

UNIVERSITÀ
DEGLI STUDI
DI PADOVA



DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

LAUREA TRIENNALE IN INGEGNERIA ELETTRONICA

Progettazione di un controllore PID per un balancing robot in ambiente simulato e sperimentale

LAUREANDO

Giacomo Garbuggio Bigaglia

Matricola 2000107

RELATORE

Prof. Mattia Bruschetta

Università di Padova

CORRELATORE

Ing. Luca Caiaffa

Università di Padova

ANNO ACCADEMICO 2022/2023
Data di laurea 25 settembre 2023

Abstract

L'attività consiste nello sviluppo di un controllo Proporzionale Integrativo Derivativo (PID) per la stabilizzazione verticale e l'inseguimento di spazio di un balancing robot. Il sistema è inizialmente modellato ricavando le Equazioni del Moto (EoM) attraverso l'utilizzo della meccanica lagrangiana; tali equazioni saranno quindi linearizzate attorno al punto di equilibrio verticale.

Implementando in Simulink, il sistema sotto forma di modello di stato linearizzato, si può dunque procedere con la sintesi del controllo PID. Successivamente si applicherà il controllo ad un modello non lineare risolvendo quindi le semplificazioni che erano state introdotte dal modello lineare. A seguito dell'implementazione del controllore in simulazione, il sistema viene testato in ambiente sperimentale, mettendo in evidenza le differenze rispetto al contesto simulativo.

Indice

Indice Figure	vii
Indice Tabelle	ix
Indice Acronimi	xi
1 Introduzione	1
2 Modellizzazione del sistema balancing robot	3
2.1 Descrizione dei parametri del robot	3
2.2 Approccio Lagrangiano per sintetizzare le equazioni del moto	10
2.3 Scrittura delle equazioni del moto ricavate dalla lagrangiana	15
2.3.1 Linearizzazione del sistema attorno al punto di equilibrio	16
2.3.2 Modellizzazione del sistema di attuazione	18
2.4 Derivazione delle funzioni di trasferimento dal modello linearizzato in spazio di stato	19
3 Sintesi dei controllori PID	23
3.1 Controllo dell'angolo ϑ	24
3.2 Controllo dell'angolo γ	26
3.3 Sviluppo in ambiente simulato di un modello più simile al robot reale	29
3.3.1 Modello non lineare	30
3.3.2 Modellizzazione della sensoristica del robot e stima degli angoli ϑ e γ	30
3.3.3 Trasformazione a tempo discreto della parte di stima e di controllo	34
3.3.4 Aggiunta saturatore	36
3.3.5 Aggiunta sistema "Anti-Wind Up"	37
3.4 Analisi in ambiente simulato delle risposte ottenute dal modello completo	40
4 Verifica dei controllori sul sistema balancing robot reale	45
4.1 Analisi taratura A applicata al sistema reale	46
4.2 Analisi taratura C applicata al sistema reale	47

INDICE

5 Conclusioni **53**

Bibliografia **55**

Indice Figure

2.1	Schema che rappresenta il robot in pianta	4
2.2	Schema che rappresenta la sezione frontale del robot	4
2.3	Schema che rappresenta la sezione laterale del robot	5
2.4	Schema che rappresenta una visione 3D del robot	5
2.5	Schema circuitale e parametri motore DC del robot	18
2.6	Diagrammi di Bode delle Funzioni di Trasferimento (FdT) del sistema	22
3.1	Config. PID parallelo	23
3.2	Config. PID serie	23
3.3	Schema Simulink per la sintesi del controllo di ϑ	25
3.4	Risposta dell'angolo ϑ rispetto al sistema lineare	25
3.5	Schema Simulink per la sintesi del controllo di γ	27
3.6	Risposte dell'angolo ϑ e dell'angolo γ , convertito in spostamento, rispetto al sistema lineare	28
3.7	Grafici dell'andamento dei segnali di controllo	29
3.8	Derivata discreta con filtro a media mobile	36
3.9	Grafico in cui viene rappresentato il segnale saturato assieme all'intervallo di lavoro	36
3.10	Schema Anti-Wind Up in forma di Clamping	37
3.11	Schema completo e discretizzato del modello non lineare del robot	38
3.12	Controllo discreto di ϑ	39
3.13	Controllo discreto di γ	39
3.14	Sensori a bordo del robot	39
3.15	Giroscopio e accelerometro	39
3.16	Parte di stima dei dati	39
3.17	Stima dell'angolo ϑ	40
3.18	Risposta del sistema con le tarature dei PID del sistema lineare riportate nella tabella 3.3	40
3.19	Grafico segnali filtraggio complementare	42

INDICE FIGURE

3.20	Schematizzazione encoder incrementale in Simulink	42
3.21	Risposte dell'angolo ϑ e dell'angolo γ , convertito in spostamento, rispetto al sistema non lineare e segnale di controllo u	43
4.1	Risposte ottenute dal sistema reale con taratura A dei PID	46
4.2	Risposte ottenute dal sistema reale con taratura C dei PID	48
4.3	Risposte ottenute dal sistema reale con taratura C dei PID volutamente disturbata	49
4.4	Risposte ottenute dal sistema reale con taratura C dei PID e segnale di inseguimento a rampa	51
4.5	Risposte ottenute dal sistema reale con taratura C dei PID e segnale di inseguimento sinusoidale	51

Indice Tabelle

2.1	Parametri geometrici e inerziali del robot	7
2.2	Valori coefficienti d'attrito del robot	15
3.1	Influenza dei coefficienti del PID sulla risposta del sistema	25
3.2	Tarature del PID_{θ}	25
3.3	Tarature sperimentali dei PID	28
3.4	Parametri geometrici della Motion Processing Unit (MPU)	31
3.5	Taratura aggiuntiva dei PID	43

Indice Acronimi

CoM Centro di Massa

CPU Central Processing Unit

EoM Equazioni del Moto

FdT Funzioni di Trasferimento

SIMO Singol Input Multi Output

MoI Momento d’Inerzia

MPU Motion Processing Unit

PID Proporzionale Integrativo Derivativo

SISO Singol Input Singol Output

Capitolo 1

Introduzione

Il balancing robot è un particolare tipo di "pendolo inverso su carrello", eredita dunque le tipiche problematiche di controllo che emergono quando si vuole studiare il suddetto sistema, per progettarne il controllo (vedi Bibliografia [4]).

La problematica principale che i due sistemi condividono è il fatto che i sistemi sono intrinsecamente instabili, ed hanno una significativa dinamica non lineare, appartengono inoltre all'insieme di sistemi meccanici Singol Input Multi Output (SIMO), ovvero quei sistemi caratterizzati da un solo ingresso e più uscite. Infatti lo studio della dinamica del sistema balancing robot parte dalla considerazione che il sistema ha due gradi di libertà: la posizione dell'angolo del robot rispetto alla verticale e il fatto che le due ruote, a contatto con il terreno, possano girare facendo quindi avanzare o retrocedere il robot. In questa trattazione si studia un controllo rispetto alla variabile ϑ (angolo di inclinazione rispetto alla verticale) ed alla variabile γ (angolo con cui le ruote girano rispetto alla posizione di partenza).

Il complesso appartiene inoltre alla classe di sistemi meccanici di tipo "sotto-attuati", perché il sistema è controllabile e può essere stabilizzato solo in alcune configurazioni, ma non in una arbitraria. Come si vedrà nella trattazione riportata nei capitoli successivi, si cercherà di progettare un controllo che stabilizzi il robot in una determinata posizione.

Per tutte le motivazioni sopra elencate è impegnativo cercare di controllare con successo tutte le variabili di uscita del sistema e tale situazione può considerarsi interessante per provare a studiare e progettare un sistema di controllo che stabilizzi il sistema.

Per stabilizzare ϑ ed eventualmente inseguire un riferimento di γ , è necessario progettare un controllo che dosi la quantità di forza che i motori devono imprimere alle ruote per evitare la caduta del robot, ciò avviene mantenendo dunque l'angolo ϑ minore di un dato valore.

Molti dispositivi e apparecchiature reali, come il lancio e l'atterraggio di razzi, i Segway, bracci robotici o robot umanoidi, presentano le medesime problematiche sopra elencate ed

inoltre hanno la necessità di essere stabilizzati verticalmente, dunque possono essere, in maniera molto semplificata, ricondotti allo studio del pendolo inverso, che diventa quindi dal punto di vista didattico una delle basi della teoria dei controlli.

In questa analisi, per la stabilizzazione e il tracking del sistema, si è fatto uso di un doppio controllore PID posto in parallelo; ciascuno combina, in modo pesato, tre differenti azioni sull'errore prodotto dalle differenze tra i valori di riferimento e le retroazioni del sistema.

Capitolo 2

Modellizzazione del sistema balancing robot

Il balancing robot è un robot che muovendosi sulle sue due ruote coassiali cerca di mantenersi in equilibrio rispetto alla verticale. Lo studio della dinamica di questo robot viene svolto considerando queste due semplificazioni importanti:

1. il robot si può muovere solo lungo una direzione, le due ruote si muoveranno quindi allo stesso identico modo;
2. il robot viene messo in condizioni di muoversi solo su un piano orizzontale, mai in un piano inclinato.

2.1 Descrizione dei parametri del robot

Il robot può essere rappresentato come un sistema "multi-body" composto da tre corpi rigidi principali: le due ruote, i due rotori del motore DC, ed il corpo principale del robot. Quest'ultimo comprende, oltre che al suo telaio anche la batteria, l'insieme delle schede elettroniche, le staffe per i motori ed i due statori (destra/sinistra) del motore in DC.

Si introducono i sistemi di coordinate di riferimento per descrivere la struttura e la configurazione del robot (al fine di rendere chiara la successiva trattazione e descrizione si prega di prendere visione degli elaborati grafici descrittivi della macchina. Vedi figure 2.2, 2.3, 2.1 e 2.4):

- **Sistema coordinate Telaio $\{b\}$:** Il sistema è situato nel centro dell'asse di rotazione delle ruote ed è solidale con il robot. Il suo asse y è diretto lungo l'asse di rotazione della ruota con direzione positiva verso la ruota sinistra. L'asse z passa attraverso il Centro di Massa (CoM) con orientamento positivo dalla base verso il CoM del corpo. La direzione e l'orientamento dell'asse x sono determinati dalla regola della mano destra.

2.1. DESCRIZIONE DEI PARAMETRI DEL ROBOT

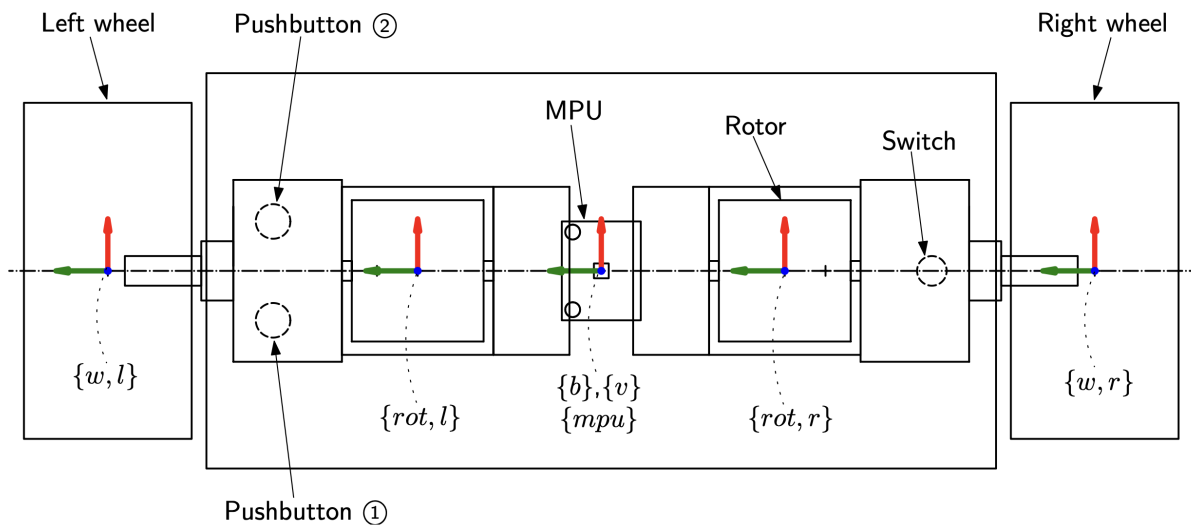


Figura 2.1: Schema che rappresenta il robot in pianta

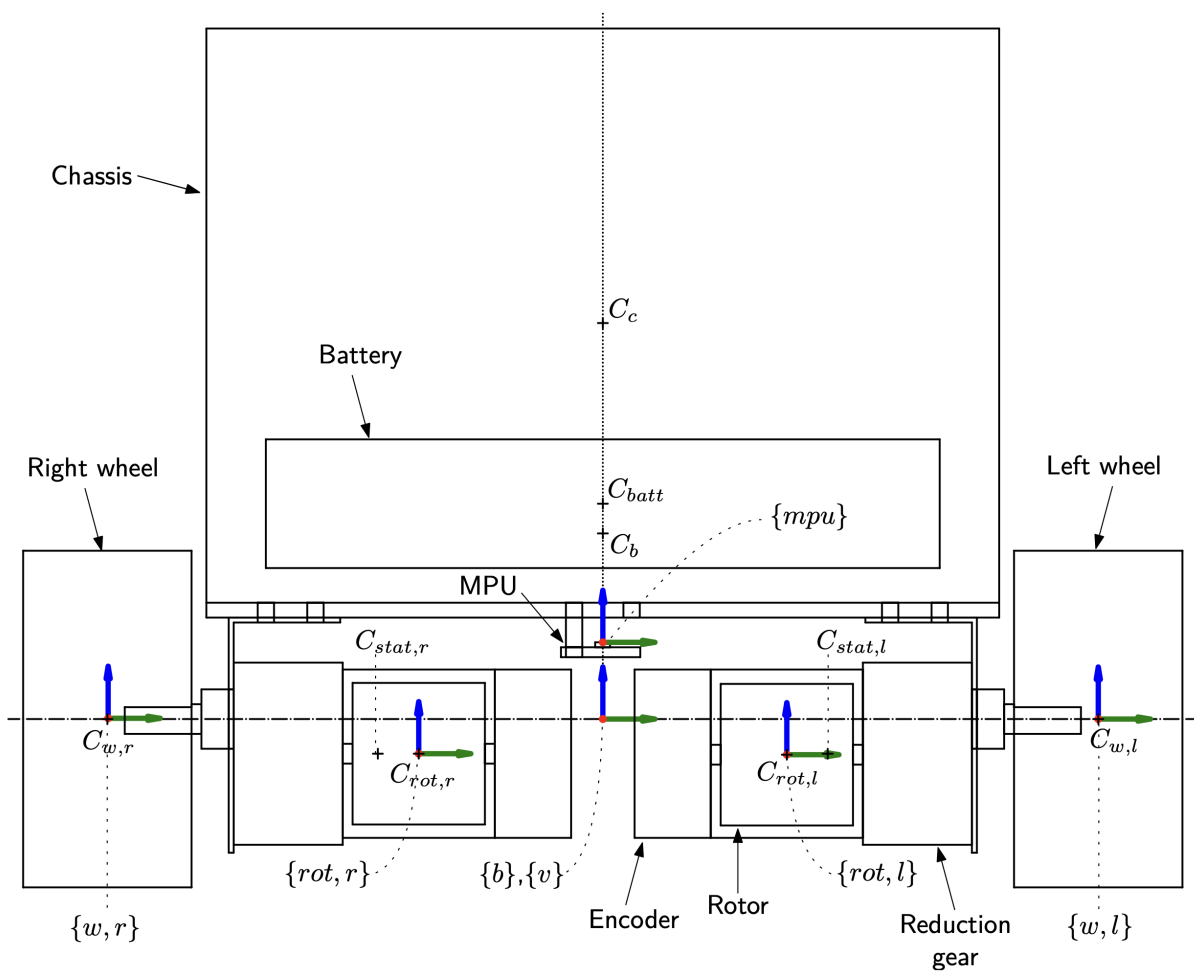


Figura 2.2: Schema che rappresenta la sezione frontale del robot

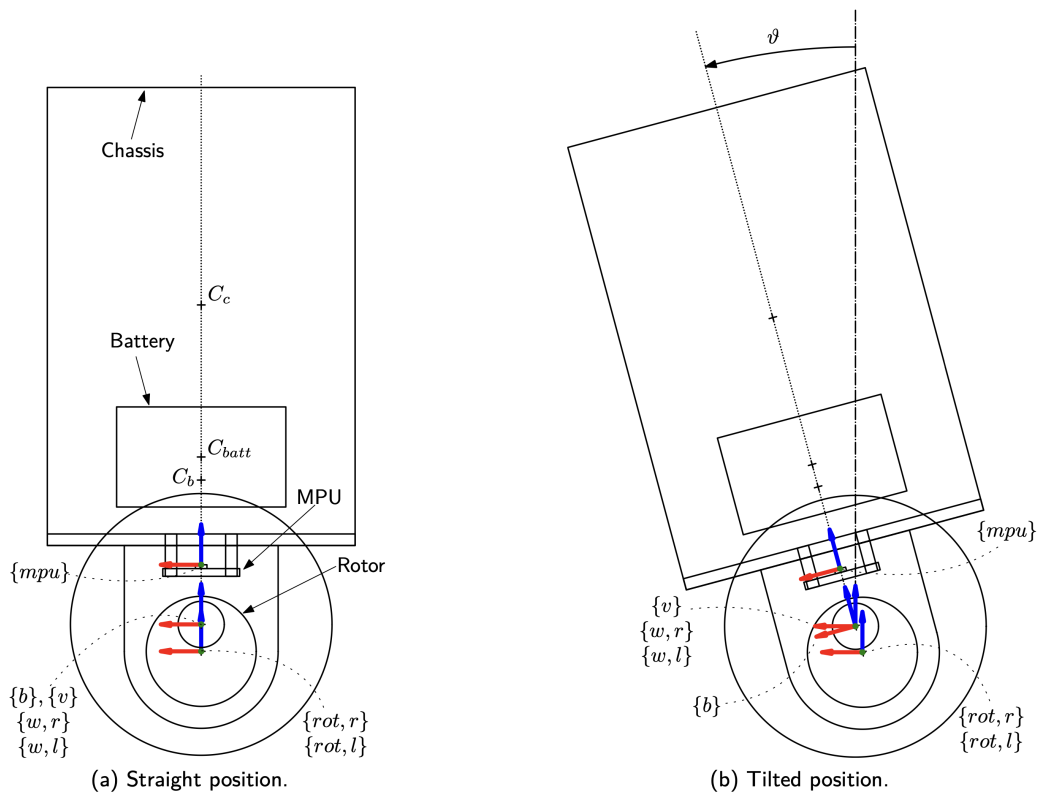


Figura 2.3: Schema che rappresenta la sezione laterale del robot

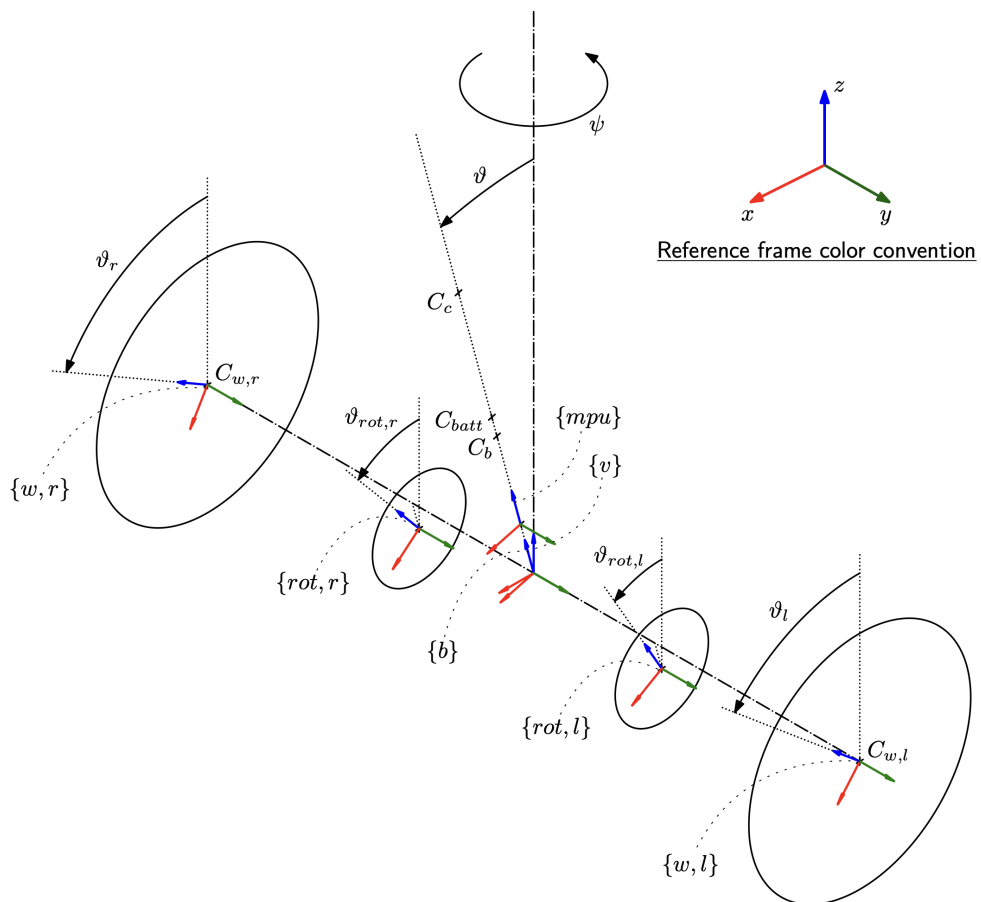


Figura 2.4: Schema che rappresenta una visione 3D del robot

2.1. DESCRIZIONE DEI PARAMETRI DEL ROBOT

- **Sistema coordinate Veicolo $\{v\}$:** Il sistema non è solidale al robot. Il centro e l'asse y sono i medesimi del sistema telaio $\{b\}$. Il suo asse z è sempre orientato in modo opposto all'accelerazione di gravità. La direzione e l'orientamento dell'asse x sono determinati dalla regola della mano destra.
- **Sistema coordinate Terra $\{o\}$:** La sua posizione e orientamento sono solidali alla terra e il riferimento iniziale coincide con la posizione e l'orientamento iniziale del sistema veicolo $\{v\}$.
- **Sistema coordinate Ruote (destra/sinistra) $\{w, r\}$ $\{w, l\}$:** I due sistemi sono solidali alle ruote e centrati nei rispettivi CoM. Gli assi sono inizialmente allineati con quelli del sistema veicolo $\{v\}$.
- **Sistema coordinate Rotori (destra/sinistra) $\{rot, r\}$ $\{rot, l\}$:** I due sistemi sono solidali ai corpi dei rotor e centrati nei propri CoM. Gli assi sono inizialmente allineati con quelli del sistema veicolo $\{v\}$.

Nella Tabella 2.1 (elencata nella pagina successiva) vengono riportati i parametri geometrici e inerziali di ciascuna componente del robot. È importante notare che:

- La massa totale di un singolo motore DC è la somma tra la massa del rotore e la massa dello statore ($m_{mot} = m_{rot} + m_{stat} = 215$ g).
- La massa totale del corpo del robot è la somma tra le componenti più pesanti che la compongono, questi sono: la massa propria del telaio, la massa della batteria e la massa dei due statori ($m_b = m_c + m_{batt} + 2m_{stat} = 215$ g).
- La posizione del CoM dello statore è stata determinata dopo aver notato (sperimentalmente) che il CoM del motore è spostato di 6 [mm] dal centro geometrico, nella direzione del cambio del motore.
- La coordinata z del CoM del corpo del robot è stata determinata nel seguente modo: $z_b^b = \frac{1}{m_b}(m_c z_c^b + m_{batt} z_{batt}^b + 2m_{stat} z_{stat}^b)$. Le coordinate x e y sono pari a zero per la simmetria del sistema.
- Il Momento d'Inerzia (MoI) del telaio e della batteria del robot sono stati determinati entrambi considerando che i corpi abbiano distribuzione di massa uniforme e siano dei parallelepipedi perfetti. La posizione dell'asse di rotazione è stata assunta uguale alla posizione del CoM di ciascuno dei due corpi.

$$I_{xx} = \frac{m}{12}(w^2 + h^2), \quad I_{yy} = \frac{m}{12}(d^2 + h^2), \quad I_{zz} = \frac{m}{12}(w^2 + d^2)$$

dove w , h , d , sono rispettivamente larghezza, altezza e profondità del parallelepipedo, mentre m è la sua massa.

Corpo del robot			
Coordinate del centro di massa rispetto al sistema $\{b\}$	x_b^b, y_b^b, z_b^b	0, 0, 46.05	[mm]
Massa	m_b	1.06	[kg]
Momento d'inerzia	$I_{b,xx}, I_{b,yy}, I_{b,zz}$	4.22, 2.20, 2.65	[gm ²]
Talaio del robot			
Dimensioni (larghezza, altezza, profondità)	w_c, h_c, d_c	160, 119, 80	[mm]
Coordinate del centro di massa rispetto al sistema $\{b\}$	x_c^b, y_c^b, z_c^b	0, 0, 80	[mm]
Massa	m_c	456	[g]
Momento d'inerzia	$I_{c,xx}, I_{c,yy}, I_{c,zz}$	1.5, 0.78, 1.2	[gm ²]
Batteria			
Dimensioni (larghezza, altezza, profondità)	$w_{batt}, h_{batt}, d_{batt}$	136, 29, 44	[mm]
Coordinate del centro di massa rispetto al sistema $\{b\}$	$x_{batt}^b, y_{batt}^b, z_{batt}^b$	0, 0, 44	[mm]
Massa	m_{batt}	320	[g]
Momento d'inerzia	$I_{batt,xx}, I_{batt,yy}, I_{batt,zz}$	0.51, 0.07, 0.06	[gm ²]
Statore del motore DC			
Dimensioni (altezza, raggio)	h_{stat}, r_{stat}	68.1, 17	[mm]
Coordinate del centro di massa rispetto al sistema $\{b\}$	$x_{stat}^b, y_{stat}^b, z_{stat}^b$	0, ± 52.1 , -7	[mm]
Massa	m_{stat}	139.75	[g]
Momento d'inerzia	$I_{stat,xx}, I_{stat,yy}, I_{stat,zz}$	0.064, 0.064, 0.02	[gm ²]
Rotore del motore DC			
Dimensioni (altezza, raggio)	h_{rot}, r_{rot}	30.7, 15.3	[mm]
Coordinate del centro di massa rispetto al sistema $\{b\}$	$x_{rot}^b, y_{rot}^b, z_{rot}^b$	0, ± 42.7 , -7	[mm]
Massa	m_{rot}	75.25	[g]
Momento d'inerzia	$I_{rot,xx}, I_{rot,yy}, I_{rot,zz}$	0.01, 0.01, 0.009	[gm ²]
Ruote			
Dimensioni (altezza, raggio)	h_w, r_w	26, 34	[mm]
Coordinate del centro di massa rispetto al sistema $\{b\}$	x_w^b, y_w^b, z_w^b	0, ± 100 , 0	[mm]
Massa	m_w	50	[g]
Momento d'inerzia	$I_{w,xx}, I_{w,yy}, I_{w,zz}$	0.017, 0.017, 0.029	[gm ²]

Tabella 2.1: Parametri geometrici e inerziali del robot

- I MoI delle ruote e del motore (sia statore che rotore) vengono determinati assumendo i corpi come dei cilindri perfetti, con distribuzione di massa uniforme. L'asse di rotazione si suppone sempre sia posizionato nel CoM degli oggetti.

$$I_{xx} = I_{zz} = \frac{m}{12}(3r^2 + h^2), \quad I_{yy} = \frac{m r^2}{2}$$

dove h e r sono rispettivamente l'altezza e il raggio, mentre m è la sua massa.

- Come detto sopra e come elencato nella tabella di valori, il corpo del robot è formato da 3 parti. Per determinare le componenti del MoI totale I_b rispetto al suo sistema di coordinate di riferimento $\{b\}$ è necessario ricorrere al teorema di Huygens-Steiner

2.1. DESCRIZIONE DEI PARAMETRI DEL ROBOT

(o teorema dell'asse parallelo), che permette di calcolare il momento d'inerzia I' di un corpo di massa m , che ruota attorno a un asse z' spostato dal suo CoM di una distanza d . Il teorema afferma che $I' = I + md^2$, dove I è il MoI del corpo rispetto a un asse z parallelo a z' e passante per il suo CoM. Tale teorema si applica per tutte le tre componenti di I_b che pertanto diventano:

$$\begin{aligned}
 I_{b,xx} &= I_{c,xx} + m_c (z_b^b - z_c^b)^2 + I_{batt,xx} + m_{batt} (z_b^b - z_{batt}^b) + \dots \\
 &\quad \dots + I_{stat,xx} + m_{stat} \left[(y_b^b - y_{stat,r}^b)^2 + (z_b^b - z_{stat,r}^b)^2 \right] + \dots \\
 &\quad \dots + I_{stat,xx} + m_{stat} \left[(y_b^b - y_{stat,l}^b)^2 + (z_b^b - z_{stat,l}^b)^2 \right] \\
 I_{b,yy} &= I_{c,yy} + m_c (z_b^b - z_c^b)^2 + I_{batt,yy} + m_{batt} (z_b^b - z_{batt}^b) + \dots \\
 &\quad \dots + 2I_{stat,yy} + 2m_{stat} (z_b^b - z_{stat,l}^b)^2 \\
 I_{b,zz} &= I_{c,zz} + I_{batt,zz} + 2I_{stat,zz} + 2m_{stat} (y_{stat}^b)^2
 \end{aligned}$$

Dopo aver introdotto i parametri geometrici ed inerziali ed aver descritto il modo in cui vengono calcolati è necessario introdurre un sistema di coordinate generalizzate per riuscire a descrivere in modo univoco il robot nello spazio:

- **Posizione del robot** $p_v^o = (x_v, y_v, z_v)$: coordinate del sistema veicolo $\{v\}$ rispetto al sistema di coordinate terra $\{o\}$.
- **Angolo di inclinazione del robot** ϑ : angolo di inclinazione tra l'asse z del sistema di coordinate telaio $\{b\}$ e l'asse z del sistema di coordinate veicolo $\{v\}$.
- **Angoli delle ruote** ϑ_r e ϑ_l : angoli tra il sistema di coordinate ruota, rispettivamente destra $\{w, r\}$ e sinistra $\{w, l\}$, ed il sistema di coordinate veicolo $\{v\}$.
- **Angolo di rotazione del robot** ψ : angolo tra il sistema di coordinate veicolo $\{v\}$ ed il sistema di coordinate terra $\{o\}$.
- **Angolo dei rotori** $\vartheta_{rot,r}$ e $\vartheta_{rot,l}$: angoli tra il sistema di coordinate rotore, rispettivamente destro $\{rot, r\}$ e sinistro $\{rot, l\}$, ed il sistema di coordinate veicolo $\{v\}$.

È importante sottolineare che le ultime due coordinate introdotte (ψ e $\vartheta_{rot,*}$) sono delle variabili dipendenti. Possono essere infatti ricavate dalle 3 coordinate precedenti, questo è possibile grazie a delle semplificazioni che si ipotizzano nell'ambito della cinematica del sistema:

• **Funzionamento meccanico del cambio:**

$$\Delta\vartheta_l = \vartheta_l - \vartheta, \quad \Delta\vartheta_{rot,l} = \vartheta_{rot,l} - \vartheta, \quad (2.1)$$

tali differenze evidenziano gli spostamenti angolari della ruota e del rotore sinistro rispetto al telaio del corpo. Poiché lo statore del motore è parte del telaio del corpo, dunque è rigidamente attaccato ad esso, $\Delta\vartheta_l$ e $\Delta\vartheta_{rot,l}$ sono gli angoli con cui l'albero di uscita del motoriduttore e il rotore ruotano rispetto allo statore. L'accoppiamento meccanico dovuto al cambio impone:

$$\Delta\vartheta_{rot,l} = N \Delta\vartheta_l \quad (2.2)$$

dove N indica il rapporto del cambio, quindi il rapporto tra il numero di giri del rotore e l'albero di uscita collegato rigidamente alla ruota. Sostituendo ora alla (2.2) la (2.1) troviamo che:

$$\vartheta_{rot,l} = \vartheta + N (\vartheta_l - \vartheta) \quad (2.3)$$

La medesima espressione sarà valida per l'angolo del rotore destro $\vartheta_{rot,r}$.

• **Ipotesi di puro rotolamento della ruota, mai slittamento:** lo slittamento può essere generato da due diverse condizioni:

- (i). la mancanza di parallelismo tra il piano di rotazione della ruota con la direzione del vettore velocità della stessa.
- (ii). qualora la velocità di rotazione dello pneumatico risulti maggiore della velocità di rotolamento libero.

Si assume quindi che il vettore velocità applicato al centro della ruota sia parallelo al piano saggittale della ruota e sia proporzionale alla velocità di rotazione della ruota così da eliminare lo slittamento in entrambi i casi.

Le condizioni imposte implicano:

$$v_l = r \dot{\vartheta}_l, \quad v_r = r \dot{\vartheta}_r, \quad w \dot{\psi} = v_r - v_l \quad (2.4)$$

dove $w = 2 |y_w^b|$ è la distanza tra i due CoM delle ruote, r è il raggio della ruota, v_l e v_r sono le velocità rispettivamente della ruota sinistra e destra. Mettendo a sistema le tre equazioni sopra riportate si ricava:

$$\dot{\psi} = \frac{r}{w} (\dot{\vartheta}_r - \dot{\vartheta}_l) \quad \xrightarrow[\text{nel tempo}]{\text{integro}} \quad \psi = \psi(0) + \frac{r}{w} (\vartheta_r - \vartheta_l) \quad (2.5)$$

dove $\psi(0) = 0$ perché il sistema veicolo $\{v\}$ al tempo $t = 0$ è allineato con il sistema

terra $\{o\}$.

Grazie alla prima semplificazione enunciata all'inizio del capitolo, si considera che il robot si possa muovere solo lungo una linea retta, così facendo si limita l'analisi solo alla dinamica longitudinale, imponendo dunque l'angolo ψ costante. Ne deriva che ϑ_r e ϑ_l siano sempre indentici, definiamo allora γ come variabile comune all'angolo di rotazione delle due ruote.

Oltre ad assumere ψ costante è possibile assumerlo pari a zero, così facendo anche y_v sarà pari a zero ed il moto del robot resterà solo confinato al piano x - z del sistema di coordinate terra $\{o\}$. È importante ricordare che, grazie alla semplificazione applicata, le coppie erogate del motore sinistro e destro debbano essere sempre identiche al fine di ottenere la stabilizzazione del robot.

Per semplificare ulteriormente l'analisi, si considera la seconda semplificazione enunciata all'inizio del capitolo; questa ipotizza che il movimento del robot possa avvenire solo su una superficie piana e orizzontale (cioè il vettore di gravità resti perpendicolare alla superficie del terreno) in modo da ottenere che la coordinata $z_v = 0$, inoltre grazie alla condizione di puro rotolamento imposta alle ruote la coordinata $x_v = r\gamma$.

2.2 Approccio Lagrangiano per sintetizzare le equazioni del moto

Per ricavare le EoM si utilizza la meccanica razionale, in particolare adottando l'approccio della meccanica lagrangiana (una formulazione della meccanica introdotta nel XVII secolo da Joseph-Louis Lagrange come riformulazione della meccanica newtoniana) (vedi Bibliografia [2]).

Viene dunque introdotta la Lagrangiana di un sistema fisico che è una funzione che caratterizza la dinamica; per i sistemi meccanici tale funzione è definita come differenza tra l'energia cinetica $T(q, \dot{q})$ e l'energia potenziale $U(q)$. L'espressione della lagrangiana che ne deriva è:

$$\mathcal{L}(q, \dot{q}) = T(q, \dot{q}) - U(q) \quad (2.6)$$

dove $q \in \mathbb{R}^n$ sono le coordinate generalizzate, \dot{q} sono le rispettive velocità. Nella meccanica razionale le coordinate generalizzate, sono un sistema di coordinate di numero pari ai gradi di libertà del sistema; queste determinano univocamente tutte le configurazioni possibili del sistema. Nel caso in esame $q = [\vartheta, \gamma]$, $\dot{q} = [\dot{\vartheta}, \dot{\gamma}]$.

Affinchè si riesca a ricavare le EoM del sistema è necessario introdurre l'azione (\mathcal{S}): una quantità scalare avente dimensioni pari al prodotto tra energia e tempo; questa grandezza caratterizza lo stato in generale e l'evoluzione di un sistema permettendo di studiarne il

moto. Matematicamente è definita come l'integrale rispetto al tempo della lagrangiana.

$$\mathcal{S} = \int_{t_1}^{t_2} \mathcal{L}(q, \dot{q}) dt \quad (2.7)$$

La definizione di tale grandezza è fondamentale per poter applicare il principio di minima azione di Hamilton, secondo il quale ogni sistema meccanico in moto tra due punti segue il cammino che, tra tutti i percorsi possibili, minimizza l'azione. Applicando questo principio alla (2.7) è possibile ricavare le quazioni di Eulero-Lagrange che rappresentano le EoM del sistema.

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}} \right] - \frac{\partial \mathcal{L}(q, \dot{q})}{\partial q} = 0 \quad (2.8)$$

L'approccio lagrangiano rispetto a quello newtoniano è molto utile nel caso in cui il sistema da studiare abbia dei vincoli, il che impone che ci siano delle forze di reazione vincolari collegate ad essi. Tali forze devono essere prese in considerazione, se si vogliono ricavare le EoM attraverso la dinamica newtoniana, spesso però non è nota l'esatta formulazione di queste forze che si trasformano dunque in incognite aggiuntive, facendo così diventare le equazioni di Newton non deterministiche.

Un altro motivo per il quale la dinamica lagrangiana è uno strumento molto potente è il fatto che qualora si debba studiare un sistema composto da molti corpi, non è necessario considerare tutte le forze applicate al centro di massa di ogni singolo corpo, ma è possibile calcolare solo le energie cinetiche e potenziali dei singoli corpi. In questo modo si riducono il numero di equazioni da svolgere.

L'equazione di Lagrange inoltre, a differenza di quella di Newton, non cambia forma se si passa da un sistema di coordinate ad un altro, questo permette di scrivere più facilmente le equazioni, visto che è possibile utilizzare coordinate diverse da quelle cartesiane arrivando spesso a semplificazioni significative.

Prima di calcolare l'energia cinetica e potenziale è necessario introdurre delle notazioni utili al fine di ricavare le velocità con cui i tre corpi principali del sistema "multy-body" si muovono rispetto al sistema terra $\{o\}$.

- $\mathbf{p}_a^b = [\mathbf{x}_a^b, \mathbf{z}_a^b]^T$: vettore posizione del punto a espresso rispetto al sistema di riferimento $\{b\}$. È possibile che il sistema $\{b\}$ coincida con il sistema $\{o\}$ allora gli apici saranno omessi nelle coordinate di posizione.
- $\mathbf{r}_{a,b}^c = [\mathbf{x}_{a,b}^c, \mathbf{z}_{a,b}^c]^T$: vettore spostamento dal punto a al punto b rispetto al sistema $\{c\}$. Esempio:

$$\mathbf{r}_{a,b}^c = \mathbf{p}_b^c - \mathbf{p}_a^c \quad (2.9)$$

2.2. APPROCCIO LAGRANGIANO PER SINTETIZZARE LE EQUAZIONI DEL MOTO

- R_a^b : matrice che descrive la rotazione del sistema $\{a\}$ rispetto al sistema $\{b\}$. Se si indica con ϑ l'angolo rispetto al quale il primo sistema è ruotato rispetto al secondo, allora la matrice di rotazione è pari a:

$$R_a^b = \begin{bmatrix} \cos \vartheta & \sin \vartheta \\ -\sin \vartheta & \cos \vartheta \end{bmatrix} \quad (2.10)$$

La procedura di derivazione del vettore velocità dei tre corpi principali del sistema è sostanzialmente uguale per i tre corpi.

Si parte dalla considerazione che il vettore velocità è la derivata temporale del vettore posizione del sistema in esame rispetto al sistema terra. Per ricavare tale vettore posizione è necessario isolarlo dal vettore spostamento prodotto dalla differenza tra il vettore posizione, che si vuole isolare, ed il vettore posizione del sistema veicolo rispetto al sistema terra.

Per prima cosa si deve quindi ricavare:

1. Il vettore posizione del sistema veicolo $\{v\}$ rispetto al sistema terra $\{o\}$ che è uguale per tutti e tre i corpi ed è pari a: $p_v^o = [r \gamma, 0]^T$.
2. Il vettore spostamento come prodotto tra la matrice di rotazione, del sistema telaio $\{b\}$ rispetto al sistema terra $\{o\}$, ed il vettore posizione del CoM del sistema d'interesse rispetto al sistema telaio $\{b\}$. Tale formulazione è valida per il telaio e per il rotore; le ruote, invece, avendo il loro CoM allineato lungo l'asse y con l'origine del sistema veicolo, avranno vettore spostamento nullo (di fatto grazie alle semplificazioni, è stata eliminata tutta la dinamica lungo l'asse y).

Successivamente, come descritto sopra, si isola il vettore posizione d'interesse e lo si deriva rispetto al tempo (facendo attenzione alle regole di derivazione del prodotto). Si riportano di seguito le tre formulazioni della velocità dei tre corpi principali che compongono il robot:

$$\dot{p}_{C_b}^o = \dot{p}_b^o + \dot{r}_{v,C_b}^o = \dot{p}_b^o + \frac{dR_o^b}{dt} p_{C_b}^o + R_o^b \dot{p}_{C_b}^o = \begin{bmatrix} r \dot{\gamma} + z_b^b \cos \vartheta \dot{\vartheta} \\ -z_b^b \sin \vartheta \dot{\vartheta} \end{bmatrix} \quad (2.11)$$

$$\dot{p}_{C_w}^o = \dot{p}_b^o = \begin{bmatrix} r \dot{\gamma} \\ 0 \end{bmatrix} \quad (2.12)$$

$$\dot{p}_{C_{rot}}^o = \dot{p}_b^o + \dot{r}_{v,C_{rot}}^o = \dot{p}_b^o + \frac{dR_o^b}{dt} p_{C_{rot}}^o + R_o^b \dot{p}_{C_{rot}}^o = \begin{bmatrix} r \dot{\gamma} + z_{rot}^b \cos \vartheta \dot{\vartheta} \\ -z_{rot}^b \sin \vartheta \dot{\vartheta} \end{bmatrix} \quad (2.13)$$

Dopo aver calcolato le espressioni delle velocità dei corpi è possibile procedere con il calcolo dell'energia cinetica T e dell'energia potenziale U . Tali energie saranno la somma

algebraica delle energie dei tre rispettivi corpi principali:

$$T = T_b + T_w + T_{rot} \quad U = U_b + U_w + U_{rot} \quad (2.14)$$

Si ricavano di seguito i contributi di energia cinetica e potenziale:

- **Energia cinetica ed energia potenziale del telaio.** L'energia cinetica viene calcolata come somma tra l'energia cinetica dovuta alla traslazione del CoM e l'energia cinetica dovuta alla rotazione del corpo rispetto al suo CoM.

$$\begin{aligned} T_b &= \frac{1}{2} m_b (\dot{p}_{C_b}^o)^T (\dot{p}_{C_b}^o) + \frac{1}{2} I_{b,yy} \dot{\vartheta}^2 \\ &= \frac{1}{2} m_b r^2 \dot{\gamma}^2 + \frac{1}{2} (I_{b,yy} + m_b l^2) \dot{\vartheta}^2 + m_b r l \cos \vartheta \dot{\gamma} \dot{\vartheta} \end{aligned} \quad (2.15)$$

L'energia potenziale è pari a:

$$U_b = m_b g z_{C_b} = m_b g l \cos \vartheta \quad (2.16)$$

- **Energia cinetica ed energia potenziale delle ruote.** Il calcolo dell'energia cinetica delle ruote, avviene anche in questo caso, come somma tra l'energia cinetica di traslazione e l'energia cinetica di rotazione.

$$\begin{aligned} T_w &= \frac{1}{2} (2m_w) (\dot{p}_{C_w}^o)^T (\dot{p}_{C_w}^o) + \frac{1}{2} (2I_{w,yy}) \dot{\gamma}^2 \\ &= (I_{w,yy} + m_w r^2) \dot{\gamma}^2 \end{aligned} \quad (2.17)$$

L'energia potenziale delle ruote è pari a zero, essendo entrambe a contatto con il terreno: $U_w = (2m_w) g z_{C_w} = 0$.

- **Energia cinetica ed energia potenziale del rotore.** Ricordando la (2.3) e avendo definito $\gamma = \vartheta_l = \vartheta_r$, l'angolo $\vartheta_{rot} = \vartheta + N (\gamma - \vartheta)$. Dunque la sua energia cinetica è pari a:

$$\begin{aligned} T_{rot} &= \frac{1}{2} (2m_{rot}) (\dot{p}_{C_{rot}}^o)^T (\dot{p}_{C_{rot}}^o) + \frac{1}{2} (2I_{rot,yy}) \dot{\vartheta}_{rot}^2 \\ &= (N^2 I_{rot,yy} + m_{rot} r^2) \dot{\gamma}^2 + \left[(1 - N)^2 I_{rot,yy} + m_{rot} (z_{rot}^b)^2 \right] \dot{\vartheta}^2 + \dots \\ &\quad \dots + 2 \left[N(1 - N) I_{rot,yy} + m_{rot} r z_{rot}^b \cos \vartheta \right] \dot{\gamma} \dot{\vartheta} \end{aligned} \quad (2.18)$$

L'energia potenziale è pari a:

$$U_{rot} = (2m_{rot}) g z_{C_{rot}} = (2m_{rot}) g z_{rot}^b \cos \vartheta \quad (2.19)$$

Si richiama la lagrangiana che è quindi pari a:

$$\begin{aligned}
 \mathcal{L} = T - U &= (T_b + T_w + T_{rot}) - (U_b + U_w + U_{rot}) \\
 &= \left[I_{w,yy} + N^2 I_{rot,yy} + \left(\frac{1}{2} m_b + m_w + m_{rot} \right) r^2 \right] \dot{\gamma} + \dots \\
 &\dots + \left[\frac{1}{2} I_{b,yy} + (1 - N)^2 I_{rot,yy} + \frac{1}{2} m_b l^2 + m_{rot} (z_{rot}^b)^2 \right] \dot{\vartheta}^2 + \dots \\
 &\dots + \left[2N(1 - N) I_{roy,yy} + (m_b l + 2m_{rot} z_{rot}^b) r \cos \vartheta \right] \dot{\gamma} \dot{\vartheta} - \dots \\
 &\dots - \left(m_b l + 2m_{rot} z_{rot}^b \right) g \cos \vartheta
 \end{aligned} \tag{2.20}$$

Le equazioni del moto che descrivono la dinamica longitudinale del robot sono calcolate attraverso l'equazione di Eulero-Lagrange (2.8) valutata prima per ϑ e poi per γ . È importante notare che, essendoci forze o coppie non conservative (come gli attriti), oppure forze o coppie esterne al sistema (come la coppia generata dai motori), le equazioni di Eulero-Lagrange non saranno più uguagliate a zero, ma saranno uguali alla somma di tali forze.

$$\begin{aligned}
 \frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{\vartheta}} \right] - \frac{\partial \mathcal{L}}{\partial \vartheta} &= \xi_{\vartheta} \\
 \frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{\gamma}} \right] - \frac{\partial \mathcal{L}}{\partial \gamma} &= \xi_{\gamma}
 \end{aligned} \tag{2.21}$$

Infatti ξ_{ϑ} e ξ_{γ} sono il risultato della somma delle forze non conservative, o esterne al sistema, associate rispettivamente alla variabile ϑ e γ . Nel sistema "balancing robot", i contributi da considerare sono la coppia del motore τ generata all'albero di uscita e le due coppie di attrito viscoso dominanti τ'_f e τ''_f .

Si considerano ora i due contributi di attrito viscoso, la coppia del motore τ viene analizzata nella sezione 2.3.2 di questo capitolo, per ora dunque la si esprime solo come parametro.

$$\tau'_f = B(\dot{\gamma} - \dot{\vartheta}) \qquad \tau''_f = B_w \dot{\gamma} \tag{2.22}$$

Il primo coefficiente è dovuto all'attrito interno del cambio del motore ed è proporzionale alla velocità di rotazione dell'albero di uscita del motore rispetto allo statore.

Il secondo è invece la coppia di attrito viscoso dovuto al rotolamento della ruota, ed è quindi proporzionale alla velocità di quest'ultima.

Considerando quindi che questi tre contributi (τ , τ'_f e τ''_f) agiscono in modo identico, sia sul lato sinistro che su quello destro, possiamo quindi scrivere che:

$$\begin{aligned}
 \xi_{\vartheta} &= -2\tau + 2\tau'_f = -2\tau + 2B\dot{\gamma} - 2B\dot{\vartheta} \\
 \xi_{\gamma} &= 2\tau - 2\tau'_f - 2\tau''_f = 2\tau - 2(B + B_w)\dot{\gamma} + 2B\dot{\vartheta}
 \end{aligned} \tag{2.23}$$

La stima dei valori dei coefficienti d'attrito B e B_w sono riportati nella Tabella 2.2

Coefficienti d'attrito			
Coefficiente d'attrito del cambio del motore	B	25×10^{-3}	[Nm/(rad/s)]
Coefficiente d'attrito delle ruote	B_w	1.5×10^{-3}	[Nm/(rad/s)]

Tabella 2.2: Valori coefficienti d'attrito del robot

2.3 Scrittura delle equazioni del moto ricavate dalla lagrangiana

Dopo aver sostituito la (2.22) e la (2.23) nella (2.21) si ricavano le EoM al variare delle sei coordinate generalizzate $(\vartheta, \gamma, \dot{\vartheta}, \dot{\gamma}, \ddot{\vartheta}, \ddot{\gamma})$:

$$\begin{aligned}
 & \left[2 I_{w,yy} + 2 N^2 I_{rot,yy} + (m_b + 2m_w + 2m_{rot}) r^2 \right] \ddot{\gamma} + 2 (B + B_w) \dot{\gamma} + \dots \\
 & \dots + \left[2 N (1 - N) I_{rot,yy} + (m_b l + 2 m_{rot} z_{rot}^b) r \cos \vartheta \right] \ddot{\vartheta} - 2B\dot{\vartheta} - \dots \\
 & \dots - (m_b l + 2 m_{rot} z_{rot}^b) r \sin \vartheta \dot{\vartheta}^2 - 2\tau = 0
 \end{aligned} \tag{2.24}$$

$$\begin{aligned}
 & \left[2 N (1 - N) I_{rot,yy} + (m_b l + 2 m_{rot} z_{rot}^b) r \cos \vartheta \right] \ddot{\gamma} - 2B\dot{\gamma} + \dots \\
 & \dots + \left[I_{b,yy} + 2 (1 - N)^2 I_{rot,yy} + m_b l^2 + 2 m_{rot} (z_{rot}^b)^2 \right] \ddot{\vartheta} + 2B\dot{\vartheta} - \dots \\
 & \dots - (m_b l + 2 m_{rot} z_{rot}^b) g \sin \vartheta + 2\tau = 0
 \end{aligned} \tag{2.25}$$

È possibile ora riscrivere tali equazioni in forma matriciale così da avere una formulazione più compatta e unica per $q = [\gamma, \vartheta]^T$, $\dot{q} = [\dot{\gamma}, \dot{\vartheta}]^T$, $\ddot{q} = [\ddot{\gamma}, \ddot{\vartheta}]^T$.

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + F_v \dot{q} + g(q) = \tau \tag{2.26}$$

dove

$$\begin{aligned}
 M(q) &= \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix}, & C(q, \dot{q}) &= \begin{bmatrix} C_{11}(q, \dot{q}) & C_{12}(q, \dot{q}) \\ C_{21}(q, \dot{q}) & C_{22}(q, \dot{q}) \end{bmatrix}, \\
 F_v &= \begin{bmatrix} F_{v,11} & F_{v,12} \\ F_{v,21} & F_{v,22} \end{bmatrix},
 \end{aligned} \tag{2.27}$$

2.3. SCRITTURA DELLE EQUAZIONI DEL MOTO RICAVATE DALLA LAGRANGIANA

con

$$\begin{aligned}
 M_{11}(q) &= 2 I_{w,yy} + 2 N^2 I_{rot,yy} + (m_b + 2 m_w + 2 m_{rot}) r^2 \\
 M_{12}(q) &= M_{21}(q) = 2 N (1 - N) I_{rot,yy} + (m_b l + 2 m_{rot} z_{rot}^b) r \cos \vartheta \\
 M_{22}(q) &= I_{b,yy} + 2 (1 - N)^2 I_{rot,yy} + m_b l^2 + 2 m_{rot} (z_{rot}^b)^2
 \end{aligned} \tag{2.28}$$

$$\begin{aligned}
 C_{11}(q, \dot{q}) &= C_{21}(q, \dot{q}) = C_{22}(q, \dot{q}) = 0 \\
 C_{12}(q, \dot{q}) &= -(m_b l + 2 m_{rot} z_{rot}^b) r \sin \vartheta \dot{\vartheta}
 \end{aligned} \tag{2.29}$$

$$\begin{aligned}
 F_{v,11} &= 2(B + B_w) \\
 F_{v,12} &= F_{v,21} = -2B \\
 F_{B,22} &= 2B
 \end{aligned} \tag{2.30}$$

$$g(q) = [0, -(m_b l + 2 m_{rot} z_{rot}^b) g \sin \vartheta]^T \tag{2.31}$$

$$\tau = [2\tau, -2\tau]^T \tag{2.32}$$

La rappresentazione in forma matriciale è anche molto utile perché ognuna delle matrici acquisisce un significato fisico grazie al modo in cui si sono raggruppati i vari fattori che componevano le due equazioni. Infatti $M(q)$ è la matrice dei coefficienti inerziali, $C(q, \dot{q})$ è la matrice dei coefficienti che rappresentano la forza centrifuga e la forza di Coriolis, F_v è la matrice dei coefficienti di attrito viscoso, $g(q)$ è il vettore colonna che contiene l'accelerazione di gravità, τ è il vettore colonna contenente la coppia del motore.

2.3.1 Linearizzazione del sistema attorno al punto di equilibrio

Il modello appena presentato è un modello non lineare, infatti al suo interno presenta funzioni non lineari. Nel sistema "pendolo inverso" è noto che sia presente un punto di equilibrio instabile in corrispondenza dell'angolo $\vartheta = 0^\circ$, dunque con il pendolo posto esattamente sulla verticale del vincolo. Data la somiglianza della dinamica del pendolo inverso con la dinamica del robot in esame e visto che si vuole stabilizzare il robot lungo l'asse verticale facendolo comunque partire da angoli iniziali ϑ_0 non molto distanti dal valore nullo, è decisamente conveniente studiare la dinamica del nostro sistema in un intorno contenuto del punto di equilibrio.

L'idea è quindi quella di approssimare un sistema non lineare con un sistema lineare facendolo lavorare in un intervallo sufficientemente ristretto di valori in modo che non si

distanzi troppo dal modello originario.

Si considera dunque il punto di equilibrio instabile $P_0 = (q_0, \dot{q}_0, \ddot{q}_0, \tau_0)$ con $q_0 = [\gamma_0, 0]^T$, $\dot{q}_0 = \ddot{q}_0 = [0, 0]^T$ e $\tau_0 = 0$ (per ogni valore di $\gamma_0 \in \mathbb{R}$). La linearizzazione della (2.26) attorno al punto di equilibrio P_0 porta a:

$$f(P_0) + \frac{\partial f(P_0)}{\partial q} \delta q + \frac{\partial f(P_0)}{\partial \dot{q}} \delta \dot{q} + \frac{\partial f(P_0)}{\partial \ddot{q}} \delta \ddot{q} + \frac{\partial f(P_0)}{\partial \tau} \delta \tau = 0 \quad (2.33)$$

dove

$$f(q, \dot{q}, \ddot{q}, \tau) = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F_v\dot{q} + g(q) - \tau \quad (2.34)$$

e δq , $\delta \dot{q}$, $\delta \ddot{q}$ e $\delta \tau$ sono le piccole deviazioni attorno ai valori specificati dal punto di equilibrio P_0 (es. $\delta q = q - q_0$). È importante notare che si usano le espansioni di Taylor al primo ordine per linearizzare le funzioni non lineari come seno e coseno. Applicando la linearizzazione descritta dalla (2.33) alle due EoM (2.24) e (2.25) si ottiene:

$$\begin{aligned} & \left[2 I_{w,yy} + 2 N^2 I_{rot,yy} + (m_b + 2m_w + 2m_{rot})r^2 \right] \ddot{\gamma} + 2(B + B_w)\dot{\gamma} + \dots \\ & \dots + \left[2N(1-N)I_{rot,yy} + (m_b l + 2m_{rot} z_{rot}^b) r \right] \ddot{\vartheta} - 2B\dot{\vartheta} - 2\tau = 0 \end{aligned} \quad (2.35)$$

$$\begin{aligned} & \left[2N(1-N)I_{rot,yy} + (m_b l + 2m_{rot} z_{rot}^b) r \right] \ddot{\gamma} - 2B\dot{\gamma} + \dots \\ & \dots + \left[I_{b,yy} + 2(1-N)^2 I_{rot,yy} + m_b l^2 + 2m_{rot} (z_{rot}^b)^2 \right] \ddot{\vartheta} + 2B\dot{\vartheta} - \dots \\ & \dots - (m_b l + 2m_{rot} z_{rot}^b) g \vartheta + 2\tau = 0 \end{aligned} \quad (2.36)$$

dove con un piccolo abuso di notazione le variabili ϑ , γ e τ vengono riusate al posto delle piccole deviazioni (δq , $\delta \dot{q}$, $\delta \ddot{q}$ e $\delta \tau$) per facilitare la lettura delle equazioni. Si riscrive ora il modello in forma matriciale come fatto per il modello non lineare.

$$M \ddot{q} + F_v \dot{q} + Gq = \tau \quad (2.37)$$

dove

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \quad (2.38)$$

con

$$\begin{aligned} M_{11} &= 2 I_{w,yy} + 2 N^2 I_{rot,yy} + (m_b + 2 m_w + 2 m_{rot}) r^2 \\ M_{12} &= M_{21} = 2 N (1 - N) I_{rot,yy} + (m_b l + 2 m_{rot} z_{rot}^b) r \\ M_{22} &= I_{b,yy} + 2 (1 - N)^2 I_{rot,yy} + m_b l^2 + 2 m_{rot} (z_{rot}^b)^2 \end{aligned} \quad (2.39)$$

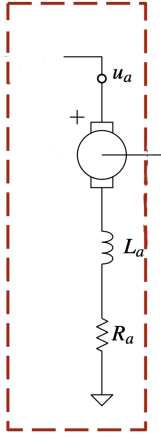
2.3. SCRITTURA DELLE EQUAZIONI DEL MOTO RICAVATE DALLA LAGRANGIANA

$$\begin{aligned} G_{11} = G_{12} = G_{21} = 0 \\ G_{22} = -(m_b l + 2 m_{rot} z_{rot}^b) g \end{aligned} \quad (2.40)$$

M e G sono la versione linearizzata attorno al punto di equilibrio P_0 della matrice inerziale $M(q)$ e del vettore di gravità $g(q)$, F_v rimane invariata.

2.3.2 Modellizzazione del sistema di attuazione

DC Motor



Coefficienti motore DC

Resistenza armatura	R_a	2.4Ω
Induttanza armatura	L_a	n.a. (trascurabile)
Costante elettromotrice	k_e	10.3×10^{-3} Vs/rad
Costante di torque	k_t	5.2×10^{-3} Nm/A
Rapporto del cambio	N	30

Figura 2.5: Schema circuitale e parametri motore DC del robot

La coppia elettomeccanica τ che viene poi trasmessa alle ruote è generata dal motore DC. A ciascuna delle due ruote viene trasmessa una coppia pari a:

$$\tau = N k_t i_a \quad (2.41)$$

dove i_a è la corrente di armatura, ossia la corrente assorbita dal motore, k_t è la costante di torque del motore. La corrente i_a deve soddisfare la seguente equazione alla maglia (vedi figura 2.5):

$$L_a \frac{di_a}{dt} + R_a i_a + k_e \frac{d\Delta\vartheta_{rot}}{dt} = u_a \quad (2.42)$$

dove R_a e L_a sono la resistenza e l'induttanza dell'armatura del motore, k_e è la costante elettrica (costante elettromotrice), $\Delta\vartheta_{rot} = \vartheta_{rot} - \vartheta = N(\gamma - \vartheta)$ è lo spostamento angolare del rotore rispetto allo statore. Dunque il prodotto tra la costante elettromotrice e la derivata temporale dello spostamento angolare, è la forza elettromotrice indotta (*f.e.m.*). Si suppone ora che la costante di tempo elettrico L_a/R_a sia trascurabile rispetto alla più piccola costante di tempo meccanica del sistema.

L'equazione differenziale (2.42) si riduce quindi alla seguente relazione algebrica:

$$i_a = \frac{1}{R_a} \left[u_a - k_e N (\dot{\gamma} - \dot{\vartheta}) \right] \quad (2.43)$$

si sostituisce ora la (2.43) alla (2.41) e si ricava che la coppia τ equivale a:

$$\tau = \frac{N k_t}{R_a} u_a - \frac{N^2 k_t k_e}{R_a} (\dot{\gamma} - \dot{\vartheta}) \quad (2.44)$$

Per trovare il modello non lineare completo si sostituisce l'equazione sopra riportata alla (2.26) e si trova che:

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + F'_v \dot{q} + g(q) = \tau' \quad (2.45)$$

dove

$$F'_v = F_v + \frac{2 N^2 k_t k_e}{R_a} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad \tau' = \frac{2 N k_t}{R_a} \begin{bmatrix} 1 \\ -1 \end{bmatrix} u_a \quad (2.46)$$

Similmente, sostituendo la (2.44) alla (2.37) si ricava il modello lineare definitivo:

$$M \ddot{q} + F'_v \dot{q} + Gq = \tau' \quad (2.47)$$

2.4 Derivazione delle funzioni di trasferimento dal modello linearizzato in spazio di stato

Ai fini della progettazione del controllo per il robot si parte dal modello linearizzato e si ricava il modello in forma di spazio di stato.

$$\begin{cases} \dot{x} = A x + B u \\ y = C x + D u \end{cases} \quad (2.48)$$

La variabile x è la variabile di stato che, grazie alla proprietà di separazione, riassume tutte le precedenti informazioni del modello necessarie e sufficienti a calcolare l'evoluzione futura del sistema. La variabile u è invece l'ingresso del sistema.

Nel nostro caso le variabili di stato sono quattro, l'angolo ϑ e l'angolo γ e le rispettive

2.4. DERIVAZIONE DELLE FUNZIONI DI TRASFERIMENTO DAL MODELLO LINEARIZZATO IN SPAZIO DI STATO

velocità angolari $\dot{\vartheta}$ e $\dot{\gamma}$. L'ingresso del sistema è pari a u_a , mentre le matrici sono:

$$A = \begin{bmatrix} 0_{2 \times 2} & I_{2 \times 2} \\ -M^{-1}G & -M^{-1}F'_v \end{bmatrix}, \quad B = \frac{2 N k_t}{R_a} \begin{bmatrix} 0_{2 \times 2} \\ M^{-1} \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad D = [0, 0, 0, 0]^T \quad (2.49)$$

La matrice A è detta "matrice dinamica", la matrice B è detta "matrice d'ingresso", la matrice C è detta "matrice d'uscita", la matrice D è la "matrice di legame diretto ingresso-uscita", nel caso in esame però il sistema non ha tale legame, dunque D è una matrice nulla.

Come accennato nell'introduzione, in questa trattazione si vogliono progettare due controlli, il primo che garantisca la stabilizzazione del robot lungo l'asse verticale; il secondo che assicuri l'inseguimento, da parte delle ruote, di un segnale di spazio fornito, senza però far sbilanciare troppo il robot dalla posizione di equilibrio verticale. Entrambi i controlli che si vogliono progettare vengono chiamati controlli di posizione, perché le variabili che si vanno a controllare sono ϑ e γ ossia i due angoli che determinano la posizione rispettivamente del corpo del robot rispetto alla verticale e delle ruote rispetto al segnale che si vuole inseguire. L'interesse dunque, in questo caso, sarà solo per le due variabili di stato del sistema che indicano una posizione (ϑ e γ) non per le altre che rappresentano una velocità ($\dot{\vartheta}$ e $\dot{\gamma}$).

Dal modello lineare in spazio di stato è possibile ricavare abbastanza facilmente le Funzioni di Trasferimento (FdT) del sistema di nostro interesse, rispettivamente una per ϑ e una per γ . Per calcolarle imponiamo a zero la terza e la quarta riga della "matrice d'uscita", così da non ottenere le FdT riguardanti $\dot{\vartheta}$ e $\dot{\gamma}$. La matrice C diventa quindi:

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.50)$$

Per poter ricavare, in modo analitico, le due FdT bisognerebbe svolgere moltiplicazioni tra matrici che potrebbero risultare tediose, perché dalla letteratura è noto che per

calcolare la FdT, dato un modello in spazio di stato, si opera nel seguente modo:

$$\begin{bmatrix} G_\vartheta \\ G_\gamma \end{bmatrix} = C (sI - A)^{-1} B + D \quad (2.51)$$

Avendo a disposizione Matlab e il tool "Control system toolbox" è possibile scrivere in uno script tutti i parametri del robot così da ricavare le matrici dello spazio di stato linearizzato A, B, C, D e tramite il comando "ss(A, B, C, D)" si ricavano direttamente le due FdT di nostro interesse.

$$G_\vartheta = \frac{-3.2537 (s + 0.808)}{(s - 7.146) (s + 8.186) (s + 5.15)} \quad (2.52)$$

$$G_\gamma = \frac{4.8693 (s - 9.163) (s + 9.163)}{s (s - 7.146) (s + 8.186) (s + 5.15)} \quad (2.53)$$

Dalla formulazione in zeri e poli delle due FdT è possibile notare che in ognuna è presente un polo a parte reale positiva che determina l'instabilità dei sistemi. Questo non è strano perché il sistema "pendolo inverso", a cui possiamo ricondurre il nostro robot, è un sistema instabile.

Dopo aver ricavato le due FdT è possibile analizzarle ulteriormente attraverso i diagrammi di Bode, i quali ci mostrano il comportamento del modulo e della fase delle funzioni al variare della frequenza.

Dai diagrammi di Bode (vedi figura 2.6) è possibile confrontare i vari punti di spezzamento con i poli e gli zeri messi a disposizione della formulazione (2.52) e (2.53) delle FdT; dagli stessi diagrammi è anche possibile osservare come il comportamento delle due FdT sia diverso al variare della frequenza. Per quanto riguarda il modulo si nota che la funzione di γ ha una pulsazione di attraversamento circa pari ad uno, mentre la funzione di ϑ non attraversa mai gli zero dB e resta sempre sotto tale valore. Anche sulla fase si possono notare delle grandi differenze tra le due funzioni, infatti quella di γ è sempre positiva, mentre quella di ϑ è sempre negativa.

2.4. DERIVAZIONE DELLE FUNZIONI DI TRASFERIMENTO DAL MODELLO LINEARIZZATO IN SPAZIO DI STATO

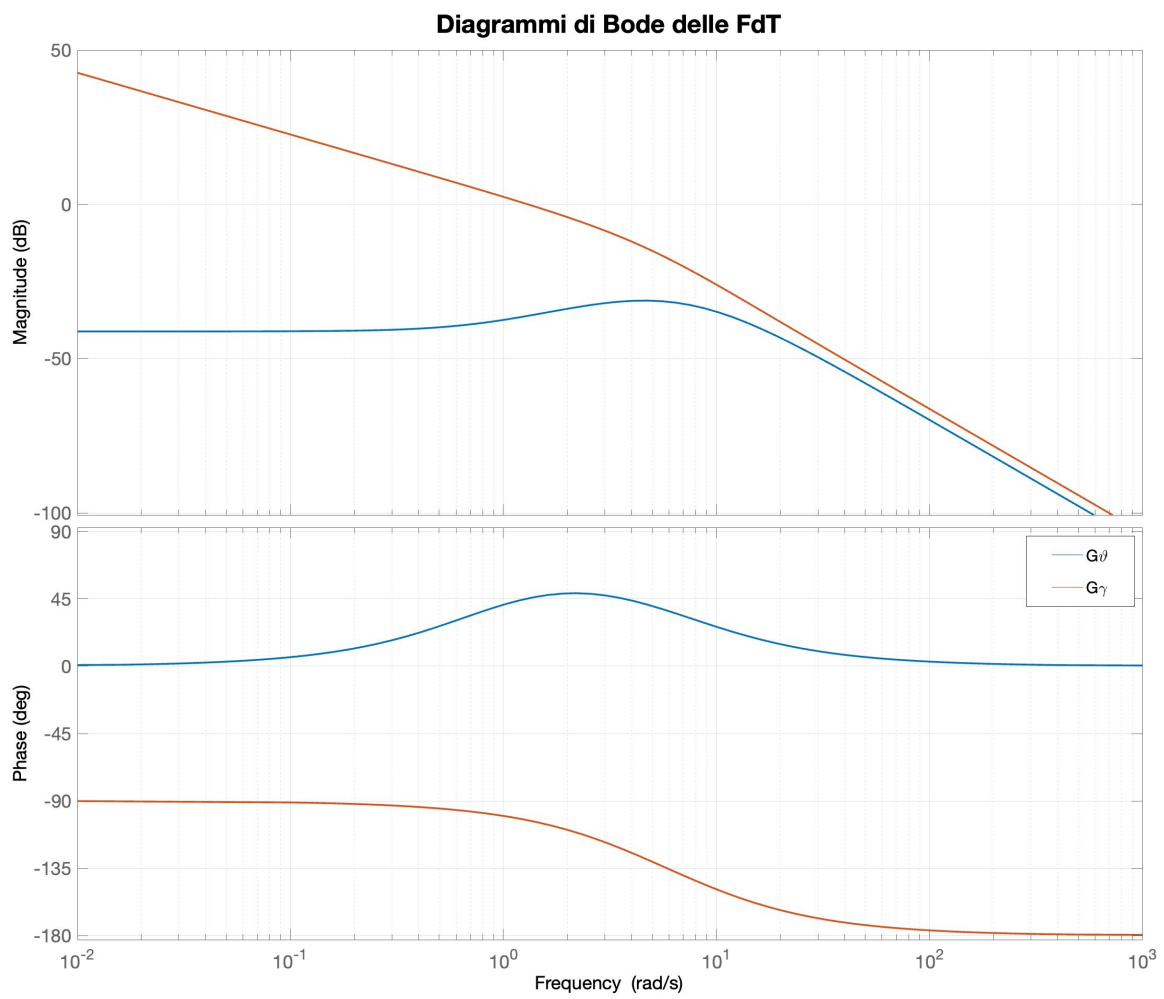


Figura 2.6: Diagrammi di Bode delle Funzioni di Trasferimento (FdT) del sistema

Capitolo 3

Sintesi dei controllori PID

In questo capitolo si analizza come sintetizzare il controllo del sistema balancing robot, precedentemente modellizzato, attraverso l'utilizzo di due controllori di tipo PID:

- Il primo per la stabilizzazione verticale del robot PID_{ϑ} , il quale deve essere in grado di mantenere, per il maggior tempo possibile, il robot in una posizione accettabilmente verticale.
- Il secondo PID_{γ} che deve riuscire a garantire un corretto inseguimento di posizione da parte delle ruote, qualora sia imposto un segnale di riferimento che implica uno spostamento del robot grazie ad una coppia impressa alle ruote da parte del motore DC.

I due controllori PID, in genere, possono essere disposti in due configurazioni: in parallelo (vedi figura 3.1) o in serie (vedi figura 3.2). Dopo aver consultato la pubblicazione in cui viene trattata la differenza tra le due strutture (vedi Bibliografia [3]), si è preferita la disposizione dei due PID in parallelo. Questa scelta progettuale consente di raggiungere prestazioni dinamiche ed una precisione che sono migliori rispetto alla disposizione in serie, ciò vale sia per il controllo in velocità sia per il controllo di posizione che è quello utilizzato in questa trattazione.

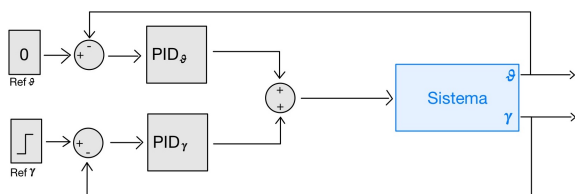


Figura 3.1: Config. PID parallelo

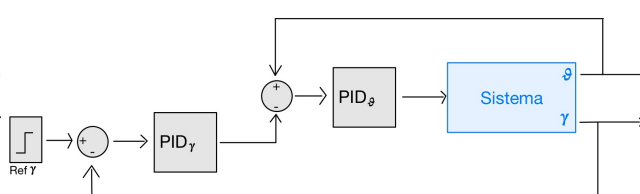


Figura 3.2: Config. PID serie

Prima di cominciare con la progettazione del controllo in ambiente simulato è importante fare alcune considerazioni riguardo al sistema.

Al balancing robot è possibile applicare solo un segnale di riferimento di posizione che attraverso il motore fa muovere le due ruote coassiali, il movimento imposto alle ruote,

3.1. CONTROLLO DELL'ANGOLO ϑ

oltre a far muovere il robot nella direzione parallela al suolo, è obbligatoriamente associato ad un sbilanciamento del telaio del robot nella direzione opposta al moto, facendo così allontanare il corpo del robot dalla posizione di equilibrio verticale.

È importante dunque, prima pensare ad un controllore per la stabilizzazione verticale del robot attorno al valore nullo di ϑ disinteressandosi totalmente del movimento delle ruote, successivamente si passa alla progettazione del controllo per l'inseguimento di spazio assumendo che il controllore di ϑ sia parte integrante del sistema. Applicando questo metodo si semplifica anche la problematica della sintesi di un controllo per un sistema SIMO, perché è come se si svolgesse la progettazione dei due controlli per due sistemi Singol Input Singol Output (SISO) separati.

Questa modalità di risoluzione del problema è validata anche dal punto di vista teorico, infatti le due funzioni di trasferimento hanno una dinamica diversa, come si vede dai diagrammi di Bode (vedi figura 2.6). La funzione G_γ ha una dinamica più veloce rispetto a G_ϑ , dunque se si stabilizza prima il robot lungo l'asse verticale e poi si pensa ad inseguire un riferimento di spazio, la stabilizzazione di γ non influenzerà eccessivamente la stabilizzazione dell'angolo, affrontare il problema diversamente sarebbe invece molto più complicato.

3.1 Controllo dell'angolo ϑ

Per la progettazione del controllo dell'angolo che stabilizzi il corpo del robot nella posizione di equilibrio instabile ($\vartheta = 0^\circ$) si comincia implementando in ambiente Simulink il modello lineare del sistema in modo da poter cominciare l'elaborazione del controllo in un ambiente dove è possibile descrivere i sistemi direttamente a tempo continuo.

Dato che si vuole pensare solo alla sintesi del controllo di ϑ è necessario imporre la "matrice d'uscita" $C = [0 \ 1 \ 0 \ 0]$ così da avere la sola variabile d'interesse come uscita del sistema e poterla quindi retroazionare. Grazie al blocco "state - space", disponibile in Simulink, è possibile scrivere al suo interno direttamente le matrici di stato A, B, C, D ; inoltre questo blocco dà la possibilità di aggiungere una condizione iniziale al sistema. Nel caso in esame si suppone che, prima di accendere l'automa, questo venga collocato da un operatore esterno, in una posizione adiacente alla verticale.

La posizione in questione può discostarsi dal punto di equilibrio di un valore massimo di $\vartheta_0 = \pm 5^\circ$. Vista la simmetria del sistema si imposta il valore positivo di ϑ_0 come angolo di partenza nel blocco "state - space" ($\vartheta_0 = 5^\circ \simeq 0,0873$ [rad]).

Per l'implementazione del controllore PID in spazio simulato, si fa uso del blocco PID(s) nel quale si possono andare a modificare facilmente i tre parametri di controllo che andranno ad agire, in modo pesato, sull'errore del sistema a tempo continuo.

Si riporta di seguito lo schema Simulink costruito:

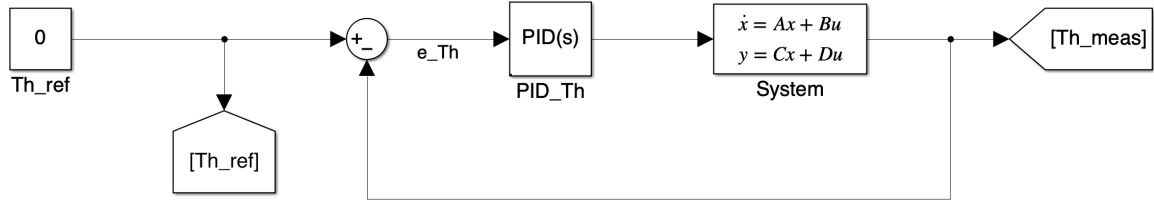


Figura 3.3: Schema Simulink per la sintesi del controllo di ϑ

Per la taratura dei parametri del Controllore PID_{ϑ} ci si è basati, in prima istanza, sul metodo euristico Ziegler-Nichols, successivamente sapendo come influiscono i tre parametri del controllore PID sulla risposta del sistema (vedi tabella 3.1), si sono eseguite varie prove e si sono ottenute due tarature: una più aggressiva ed una meno aggressiva.

	Tempo di salita	Sovra/Sottoelongazione	Errore a regime	Stabilità
$K_P \nearrow$	\searrow	\nearrow	\searrow	\searrow
$K_I \nearrow$	\searrow	\nearrow	"Annulla"	\searrow
$K_D \nearrow$	Dipende dal caso	\searrow	Inalterato	\nearrow

Tabella 3.1: Influenza dei coefficienti del PID sulla risposta del sistema

Si riportano ora i valori dei parametri delle due tarature per il PID_{ϑ} affiancati alle rispettive risposte del sistema.

Taratura meno aggressiva	
K_P	-80
K_I	-282
K_D	-5.3

Taratura più aggressiva	
K_P	-150
K_I	-400
K_D	-6.5

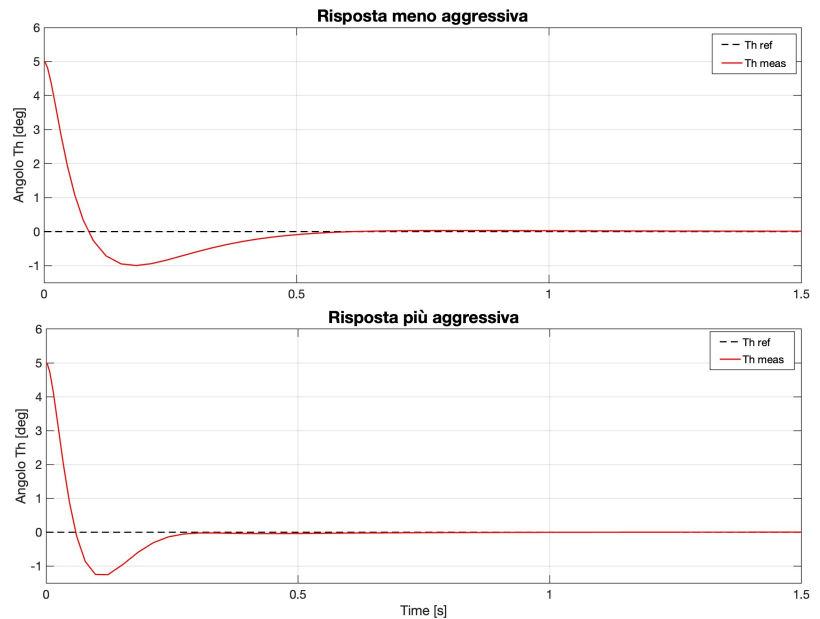


Tabella 3.2: Tarature del PID_{ϑ}

Figura 3.4: Risposta dell'angolo ϑ rispetto al sistema lineare

3.2 Controllo dell'angolo γ

A seguito della stabilizzazione, nel punto di equilibrio, dell'angolo ϑ è possibile procedere con la progettazione del controllo dell'angolo γ per garantire la stabilità verticale anche in una determinata posizione dello spazio, oppure riuscire a far muovere il robot avanti e/o indietro, nel piano $x-z$ del sistema di coordinate terra $\{o\}$, in modo che tali movimenti non causino un distanziamento troppo elevato dalla posizione di equilibrio verticale facendo cadere la macchina.

Si implementa questa caratteristica di controllo perché, oltre ad aggiungere una specifica per rendere il sistema finale più completo da un punto di vista controllistico, si desidera anche vedere come il controllo dell'angolo ϑ reagisce ad eventuali disturbi che lo portano fuori dalla posizione di equilibrio precedentemente acquisita.

È necessario che venga applicato rispetto alla variabile γ , il segnale di posizione che si genera per far avvenire uno spostamento. Questo perché è quest'ultima variabile quella che indica il movimento delle ruote in riferimento alla posizione di partenza. Diventa quindi fondamentale avere in uscita dal sistema lineare, oltre all'angolo ϑ , anche l'angolo γ per svolgere la retroazione di entrambe le variabili d'uscita così da ottenere un sistema in catena chiusa che garantisce il calcolo dell'errore sul quale lavora poi l'azione di controllo.

Affinchè si ottengano le due variabili di controllo in uscita dal sistema è necessario modificare lo script Matlab in modo che "matrice d'uscita" C abbia due righe.

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Per lo spostamento del robot viene scelto, nella fase di test, un segnale a gradino che agisce sul sistema solamente dopo l'avvenuta stabilizzazione iniziale del robot lungo la verticale. Il segnale scelto non è casuale, perché impone un cambiamento istantaneo del valore di γ mettendo dunque alla prova tutto il sistema, l'ampiezza di tale segnale non deve essere quindi troppo elevata, perché dato il cambiamento repentino di γ , il robot non riuscirebbe nella realizzazione del brusco spostamento.

L'ampiezza scelta per il segnale a gradino è pari quindi ad un radiante; quest'ultimo moltiplicato per il raggio della ruota corrisponde ad un movimento del robot di circa 0.0340 [m]. Questa operazione viene effettuata al fine di riuscire a capire quanto effettivamente il segnale sposti l'automa.

Successivamente, come fatto per la progettazione del controllo precedente, si implementa il sistema linearizzato in spazio di stato in Simulink e si aggiungono i blocchi necessari a rappresentare il sistema in catena chiusa con i due controllori PID.

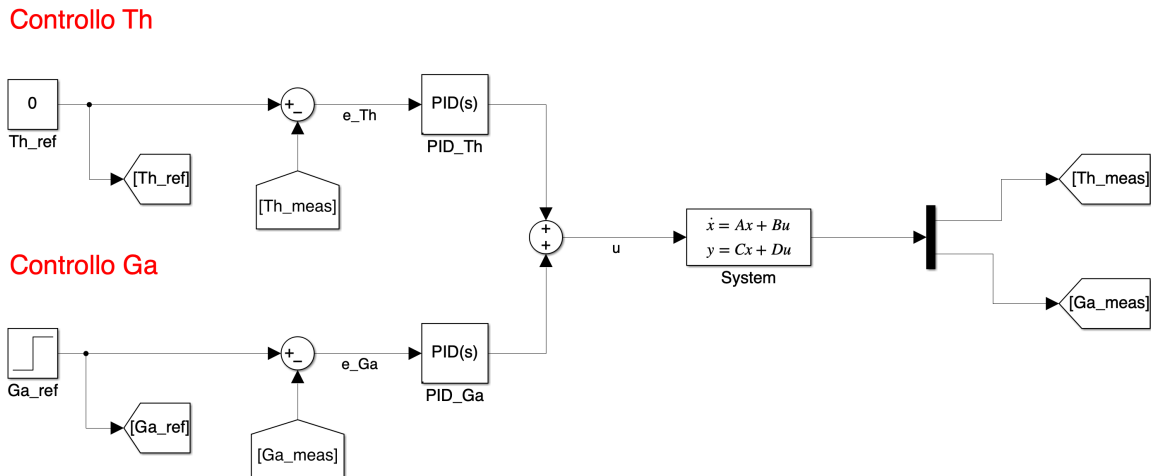


Figura 3.5: Schema Simulink per la sintesi del controllo di γ

Per la taratura del controllore PID_γ è necessario ricordare quanto scritto all’inizio del capitolo, ossia che il movimento del robot si oppone alla stabilizzazione dello stesso lungo l’asse verticale. È proprio per questo motivo che, come riportato nel paragrafo precedente, è necessario impostare che il riferimento a gradino di γ avvenga solo dopo che l’angolo ϑ si è stabilizzato, inoltre, sempre per il medesimo motivo, non è possibile tarare in modalità aggressiva entrambi i controlli assieme (soprattutto per quanto riguarda il termine integrale), perché ciò potrebbe portare entrambi i sistemi all’instabilità, oppure si otterrebbe una risposta con una sovraelongazione troppo ampia; questo non sarebbe sicuramente ammissibile dal sistema reale.

Dalla figura 3.6 e dalla tabella 3.3 (riportate nella pagina successiva), si possono esaminare le tarature migliori e le rispettive risposte, riferite al sistema riportato in figura 3.5. Queste sono state trovate a seguito di una serie di prove ripetute facendo, ancora una volta, riferimento alla tabella 3.1.

Come visto nella sezione precedente, i grafici con i valori in rosso riportano le risposte dell’angolo ϑ in gradi, mentre i grafici con i valori in blu riportano lo spostamento in metri del robot. Quest’ultimo movimento è calcolato facendo il prodotto tra la risposta (in radianti) dell’angolo γ ed il valore noto del raggio della ruota (vedi tabella 2.1). Questa operazione viene svolta per dare un’idea più concreta di quanto effettivamente il robot possa spostarsi nello spazio reale.

Nella parte alta della figura 3.6 si osserva una risposta del sistema più "soft" con una minor prontezza, soprattutto per l’inseguimento di spazio da parte delle ruote, questo comporta una sovraelongazione minore sia per ϑ che per γ . La motivazione di questa risposta (particolarmente favorevole) deve essere attribuita alla maggiore aggressività dei coefficienti nel controllo di ϑ .

Nella parte bassa invece si osserva una risposta con una prontezza maggiore, soprattutto

3.2. CONTROLLO DELL'ANGOLO γ

Taratura A			
PID $_{\vartheta}$		PID $_{\gamma}$	
K_P	-150	K_P	-4.86
K_I	-400	K_I	-3.86
K_D	-6.5	K_D	-0.38

Taratura B			
PID $_{\vartheta}$		PID $_{\gamma}$	
K_P	-80	K_P	-3.86
K_I	-282	K_I	-7.86
K_D	-5.3	K_D	-0.5

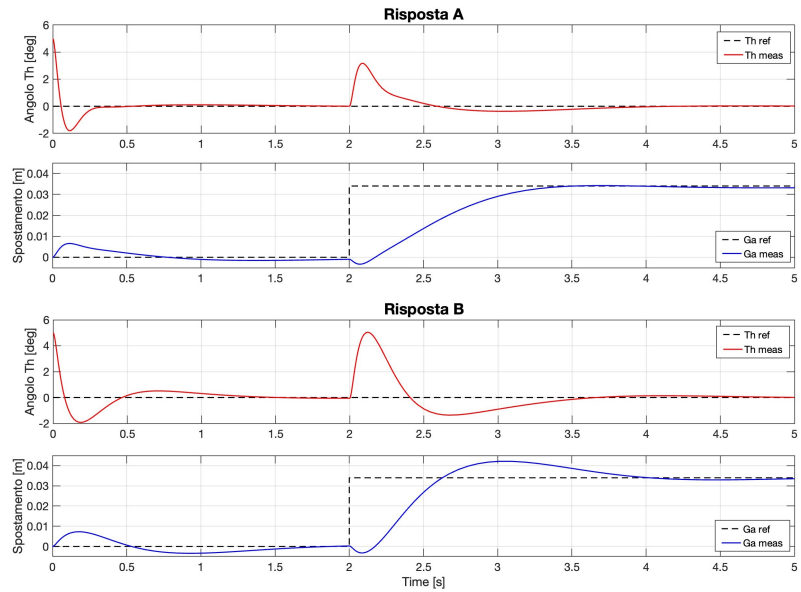


Tabella 3.3: Tarature sperimentali dei PID

Figura 3.6: Risposte dell'angolo ϑ e dell'angolo γ , convertito in spostamento, rispetto al sistema lineare

per γ , dunque una velocità più alta nel raggiungere il valore di regime, ma il compromesso per ottenere questa velocità è quello di avere una maggior sovraelongazione in entrambe le risposte.

È interessante notare anche come dall'analisi sperimentale dalle due risposte del sistema, venga confermato quanto anticipato all'inizio del capitolo: il movimento delle ruote porta il robot fuori dall'equilibrio verticale.

Oltre alle risposte del sistema è anche molto importante monitorare i valori assunti dal segnale u (segnale di controllo o di ingresso al sistema) ossia il risultato dalla somma dei due segnali uscenti dai controllori (vedi figura 3.5).

In figura 3.7 vengono riportati due grafici dove viene rappresentato l'andamento di tale segnale nel tempo per le due rispettive tarature trovate. Come si può osservare dai grafici il valore del segnale di controllo è espresso in volt, perché porta l'informazione di tensione che deve essere fornita ai motori per garantire al robot di muoversi come riportato nella figura 3.6.

Mettendo in relazione i grafici delle tensioni di controllo con i grafici delle risposte del sistema salta subito all'occhio che il segnale di controllo assume il valore nullo nei tempi in cui il robot è praticamente fermo e stabile. Al contrario all'inizio della simulazione e nell'istante in cui viene fornito il segnale a gradino (per l'inseguimento di spazio), il segnale di controllo u in entrambe le tarature, è molto distante dal valore nullo assumendo voltaggi molto elevati. Questo accade perché l'azione di controllo imporrebbe che il motore si muovesse in modo repentino al fine di riportare il robot in una posizione di stabilità, ma se nel robot fisicamente realizzato si verificassero sbalzi di tensione così elevati, tutti

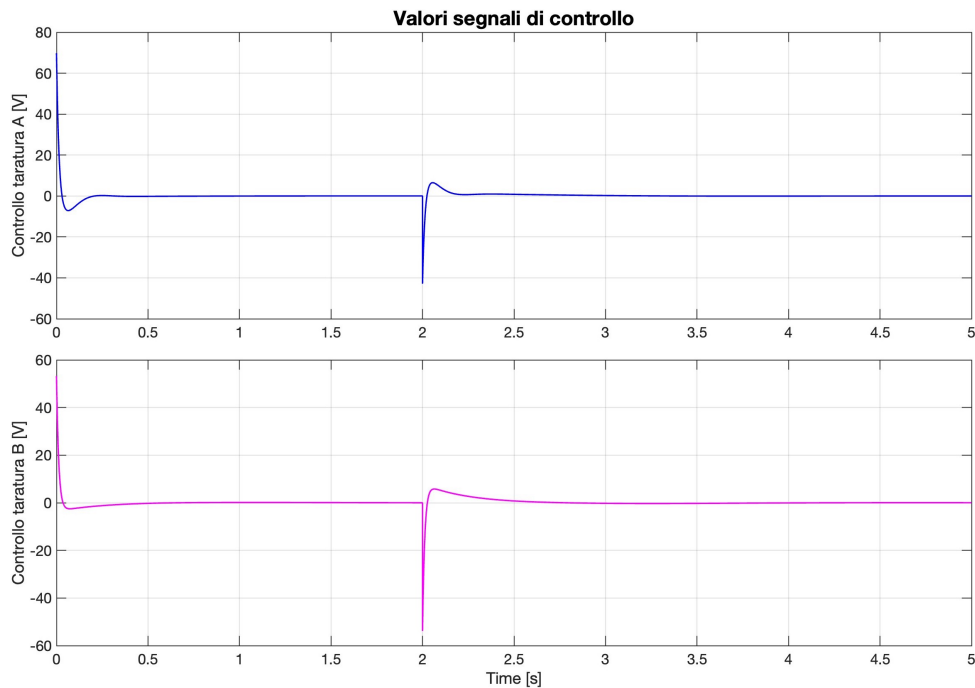


Figura 3.7: Grafici dell'andamento dei segnali di controllo

i componenti della macchina che non sono stati progettati per lavorare a tensioni così elevate verrebbero sicuramente danneggiati. Fortunatamente il robot reale (sotto esame) ha, al suo interno, una limitazione sulla tensione che può essere erogata dalla batteria; questo per consentire la protezione di tutti i componenti (nella prossima sezione si andrà infatti ad analizzare l'impatto di tale limitazione sull'azione di controllo).

3.3 Sviluppo in ambiente simulato di un modello più simile al robot reale

Per affinare la progettazione dei due controllori PID posti in parallelo, si vuole ora lavorare su un sistema più simile a quello reale, per fare ciò si modifica il modello Simulink, eliminando le semplificazioni che si erano ipotizzate nella prima fase per il calcolo del modello lineare. Di seguito vengono elencate le varie modifiche che si apportano allo spazio di simulazione:

- passaggio dal modello lineare al modello non lineare del robot;
- aggiunta al ramo di retroazione di una modellizzazione dei sensori che il robot reale utilizza per stimare i due angoli d'interesse;
- discretizzazione della parte di stima dell'angolo ϑ e creazione dei controllori PID a tempo discreto;

3.3. SVILUPPO IN AMBIENTE SIMULATO DI UN MODELLO PIÙ SIMILE AL ROBOT REALE

- aggiunta di un saturatore per limitare i volt forniti dai controllori per stabilire una soglia limitata di valori che il segnale d'ingresso del sistema può assumere;
- aggiunta di un sistema Anti-Wind Up per limitare le sovraelongazioni delle risposte del sistema, che vengono incrementate a causa dell'aggiunta del saturatore.

Dopo aver elencato le varie modifiche da apportare al modello simulativo è necessario approfondirle e spiegarle più nel dettaglio una per volta.

3.3.1 Modello non lineare

Nella sezione 2.3.1 del capitolo 2 si sono linearizzate le EoM attorno al punto di equilibrio instabile P_0 . Questa operazione, ci ha garantito una maggiore facilità nella realizzazione del modello Simulink e ci ha permesso di svolgere una taratura preliminare dei controlli. La linearizzazione (essendo comunque teoricamente valida per angoli ϑ piccoli) è pur sempre però una semplificazione, infatti attraverso i metodi di derivazione in P_0 e le espansioni di Taylor al primo ordine si sono eliminate le funzioni non lineari. Se il valore dell'angolo ϑ comincia a distaccarsi maggiormente dal valore nullo, la procedura di linearizzazione non approssima più al meglio il sistema reale diventando quindi inutilizzabile. Volendo quindi migliorare la qualità del modello diventa necessario abbandonare il modello lineare a favore dell'implementazione del modello non lineare. Si scrivono dunque, in uno script Matlab, le equazioni del modello non lineare del robot (2.24) e (2.25) calcolate nel capitolo precedente. Infine si rappresenta in Simulink tale modello grazie all'utilizzo di funzioni specifiche di Matlab (ad es. la funzione "Block").

Dall'immagine 3.11 (vedi pagina 38) è possibile osservare il blocco Simulink dove è stato implementato il sistema non lineare. Si può inoltre notare dalla figura che i segnali del sistema sono:

- in ingresso il solo segnale di controllo;
- in uscita sei segnali: i primi quattro rappresentano le variabili di stato, gli ultimi due corrispondono alle rispettive accelerazioni $\ddot{\vartheta}$ e $\ddot{\gamma}$ (che saranno fondamentali per la parte di stima degli angoli ϑ e γ spiegata nella sezione successiva).

3.3.2 Modellizzazione della sensoristica del robot e stima degli angoli ϑ e γ

Per stimare l'angolo di inclinazione rispetto alla verticale, il robot utilizza due sensori: un accelerometro che misura l'accelerazione lineare ed un giroscopio che misura la velocità angolare. Questi due sensori possono essere schematizzati in un'unica unità che viene chiamata Motion Processing Unit (MPU), tale unità è posizionata appena sopra l'asse delle ruote e fa parte del telaio del robot (vedi figure 2.1 ÷ 2.4 a pagina 4 e 5).

Il suo sistema di riferimento $\{mpu\}$, con centro il CoM dei sensori, è solidale al sistema di coordinate telaio $\{b\}$. Nella tabella 3.4 si trovano le coordinate del centro, C_{mpu} , rispetto a quelle del telaio $\{b\}$.

Motion Processing Unit

Coordinate del centro di massa rispetto al sistema $\{b\}$ $x_{mpu}^b, y_{mpu}^b, z_{mpu}^b, 0, 0, 13.5$ [mm]

Tabella 3.4: Parametri geometrici della Motion Processing Unit (MPU)

La misurazione dell'accelerazione è equivalente all'accelerazione lineare effettiva sperimentata dal corpo del robot (misurata dal sensore) sommata al vettore di accelerazione gravitazionale che è costante (A). Invece, la misurazione della velocità angolare è fornita dal giroscopio e coincide con l'effettiva velocità angolare del corpo del robot (B).

- A. Per determinare l'espressione di uscita dell'accelerometro in funzione delle coordinate generalizzate del robot (e delle loro derivate), è necessario predeterminare il vettore di posizione del centro del sensore rispetto al sistema di riferimento terra e derivarlo due volte per ottenere l'accelerazione reale. Tale operazione è esattamente uguale a quanto eseguito per ricavare le velocità dei corpi rigidi svolta nella sezione 2.2 del precedente capitolo; l'unica differenza è che per ottenere l'accelerazione va derivato una seconda volta.

Si riporta di seguito il valore del vettore dell'accelerazione reale.

$$\ddot{p}_{C_{mpu}}^o = \ddot{p}_v^o + \frac{d^2 R_b^o}{dt^2} p_{C_{mpu}}^b + 2 \frac{dR_b^o}{dt} \dot{p}_{C_{mpu}}^b + R_b^o \ddot{p}_{C_{mpu}}^b = \begin{bmatrix} r \ddot{\gamma} + z_{mpu}^b (-\sin \vartheta \dot{\vartheta}^2 + \cos \vartheta \ddot{\vartheta}) \\ -z_{mpu}^b (-\cos \vartheta \dot{\vartheta}^2 + \sin \vartheta \ddot{\vartheta}) \end{bmatrix} \quad (3.1)$$

Successivamente si aggiunge il contributo del vettore gravità calcolato rispetto al sistema terra ($g^o = [0, -g]^T$, con $g = 9.81$ [m/s²]).

Una volta ottenuta la somma si deve riproiettarla, attraverso la matrice di rotazione R_o^b , sul sistema di riferimento solidale al sensore per ottenere l'output effettivo. Definendo l'output finale dell'accelerometro come un vettore lungo l'asse y prodotto dalle coordinate x e z si ottiene:

$$y_a = \begin{bmatrix} x_a^{mpu} \\ z_a^{mpu} \end{bmatrix} = R_o^b (\ddot{p}_{C_{mpu}}^o + g^o) = \begin{bmatrix} r \ddot{\gamma} \cos \vartheta + z_{mpu}^b \ddot{\vartheta} + g \sin \vartheta \\ r \ddot{\gamma} \sin \vartheta - z_{mpu}^b \dot{\vartheta}^2 - g \cos \vartheta \end{bmatrix} \quad (3.2)$$

- B. Il risultato del giroscopio è invece molto più banale, perché è semplicemente la derivata nel tempo dell'angolo ϑ .

$$y_g = \dot{\vartheta} \quad (3.3)$$

3.3. SVILUPPO IN AMBIENTE SIMULATO DI UN MODELLO PIÙ SIMILE AL ROBOT REALE

Fino ad ora si è spiegato come vengono calcolate le uscite dei due sensori, ma non come ricavare gli angoli ϑ e γ che sono i parametri d'interesse. L'angolo ϑ può però essere stimato proprio utilizzando i dati forniti dal MPU, visto che è noto il modo in cui i due sensori operano per calcolare le rispettive uscite.

Si inizia considerando l'espressione d'uscita dell'accelerometro (3.2) ed ipotizzando che il movimento del robot sia lento così che le tre accelerazioni:

- l'accelerazione lineare $r \ddot{\gamma}$;
- l'accelerazione tangenziale $z_{mpu}^b \ddot{\vartheta}$;
- l'accelerazione centripeta $z_{mpu}^b \dot{\vartheta}^2$;

siano quantità irrilevanti perché infinitesime. Sotto tale ipotesi l'equazione dell'accelerometro che produce il valore di accelerazione è pari a:

$$y_a = \begin{bmatrix} x_a^{mpu} \\ z_a^{mpu} \end{bmatrix} \approx \begin{bmatrix} g \sin \vartheta \\ -g \cos \vartheta \end{bmatrix} \quad (3.4)$$

Tale formulazione consiste nella proiezione dell'accelerazione di gravità lungo i due assi (x e z) del sensore. È quindi possibile ricavare, attraverso l'utilizzo della funzione trigonometrica arcotangente2, una stima dell'angolo ϑ , che indicheremo con $\hat{\vartheta}_a$.

La funzione trigonometrica arcotangente2 ($\text{atan2}(x, y)$) indica l'angolo in radianti tra il semiasse positivo delle X di un piano cartesiano ed un punto di coordinate (x, y) giacente su di esso. (N.B. gli argomenti reali x e y non devono essere entrambi nulli).

$$\hat{\vartheta}_a = \text{atan2}(x_a^{mpu}, -z_a^{mpu}) \quad (3.5)$$

Usando la formulazione sopra per stimare ϑ si verificano i due inconvenienti di seguito descritti.

1. la stima è sensibile alle accelerazioni del corpo del robot e dunque potrebbe essere inaffidabile qualora il movimento non sia abbastanza lento da poter ancora trascurare le tre componenti di accelerazione;
2. la stima dell'angolo è alterata a causa del rumore di uscita dell'accelerometro, che è piuttosto grande, soprattutto ad alta frequenza.

Per risolvere questi problemi è possibile effettuare la stima sfruttando un metodo diverso, che consiste nell'usufruire del secondo sensore presente nella MPU. Per svolgere la stima utilizzando il giroscopio basta semplicemente far riferimento alla sua uscita (3.3) ed integrarla rispetto al tempo come riportato di seguito:

$$\hat{\vartheta}_g = \hat{\vartheta}_g(0) + \int_0^t y_g(\tau) d\tau \quad (3.6)$$

Questa stima non è sicuramente dipendente dall'accelerazione del corpo del robot e risulta anche meno influenzata dal rumore del sensore che viene filtrato dall'azione integrale.

Anche in questo caso però si va incontro a delle problematiche:

1. per calcolare la stima dell'angolo finale è necessario conoscere l'angolo di partenza $\hat{\vartheta}_g(0)$.
2. l'uscita del giroscopio è tipicamente influenzata da due errori: un errore sistematico (costante) ed un errore di deriva (rampa lineare) che permette al termine integrale di crescere senza limiti nel tempo.

Dalla descrizione delle problematiche dei due sensori è possibile notare che la stima sostenuta sul valore di uscita dell'accelerometro ($\hat{\vartheta}_a$) è maggiormente attendibile a bassa frequenza, mentre la valutazione basata sull'uscita del giroscopio ($\hat{\vartheta}_g$) è prevalentemente affidabile ad alta frequenza, si può dunque ricorrere all'utilizzo di entrambi i sensori applicando un approccio di *filtraggio complementare* per combinare assieme i due differenti risultati provenienti dai due sensori (vedi Bibliografia [1]). Si ottiene così una stima finale $\hat{\vartheta}$ sufficientemente accurata ad ogni frequenza.

Per implementare il metodo di *filtraggio complementare* è necessario innanzitutto considerare un filtro passa-basso $H(s)$, con una frequenza di taglio w_c e guadagno unitario in DC. Successivamente si analizza il filtro $H'(s)$ complementare ad $H(s)$ in modo che $H(s) + H'(s) = 1$, ossia un filtro passa-alto con formulazione pari a $H'(s) = 1 - H(s)$. Il filtro passa-basso viene utilizzato per rimuovere la maggior parte del rumore ad alta frequenza che influenza la stima basata sull'accelerometro $\hat{\vartheta}_a$, mentre il filtro passa-alto è impiegato per eliminare gli errori a bassa frequenza che condizionano la valutazione svolta dal giroscopio $\hat{\vartheta}_g$.

Si riportano di seguito le formulazioni delle stime filtrate:

$$\hat{\vartheta}_{a,f} = H(s) \hat{\vartheta}_a, \quad \hat{\vartheta}_{g,f} = [1 - H(s)] \hat{\vartheta}_g \quad (3.7)$$

Queste stime possono essere considerate valide su due intervalli di frequenza disgiunti e complementari. Il primo risultato vale per frequenze comprese tra $[0, w_c)$, mentre per il secondo le frequenze devono appartenere all'intervallo $[w_c, +\infty)$.

È possibile sommare i due contributi di stima dell'angolo ϑ in modo da ottenere un'unica espressione valida per l'intera gamma di frequenze $[0, +\infty)$

$$\hat{\vartheta} = \hat{\vartheta}_{a,f} + \hat{\vartheta}_{g,f} = H(s) \hat{\vartheta}_a + [1 - H(s)] \hat{\vartheta}_g \quad (3.8)$$

La scelta dell'ordine e della frequenza di taglio della coppia di filtri complementari è necessariamente ottenuta come compromesso tra due requisiti contrastanti:

- l'attenuazione del rumore di uscita dell'accelerometro ad alta frequenza;

3.3. SVILUPPO IN AMBIENTE SIMULATO DI UN MODELLO PIÙ SIMILE AL ROBOT REALE

- la reiezione degli errori sistematici e di deriva a bassa frequenza dovuti al giroscopio.

Nel caso in esame si è scelto un filtro del primo ordine con una frequenza di taglio pari a $w_c = 0.35$ [Hz].

La stima dell'angolo ϑ è fondamentale anche per il calcolo dell'angolo γ che viene svolta attraverso un encoder incrementale; questo misura lo spostamento angolare, del rotore rispetto allo statore che è rigidamente attaccato al telaio del robot. È necessario, prima di utilizzare il valore fornito dall'encoder, effettuare le conversioni adatte perché l'uscita del dispositivo è fornita nell'unità di misura "[pulse per second]". Dopo che viene acquisito il segnale d'uscita dal sensore, la stima di γ viene svolta ricordando il funzionamento del cambio spiegato nel capitolo 2, in particolare isolando γ dalla formula $\Delta\vartheta_{rot} = N(\gamma - \vartheta)$.

È possibile osservare le immagini 3.11, 3.14, 3.15, 3.16 e 3.17 (vedi pagine 38, 39, 40) per comprendere meglio come è avvenuta l'implementazione dei sensori e della parte di stima in Simulink.

3.3.3 Trasformazione a tempo discreto della parte di stima e di controllo

Fino ad ora nel modellizzare il sistema non si è mai parlato di "sistema a tempo discreto", ma sempre di "sistema a tempo continuo". Questo, ancora una volta, perché si era in una fase di prima modellazione dove si desiderava semplificare (si era dunque supposto il tempo continuo per l'intero sistema). In realtà questa semplificazione non era poi del tutto sbagliata in quanto il "robot reale" pur essendo un sistema digitale, dunque un sistema discreto, ha una Central Processing Unit (CPU) con una frequenza di clock sufficientemente alta da poterlo approssimare ad un sistema continuo.

Per quanto riguarda la parte di stima e di controllo, è necessario invece che si passi dal tempo continuo al tempo discreto, il tempo di campionamento che viene scelto è pari a $Ts = 10^{-2}$ [s], ossia il tempo di campionamento a cui la MPU raccoglie i dati.

Per la discretizzazione della parte di stima basta semplicemente discretizzare i filtri $H(s)$ e $H'(s)$ attraverso la discretizzazione di "Eulero in avanti", ponendo dunque $s = (z-1)/Ts$. Per la parte di controllo è necessario ricostruire da zero i due controllori abbandonando quindi il blocco PID(s) e ricostruendo i due PID da adottare con i tre contributi disposti separatamente; questi ultimi andranno quindi a sommarsi tra loro.

Visto che si vuole costruire un controllore discreto, i termini da rendere discreti dei due PID sono i termini integrali e i termini derivativi, perché i termini proporzionali sono fattori puramente moltiplicativi dell'errore precedentemente calcolato.

Per la creazione di un termine integrale discreto, si fa sempre uso della discretizzazione di "Eulero in avanti" del termine integrale a tempo continuo ($1/s$), con il medesimo tempo di campionamento Ts usato nella discretizzazione della stima. Tale operazione è facilmente

realizzabile in simulazione perché nella libreria di blocchi di Simulink è presente il blocco "Discrete – Time Integrator" nel quale è già presente l'azione integrativa discretizzata, basta solo scegliere il tipo di discretizzazione assieme al tempo di campionamento desiterato.

Relativamente invece alla realizzazione del termine derivativo si è scelta un derivata discreta con l'aggiunta di un filtro a media mobile. Questo filtro è il più semplice filtro lineare e riesce a calcolare la media di N valori, dunque applicandolo assieme all'azione derivativa è possibile calcolare la derivata sulla media di N valori letti, al posto di applicarla ad ogni singolo valore. Tale operazione ha il compito di filtrare ulteriormente il valore di ingresso al termine derivativo che è stato calcolato dal sistema di stima (grazie all'azione di acquisizione dei sensori).

Come tempo di campionamento si rimane coerenti con la scelta di T_s , mentre si è scelto $N = 3$ per un motivo di "design di filtro".

Di seguito si riportano i passaggi matematici che si sono svolti per la creazione della derivata filtrata a tempo discreto.

$$\begin{aligned}
 Y(z) &= D(z) U(z) = \frac{1 - z^{-N}}{NT_s} = \frac{U(z) - z^{-3}U(z)}{3T_s} \\
 &\text{Antitrasformo} \\
 &\Downarrow \\
 Y(k) &= D(k) U(k) = \frac{U(k) - U(k - 3)}{3T_s} = \dots \\
 &\text{Aggiungo e tolgo } U(k - 1) \text{ e } U(k - 2) \\
 &\Downarrow \\
 \dots &= \frac{U(k) + U(k - 1) - U(k - 1) + U(k - 2) - U(k - 2) - U(k - 3)}{3T_s} = \dots \\
 \dots &= \frac{1}{3} \left[\frac{U(k) - U(k - 1)}{T} + \frac{U(k - 1) - U(k - 2)}{T} + \frac{U(k - 2) - U(k - 3)}{T} \right]
 \end{aligned} \tag{3.9}$$

Dall'ultima formulazione si vede chiaramente che si sta eseguendo la media delle derivate in: $k, k - 1$ e $k - 2$.

L'implementazione di tale derivata filtrata, in spazio simulato, può essere rappresentata nella figura sotto riportata.

3.3. SVILUPPO IN AMBIENTE SIMULATO DI UN MODELLO PIÙ SIMILE AL ROBOT REALE

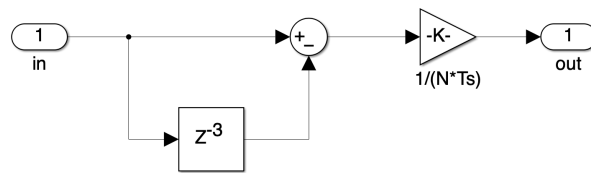


Figura 3.8: Derivata discreta con filtro a media mobile

3.3.4 Aggiunta saturatore

Per motivi di sicurezza al robot reale è stata imposta una limitazione rispetto ai volt che il controllore può fornire ai motori DC; in questo modo si ottiene la protezione di tutti i vari componenti elettronici dalle sovratensioni che potrebbero verificarsi. Diventa quindi necessario implementare tale limitazione anche nella nostra modellizzazione del sistema, perché la stessa influenza l'azione di controllo.

Il segnale d'ingresso al sistema deve dunque appartenere all'intervallo $[-11, +11][V]$, di conseguenza la somma dei due segnali di controllo, uscenti dai due PID, non deve mai eccedere da tale intervallo.

Di seguito si riporta un grafico dove è possibile apprezzare la saturazione del segnale di controllo quando quest'ultimo esce dall'intervallo di sicurezza.

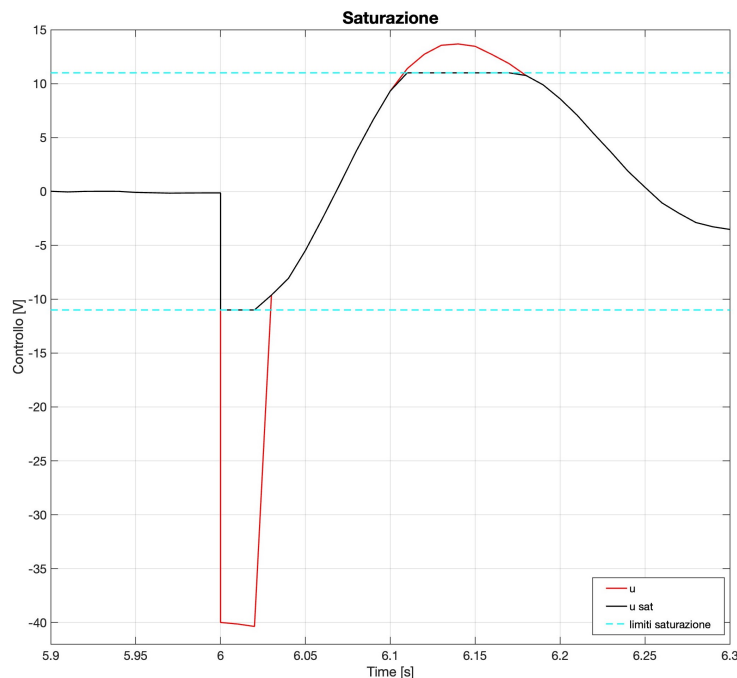


Figura 3.9: Grafico in cui viene rappresentato il segnale saturato assieme all'intervallo di lavoro

Per garantire ciò in simulazione è necessario aggiungere al modello Simulink un blocco di saturazione, sul segnale di input al sistema, selezionando l'intervallo d'interesse. Si veda

la successiva figura 3.11 (vedi pagina 38) per prendere visione di come si è realizzato lo schema con l'aggiunta di tale blocco.

L'azione saturativa, oltre a limitare i volt in entrata al sistema e quindi aiutare a garantire la sicurezza, provoca purtroppo un'azione indesiderata: il fenomeno del "Wind Up" che sarà analizzato nella sezione successiva del capitolo.

3.3.5 Aggiunta sistema "Anti-Wind Up"

Il Wind Up è un fenomeno indesiderato che consiste in un aumento significativo della sovraelongazione della risposta del sistema. Questa problematica è causata dall'accumulo del termine integrale del PID quando il segnale di controllo supera la soglia di lavoro stabilita e rende quindi necessario l'intervento dell'azione saturativa per limitarne l'ampiezza. L'accumulo del termine integrale deve però scaricarsi una volta che si è rientrati nell'intervallo limite del saturatore, ma questo da origine a sovraelongazioni non trascurabili nella risposta del sistema.

È possibile contenere questo fenomeno aggiungendo degli schemi studiati appositamente per limitare l'azione integrativa quando entra in funzione il blocco saturativo. Visto che il sistema sotto esame è un sistema digitale si è pensato di realizzare un sistema di "Anti-Wind Up" digitale.

Il modello che si utilizza è chiamato "*Schema di Clamping*", grazie all'immagine riportata di seguito, oltre a osservare come avviene l'implementazione della protezione nello spazio simulato, è possibile capire la metodologia di funzionamento.

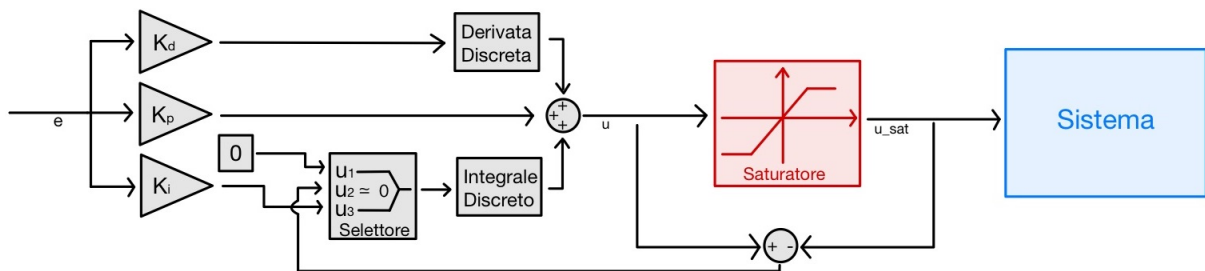


Figura 3.10: Schema Anti-Wind Up in forma di Clamping

Per comprendere il funzionamento di questo schema è necessario partire dall'operazione di differenza che viene svolta tra il valore del segnale in ingresso al saturatore e quello del segnale d'uscita; segnale d'ingresso al sistema. L'operazione di differenza ha in realtà il ruolo di essere considerata come un'operazione di confronto tra i due segnali, infatti il segnale nel quale viene calcolata la differenza diventa entrata di un blocco selettore, che in base al valore assunto del segnale differenza inibisce o meno l'azione del termine integrale. Il blocco selettore ha un funzionamento molto semplice:

3.3. SVILUPPO IN AMBIENTE SIMULATO DI UN MODELLO PIÙ SIMILE AL ROBOT REALE

- se il segnale \mathbf{u}_2 , detto anche segnale di controllo, soddisfa la condizione inserita, allora il selettore abilita il passaggio del segnale \mathbf{u}_3 ;
- se la condizione non è verificata il selettore abilita la trasmissione del segnale \mathbf{u}_1 .

Prendendo in considerazione il nostro caso, se il segnale di controllo è pari a zero, dunque il segnale di ingresso al sistema ottempera all'intervallo di valori desiderato, il controllore va ad agire normalmente apportando anche il contributo del termine integrale. Nel caso opposto, in cui la differenza tra i due segnali non è più pari a zero (il che indica l'entrata in funzione del saturatore), il selettore inibisce l'azione del termine integrale moltiplicandola per zero.

Per comprendere meglio come vengono ad essere implementati i due controllori PID discreti ed i rispettivi sistemi di "Anti-Wind Up" digitali si può fare riferimento alle figure 3.11, 3.12 e 3.13.

Dopo aver spiegato una per volta le modifiche che si vanno ad effettuare sul sistema in ambiente simulato, per renderlo il più possibile simile al sistema reale, è necessario riportare una documentazione fotografica dalla quale è possibile comprendere anche visivamente come sia stata eseguita la costruzione del modello e di tutte le sue parti.

Si riporta dunque prima un immagine dove viene rappresentato lo schema completo del sistema e successivamente una serie di immagini consecutive dove è possibile apprezzare il modo in cui sono stati implementati i vari blocchi che compongono il sistema nella sua interezza.

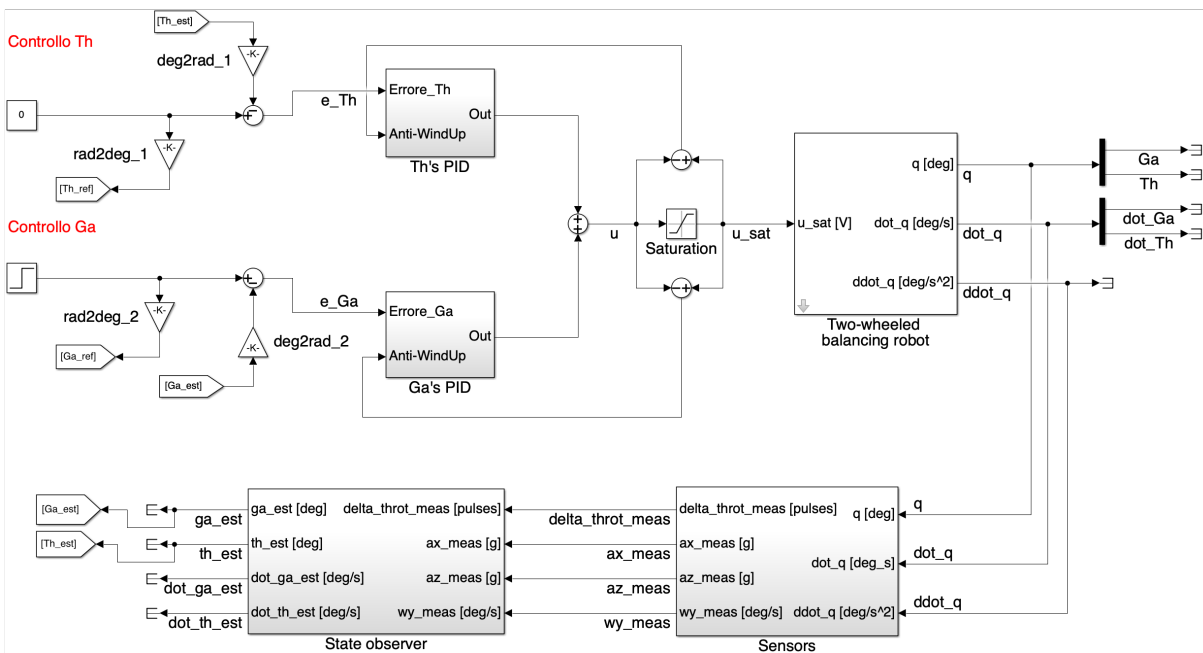


Figura 3.11: Schema completo e discretizzato del modello non lineare del robot

Th's_PID

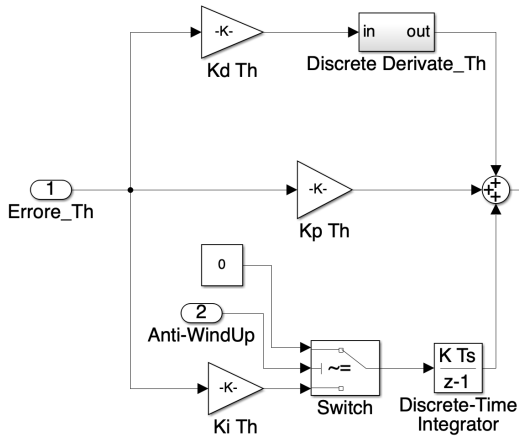


Figura 3.12: Controllo discreto di ϑ

Ga's_PID

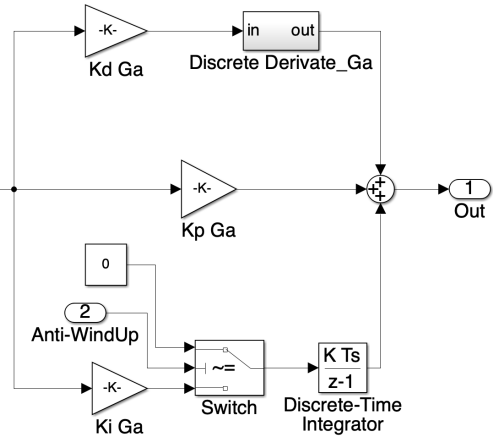


Figura 3.13: Controllo discreto di γ

Sensors

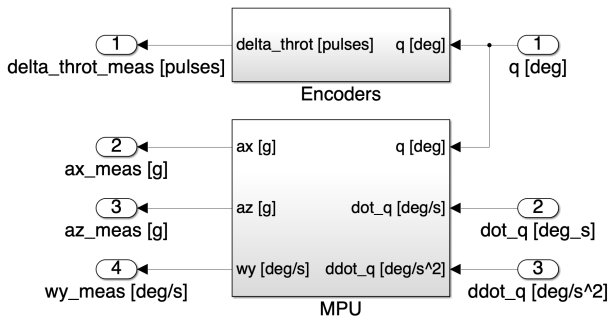


Figura 3.14: Sensori a bordo del robot

MPU

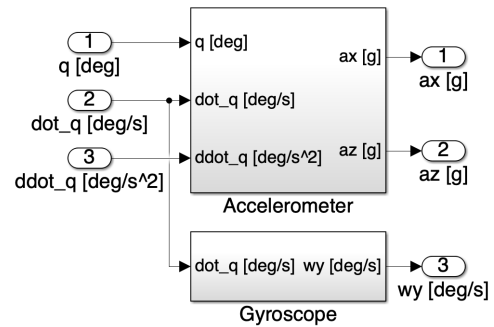


Figura 3.15: Giroscopio e accelerometro

State observer

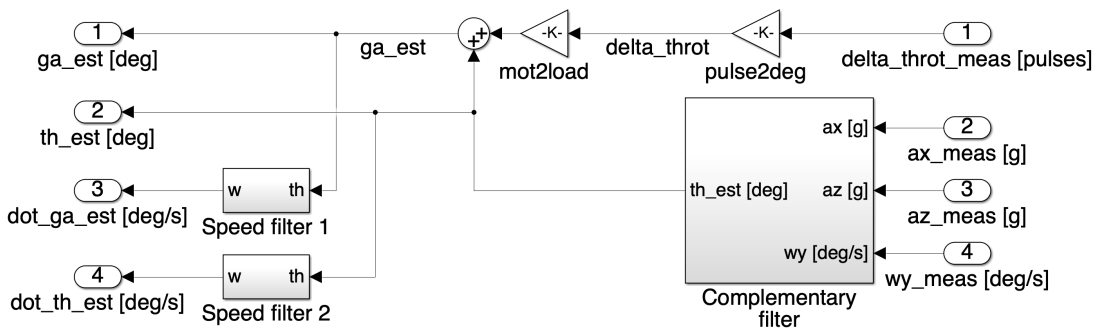


Figura 3.16: Parte di stima dei dati

3.4. ANALISI IN AMBIENTE SIMULATO DELLE RISPOSTE OTTENUTE DAL MODELLO COMPLETO

Complementary filter

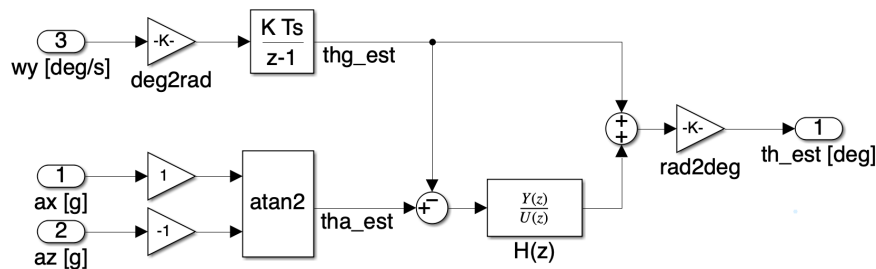


Figura 3.17: Stima dell'angolo ϑ

3.4 Analisi in ambiente simulato delle risposte ottenute dal modello completo

Dopo che si è discusso delle modifiche da apportare al modello e si è visto come si sono svolte le implementazioni in spazio simulato, è necessario analizzare e discutere se le risposte del sistema ottenute, al variare delle due tarature svolte precedentemente per il modello lineare, siano ancora considerate accettabili anche per il sistema non lineare. Si riportano di seguito le due risposte del sistema ottenute al variare delle tarature A e B (vedi tabella 3.3).

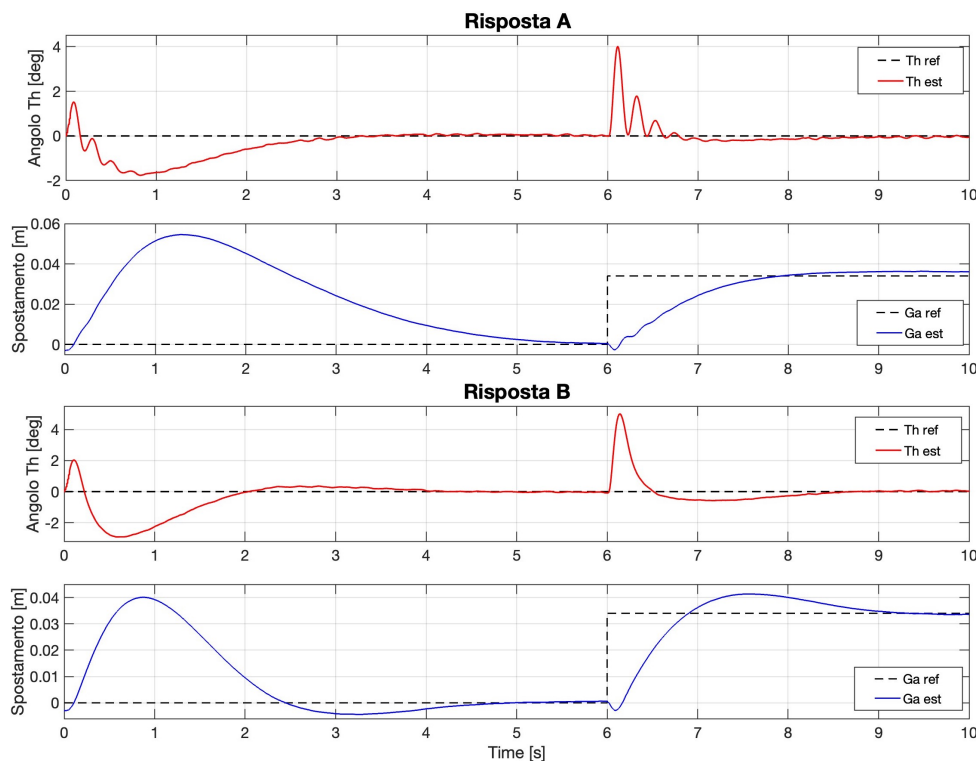


Figura 3.18: Risposta del sistema con le tarature dei PID del sistema lineare riportate nella tabella 3.3

Le risposte sopra riprodotte si differenziano in modo simile a come differivano nella modellizzazione lineare.

È importante sottolineare le diversità che le due risposte prodotte dal sistema non lineare, hanno con le rispettive risposte ottenute dal modello linearizzato (vedi figura 3.6); si ricorda che si è scelto di mantenere le medesime tarature. Le discrepanze più evidenti quando si mettono a confronto i grafici sono quelle di seguito illustrate. Nelle risposte del modello lineare sia ϑ che γ partono dai valori noti $\vartheta_0 = 5^\circ$ e $\gamma_0 = 0^\circ$, mentre in entrambe le risposte del modello non lineare le due variabili non partono più dalle medesime quantità (nonostante siano state comunque impostate all'interno del modello non lineare). Questa asimmetria è dovuta ai sensori ed al metodo di misura utilizzati dal robot per la stima delle variabili.

Nella sezione 3.3.2 di questo capitolo si sono spiegate le diverse difficoltà che si verificavano se si utilizzava solo uno dei due sensori per la stima dell'angolo ϑ , si era allora scelto di utilizzare entrambi i sensori per compensare le diverse problematrichie. Per fare ciò era necessario realizzare un'azione di *filtraggio complementare* dei segnali provenienti dai sensori.

Quest'ultima azione, pur permettendo l'utilizzo dei due sensori e aumentando in questo modo l'affidabilità del risultato, è responsabile dell'errore commesso sul valore di partenza di ϑ . Questo perché, nonostante l'accelerometro sia riuscito a campionare il valore di partenza con un errore di misura di circa 1.5° , il filtro elimina ugualmente il valore campionato perché considerato di disturbo.

Per risolvere il problema si dovrebbe modificare il filtro, ma così facendo si andrebbe a compromettere tutta la stima di valori successiva che, con la scelta del filtro effettuata precedentemente, risulta corretta a meno del valore di partenza.

Facendo riferimento, prima alla figura 3.17 (dove è riportato lo schema del *filtraggio complementare*) e successivamente al grafico dei segnali (riportato a pagina successiva in figura 3.19), è possibile apprezzare la funzione caratteristica del filtro sull'errore iniziale commesso sulla stima dell'angolo ϑ .

Per quanto riguarda invece l'errore sul valore iniziale di γ , questo è generato sia dai sensori che del sistema di stima.

L'encoder incrementale, come riportato alla fine della sezione 3.3.2, calcola lo spostamento angolare come prodotto tra il fattore di conversione del cambio N e la differenza tra il valore dell'angolo ϑ e quello dell'angolo γ provenienti dall'uscita del sistema non lineare. Tale operazione produce un valore iniziale di γ negativo, a causa di $\vartheta_0 = 5^\circ$. Il segnale inoltre deve poi essere campionato, quantizzato e convertito in [pulse], in modo da essere omogeneo con l'output del dispositivo reale. Con il valore dello spostamento angolare del rotore rispetto allo statore, calcolato dall'encoder, si procede a calcolare il valore

3.4. ANALISI IN AMBIENTE SIMULATO DELLE RISPOSTE OTTENUTE DAL MODELLO COMPLETO

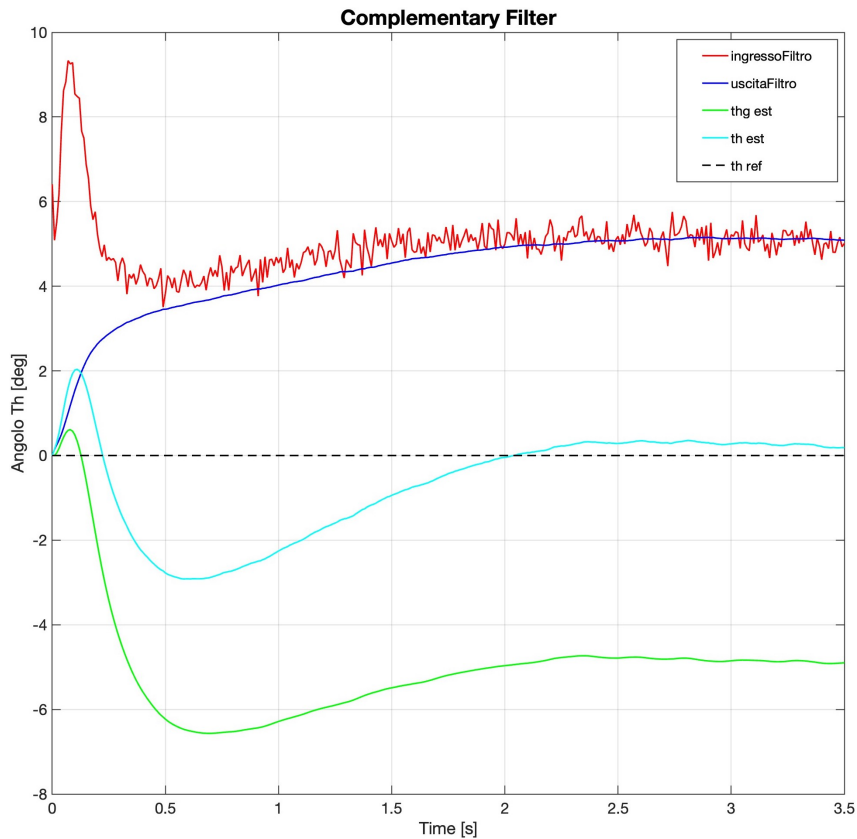


Figura 3.19: Grafico segnali filtraggio complementare

del segnale di γ nella parte dello state observer, (vedi figura 3.16). Il segnale di uscita dell'encoder subisce, prima due conversioni di unità di misura, a questo valore dovrà quindi essere sommato il valore dell'angolo ϑ stimato dal filtro complementare (in modo da andare a bilanciare la sottrazione precedentemente svolta dall'encoder incrementale). L'errore sul valore iniziale di γ ha origine dal fatto che il valore uscente dal filtro complementare per la stima di ϑ inizialmente non coincide, come visto sopra, con il valore reale e dunque si ottiene un valore iniziale di γ negativo.

Di seguito si riporta un'immagine che rappresenta come viene schematizzato l'encoder incrementale in Simulink così da comprenderne meglio il funzionamento.

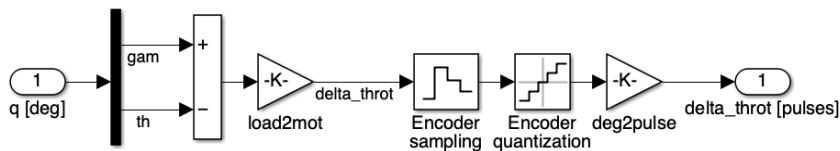


Figura 3.20: Schematizzazione encoder incrementale in Simulink

Un'altra evidente differenza, tra le risposte del sistema lineare (vedi figura 3.6) e quelle del sistema non lineare (vedi figura 3.18), è la presenza di una sovraelongazione molto maggiore soprattutto nei primi secondi di entrambe le risposte di γ del sistema non li-

neare, nonostante nel sistema sia presente lo schema di Anti-Wind Up. Questo fatto ha comportato che si è dovuto spostare in avanti l'istante di avvio del segnale di riferimento a gradino per l'inseguimento di spazio, per garantire la stabilizzazione del sistema prima dell'avvio di tale segnale.

La presenza di questa sovraelongazione nelle risposte del sistema non lineare è sicuramente generata dalle semplificazioni applicate al modello lineare; queste ultime rendevano la prima modellizzazione meno sensibile a tale fenomeno.

Al fine di provare a ridurre il fenomeno indesiderato della sovraelongazione su γ è necessario eseguire una nuova taratura di entrambi i controllori PID, il cambiamento più evidente sarà quello di rendere più "aggressivo" il controllo di γ , di conseguenza si dovrà cambiare anche la taratura del PID $_{\vartheta}$.

A seguito di svariati tentativi si è ottenuta una risposta abbastanza soddisfacente, anche se per ridurre la sovraelongazione su γ si è dovuto necessariamente variare in negativo la risposta di ϑ .

Di seguito si riporta la tabella con la taratura trovata; questa viene affiancata dai due grafici delle risposte del sistema e dal grafico del segnale di controllo u ; quest'ultimo indica inoltre quando il saturatore entra in funzione.

Taratura C			
PID $_{\vartheta}$		PID $_{\gamma}$	
K_P	-125.5	K_P	-9.86
K_I	-282.3	K_I	-15.86
K_D	-5.3	K_D	-0.9

Tabella 3.5: Taratura aggiuntiva dei PID

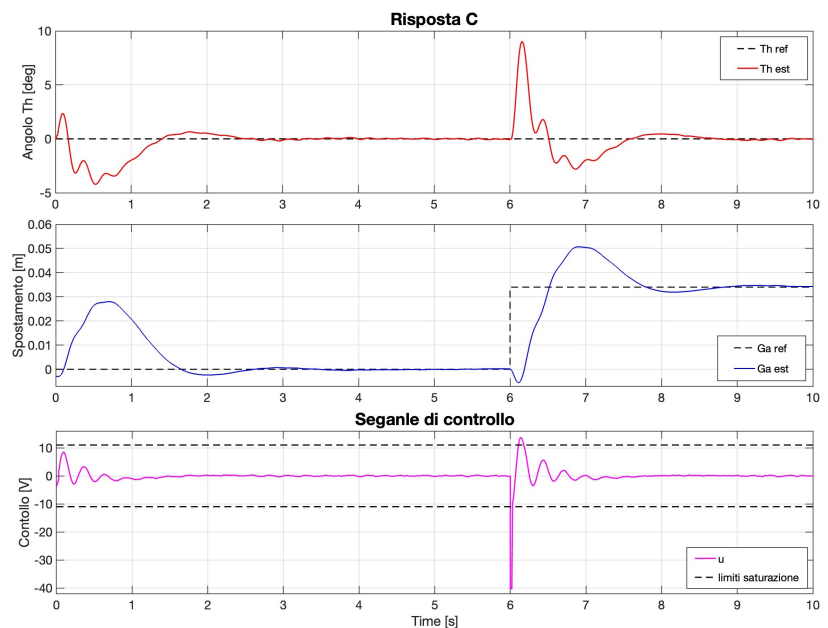


Figura 3.21: Risposte dell'angolo ϑ e dell'angolo γ , convertito in spostamento, rispetto al sistema non lineare e segnale di controllo u

Dai sopra riportati grafici delle risposte del sistema, si può concludere che la nuova taratura trovata migliora le risposte ottenute rispetto ai risultati ottenuti con le precedenti tarature (questo anche se non si arriva ad uguagliare i risultati ricavati dal sistema lineare).

3.4. ANALISI IN AMBIENTE SIMULATO DELLE RISPOSTE OTTENUTE DAL MODELLO COMPLETO

Si decide dunque di conservare la taratura C che nella prova del sistema reale andrà a rimpiazzare la taratura B, così da avere a disposizione per i successivi test, riportati nel prossimo capitolo, le tarature A e C sufficientemente diverse.

Capitolo 4

Verifica dei controllori sul sistema balancing robot reale

Dopo aver studiato la dinamica del robot per poter creare un modello, in ambiente simulato, nel modo più verosimile possibile al sistema reale ed aver successivamente svolto la progettazione dei controlli, è necessario concludere il lavoro con delle prove sul sistema reale dei controlli sintetizzati per verificare l'effettiva riuscita del progetto.

Il robot reale è basato su Arduino e grazie ad un "tool" Matlab, che riesce a convertire il codice Matlab-Simulink in codice Arduino, è possibile trasportare molto facilmente la progettazione dei controlli avvenuta in Simulink. La sintesi dei controllori PID, effettuata nel capitolo precedente, viene dunque convertita e spostata all'interno del robot solamente effettuando un collegamento, attraverso cavo, tra la scheda Arduino ed una porta USB del computer.

Il robot inoltre è anche in grado di inviare dati all'esterno, in tempo reale, tramite data logger e connessione bluetooth. Per merito di questa struttura è possibile quindi visualizzare e salvare i dati che rappresentano le grandezze misurate dai sensori in tempo reale, come se si stesse facendo una simulazione in Simulink.

Grazie alle specifiche del data logger ed alla "buona banda" consentita dalla connessione bluetooth, è possibile trasmettere e successivamente memorizzare sei segnali presenti all'interno della parte di controllo del robot. I segnali da salvare sono necessariamente:

- la stima dell'angolo ϑ [deg]
- la valutazione dell'angolo γ convertita in spostamento [m] (sul riferimento di spazio che le ruote dovranno inseguire);
- il controllo u [V] (non saturato);
- le due velocità angolari $\dot{\vartheta}$ [deg/s] e $\dot{\gamma}$ [deg/s] (in modo da riuscire ad ottenere una visione completa delle quattro variabili di stato del sistema reale).

Nelle pagine successive di questo capitolo si andranno a visualizzare ed analizzare le riposte

ottenute al variare delle tarature A e C sintetizzate nel capitolo 3.

4.1 Analisi taratura A applicata al sistema reale

Si traspone dunque nel sistema reale il controllo progettato con la taratura A dei PID riportata in tabella 3.3.

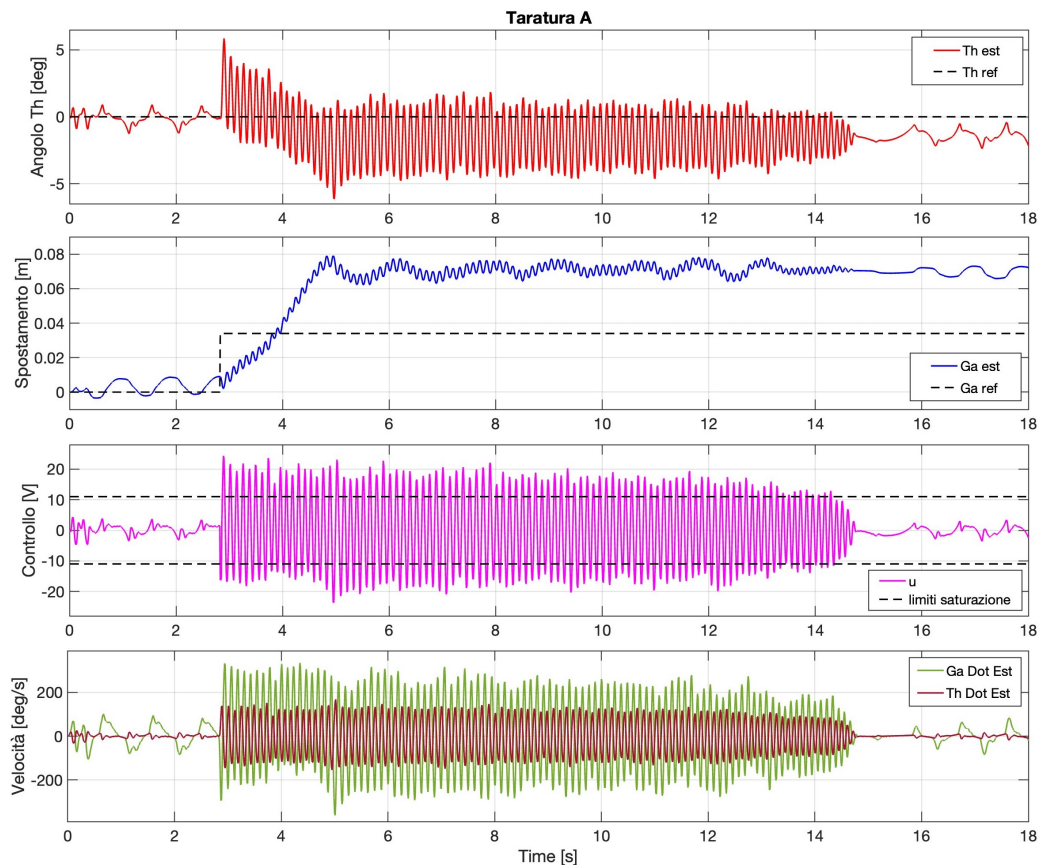


Figura 4.1: Risposte ottenute dal sistema reale con taratura A dei PID

Analizzando i primi secondi della simulazione dell'angolo ϑ si nota che il robot viene disposto dall'operatore esterno in una posizione pressoché verticale favorendo l'azione di controllo nella stabilizzazione che avviene con successo. Esaminando invece il comportamento di γ , sempre nei primi istanti, non si evidenziano fortunatamente sovraelongazioni eccessive come in ambiente simulato, ma si verificano delle oscillazioni in un intervallo il cui valore di un estremo è il valore nullo, dunque la specifica di controllo in questo caso non viene soddisfatta.

Sia dai movimenti del robot che si sono potuti osservare ad occhio nudo durante la simulazione come nelle risposte sopra riportate, si nota che il robot, dopo l'applicazione del segnale a gradino (per l'inseguimento di spazio da parte delle ruote), non riesce ad

inseguire tale segnale correttamente. Infatti il robot mantiene un errore a regime costante di qualche centimetro dalla posizione desiderata. La macchina, inoltre, non riesce nemmeno a stabilizzarsi rispetto alla posizione di equilibrio dell'angolo ϑ , continuando ad oscillare molto velocemente per un tempo di circa 15 secondi.

Dopo questo tempo l'oscillazione si smorza in modo evidente, ma non si svolge comunque attorno al valore nullo e l'errore a regime rispetto a γ persiste.

Come si vede dal grafico dove è riportato il segnale di controllo non saturato, durante i 15 secondi di oscillazione molto veloce, il controllo vorrebbe fornire tensioni ai motori abbastanza elevate (queste si aggirano intorno a circa il doppio del limite imposto dalla saturazione), al fine di riuscire a riportare il robot alla stabilità. È proprio in questi casi che l'azione saturativa è fondamentale per garantire l'integrità di tutti i componenti e poter quindi ripetere il test modificando i valori dei coefficienti di controllo.

Dall'analisi delle risposte si può concludere che la taratura A, nonostante in ambiente simulato potesse essere considerata valida, venga considerata in modo ugualmente positivo nel test reale; ciò perché non soddisfa nessuna specifica di controllo stabilità.

La causa di questi mal funzionamenti potrebbe essere attribuita:

- alla troppa "aggressività" dei termini del controllore PID_{ϑ} (che probabilmente causano la vibrazione del robot rispetto alla verticale in modo molto accentuato);
- alla scarsa "aggressività" del termine integrale all'interno del PID_{γ} (che comporta il mancato annullamento dell'errore rispetto a γ).

Si può quindi concludere che il problema sta nell'errato bilanciamento "dell'aggressività" dei due controlli.

4.2 Analisi taratura C applicata al sistema reale

A seguito del mal funzionamento del sistema di controllo rispetto al test del robot reale si sceglie di eseguire la medesima prova modificando la taratura dei controlli PID. Si passa dunque dalla taratura A alla taratura C visibile nella tabella 3.5 alla fine del capitolo precedente.

Dalla figura 4.2 (visibile nella pagina successiva), dove vengono riportate le risposte del sistema, si può subito notare la validità di questa risposta rispetto alla precedente.

Come si può vedere dal primo grafico dell'immagine, l'angolo ϑ pur non stabilizzandosi sul valore nullo, oscilla intorno ad esso in un intervallo di valori che va da circa meno un grado a circa più un grado.

Anche rispetto alla variabile γ si ottiene una risposta soddisfacente perché persino in questo caso il robot oscilla, in un intervallo sempre contenuto di valori, intorno alla posizione esatta non riuscendo a stabilizzarsi in modo completo. È possibile osservare il sostanziale

4.2. ANALISI TARATURA C APPLICATA AL SISTEMA REALE

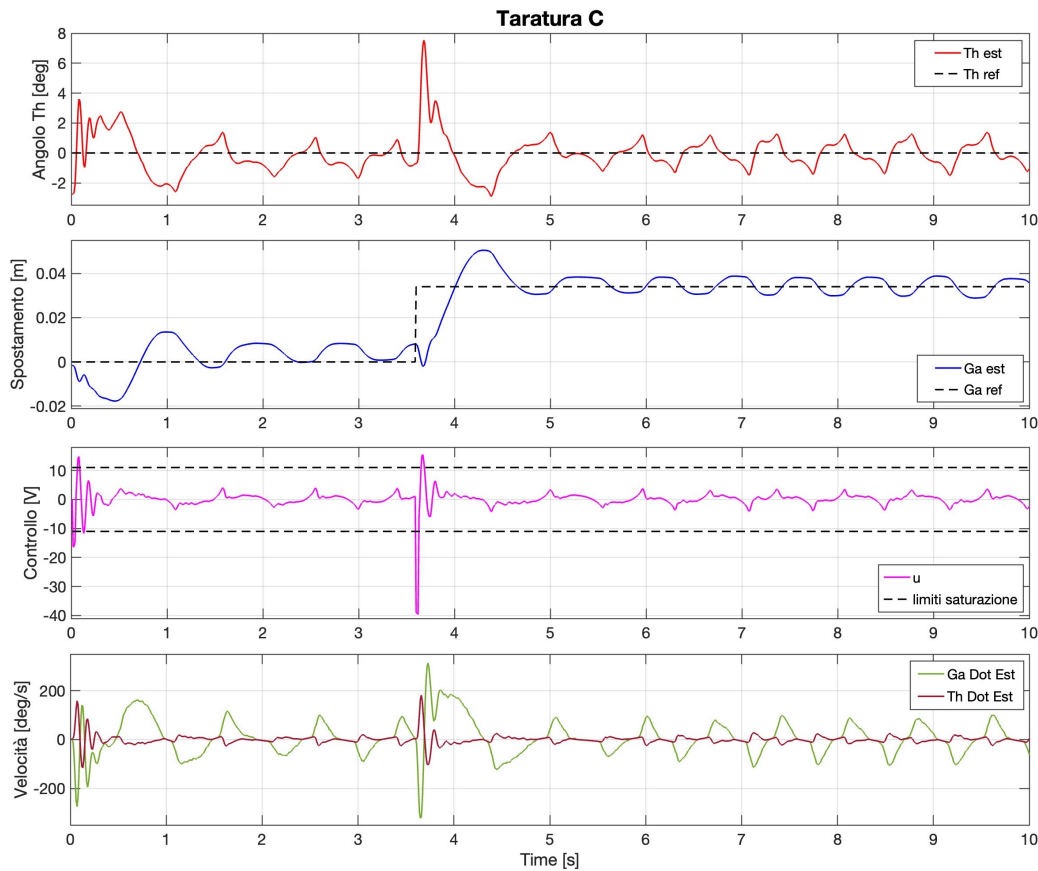


Figura 4.2: Risposte ottenute dal sistema reale con taratura C dei PID

cambiamento generato da questa taratura mettendo a confronto le due risposte dall'istante in cui viene applicato il segnale per l'inseguimento di spazio. Infatti, nella seguente prova, non si verifica più un'oscillazione incontrollata rispetto a ϑ , ma al contrario, il robot ha un'oscillazione più pronunciata nel momento dello spostamento per poi tornare a stabilizzarsi nell'oscillazione attorno al valore d'equilibrio come avveniva negli istanti precedenti.

Per quanto riguarda la variabile γ si ha un inseguimento abbastanza pronto del segnale a gradino, denunciando solo una lieve sovraelongazione che poi va ad annullarsi (il robot si ristabilizza poi nell'oscillazioni attorno al valore d'equilibrio). Non si verifica la persistenza di errore a regime.

Se si analizza invece la differenza tra i valori assunti dai due segnali di controllo rispetto alle due tarature, risulta evidente quanto sia più "aggressivo" ed efficace il segnale di controllo u della taratura C perché riesce attraverso due soli impulsi (che vengono comunque saturati), a riportare il robot nelle zone di equilibrio.

Dalle considerazioni riportate sopra è possibile valutare in modo decisamente positivo la taratura C, proprio per questo motivo si desidera mettere sotto stress il sistema. Ciò avverrà andando a verificare come il controllo reagisce ad un eventuale disturbo esterno,

che nel nostro caso abbiamo simulato con una lieve spinta del robot cercando di farlo cadere (portandolo fuori dalla posizione di equilibrio verticale).

Si riporta in figura 4.3 le risposte ottenute dal sistema successivamente alla prova sopra descritta.

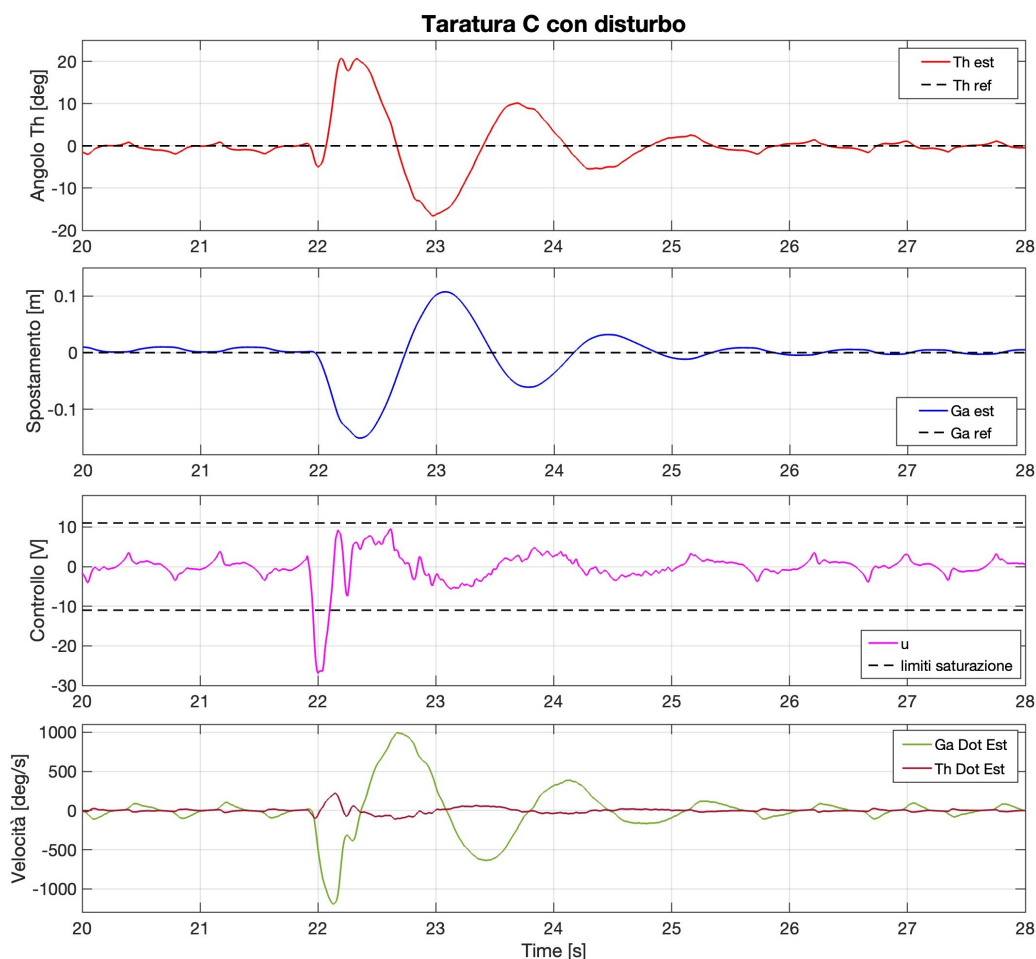


Figura 4.3: Risposte ottenute dal sistema reale con taratura C dei PID volutamente disturbata

Com'è possibile osservare dalla figura il test è andato a buon fine, il robot ha resistito alla spinta, ha oscillato sia sul piano verticale che sul piano orizzontare per qualche secondo, ma poi si è riportato nella situazione di equilibrio prefissata.

Si può quindi affermare che il sistema di controllo progettato è robusto e riesce a reiettare lievi spinte provenienti dall'esterno.

A seguito del successo della prova svolta precedentemente si decide di voler eseguire altri due ulteriori test sul sistema reale per verificare che il controllo progettato sia efficace anche adottando diversi segnali di inseguimento di spazio.

Il primo test si svolge applicando un segnale di spazio a rampa, prima ascendente poi discendente, questo implica il movimento del robot a velocità costante prima in un senso

4.2. ANALISI TARATURA C APPLICATA AL SISTEMA REALE

e poi in quello opposto.

Il secondo segnale che si vuole far insorgere dal robot è invece una sinusoide, per far muovere continuamente il robot avanti ed indietro a velocità differenti.

Entrambi i segnali comunque non faranno spostare il robot in modo brusco e veloce, perché le rampe avranno coefficienti angolari rispettivamente di 2 [m/s] e -2 [m/s], mentre la sinusoide avrà una frequenza di 0.1 [Hz] ed una ampiezza di 5 [rad].

Dalle risposte del sistema riportate in figura 4.4 e 4.5 (vedi pagina successiva) è possibile osservare che in entrambe le prove il robot segue il riferimento di spazio in modo abbastanza rigoroso e soprattutto senza cadere.

La cosa che emerge, se si osserva attentamente soprattutto la risposta con il segnale a rampa, è che quando il robot si muove in avanti, con velocità positiva facendo riferimento ai grafici, segue più rigorosamente il riferimento, rispetto a quando si muove nel senso opposto. Si nota infatti che viene introdotto un lieve ritardo, in entrambe le risposte, nel momento in cui il movimento del robot assume velocità negativa; questo non aiuta l'inseguimento del segnale che viene effettuato con una minor precisione.

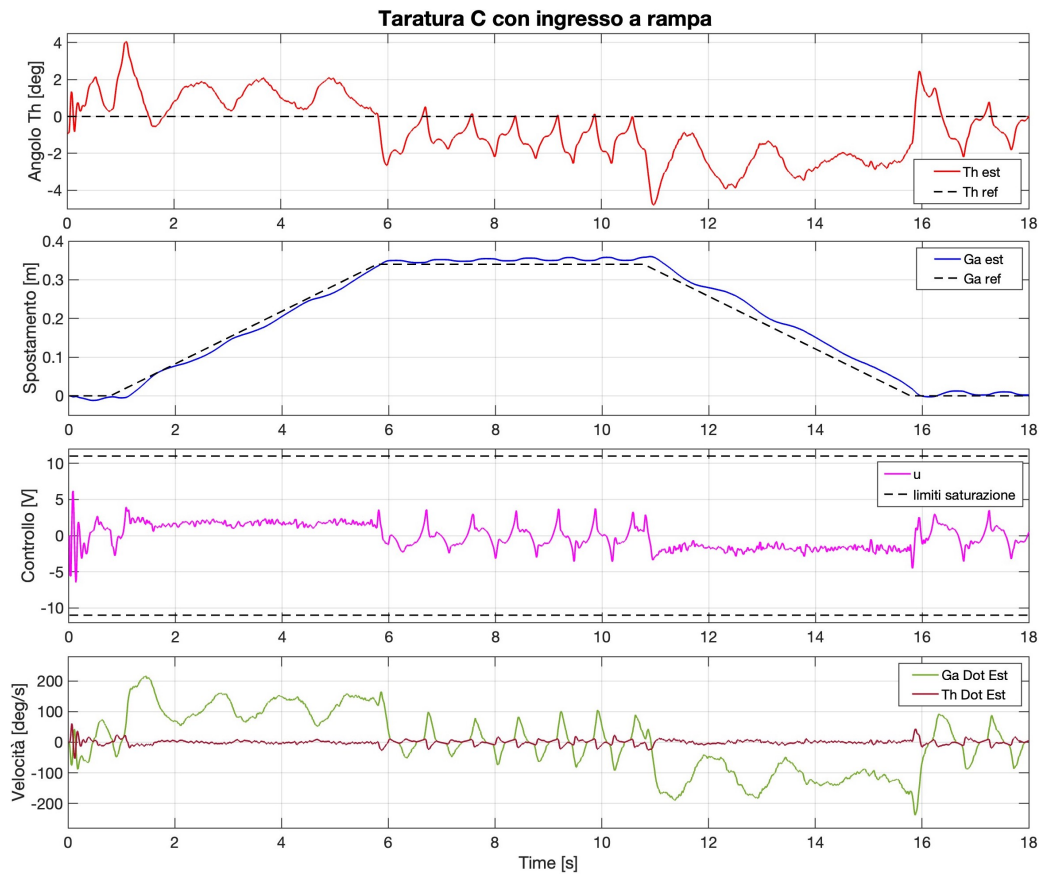


Figura 4.4: Risposte ottenute dal sistema reale con taratura C dei PID e segnale di inseguimento a rampa

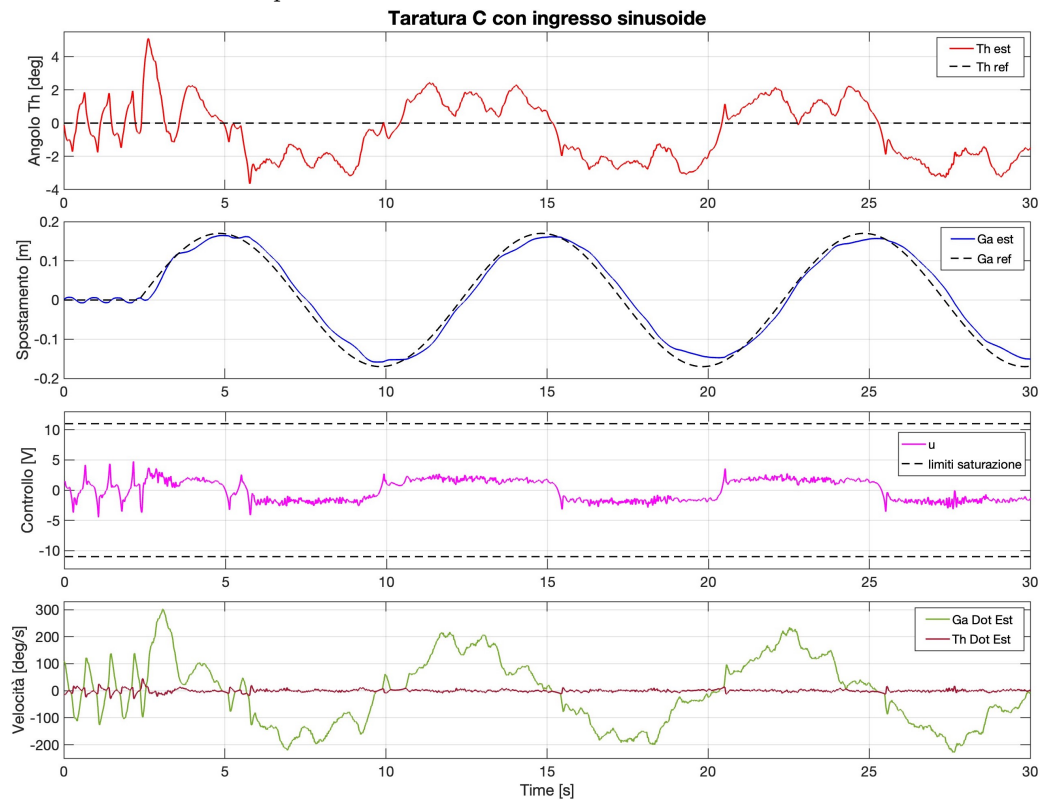


Figura 4.5: Risposte ottenute dal sistema reale con taratura C dei PID e segnale di inseguimento sinusoidale

Capitolo 5

Conclusioni

Dopo aver svolto la fase di test sul robot reale ed aver analizzato le risposte ottenute, si può valutare positivamente la progettazione del controllo.

Nonostante siano emerse delle differenze tra il robot reale ed il modello creato in ambiente simulato, come era plausibile, è stato comunque possibile ottenere un controllo che ottempera, anche se non in modo perfetto, le specifiche prefissate all'inizio della trattazione.

Il controllo progettato è in aggiunta, come esplorato nel capitolo precedente, robusto ad eventuali lievi disturbi esterni e funzionante anche per altri possibili segnali d'inseguimento di spazio.

Probabilmente si potrebbe perfezionare l'azione di controllo in modo da garantire una migliore stabilizzazione per la variabile γ . Infatti, eseguiti i test sul sistema reale, si è constatato che, oltre al diverso comportamento dei motori nei due sensi di rotazione, è presente anche una leggera rotazione del robot su se stesso. Quest'ultima rotazione avviene in senso antiorario rispetto all'asse longitudinale ed è visibile ad occhio nudo quando il robot avanza; la rotazione è probabilmente provocata dalla maggior spinta del motore destro. Nel caso in analisi viene quindi a mancare la prima ipotesi enunciata all'inizio del capitolo 2 e di conseguenza non si potrebbero più applicare le semplificazioni effettuate grazie a tale ipotesi. Per porre rimedio alla presenza di questo mismatch in ambiente sperimentale, sarebbe quindi necessario progettare un ulteriore controllore PID agente sulla differenza relativa tra le due posizioni dei motori.

Bibliografia

- [1] Ahmed Fahem Albaghdadi. «An Optimized Complementary Filter For An Inertial Measurement Unit Contain MPU6050 Sensor». In: *Iraqi Journal of Electrical and Electronic Engineering* 15, No. 2 (2019). URL: <https://www.iasj.net/iasj/download/f794c16f8da73924>.
- [2] Guido Gentile. «Meccanica lagrangiana». In: *Introduzione ai sistemi dinamici - Volume 2: Meccanica lagrangiana e hamiltoniana*. Milano: Springer Milan, 2022, pp. 1–70. ISBN: 978-88-470-4014-4. DOI: 10.1007/978-88-470-4014-4_1. URL: https://doi.org/10.1007/978-88-470-4014-4_1.
- [3] Wang Jia-Jun. «Position and speed tracking control of inverted pendulum based on double PID controllers». In: *2015 34th Chinese Control Conference (CCC)*. 2015, pp. 4197–4201. DOI: 10.1109/ChiCC.2015.7260286.
- [4] Matteo Dal Nevo. «Controllo classico di un pendolo inverso». Università di Padova, 2021. URL: <https://hdl.handle.net/20.500.12608/31883>.

