin{cases} Q = \text{diag}([10, 1, 1, 1]) \\ R = 0.01 \end{cases} \quad \{2\} = \begin{cases} Q = \text{diag}([1000, 1, 1, 1]) \\ R = 0.001 \end{cases} \quad (3.13)$$



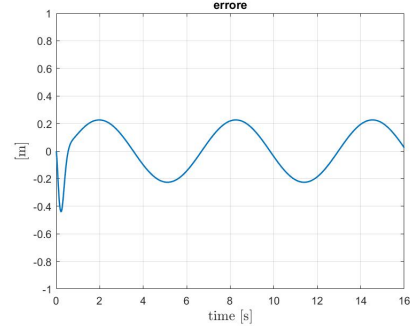
(a) Coppia in ingresso con parametri $\{1\}$



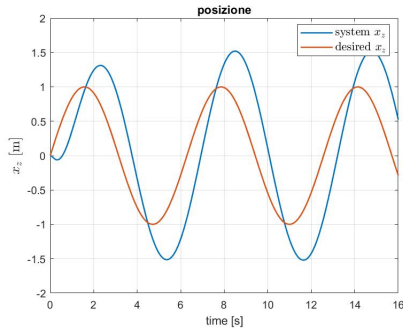
(b) Coppia in ingresso con parametri $\{2\}$



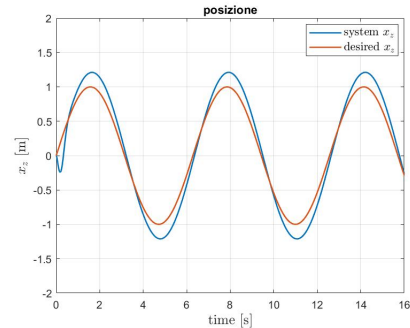
(c) Errore con parametri $\{1\}$



(d) Errore con parametri $\{2\}$



(e) Posizione con parametri $\{1\}$



(f) Posizione con parametri $\{2\}$

Figura 3.7. Stabilizzazione intorno ad una traiettoria sinusoidale mediante LQR

Confrontando i grafici 3.7a e 3.7b è evidente che lo sforzo di controllo iniziale nel caso della scelta dei parametri $\{2\}$ è notevolmente superiore rispetto al caso $\{1\}$. Un maggiore sforzo di controllo è tuttavia associato ad un minore errore nell'inseguimento della traiettoria desiderata (grafici 3.7c e 3.7d), sia nella fase che nel modulo dell'oscillazione (grafici 3.7e e 3.7f)

3.6 Considerazioni sull'LQR

Come abbiamo visto, la tecnica dell'LQR mostra performance soddisfacenti nelle sezioni 3.2, 3.3, 3.4, mentre ha prestazioni visibilmente peggiori nella sezione 3.5.

Nei primi due casi la traiettoria di riferimento è costante (tempo invariante), un opportuno feedback dallo stato è dunque sufficiente per stabilizzare il sistema portando l'errore $(x(t) - x_d)$ a zero.

Nel caso con velocità costante, la traiettoria desiderata diventa una funzione del tempo. A livello teorico, il controllo LQR non sarebbe in grado di stabilizzare efficacemente questo tipo di sistemi. Tuttavia, il WIP è un sistema non lineare che presenta infiniti punti di equilibrio corrispondenti alle infinite posizioni che può assumere lungo l'ascissa, mantenendo orientamento verticale e velocità nulla. Se si considerano due sistemi di riferimento, in moto rettilineo uniforme l'uno rispetto all'altro, stabilizzare un sistema intorno ad una traiettoria a velocità costante nel sistema di riferimento fisso coincide esattamente a stabilizzare lo stesso sistema intorno all'origine nel sistema di riferimento in moto. Poiché la stabilità di un punto di equilibrio non dipende dal sistema di riferimento considerato, anche in questo caso il un feedback K sarà in grado di garantire stabilità.

Nella tecnica LQR, il calcolo del vettore K fa riferimento ad un punto di equilibrio costante nel tempo. Se tale controllore viene applicato ad un sistema in cui la traiettoria desiderata varia nel tempo, con un accelerazione non nulla, le sue performance nel mantenere la traiettoria desiderata sono scadenti, e presenteranno ritardi, come vediamo in Figura 3.7. Un controllo basato solo su feedback non è in grado di replicare una traiettoria desiderata con accelerazione non costante. Il motivo può essere spiegato intuitivamente considerando come viene generato l'input del sistema (la coppia applicata alle ruote) $u = -K(x(t) - x_d(t))$. Nel momento in cui il robot riuscisse a trovarsi esattamente sulla traiettoria desiderata, l'errore sullo stato sarebbe nullo, e conseguentemente anche u varrebbe zero. Per il primo principio della dinamica, un corpo non soggetto a forze esterne si muove a velocità costante. L'accelerazione del robot allora diventerebbe nulla, causando l'allontanamento immediato dalla traiettoria desiderata.

Come visto nella sezione 3.5 la performance può essere migliorata modificando i valori dei parametri dell'LQR, tuttavia per quanto detto finora il tracking basato solamente su *feedback* non potrà mai essere esatto.

Capitolo 4

Intrinsically-Stable Model Predictive Control

In questo capitolo verranno affrontati i problemi di balancing e trajectory tracking, già incontrati nel Capitolo 3, mediante la metodologia dell’Intrinsically-Stable Model Predictive Control (IS-MPC). Questo approccio combina tre tecniche principali:

- *model predictive control*, per l’inseguimento della traiettoria desiderata;
- *input-output feedback linearization*, per la linearizzazione del modello;^[4]
- *linear stable inversion*, per la stabilità della zero dynamics.^[2]

Il MPC (Model Predictive Control) è una tecnica per il controllo ottimo, adatta per real-time applications. Il suo funzionamento si basa sulla risoluzione di un problema di ottimizzazione in un certo intervallo di tempo nel futuro, che prende il nome di *receding horizon*. La soluzione corrisponde alla sequenza ottimale di input da fornire al sistema tali da minimizzare una certa funzione di costo; dopodiché viene applicato al sistema solamente il primo ingresso della sequenza. Ad ogni iterazione lo stato viene aggiornato, il receding horizon viene spostato in avanti e il ciclo si ripete con una nuova ottimizzazione.

Il vantaggio dell’MPC è la possibilità di specificare vincoli di uguaglianza e disuguaglianza. La soluzione è un compromesso tra minimizzazione dell’errore e degli sforzi di controllo. Il problema della stabilità della zero dynamics viene risolto proprio inserendo un vincolo che, come vedremo, è ottenuto dalla *linear stable inversion*.

Un problema di ottimizzazione è solitamente espresso come una funzione di costo da minimizzare ed una serie di vincoli di uguaglianza e disuguaglianza:

$$\begin{aligned} \chi^* = \operatorname{argmin}_{\chi} J(\chi) &= \operatorname{argmin}_{\chi} \frac{1}{2} \chi^T H \chi + c^T \chi \\ &\begin{cases} A_{disug} \cdot \chi \leq b_{disug} \\ A_{ug} \cdot \chi = b_{ug} \end{cases} \end{aligned} \tag{4.1}$$

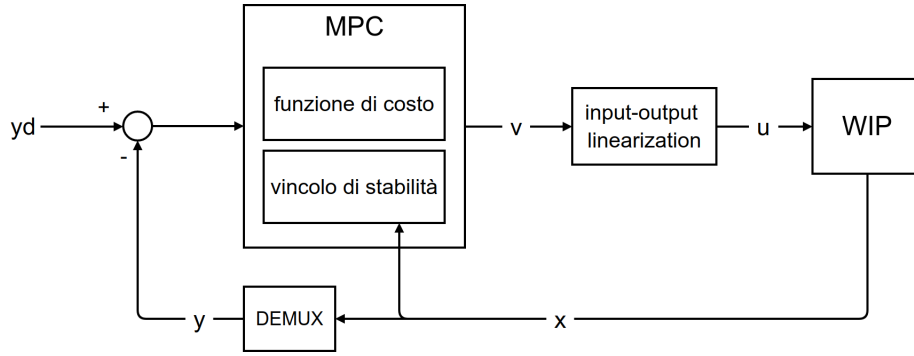


Figura 4.1. Schema di controllo per il trajectory tracking mediante IS-MPC

Lo schema di controllo applicato è riportato in Figura 4.1. L'ottimizzazione avviene all'interno del blocco 'MPC', che contiene al suo interno la funzione di costo (sezione 4.5) e il vincolo di stabilità (sezione 4.6). Dall'MPC viene generata una sequenza ottimale di input di cui solo il primo (v) verrà applicato al sistema. Come spiegato nella sezione 4.1, trattandosi di un sistema non lineare, è necessario trasformare l'ingresso v nell'ingresso reale u da applicare al sistema. Lo stato del sistema, interamente osservabile, è poi usato dall'ottimizzatore per la successiva iterazione.

4.1 Trasformazione di coordinate e linearizzazione

Consideriamo il sistema non ancora linearizzato, ricavato aggiungendo un'uscita alla 3.2:

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) = x_1 = x_z \end{cases} \quad (4.2)$$

Attraverso una trasformazione di coordinate è possibile riscriverlo nella forma normale di Byrnes-Isidori. [1]

$$\zeta = \begin{pmatrix} z_1 \\ z_2 \\ \eta_1 \\ \eta_2 \end{pmatrix} = T(x) = \begin{pmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{pmatrix} \quad (4.3)$$

Per non appesantire la notazione, introduciamo la derivata di Lie. Il sistema può essere scritto come:

$$\begin{aligned} \dot{z} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} &= \begin{pmatrix} L_f^2 h(x) + u \cdot L_f L_g h(x) \\ z_2 \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \dot{y} \end{pmatrix} \\ \dot{\eta} = \begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} &= a_1(z, \eta) + a_2(z, \eta)u \end{aligned} \quad (4.4)$$

dove $a_1(z, \eta)$ e $a_2(z, \eta)$ non sono riportate esplicitamente per semplicità.

Il sistema resta non lineare, tuttavia in questa forma vengono messe in evidenza la dinamica ingresso-uscita (equazioni in z) e la zero dynamics (equazioni in η). Useremo le prime per realizzare il trajectory tracking sull'uscita, e le seconde per garantire la non-divergenza delle dinamiche interne.

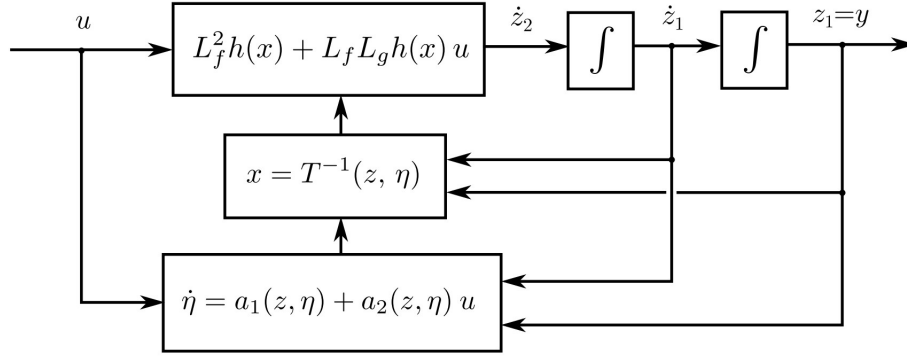


Figura 4.2. Schema di controllo nella forma normale di Byrnes-Isidori

4.2 Trajectory tracking

Il problema del trajectory tracking consiste nel formulare un controllo in grado di far convergere l'uscita del sistema (posizione lungo l'asse delle ascisse) ad una traiettoria desiderata. Come accennato, le seguenti equazioni non lineari rappresentano il legame tra ingresso e uscita.

$$\dot{z} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} z_2 \\ L_f^2 h(x) + u \cdot L_f L_g h(x) \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \ddot{y} \end{pmatrix} \quad (4.5)$$

Per eliminare la non linearità utilizzo una tecnica chiamata "*input-output feedback linearization*". [4] Introduco la variabile v tale che:

$$u = \frac{1}{L_f L_g h(x)} (v - L_f^2 h(x)) \quad (4.6)$$

Nel grafico 4.2, v rappresenta l'intero blocco a sinistra dei due integratori. Questa sostituzione ci permette di scrivere la relazione lineare $\dot{z}_2 = v = \ddot{x}_z = \ddot{y}$ che lega l'ingresso modificato v e la derivata seconda dell'uscita. Il sistema diventa quindi un doppio integratore e può essere riscritto come:

$$\dot{z} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + v \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \ddot{y} \end{pmatrix} \quad (4.7)$$

Se dunque chiamiamo $x_z^*(t)$ la traiettoria desiderata, possiamo fare in modo che sia inseguita dal sistema scegliendo come ingresso $v^*(t) = \ddot{x}_z^*(t)$. Usando la formula 4.6 potremo calcolarci il conseguente ingresso da fornire al sistema $u^*(t)$ per ottenere l'accelerazione desiderata.

Riassumendo, abbiamo scomposto il problema del trajectory tracking in due parti:

1. Linearizzazione della dinamica ingresso-uscita con la tecnica dell'*input-output feedback linearization*
2. Controllo del sistema lineare mediante *feedforward*, fornendo direttamente l'accelerazione desiderata, generata dall'MPC, in input.

4.3 Stabilità della Zero Dynamics

Riprendendo la formula 4.4 dopo l'introduzione della variabile v :

$$\dot{\eta} = \begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} = \begin{pmatrix} \eta_2 \\ \frac{m_b l (g + R \eta_2^2) \sin(\eta_1) - v((m_b + m_w^{eq})R + m_b l \cos(\eta_1))}{I_b + m_b l (l + R \cos(\eta_1))} \end{pmatrix} \quad (4.8)$$

Queste equazioni prendono il nome di *forced zero dynamics*. Per definizione la *zero dynamics* è la dinamica di η quando l'uscita è nulla. Per quanto affermato nella sezione 4.2, questa condizione è verificata per $v = 0$. La zero dynamics è la seguente:

$$\dot{\eta} = \begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} = \begin{pmatrix} \eta_2 \\ \frac{m_b l (g + R \eta_2^2) \sin(\eta_1)}{I_b + m_b l (l + R \cos(\eta_1))} \end{pmatrix} \quad (4.9)$$

Il punto di equilibrio corrisponde ad $\eta = 0$, in corrispondenza del quale il sistema è intrinsecamente instabile. Ciò può essere facilmente dimostrato linearizzando la formula 4.8 intorno al punto di equilibrio ed osservando che la matrice dinamica del sistema linearizzato presenta un autovalore a parte reale positiva.

$$\dot{\eta} = \begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \alpha^2 & 0 \end{pmatrix} \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} + v \begin{pmatrix} 0 \\ -\beta \end{pmatrix} = A_0 \eta + B_0 v \quad (4.10)$$

dove $\alpha = \sqrt{\frac{m_b l g (m_b + m_w^{eq})}{\det(M_0)}}$ e $\beta = \frac{[m_b(l+R) + m_w^{eq}R]}{\det(M_0)}$. Gli autovalori della *linearized forced zero dynamics* sono $\lambda_{1,2} = \pm\alpha$, di cui uno a parte reale positiva.

Attraverso una opportuna trasformazione di coordinate è possibile separare la componente stabile da quella instabile.

$$\eta = T^{-1} \tilde{\eta} = T^{-1} \begin{pmatrix} \eta_s \\ \eta_u \end{pmatrix} \quad (4.11)$$

dove T è tale che:

$$T A_0 T^{-1} = \begin{pmatrix} A_s & 0 \\ 0 & A_u \end{pmatrix} \quad (4.12)$$

A_s (stable) contiene solo gli autovalori a parte reale negativa, mentre A_u (unstable) i rimanenti.

La legge di controllo che realizza il trajectory tracking deve anche garantire la non divergenza della componente instabile della zero dynamics. A tal fine useremo una tecnica conosciuta come *linear stable inversion* [2], che possiamo riassumere come segue.

Linear Stable Inversion La dinamica di η_u è nota essere:

$$\begin{aligned} \eta_u(t) &= e^{A_u t} \eta_u(0) + \int_0^t e^{A_u(t-\tau)} B_u v(\tau) d\tau \\ &= e^{A_u t} \left(\eta_u(0) + \int_0^t e^{-A_u(\tau)} B_u v(\tau) d\tau \right) \end{aligned} \quad (4.13)$$

Dal momento in cui $v(t) = \ddot{x}_{des}(t)$, tutti i termini della parte destra dell'equazione sono noti, ad eccezione di $\eta_u(0)$. Se scelgo la seguente condizione iniziale particolare

$$\eta_u^*(0) = - \int_0^\infty e^{-A_u \tau} B_u v(\tau) d\tau \quad (4.14)$$

la dinamica di η_u diventa la seguente:

$$\eta_u^*(t) = - \int_t^\infty e^{A_u(t-\tau)} B_u v(\tau) d\tau = - \int_0^\infty e^{-A_u \nu} B_u v(t + \nu) d\nu \quad (4.15)$$

Sebbene si tratti di un'integrazione su un intervallo di tempo infinito, se l'input (noto) non diverge non lo farà nemmeno $\eta_u^*(t)$, poiché l'ingresso viene moltiplicato per un esponenziale con esponente negativo. Dunque con questa particolare condizione iniziale otteniamo la non divergenza della zero dynamics.

4.4 Discretizzazione del sistema

L'MPC opera ad istanti di tempo discreti, è dunque necessario discretizzare il sistema. Consideriamo l'equazione 4.7, che riscriviamo per chiarezza.

$$\dot{z} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + v \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \ddot{y} \end{pmatrix} \quad (4.16)$$

Assumendo l'ingresso costante nell'intervallo di tempo unitario, il sistema discretizzato è definito come segue, scegliendo opportunamente le matrici \bar{A} e \bar{B} .

$$z^{k+1} = \bar{A}z^k + \bar{B}v^k \quad (4.17)$$

Per applicare l'MPC deve essere definita una funzione di costo e dei vincoli.

4.5 Funzione di costo

La funzione di costo sarà ottenuta dalla somma di due componenti principali.

- Errore di tracciamento atteso.

Considero un numero N_c di istanti di tempo del receding horizon, e calcolo l'output per ciascuno di essi.

$$Y^k = \begin{pmatrix} y^{k+1} \\ y^{k+2} \\ \vdots \\ y^{k+N_c} \end{pmatrix} = \begin{pmatrix} C\bar{A}z^k + C\bar{B}v^k \\ C\bar{A}^2z^k + C\bar{A}\bar{B}v^k + C\bar{B}v^{k+1} \\ \vdots \\ C\bar{A}^{N_c}z^k + C\sum_{i=0}^{N_c-1} \bar{A}^{N_c-1-i} \bar{B}v^{k+1} \end{pmatrix} = \quad (4.18)$$

$$= \Omega z^k + \Gamma V_k$$

Il vettore V_k è proprio quello che desidero ottimizzare. Calcolo il quadrato della differenza tra l'output desiderato e output reale. Questo errore viene poi moltiplicato per un fattore scalare che ne indica il peso.

$$J_y = \frac{1}{2} w_y \|Y^k - Y_d^k\|^2 \quad (4.19)$$

- Sforzo di controllo. Questo termine è necessario per evitare la divergenza dell'input.

$$J_e = \frac{1}{2} w_e \|V^k\|^2 \quad (4.20)$$

Per rendere la dinamica più fluida potrebbe inoltre essere aggiunto un termine che minimizza la differenza tra input adiacenti. I pesi da attribuire a ciascun termine dipendono dalle esigenze di performance e dallo sforzo di controllo a disposizione.

$$J = J_y + J_e \quad (4.21)$$

4.6 Vincolo di stabilità

L'aggiunta di vincoli implica che non è garantito il raggiungimento di una soluzione, tuttavia essi sono necessari per la stabilità del sistema. In questa sede ci limiteremo a definire il vincolo di stabilità, ma ulteriori vincoli su ingresso e uscita potrebbero essere specificati.

Il vincolo di stabilità è ottenuto a partire dalla formula 4.14, che riportiamo per chiarezza.

$$\eta_u^*(0) = - \int_0^\infty e^{-A_u \tau} B_u v(\tau) d\tau \quad (4.22)$$

$\eta_u^*(0)$ rappresenta la condizione iniziale a partire dalla quale $\eta_u(t)$ non diverge. In questa formula, $v(t)$ rappresenta l'incognita da determinare. Ponendo questo vincolo, la soluzione (se esiste) che verrà restituita dall'ottimizzatore, sarà una sequenza ottimale di ingressi per cui è rispettata questa condizione. La *linear stable inversion* viene utilizzata per elaborare un vincolo che garantisca la non divergenza della zero dynamics.

Adattando il problema al caso discreto, la condizione iniziale 4.22 diventa:

$$\eta_u(t_k) = - \underbrace{\int_{t_k}^{t_k+T_c} e^{-A_u(\tau-t_k)} B_u v(t) d\tau}_{\text{orizzonte di controllo}} - \underbrace{\int_{t_k+T_c}^\infty e^{-A_u(\tau-t_k)} B_u v(t) d\tau}_{\text{coda}} \quad (4.23)$$

In questa tesi ignoreremo il contributo della coda. Questo può essere fatto per la presenza del termine esponenziale, che rende il suo peso sufficientemente piccolo da essere trascurato. Sotto l'ipotesi di input costante nel singolo intervallo di tempo, possiamo discretizzare l'integrale, e riscrivere la 4.22 nella seguente forma matriciale compatta:

$$A_{eq} \cdot V^k = \eta_u(t_k) \quad (4.24)$$

Questa equazione rappresenta il vincolo di stabilità da inserire nel problema di ottimizzazione per garantire la stabilità della zero dynamics.

4.7 Traiettoria costante

In questa sezione vengono analizzate le performance del controllore nel replicare la traiettoria costante $x_d = (2, 0, 0, 0)$, come visto nella sezione 3.3. A tal fine però verrà utilizzata la tecnica dell'IS-MPC invece dell'LQR. In Figura 4.3 viene mostrata l'evoluzione della coppia in ingresso, dell'inclinazione, della posizione, nonché dell'errore di tracking.

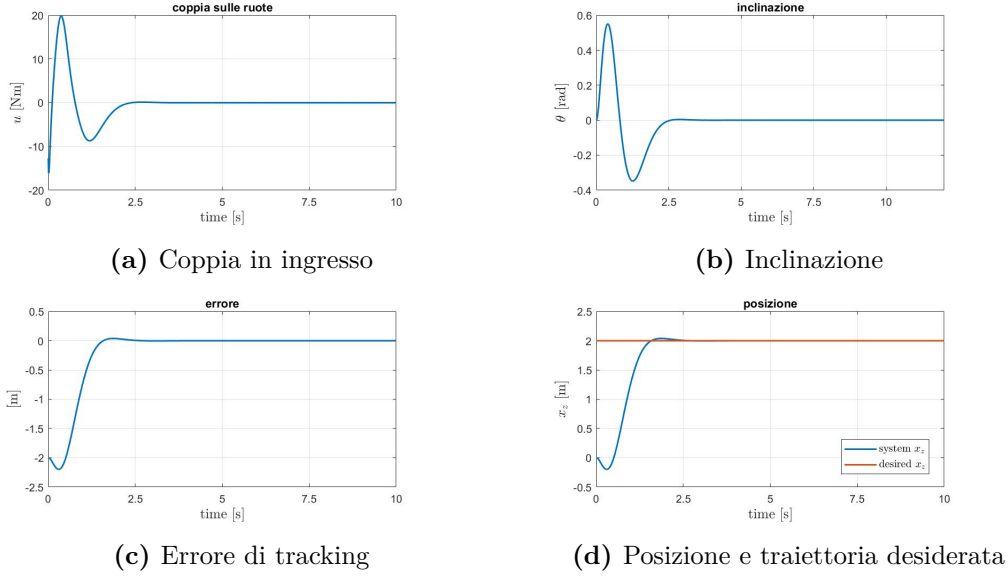


Figura 4.3. Stabilizzazione intorno ad una traiettoria costante mediante IS-MPC

Il controllore impiega circa 2.5 secondi per far convergere l'errore a zero (grafico 4.3c). Confrontando le Figure 4.3 e 3.4 osserviamo che l'IS-MPC garantisce una convergenza più rapida rispetto al LQR, al costo però di un maggiore sforzo di controllo. Entrambe le tecniche però garantiscono la convergenza del transitorio per questo tipo di traiettoria.

4.8 Traiettoria sinusoidale

In questa sezione studiamo un'azione di controllo finalizzata all'inseguimento di una traiettoria sinusoidale. In Figura 4.4 e 4.5 vengono riportate le performance del controllore al variare dei pesi w_y e w_e nella funzione di costo. Nel caso {2} è stato ridotto il contributo dello sforzo di controllo alla funzione di costo complessiva.

$$\{1\} = \begin{cases} w_y = 1 \\ w_e = 0.01 \end{cases} \quad \{2\} = \begin{cases} w_y = 1 \\ w_e = 0.001 \end{cases} \quad (4.25)$$

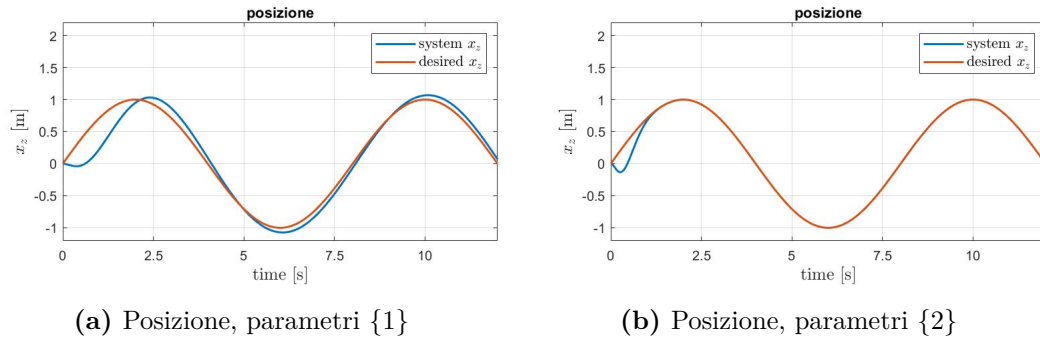


Figura 4.4. Traiettoria reale e desiderata, con parametri {1} (a sinistra) e {2} (a destra)

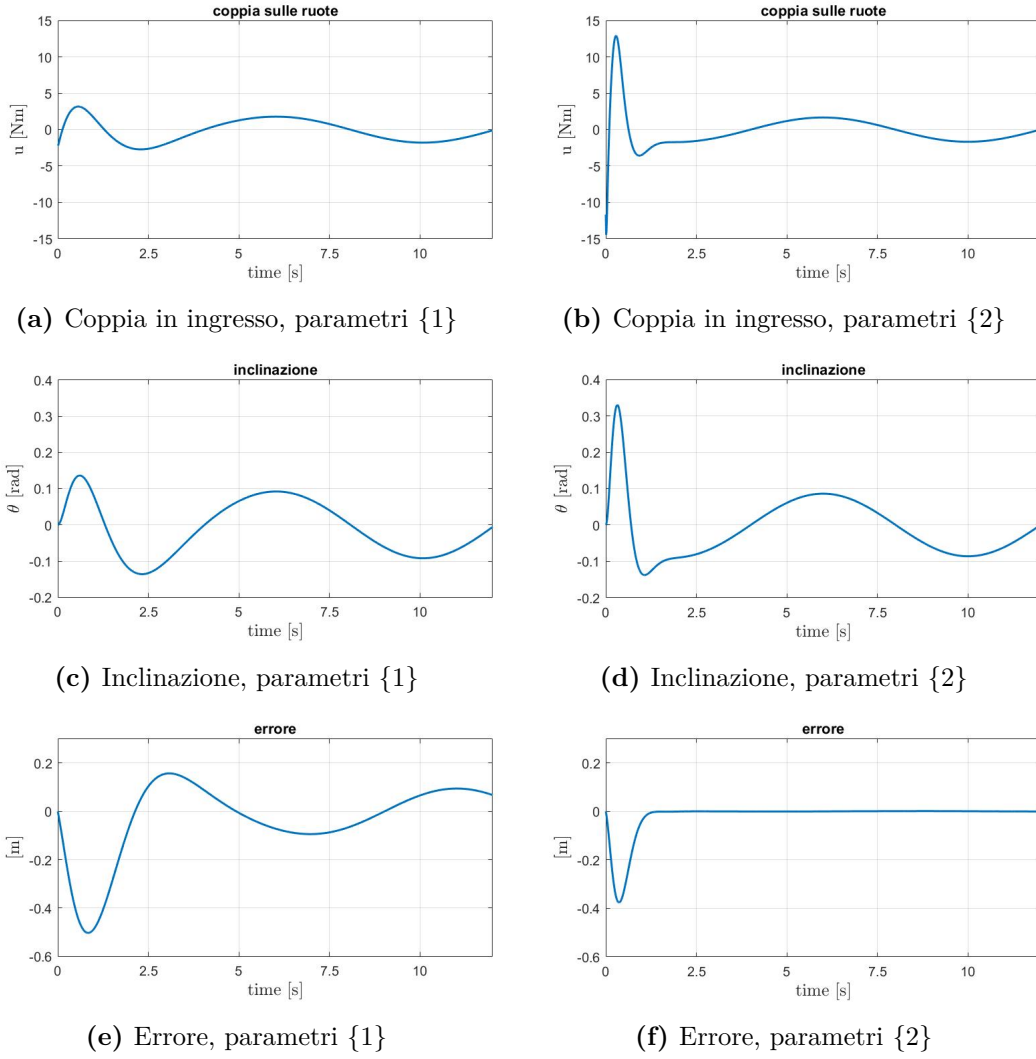


Figura 4.5. Stabilizzazione intorno ad una traiettoria sinusoidale mediante IS-MPC, con parametri {1} (a sinistra) e {2} (a destra)

Come è possibile osservare dai grafici 4.4a e 4.4b, i pesi w_y e w_e hanno un'influenza rilevante sulle performance del controllore. Riducendo il peso dell'ingresso infatti (caso {2}) l'errore di tracking può essere reso nullo (grafico 4.5f) grazie ad uno sforzo di controllo superiore (grafici 4.5a e 4.5b).

Capitolo 5

Conclusioni

In questa tesi è stato affrontato il problema del controllo di un Wheeled Inverted Pendulum. Gli obiettivi di controllo sono stati il bilanciamento e il trajectory tracking e le tecniche utilizzate sono state il 'Linear Quadratic Regulator' (Capitolo 3) e il 'Intrinsically-Stable Model Predictive Control' (Capitolo 4).

Linear Quadratic Regulator Il controllo è stato realizzato mediante un feedback che prende in considerazione l'errore tra lo stato desiderato e lo stato reale. La matrice del guadagno K è stata calcolata con la tecnica dell'LQR.

IS - Model Predictive Control Il problema di controllo viene trasformato in un problema di ottimizzazione. La funzione di costo (da ottimizzare) dipende dall'input e dall'errore tra l'uscita prevista e quella desiderata (non dall'intero stato).

La differenza sostanziale tra le due tecniche risiede nella funzione di costo e nella stabilizzazione della zero dynamics.

Nella caso dell'LQR, la funzione di costo prende in considerazione l'intero stato e l'ingresso. Lo stato ricordiamo essere definito come $x = (x_z, \theta, \dot{x}_z, \dot{\theta})$. Ad ognuna delle componenti di x è associato un peso, che indica il contributo che l'errore su quella specifica componente ha sulla funzione di costo complessiva. La stabilità della zero dynamics è ottenuta come conseguenza del fatto che θ e $\dot{\theta}$ hanno un peso non nullo nella funzione di costo. Il calcolo del guadagno K dunque sarà tale (entro certi limiti) da evitare la divergenza dell'angolo e della sua derivata.

L'IS-MPC invece separa i problemi di trajectory tracking e stabilità della zero dynamics. La funzione di costo che si vuole minimizzare non tiene conto dell'intero, bensì solo del discostamento dell'uscita (ascissa) rispetto all'uscita desiderata. Mentre uno specifico vincolo, ottenuto dalla *linear stable inversion*, viene introdotto per garantire la non divergenza delle dinamiche interne.

La presenza o meno della zero dynamics nella funzione di costo ha un impatto sulle performance di controllo e sulle traiettorie che è possibile inseguire. Il LQR cerca di minimizzare l'inclinazione e la velocità angolare per evitare che divergano, ottenendo come risultato però delle traiettorie più 'conservative' e dei tempi di convergenza più lunghi. Il IS-MPC invece vuole minimizzare solamente l'errore sulla traiettoria (e lo sforzo di controllo), ignorando l'inclinazione che il robot raggiunge fintanto che quest'ultima rispetti il vincolo di stabilità. Questo secondo approccio consente l'inseguimento di traiettorie più 'aggressive' e conseguentemente tempi di convergenza inferiori.

Bibliografia

- [1] Marco Kannevorf. *Stable trajectory control for wheeled inverted pendulum vehicles*. Sapienza, Università di Roma, 2021
- [2] Leonardo Lanari and Seth Hutchinson. *Inversion-based gait generation for humanoid robots*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 1592–1598. IEEE, 2015
- [3] Andrea Cristofaro. *Attuatori e controllo del moto*. Slides del corso di "Laboratorio di Automatica" presso Sapienza, Università di Roma, 2022.
- [4] Alberto Isidori. *Nonlinear control systems*. Springer Science Business Media, 2013.