



**UNIVERSITÀ DEGLI STUDI DI ROMA
TOR VERGATA**

FACOLTÀ DI INGEGNERIA

**CORSO DI LAUREA MAGISTRALE IN INGEGNERIA
DELL'AUTOMAZIONE**

A.A. 2019/2020

Tesi di Laurea Magistrale

**REALIZZAZIONE DI UN ROBOT OMNIDIREZIONALE
DOTATO DI SENSORI A ULTRASUONI E SUA
LOCALIZZAZIONE IN AMBIENTI INDOOR**

RELATORE

Prof. Francesco Martinelli

CANDIDATO

Caterina Grimaldi

*A tutte le persone che mi sono state vicine nela strada percorsa,
e a quelle che divideranno con me questa salita ripida
quanto meravigliosa avventura chiamata vita.*

Ed in ultimo, alla mia capacità di fare "Scaccola tre!"

Indice

Ringraziamenti	1
1 Introduzione e obiettivo del progetto	2
2 Descrizione del robot e prototipo	6
2.1 Caratteristiche del robot	9
2.2 Ruote mecanum	9
2.3 Design meccanico ed elettronico	12
2.3.1 Progettazione <i>chassis</i>	12
2.3.2 Schema elettrico	13
2.4 Componenti elettroniche	16
2.4.1 Arduino	16
2.4.2 Batteria	17
2.4.3 Sensori	18
2.4.4 Ponte H	23
2.4.5 DC converter	24
2.5 Componenti meccaniche	25
2.5.1 Motori	25
3 Cinematica omnidirezionale	28

3.1	Cinematica	29
3.2	Movimento omnidirezionale	35
4	Localizzazione	38
4.1	<i>Dead Reckoning</i> indoor	39
4.2	Filtro di Kalman esteso	40
5	Risultati	45
5.1	Risultati simulazione	46
5.1.1	Risultati del sistema	47
5.1.2	Risultati con diverse condizioni iniziali	49
5.1.3	Risultati al variare della matrice di covarianza	52
5.1.4	Risultati al variare del raggio di una ruota	55
5.2	Risultati sperimentali	58
5.2.1	Risultati sperimentali del sistema	59
5.2.2	Risultati con diverse condizioni iniziali	62
5.2.3	Risultati al variare della matrice di covarianza	64
6	Conclusioni e sviluppi futuri	66
	Elenco delle figure	69
	Bibliografia	71

Ringraziamenti

È per me doveroso dedicare questo spazio del mio elaborato alle persone che hanno contribuito, con il loro instancabile supporto, alla realizzazione dello stesso.

In primis, un ringraziamento speciale al mio relatore, il prof Martinelli, per la sua immensa e smisurata pazienza, per i suoi indispensabili consigli, per l'umano, quanto raro, supporto nelle difficoltà e per la passione che mi ha trasmesso fin dalla mia prima lezione: è stato davvero un onore, per me, poter imparare e collaborare.

Ringrazio i miei genitori che mi hanno sempre sostenuto, appoggiando ogni mia decisione, e insegnato la bellezza e l'amore per lo studio.

Ringrazio chi mi ha dato il coraggio che cercavo e la spinta giusta per non arrendermi mai e andare insieme all'arrembaggio di questo giorno.

Ringrazio il Roma2LUG, la mia casa appena iniziata l'università, che mi ha permesso di imparare molto ma soprattutto di intraprendere un percorso nuovo con compagni di viaggio davvero speciali, nonchè amici.

E sempre restando nella sfera degli amici, vorrei ringraziarli tutti per essere parte della mia vita personale, universitaria e scoutistica.

Capitolo 1

Introduzione e obiettivo del progetto

I robot mobili sono in grado di muoversi autonomamente e di svolgere i compiti assegnati. Oltre alla mobilità, caratteristica principale, gioca un ruolo importante l'autonomia, ovvero l'indipendenza del robot dall'intervento umano; l'autonomia energetica è definita prevalentemente in funzione dell'alimentazione elettrica del robot.

Nonostante la robotica mobile sia in uno stadio di sviluppo precoce, prototipi e primi prodotti in serie hanno iniziato a diffondersi da tempo nei diversi settori; ad esempio, con l'aspirapolvere mobile e quasi completamente autonomo, la robotica mobile è entrata di diritto a prendere parte alla vita domestica.

Le innovazioni avute nello sviluppo dell'intelligenza artificiale e la loro interconnessione con la robotica porterà in futuro ad aumentare la rilevanza e l'impiego dei robot mobili autonomi.

La robotica mobile, rispetto alla robotica classica, rappresenta l'evoluzione dei sistemi esistenti, la trasformazione è soprattutto data dalla rilevanza e l'impiego dei sensori in molti campi applicativi.

I robot classici si basano su un modello predeterminato, che li rende non autosufficienti; l'utilità sensoristica viene a mancare, in quanto si assiste ad un basso grado di incertezza sopportabile.

La robotica mobile, invece, si affida totalmente alla moltitudine di sensori, come una telecamera, un sistema con scanner laser, ultrasuoni e altro: i sensori diventano fondamentali perchè in grado di reagire autonomamente agli eventi improvvisi.

Da un punto di vista di programmazione, sussiste una differenza sostanziale: la robotica mobile è malleabile nonostante sia basata su regole predefinite e sia orientata al compito, mentre i robot industriali classici hanno un software dotato di una catena di comandi che, in quanto non adattabili al contesto, le conferiscono rigidità.

La possibilità di avere un software in grado di adattarsi a infiniti scenari e la fusione con dei sensori rappresentano l'inizio di nuove applicazioni, tali da consentire agli apparecchi mobili di pianificare ed eseguire i loro movimenti in autonomia.

Generalmente, i robot classici ripetono sempre la stessa operazione mentre l'andamento dei movimenti di un robot mobile dipende dai cambiamenti dell'ambiente.

I robot industriali tradizionali sono dotati di una postazione fissa e per questo motivo lavorano in uno spazio conosciuto, con poche variazioni.

La robotica mobile, invece, ha un uso più versatile; utilizzando la propria percezione i robot autonomi sono in grado di esplorare autonomamente nuovi ambienti.

Quindi, riassumendo, queste sono le caratteristiche della robotica mobile:

1. percezione grazie ai sensori;
2. capacità di adattarsi ad ogni situazione ambientale;
3. navigazione, pianificazione e azione autonoma.

Alla luce di quanto detto fino a questo punto, è doveroso spiegare la novità di questa tesi, e cioè quella di unire le caratteristiche elencate di un robot mobile all'uso di ruote omnidirezionali e all'implementazione del filtro di Kalman, aprendo numerosi scenari applicativi.

I vantaggi delle ruote mecanum consistono nella maggiore manovrabilità, nella capacità di assumere omnidirezionalità o mobilità locale - rendendo possibile il raggiungimento di tre gradi di libertà nello spazio - nello scarso attrito sul suolo - tale da non provocare danni alla superficie su cui opera -, nell'adozione di una coppia di motrice inferiore che richiede un minor dispendio energetico e l'utilizzo di una potenza inferiore; all'opposto, gli svantaggi delle ruote mecanum sono limitati alla complessità della realizzazione delle ruote e alla possibilità di un più ampio margine di errori a causa di una maggiore sensibilità alle asperità del terreno.

Sono già presenti allo stato dell'arte alcuni prototipi appartenenti ad ambiti più disparati. Possiamo notare come, in ambito medico, si siano sviluppati prototipi dotati di ruote omnidirezionali[2] in grado di semplificare e rendere ancora più accessibile un luogo a persone invalide; in ambito spaziale, favoriscono l'esplorazione di pianeti ignoti tramite un robot autonomo[3] e, ancora, in campo commerciale e industriale semplifica il lavoro all'interno di un magazzino tramite un elevatore più agile[4].

Nel secondo capitolo saranno descritte le componenti utilizzate nella costruzione del prototipo e chiarite le scelte progettuali fatte, nel terzo sarà esposta la cinematica del robot dotato di ruote omnidirezionali con un riepilogo del verso di rotazione per ogni ruota in base ai movimenti da effettuare, successivamente nel capitolo 4, verrà presentata la localizzazione tramite *dead reckoning* e filtro di Kalman per correggere gli errori del sistema.

Infine nel capitolo 5, sono riportati i risultati avuti in simulazione e quelli sperimentali in modo da poter discutere quanto visto e poter individuare sviluppi futuri del prodotto.

Capitolo 2

Descrizione del robot e prototipo

In questo capitolo verrà illustrato come realizzare un robot dotato di ruote omnidirezionali mecanum, dai sensori alla scheda di controllo, dalle ruote alla struttura; saranno presentate le decisioni progettuali intraprese, dalla batteria ai motori.

Nella realizzazione del prototipo ci sono state tre fasi principali.

Si è proceduto iniziando dalla scelta della dimensione delle ruote e del numero di rulli, in grado di garantire delle specifiche caratteristiche che saranno esplicate successivamente e anche nella scelta dei motoriduttori e dei motori.

Nella seconda fase sono stati scelti i sensori di distanza e gli encoder; quest'ultimi, infatti, erano direttamente dipendenti dalla prima fase di scelta delle ruote e dei motori.

Le scelte fatte finora ci portano all'ultima fase, la fase tre.

Per poter pilotare i motori DC, si è dovuto utilizzare il ponte H, perchè in grado di comunicare con Arduino. Quest'ultimo infatti, è stato scelto per la semplicità di uso e perchè è fra le schede che meglio comunicano con una vasta gamma di sensori.

Scegliendo la batteria, creata appositamente in base alle specifiche, si è dovuto aggiungere un componente responsabile dell'alimentazione del microcontrollore e cioè la scheda per la corrente continua in grado di abbassare la tensione.

Nel diagramma che segue sono riassunte le tre fasi progettuali per la scelta dei componenti.

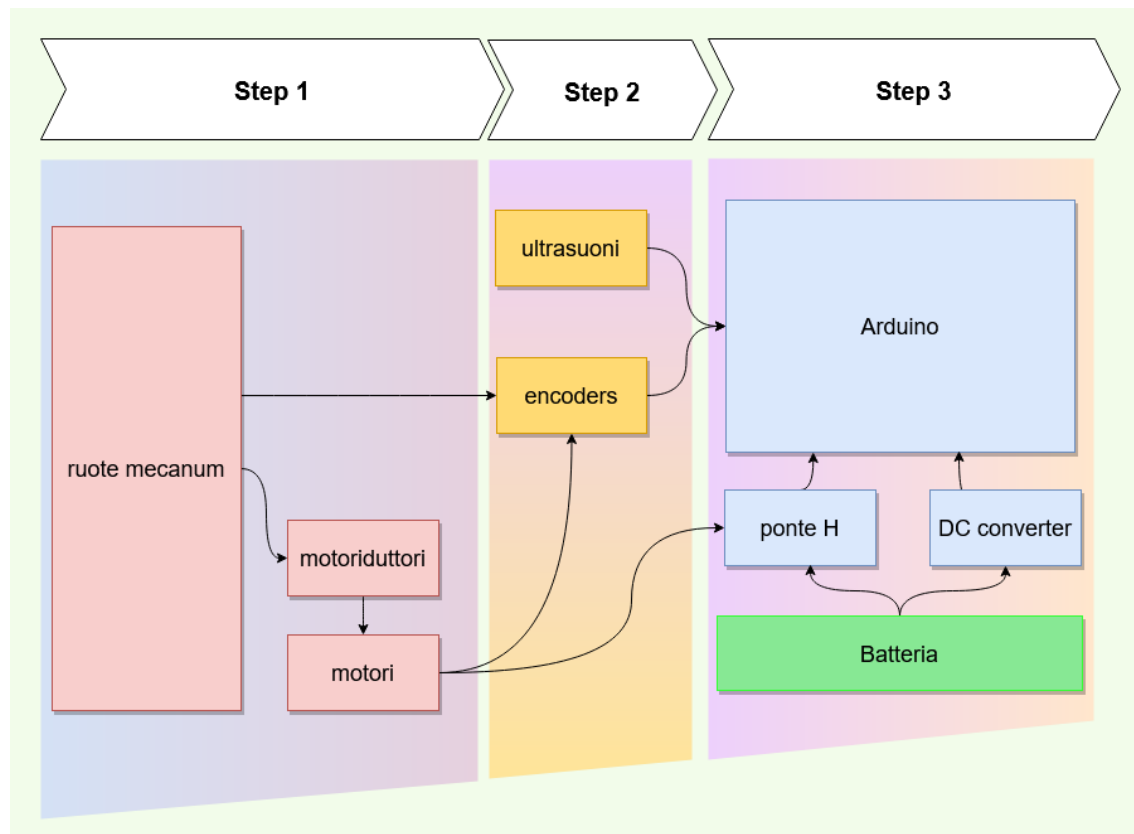


Figura 2.1: procedimento di realizzazione *bottom up*.

In figura 2.2 e 2.3 sono presentate le diverse viste del prototipo finale.



Figura 2.2: vista del prototipo finale poggiato su un piano.

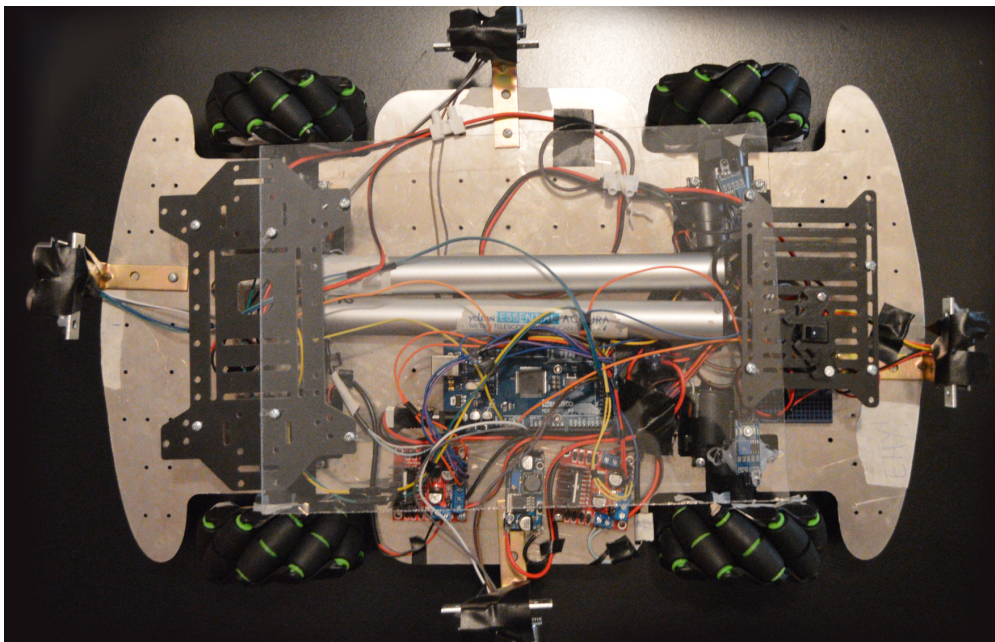


Figura 2.3: vista dall'alto del prototipo.

2.1 Caratteristiche del robot

Il robot è dotato di ruote *mecanum* che permettono lo spostamento in tutte le direzioni senza dover sterzare o senza che avvenga una rotazione su se stesso.

Infatti, la sua abilità è tale da essere impiegato in spazi ristretti e la sua velocità è un plus.

Avendo inoltre una struttura in lega di alluminio, è in grado di sopportare molto peso in rapporto alle piccole dimensioni: le sue applicazioni sono difatti infinite.

Ad esempio, è facile pensare di impiegarlo nel trasporto di un carico in un breve percorso all'interno di un magazzino o un ambiente indoor; rappresenta la base mobile per un qualsiasi braccio robotico in quanto dotato di sensori di distanza tali che il trasporto o il movimento in generale siano privi di pericoli; inoltre, dato il costo molto contenuto, un sistema di controllo semplice come Arduino, può essere realizzato facilmente.

È possibile programmare il suo spostamento e tramite il filtro di Kalman, essere certi di poter compensare errori dovuti, ad esempio, ad una pavimentazione non perfettamente liscia o ad una errata lettura dei sensori di distanza.

2.2 Ruote mecanum

Il design di un classico progetto di un robot mobile comprende l'uso di due tipologie di ruote: le ruote basculanti e le ruote sterzanti; esse hanno un carico maggiore e una notevole tolleranza per quanto riguarda le irregolarità del piano di appoggio rispetto a delle configurazioni particolari; tuttavia, a causa della loro natura non olonomica, non sono veramente ruote omnidirezionali: impiegano tempo per far muovere il robot

secondo una determinata direzione o un punto da raggiungere.

Le ruote omnidirezionali sono in grado di trasferire il movimento senza che la direzione della velocità della base rispetto alla ruota, nel punto di contatto con il suolo, corrisponda necessariamente alla velocità tangenziale della ruota misurata nello stesso punto. Sono in grado di muoversi in una direzione diversa da quella direzione imposta dal rotolamento della ruota, senza alcuno strisciamento relativo con la base, ma sfruttando il rotolamento dei rullini presenti sulla ruota: se ruota in un certo verso, avverrà un movimento passivo dei rulli nel verso opposto.

La configurazione di una ruota *mecanum*, con l'asse dei rulli posti sulla circonferenza a 45 gradi rispetto all'asse di rotazione dell'intera ruota, trasmette una parte della forza nella direzione di rotazione ad una direzione normale alla direzione dell'intera ruota. La configurazione della piattaforma è composta da quattro ruote situate in modo simile a quella di un'automobile.

Le forze dovute alla direzione e alla velocità di ciascuna delle quattro ruote possono essere sommate in un vettore di forza totale che consente la traduzione del veicolo in qualsiasi direzione. A seconda della direzione e della velocità delle singole ruote, è consentito al robot di muoversi liberamente nella direzione del vettore di forza risultante, senza cambiare l'orientamento delle ruote.

I punti di forza delle ruote *mecanum* - anche dette svedesi - sono nel design compatto e nell'alta capacità di carico. Di contro, se il suolo è irregolare, ne risentirà il moto del robot.

Nel caso delle ruote *mecanum* in uso nel prototipo costruito, in tabella sono riportate

le specifiche dimensionali e il carico totale dell'intero robot, tenuto conto che il materiale con il quale sono realizzate non permette di sfruttare al massimo il punto di forza rappresentato da un alto carico.

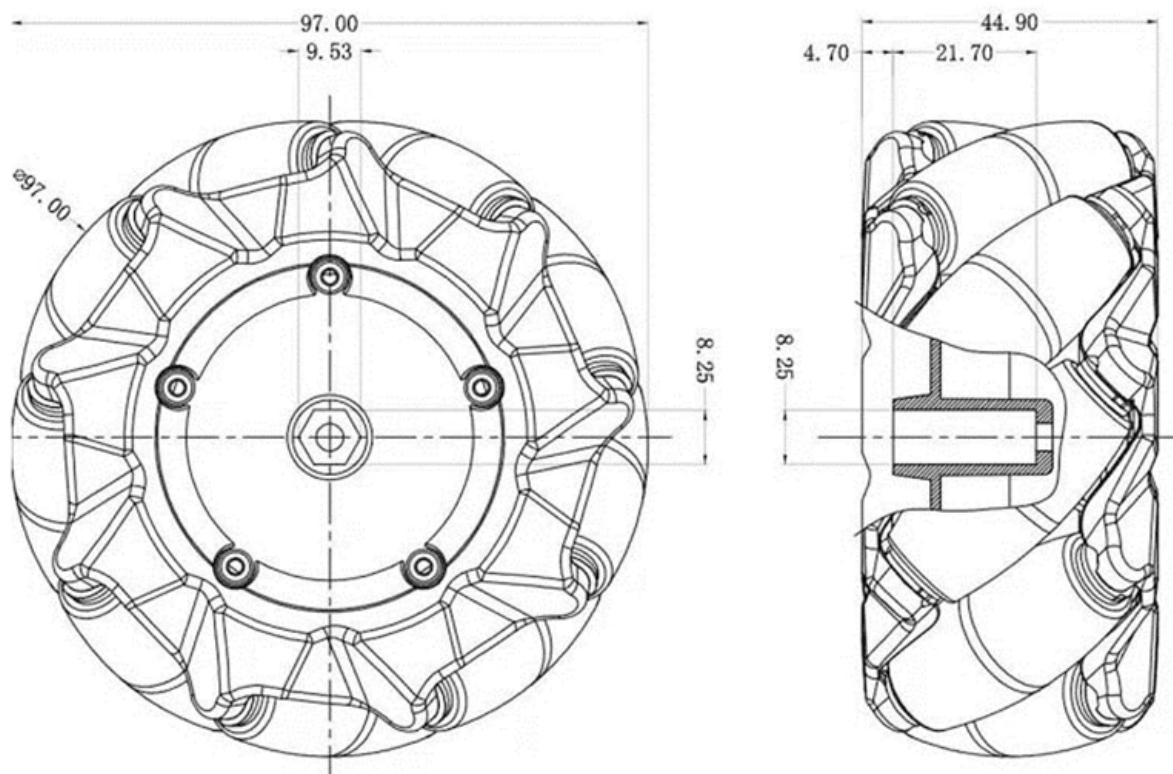


Figura 2.4: specifiche di una ruota mecanum.

dimensione interna	dimensione esterna	spessore	numero rulli	carico totale
8.25 mm	97 mm	44.9 mm	9	15 kg

Tuttavia, per un robot di piccole dimensioni, la capacità di carico totale pari a 15 kg è allo stesso modo un ottimo risultato.

In ultimo, viene affrontata in questa sezione anche la scelta sul numero di ruote da utilizzare.

In particolare, poichè l'intento è quello di costruire un robot dotato di ruote *mecanum*, appare evidente che configurazioni a due o tre ruote non porterebbero a risultati differenti dall'uso di ruote semplici; per questo motivo si è scelto di procedere con una configurazione a quattro ruote disposte come negli autoveicoli per semplicità di calcolo; configurazioni derivanti da quest'ultima ma con un numero maggiore di ruote hanno un funzionamento simile che genera solo complessità di calcolo.

In questo caso, non avendo necessità specifiche richieste dal carico o dalle dimensioni, si è optato per una configurazione a quattro ruote *mecanum*.

2.3 Design meccanico ed elettronico

In questa sezione presenteremo tutte le fasi di progettazione del robot da un punto di vista elettronico e meccanico, senza dimenticare la scelta dei componenti.

In particolare, vedremo le specifiche relative alle schede elettroniche, il "cervello" del progetto, i sensori e l'alimentazione; vedremo inoltre su quali basi poter progettare la parte meccanica e scegliere i motori.

2.3.1 Progettazione *chassis*

La specifica richiesta da soddisfare per la progettazione dello *chassis* riguarda la possibilità di poter adagiare le diverse schede in modo intercambiabile, cioè con dei fori distribuiti sull'intera superficie con passo di 2,50 mm. La larghezza e la profondità sono state scelte in base alla dimensione delle ruote: sia l'interasse che il passo sono infatti proporzionali al raggio delle ruote.

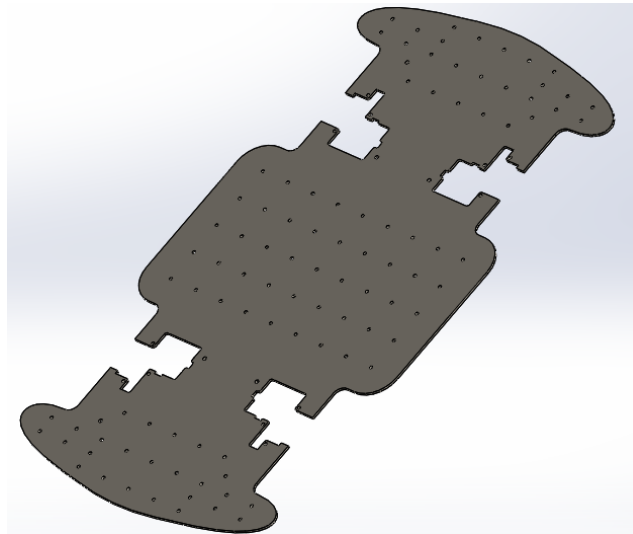


Figura 2.5: progettazione dello *chassis* in lega di alluminio.

Il materiale, l'alluminio, è stato scelto in quanto non necessita di vernici, un passaggio ulteriore non indispensabile durante la fase di prototipazione; inoltre è leggero e lavorabile e quindi risulta uno dei materiali che più si prestano a modifiche.

Lo *chassis* è stato realizzato con uno spessore pari a 2 mm: tale spessore non permette una perfetta rigidità della struttura; per questo motivo si è scelto ulteriormente di aggiungere dei listelli di collegamento con le zone più esterne.

Utilizzando dei distanziatori, dei rettangoli in fibra di vetro e del plexiglass per ricoprire l'intero robot lasciando intravedere l'interno, la struttura non flette ed è rigida.

2.3.2 Schema elettrico

Inizialmente si provvede a collegare i motori al ponte H: facendo attenzione alla polarità in modo da non invertire la rotazione dell'albero motore, si procede al cablaggio dei pin responsabili del controllo delle ruote, sia per la direzione che per l'intensità di

rotazione.

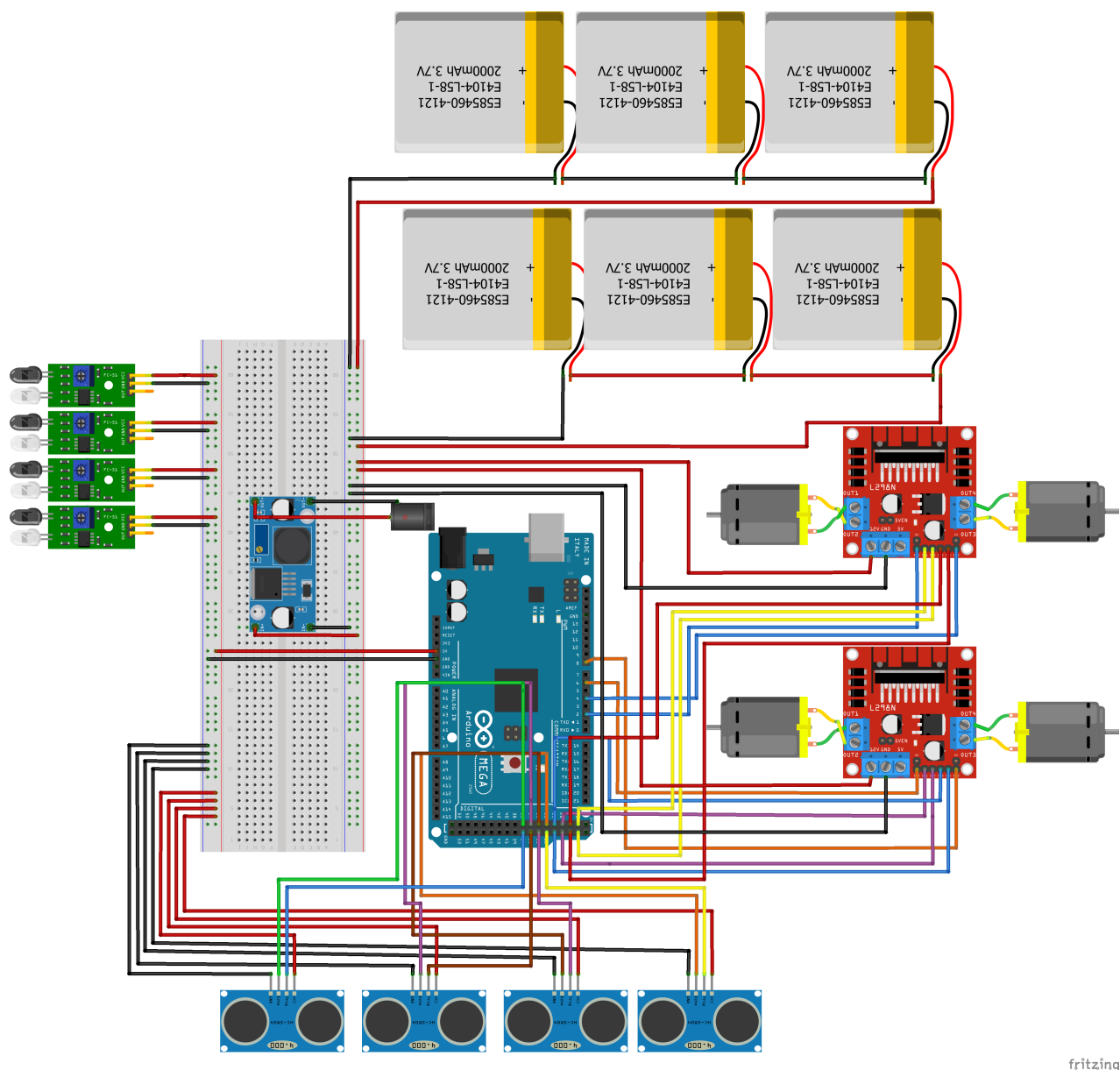
I pin di abilitazione dei motori saranno collegati al microcontrollore nei pin PWM; i pin di ingresso sul ponte H saranno diretti nei pin digitali della scheda.

A questo punto è possibile procedere con i sensori; per i sensori ad ultrasuoni, i pin esterni responsabili dell'alimentazione andranno nella breadboard, mentre i due pin centrali, TRIG ed ECHO, saranno collegati ad Arduino, in modo da poter leggere quanto misurato.

Gli encoder avranno solo due collegamenti per sensore, ossia l'alimentazione tramite la scheda forata.

Per quanto riguarda il convertitore della corrente DC/DC, le uscite saranno collegate al jack di potenza, facendo in modo che il positivo sia centrale.

Preso la batteria, possiamo collegare il polo positivo e negativo ai due ponte H nel pin di entrata a 12 V e al GND, e al convertitore DC/DC.



fritzing

Figura 2.6: schema elettrico di collegamento.

2.4 Componenti elettroniche

In questo paragrafo saranno presentati i componenti in uso del progetto; dalla scheda di controllo dei motori al "cervello" del robot, dalla batteria ai sensori.

2.4.1 Arduino

Arduino è la più famosa piattaforma *open source* di prototipazione hardware e software nel campo dei microcontrollori ¹.

Al suo interno sono presenti diverse strutture hardware per la comunicazione analogica e digitale tramite i molteplici pin di ingresso/uscita e per il controllo della tensione.

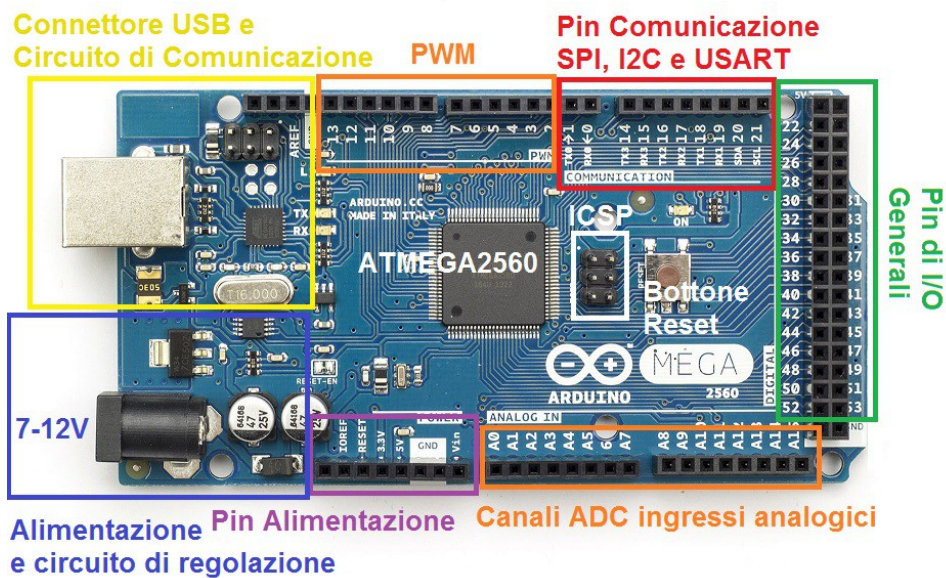


Figura 2.7: microcontrollore Arduino ATmega 2560.

La scheda utilizzata nel progetto è una ATmega 2560 in quanto ha poco più di cinquanta pin digitali e una decina di PWM: entrambe le caratteristiche sono indispensabili per la numerosità di ingressi di sensori, motori e schede di controllo.

¹Si tratta di un dispositivo elettronico integrato su singolo circuito elettronico, evoluzione diretta del microprocessore.

2.4.2 Batteria

Visto l'assorbimento dei motori e la tensione a circa 7 V, si è scelta una batteria che potesse durare a lungo e che fosse da 12 V. In particolare, sono stati creati due stick ricaricabili da 12 V e 8800 mAh .



Figura 2.8: batteria GTF 26650 li.ion .

Collegando i due stick, all'interno in serie e fra i due in parallelo, sono stati creati due alloggiamenti in alluminio.

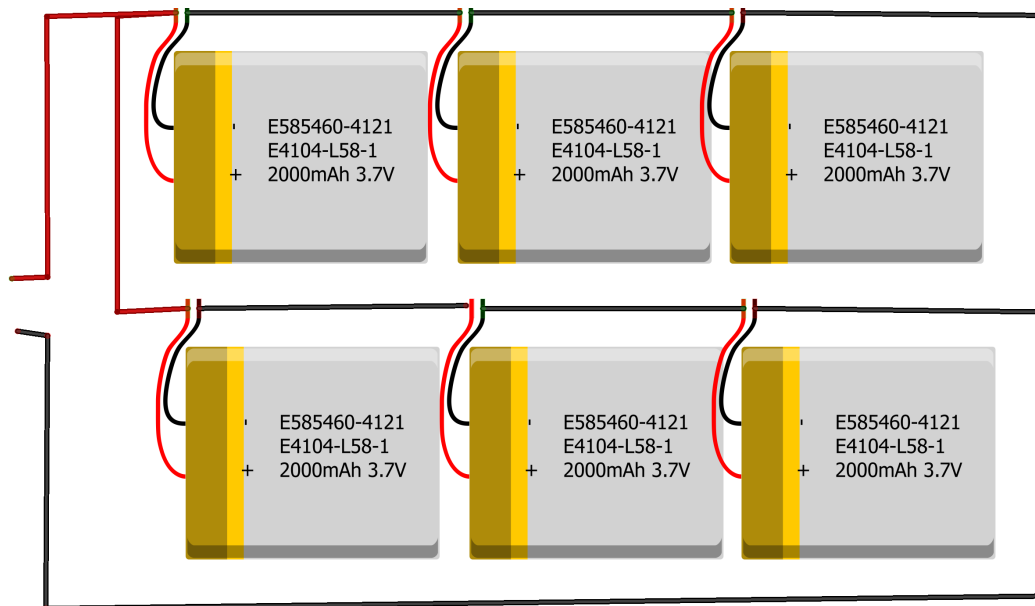


Figura 2.9: collegamento batteria.

2.4.3 Sensori

In questa sezione verranno presentate le due tipologie di sensori presenti sul robot: i sensori ad infrarossi ed i sensori a ultrasuoni.

La prima tipologia, applicata per la lettura dello spazio percorso dal robot in base al numero di giri delle ruote, è utilizzata come misura interna del sistema per poter calcolare la posizione in ogni istante di tempo tramite la tecnica dell'odometria.

La seconda tipologia, applicata per calcolare la distanza da ostacoli presenti sul percorso, viene utilizzata come misura esterna in modo da aumentare la precisione della stima della posa del robot ed evitare che il moto del robot non si intersechi con ostacoli di vario genere.

Encoder

La misura della velocità di una ruota è affidata ad un dispositivo chiamato *encoder*.

Esistono diverse tipologie raggruppabili in due macroaree:

- *Encoder assoluti*: rilevano la posizione assoluta della ruota, conoscendo in ogni istante la posizione. Sono indicati per applicazioni di alta precisione e per i casi in cui viene richiesta una conoscenza assoluta della posizione; per questo motivo sono sensori costosi.
- *Encoder incrementali*: sono in grado di calcolare la variazione di spostamento senza conoscere la posizione iniziale. Questa tipologia di encoder è semplice da realizzare in un progetto; inoltre è particolarmente indicato per il calcolo della velocità.

È possibile inoltre differenziare in base alla tecnologia (*encoder* ottici, capacitivi ...).

In questo caso utilizzeremo gli *encoder* ottici, costituiti da due parti: un emettitore e un ricevitore.



Figura 2.10: sensore di prossimità ad infrarossi FC-51.

L'emettitore utilizza una luce infrarossi che, riflettendo su una superficie che può essere bianca o nera in modo da assorbire o riflettere la luce, viene catturata dal ricevitore. Il sensore in uso FC-51 è dotato di un potenziometro in grado di far variare il range d'azione: in questo caso sarà settato ad una distanza di circa 2 cm e la sua accuratezza nella misura è pari a ± 1 %.

Caratteristica	Voltaggio	Corrente	Frequenza	Distanza min/max	Angolo
Valore	5 V	140mA	100 Hz	[0.02; 0.3] m	35 gradi

Per creare questo sistema di lettura dei tic, ossia del numero di settori in grado di riflettere la luce posti equidistanti su ogni ruota, è stato utilizzato un sensore IR.

```

void INTERRUPT_0()
{
    rev[0]++;
}
void INTERRUPT_1()
{
    rev[1]++;
}
void INTERRUPT_2()
{
    rev[2]++;
}
void INTERRUPT_3()
{
    rev[3]++;
}

```

Dopo aver utilizzato gli interrupt per avere il numero di tic letti dal sensore istantaneamente, ² è possibile definire la distanza percorsa in base al tempo impiegato dalla luce emessa a riflettere sulla ruota e ad essere letta dal ricevitore.

²L'interrupt è un segnale asincrono che indica la necessità di "attenzione" da parte di una periferica collegata ad Arduino.

Nel codice che segue viene eseguito questo processo per ogni ruota del robot.

```
void tachometer(){
  delay(600);
  for (int z = 0; z < 4; z++)
  {
    tempo[z]=millis()-previous[z];
    rpm_value[z]=(rev[z]/tempo[z])*60000;
    previous[z] = millis();
    passo = 2*PI*rpm_value[z]*radius*tempo[z]/600;
    Serial.println(passo);
    rev[z]=0;
  }
}
```

Sul mercato sono disponibili altre tipologie di *encoder* con una migliore risoluzione, come quelli a forcella ³; questi ultimi però non sono stati presi in considerazione in quanto non applicabili alla tipologia di motore in uso.

Ultrasuoni

Il sensore ad ultrasuoni fornisce un valore, in microsecondi, del tempo di volo di un'onda sonora che compie un viaggio di andata e ritorno dopo essere stata riflessa da un ostacolo.



Figura 2.11: sensore ultrasuoni HC-SR04.

³L'encoder a forcella, normalmente utilizzato come fine corsa, ha la caratteristica di avere l'emettitore LED posizionato alla stessa altezza del transistor in modo da non avere una riflessione della luce, ma solo un ON/OFF del segnale.

Viene spesso utilizzato nell'industria e nelle applicazioni relative alla domotica, sfruttando gli ultrasuoni per calcolare la distanza del robot rispetto all'ambiente.

Il sensore ha due trasduttori: una capsula in grado di trasmettere l'onda e una capsula in grado di catturarla.

Il sensore, collegato alla scheda Arduino, tramite la misura dell'intervallo di tempo impiegato nel percorrere lo spazio fra robot e ostacolo, compreso il ritorno, emette l'impulso sonoro e setta il segnale sul pin della scheda Arduino come HIGH; non appena la capsula ricevente avrà l'onda di ritorno, setterà il segnale come LOW. Il tempo intercorso fra i fronti di salita e di discesa è detto tempo di volo dell'onda.

La distanza viene ricavata secondo

$$spazio = \frac{v \cdot \Delta t}{2} \quad (2.4.1)$$

con v la velocità del suono nell'aria e Δt il periodo di tempo.

Sono riportate in tabella le caratteristiche del sensore in fatto di tempistiche di segnale, accuratezza, distanza e dimensioni.

Caratteristica	Voltaggio	Corrente	Frequenza	Distanza min/max	Angolo
Valore	5 V	15mA	40 Hz	[0.02; 4] m	15 gradi

Caratteristica	Segnale Trigger	Segnale Echo	Dimensioni	Accuratezza a 20°C
Valore	10 uS	10 uS	45 · 20 · 15 mm	±3 mm

Sul robot sono presenti quattro sensori ultrasuoni per evitare gli ostacoli presenti sul percorso. L'algoritmo è lo stesso per ogni sensore; viene riportata una porzione di codice presente su Arduino per la lettura della distanza del sensore frontale con un ostacolo.

```
void checkFrontDistance()
{
    digitalWrite(frontTriggerPin, LOW);
    delayMicroseconds(4);
    digitalWrite(frontTriggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(frontTriggerPin, LOW);
    frontDuration = pulseIn(frontEchoPin, HIGH);
    frontDistanceCm = frontDuration * 10 / 292 / 2;
}
```

2.4.4 Ponte H

Il ponte H è un circuito che, alimentato a corrente continua, permette l'inversione di corrente in una maglia in modo da cambiare il verso di rotazione di un motore DC.

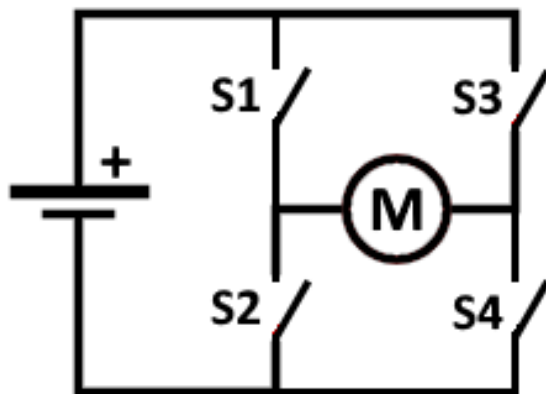


Figura 2.12: schema base di un ponte H.

Utilizzando i pin di ingresso presenti sulla scheda, è possibile inviare un segnale HIGH o LOW in modo da modificare il verso di rotazione; in ogni scheda sono presenti due enable che servono ad attivare il controllo su motori DC piuttosto che passo-passo.



Figura 2.13: ponte H L298N.

Le peculiarità di questa scheda sono nel fatto di poter gestire un ingresso fino a 35 V e poter pilotare in velocità due motori DC tramite il PWM ⁴.

2.4.5 DC converter

Fra i componenti elettronici utilizzati troviamo il DC converter. Questa scheda consente di regolare con precisione una tensione in ingresso variabile che vada da 3 a 40 V ad una tensione in uscita da un minimo di 1.25 V ad un massimo di circa 37 V con una corrente di 2A .

⁴Il PWM è una modulazione di impulsi che permette di modificare il duty cycle, il tempo nel quale un segnale cambia stato.

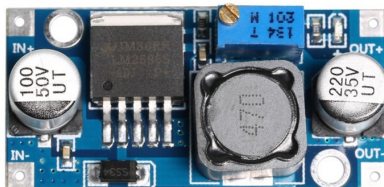


Figura 2.14: convertitore DC/DC LM2596.

Tramite il trimmer posto sulla scheda è possibile regolare l'uscita con la vite color oro in un range che da da 3 a 35 V in corrente continua.

2.5 Componenti meccaniche

Durante la costruzione del robot sono state prese delle decisioni in ambito meccanico relative alla scelta dei motori e dei motoriduttori. Di seguito è esplicitata la scelta in base ai punti di forza e debolezza.

2.5.1 Motori

Esistono diverse tipologie di motore, come i passo-passo ed i motori DC.



Figura 2.15: motore a corrente continua.

Nello sviluppo del robot si è scelto di utilizzare motori DC per la semplicità d'uso; in particolare, dovendo utilizzare un motoriduttore per la trasmissione della potenza all'albero, un motore DC si prestava ad infiniti modelli di scelta.

Inoltre, per questa tipologia di applicazione, sono usati maggiormente i motori a corrente continua piuttosto che i motori passo passo in quanto viene richiesto un movimento più fluido e di potenza.

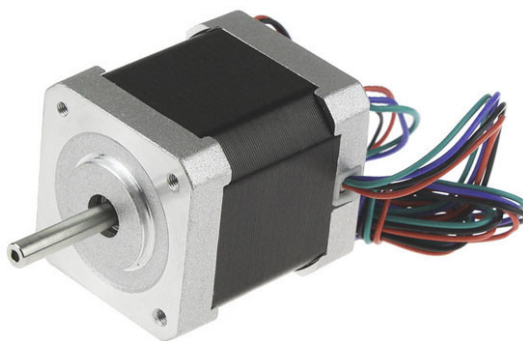


Figura 2.16: motore passo-passo.

Infatti, i motori passo-passo sono motori elettrici che offrono un controllo preciso di rotazione, anche di unità di grado: ruota in ogni direzione ed offre un maggior controllo dello spostamento effettuato.

I motori a corrente continua scelti raggiungono i 460 rpm con un voltaggio pari a 7.4 V e circa 0.6 A di assorbimento in condizioni normali⁵.

Il motoriduttore utilizzato, che viene schematizzato in figura, permette un giro della ruota ogni 25 giri dell'albero motore principale.

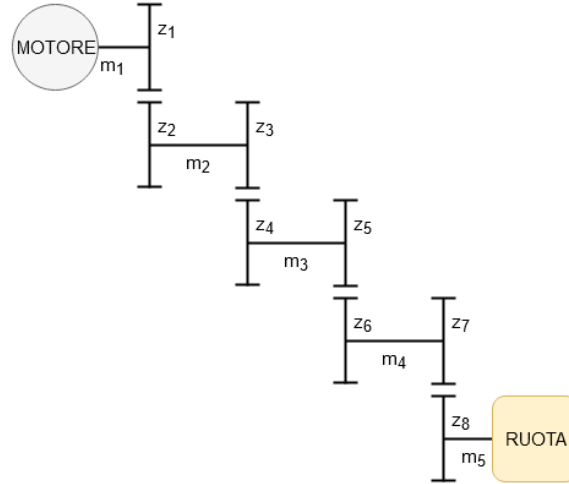


Figura 2.17: schema del motoriduttore.

Sfruttando la notazione presente, possiamo calcolare il rapporto:

$$\tau = \frac{m_5}{m_1} = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 \quad (2.5.1)$$

in cui m rappresenta gli alberi e $\tau_i = \frac{z_i}{z_{i+1}}$ con z numero di denti degli ingranaggi. Il rapporto di trasmissione τ è pari a 0.04.

⁵RPM = giri per minuto.

Capitolo 3

Cinematica omnidirezionale

In questo capitolo verrà esposta la cinematica diretta e inversa di un robot dotato di quattro ruote omnidirezionali mecanum e della combinazione dei versi di rotazione delle ruote in base al movimento da far effettuare al robot.

La cinematica è la capacità di calcolare i movimenti delle ruote attuati dal robot in funzione della velocità; viene creata una relazione univoca fra moto assoluto del robot nello spazio e, semplificando, quantità di energia da assegnare ad ogni attuatore.

L'obiettivo preposto comprende infatti il calcolo delle velocità delle singole ruote, data la velocità che il robot deve assumere: tutto ciò non è altro che determinare la cinematica inversa.

Allo stesso modo è possibile trovare una relazione fra le diverse pose, espressa tramite una terna della posizione nel sistema di riferimento e orientamento, del robot nel tempo, di cui si tratterà nel capitolo relativo alla localizzazione; qui ci focalizzeremo su quanto e come la cinematica di un robot dotato di ruote omnidirezionali *mecanum* sia conveniente per il raggiungimento di un punto dello spazio attraverso un semplice sistema di controllo: la possibilità di uso di un robot con ruote omnidirezionali che lo rende ologonico lascia spazio a infiniti scenari e usi.

3.1 Cinematica

La ruota classica è vista come un disco di raggio r in grado di ruotare senza slittamento in un piano xy con punto di contatto M appartenente al piano; la ruota è posta in modo perpendicolare al piano.

La rotazione senza slittamento implica che la velocità istantanea $\dot{r}_M$ nel punto di contatto M tra la ruota e il piano sia nulla, come in figura 3.4, ossia tale che

$$\dot{r}_M \cdot \vec{E}_x = 0 \quad (3.1.1)$$

$$\dot{r}_M \cdot \vec{E}_y = 0 \quad (3.1.2)$$

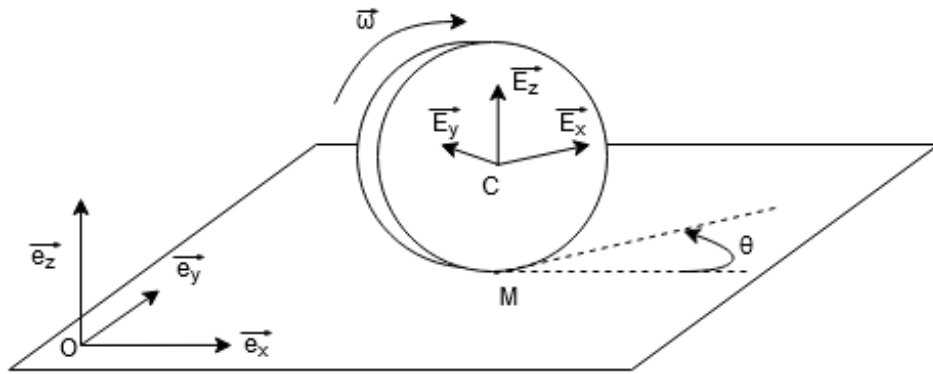


Figura 3.1: forze e sistemi di riferimento di una ruota in grado di rotolare senza strisciare.

La ruota si trova in (x_c, y_c, z_c) nel centro di massa C .

L'angolo ω definisce l'angolo di rotazione attorno all'asse $\vec{E}_y$ fissato il corpo, mentre θ è l'angolo tra il piano della ruota e $\vec{e}_x$.

Le ruote omnidirezionali sono costituite da un set di rulli in grado di ruotare liberamente, distribuiti esternamente sulla circonferenza della ruota.

L'inclinazione dell'asse di rotazione dei rulli fa cambiare il design e il funzionamento.

In particolare, è definita ruota omnidirezionale classica se l'asse di rotazione dei rulli è pari a $\pi/2$.



Figura 3.2: ruota omnidirezionale con l'asse dei rulli a 90 gradi rispetto all'asse di rotazione dell'intera ruota.

Per quanto riguarda le ruote *mecanum* che saranno prese in esame durante la trattazione, possiamo vedere come l'angolo dei rulli posizionati sulla circonferenza sia pari a $\pi/4$.



Figura 3.3: ruota mecanum con l'asse dei rulli posti a 45 gradi rispetto all'asse di rotazione dell'intera ruota.

Definendo con δ l'angolo dell'asse dei rulli, si presenta il caso generale, ricordando che $\delta = \pi/4$ nel caso di ruote *mecanum*.

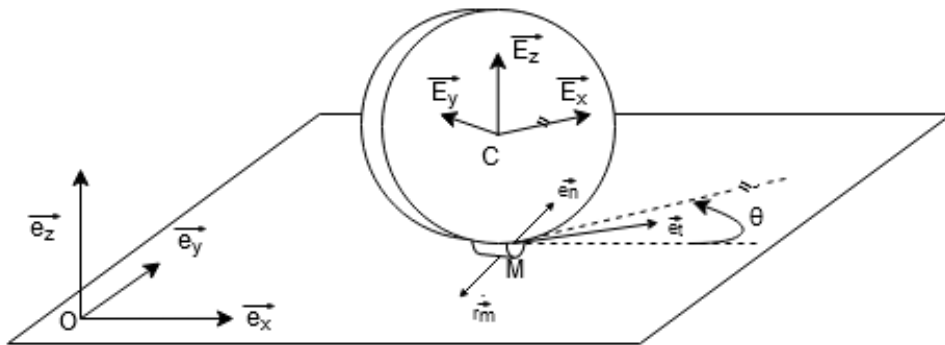


Figura 3.4: descrizione delle forze presenti su una ruota omnidirezionale.

Sia $\vec{e}_r$ il vettore unitario lungo l'asse dei rulli e δ l'angolo fra il piano della ruota e il vettore perpendicolare all'asse dei rulli, ossia fra il vettore $\vec{e}_n$ e $\vec{E}_x$.

La proiezione del vettore velocità $\vec{r}_M$ nel punto di contatto M della ruota con il piano

sull'asse $\vec{e}_\tau$ è nulla,

$$\dot{r}_M \cdot \vec{E}_\tau = 0 \quad (3.1.3)$$

$$\dot{r}_M = \dot{r}_C + \vec{\omega} \times \overrightarrow{CM} \quad (3.1.4)$$

con $\dot{r}_C$ la velocità del centro di massa C e $\vec{\omega}$ la velocità angolare del disco.

I vettori $\vec{\omega}$ e $\overrightarrow{CM}$ sono definiti come

$$\vec{\omega} = \dot{\omega} \vec{E}_y + \dot{\theta} \vec{E}_z \quad (3.1.5)$$

$$\overrightarrow{CM} = -r \vec{E}_z \quad (3.1.6)$$

Definendo x_c, y_c, r le coordinate in C,

$$\dot{r}_C = (\dot{x}_C \cos \theta + \dot{y}_C \sin \theta) \vec{E}_x + (-\dot{x}_C \sin \theta + \dot{y}_C \cos(\theta)) \vec{E}_y \quad (3.1.7)$$

$$\vec{\omega} \times \overrightarrow{CM} = -r \dot{\omega} \vec{E}_x \quad (3.1.8)$$

allora,

$$\dot{r}_M = (\dot{x}_C \cos \theta + \dot{y}_C \sin \theta - r \dot{\omega}) \vec{E}_x + (-\dot{x}_C \sin \theta + \dot{y}_C \cos \theta) \vec{E}_y \quad (3.1.9)$$

Inoltre,

$$\vec{e}_\tau = \sin \delta \vec{E}_x - \cos \delta \vec{E}_y \quad (3.1.10)$$

Unendo le ultime equazioni,

$$\begin{aligned} \dot{r}_M \cdot \vec{e}_\tau &= \dot{x}_C \cos \theta \sin \delta + \dot{y}_C \sin \theta \sin \delta \\ &\quad - r \dot{\omega} \sin \delta + \dot{x}_C \sin \theta \cos \delta - \dot{y}_C \cos \theta \cos \delta \\ &= \dot{x}_C \sin(\theta + \delta) - \dot{y}_C \cos(\theta + \delta) - r \dot{\omega} \sin \delta = 0 \end{aligned} \quad (3.1.11)$$

e sostituendo il valore di $\delta = \frac{\pi}{4}$, diventa

$$\dot{x}_C \sin(\theta + \frac{\pi}{4}) - \dot{y}_C \cos(\theta + \frac{\pi}{4}) - r\dot{\omega} \sin \frac{\pi}{4} = 0 \quad (3.1.12)$$

$$\dot{x}_C (\cos \theta + \sin \theta) + \dot{y}_C (\sin \theta - \cos \theta) - r\dot{\omega} = 0 \quad (3.1.13)$$

Fino ad ora è stato descritto il moto relativo ad una sola ruota.

Estendiamo il discorso ad un robot dotato di quattro ruote *mecanum* e definiamo con l_1 la distanza fra il centro di una ruota e la proiezione sull'asse di rotazione del centro del robot e con l_2 la distanza fra il centro del robot e la proiezione del centro della ruota sullo chassis, con δ l'angolo dei rulli presenti su ogni ruota.

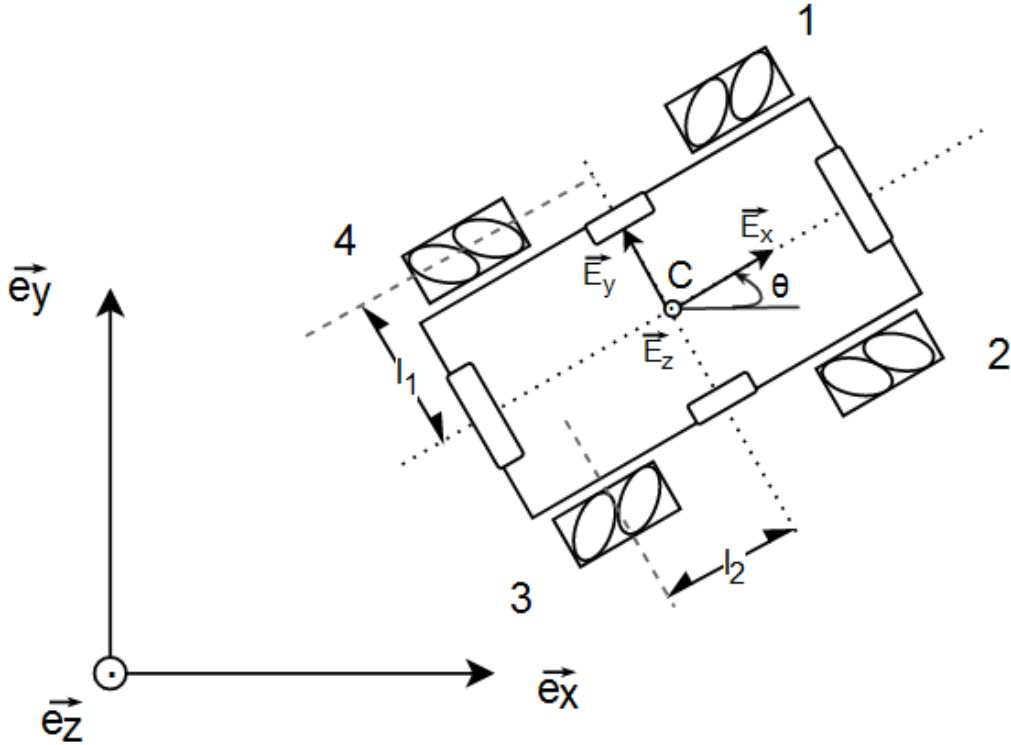


Figura 3.5: definizione del sistema di riferimento e delle grandezze di un robot omnidirezionale.

Osservando l'intero robot, possiamo notare che le quattro ruote sono posizionate, come appare in figura, in modo tale che l'intersezione dei prolungamenti degli assi a $\pi/4$ dei singoli rulli delle ruote si incontrino nel centro dello chassis.

Il veicolo, nella configurazione data da quattro ruote, è in grado di garantire il movimento in ogni direzione senza alcun sistema per sterzare.

Durante il moto le ruote *mecanum* girano con una certa velocità $\dot{\omega}_i$ e viene assunto che il raggio sia lo stesso per ogni ruota.

Quando le ruote mecanum sono azionate, i rulli angolati traducono una porzione della forza di rotazione della ruota ad una forza normale all'asse dei rulli.

Il robot è in grado di ruotare attorno al suo centro di massa C di un certo θ .

Applicando la formula precedente per ogni ruota e tenendo conto del sistema di riferimento, possiamo scrivere che:

$$\dot{x}_C(\cos \theta + \sin \theta) - \dot{y}_C(\cos \theta - \sin \theta) - \dot{\theta}(l_1 + l_2) - r\dot{\omega}_1 = 0 \quad (3.1.14)$$

$$\dot{x}_C(\cos \theta - \sin \theta) + \dot{y}_C(\cos \theta + \sin \theta) + \dot{\theta}(l_1 + l_2) - r\dot{\omega}_2 = 0 \quad (3.1.15)$$

$$\dot{x}_C(\cos \theta + \sin \theta) - \dot{y}_C(\cos \theta - \sin \theta) + \dot{\theta}(l_1 + l_2) - r\dot{\omega}_3 = 0 \quad (3.1.16)$$

$$\dot{x}_C(\cos \theta - \sin \theta) + \dot{y}_C(\cos \theta + \sin \theta) - \dot{\theta}(l_1 + l_2) - r\dot{\omega}_4 = 0 \quad (3.1.17)$$

A questo punto è possibile scrivere la cinematica inversa sotto forma di matrice,

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} \cos \theta + \sin \theta & -\cos \theta + \sin \theta & -(l_1 + l_2) \\ \cos \theta - \sin \theta & \cos \theta + \sin \theta & (l_1 + l_2) \\ \cos \theta + \sin \theta & -\cos \theta + \sin \theta & (l_1 + l_2) \\ \cos \theta - \sin \theta & \cos \theta + \sin \theta & -(l_1 + l_2) \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad (3.1.18)$$

Non essendo una matrice quadrata ed avendo termini positivi e limitati, possiamo

procedere e scrivere la cinematica diretta utilizzando la pseudoinversa e ricavando così la relazione tra la velocità del centro di massa del robot e la velocità di rotazione.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \frac{r}{4} \cdot \begin{bmatrix} \sqrt{2} \sin(\theta + \frac{\pi}{4}) & \sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) & \sqrt{2} \cos(\theta + \frac{\pi}{4}) \\ -\sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) & -\sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) \\ -\frac{1}{l_1+l_2} & \frac{1}{l_1+l_2} & \frac{1}{l_1+l_2} & -\frac{1}{l_1+l_2} \end{bmatrix} \cdot \begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} \quad (3.1.19)$$

3.2 Movimento omnidirezionale

Avendo a disposizione la cinematica inversa in 3.1.18, possiamo ricavare la velocità - ed il segno per decidere il verso - di rotazione da affidare ad ogni ruota.

In particolare, ponendo l'orientamento del robot $\theta = 0$ per facilitarci il calcolo, la 3.1.18 diventa

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & -1 & -(l_1 + l_2) \\ 1 & 1 & (l_1 + l_2) \\ 1 & -1 & (l_1 + l_2) \\ 1 & 1 & -(l_1 + l_2) \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad (3.2.1)$$

e se volessimo fare in modo che il robot vada, nel sistema di riferimento assoluto, nel verso positivo dell'asse $\vec{e}_x$, avremmo $\dot{y} = 0$ e $\dot{\theta} = 0$ e la 3.2.1 avrebbe solo la prima colonna non nulla, con coefficienti positivi, ossia un verso di rotazione positivo per ogni ruota.

Reiterando lo stesso procedimento per il verso opposto, ossia nella direzione data da $-\vec{e}_x$, avremmo il verso di rotazione per ogni ruota opposto, detto anche negativo se-

condo il nostro sistema di riferimento attuale.

Nel caso in cui volessimo percorrere la direzione data da $\vec{e}_y$, dovremmo azzerare, questa volta, $\dot{x}$ e $\dot{\theta}$

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} = \frac{1}{r} \cdot \begin{bmatrix} -1 \\ +1 \\ -1 \\ +1 \end{bmatrix} \quad (3.2.2)$$

Allo stesso modo possiamo ripetere lo stesso procedimento per il verso opposto, ossia per $-\vec{e}_y$ e constatare che il verso di rotazione per ogni ruota sia opposto alla 3.2.2.

Per quanto riguarda invece la rotazione su se stesso in senso antiorario, possiamo pensare di assumere $\dot{x} = 0$ e $\dot{y} = 0$.

In questo modo possiamo scrivere che

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} = \frac{1}{r} \cdot \begin{bmatrix} -\frac{1}{l_1+l_2} \\ \frac{1}{l_1+l_2} \\ \frac{1}{l_1+l_2} \\ -\frac{1}{l_1+l_2} \end{bmatrix} \quad (3.2.3)$$

Vengono riassunte, nel grafico che segue, le velocità delle ruote in base al tipo di spostamento del robot.

Si tenga conto che, nelle ultime due pose del robot, viene indicata anche l'intensità da assegnare ad ogni ruota: infatti, aumentando o diminuendo i giri delle ruote, si possono avere dei movimenti misti, non solo in una direzione.

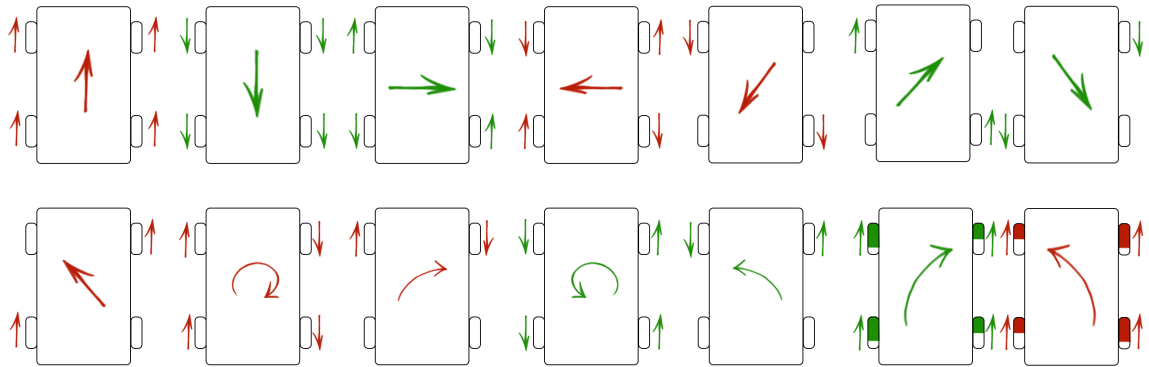


Figura 3.6: definizione del verso di rotazione di ogni singola ruota in base al movimento da effettuare.

Si noti che l'assunzione iniziale relativa a $\theta = 0$ non è condizione necessaria per avere le direzioni di rotazione delle ruote; si ripeta il procedimento per l'orientamento desiderato.

Capitolo 4

Localizzazione

*In questo capitolo sarà declinata la localizzazione tramite *dead reckoning* e tramite il filtro di Kalman applicato al robot.*

Il sistema localizzazione relativa - detta anche *dead reckoning* - viene più utilizzato rispetto ad un sistema di localizzazione assoluto. Infatti, basandosi sulla stima della posizione data dalle misure relative del moto del robot, è possibile localizzare il robot: ogni misura è ottenuta in relazione alle precedenti e per questo motivo gli errori crescono con il tempo.

La facilità di realizzazione nonché l'economicità, rendono questo sistema preferibile a sistemi più complessi e accurati, come la localizzazione assoluta.

Il sistema di localizzazione assoluta calcola la posizione del robot rispetto al sistema di riferimento generale, senza basarsi sulle misure lette precedentemente.

La posizione del robot dipende dalla misura corrente e non viene stimata utilizzando misure passate, come nei sistemi di localizzazione relativa.

Il pregio di questi sistemi è naturalmente l'assenza di errori crescenti col tempo durante il movimento del robot.

I sistemi di localizzazione assoluta si dividono in diversi metodi basati su *landmarks*, *beacons*, *active ranging systems*, *route-based* e *mapbased*.

Spesso si utilizza una soluzione mista che coniuga la localizzazione relativa e i sensori per la localizzazione assoluta. Le posizioni relative del robot possono essere acquisite generalmente tramite odometria e navigazione inerziale.

4.1 *Dead Reckoning indoor*

L'odometria si occupa di calcolare la posizione attuale di un corpo in movimento dall'ultima posizione nota e dalle condizioni iniziali.

Nota come *Dead Reckoning* ed in uso dai tempi delle spedizioni marittime, consente di calcolare la posizione attuale conoscendo sia la precedente che l'attuale velocità nonché la rotta percorsa.

Nel caso di robot mobili, conoscendo l'ingresso applicato alle ruote, si conosce anche la velocità. Utilizzando dei sensori per calcolare la distanza percorsa da ogni ruota, noti come encoder ed usandoli nell'odometria, è possibile ricostruire la distanza percorsa e l'orientamento di movimento del robot.

Gli encoder vanno a stimare il numero di tic e di giri della ruota in un certo intervallo di tempo.

Calcolando la nuova posizione utilizzando la tecnica del *Dead Reckoning* si deve tenere conto dei possibili errori cumulativi generati dai sensori ad ogni nuova computazione, come la non corretta rilevazione della velocità del moto o la mancata rilevazione della direzione.

Per comprendere l'importanza della corretta rilevazione, dato un tragitto di 1 km, la

variazione di un solo grado nella direzione comporta un errore di circa 17,5 metri.

L'odometria funziona bene nel caso di piccoli spostamenti ed è un meccanismo di rapida applicazione; garantisce, infatti, misure rapide e semplici da un punto di vista computazionale.

La sua debolezza è derivata dall'accumulazione, in ogni istante e in ogni spostamento, di errori sistematici, di disturbi del suolo, della variazione del raggio delle ruote in base alla temperatura, della risoluzione del sensore, di errori dovuti alla discretizzazione delle equazioni, del passaggio su terreno disconnesso e delle accelerazioni elevate e del conseguente scivolamento.

L'utilizzo di un robot in ambiente indoor riduce gli errori sistematici relativi al suolo perchè privo di disconnessioni. Non appena il robot si muove in un ambiente con o senza asperità, l'odometria accumula errori non compensabili.

4.2 Filtro di Kalman esteso

Come visto nella sezione precedente, l'odometria non è in grado di fornire con precisione una stima della posizione del robot, poichè soggetta a molteplici errori. L'idea è quella di affiancare all'odometria un sistema in grado di restituire la posizione assoluta del robot nel sistema di riferimento generale con una buona precisione. In questo è di aiuto il filtro di Kalman per il filtraggio e la predizione nei sistemi. Il robot, nonostante la quantità di sensori adottati, giunge a differenti misure della posizione a causa degli errori e della scarsità di precisione, dipendente totalmente dalle caratteristiche dei sensori. L'idea è quella di raccogliere le diverse misurazioni di tutti i sensori in uso e fonderle in modo da ottenere una stima sulla posa del robot che sia maggiormente precisa.

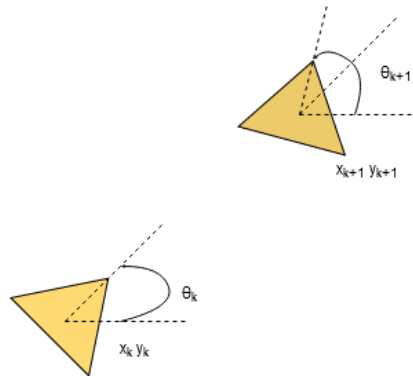


Figura 4.1: posa iniziale e posa predetta tramite filtro di Kalman esteso.

Il filtro di Kalman permette di effettuare questa operazione di fusione delle informazioni, indipendentemente dalla precisione, per stimare il valore attuale della posizione del robot.

Il filtro utilizza la funzione gaussiana di densità di probabilità per poter determinare l'affidabilità della stima.

La localizzazione mediante filtro di Kalman necessita che il problema sia formulato per la fusione delle misure dei sensori. La procedura che segue, in grado di migliorare l'affidabilità del robot con metodi probabilistici, viene svolto in due fasi, la *fase di predizione* e la *fase di correzione*, ossia il miglioramento delle previsioni e quello delle misure.

Il primo passo, e cioè la fase di predizione, consiste nell'applicare il modello di errore gaussiano al percorso misurato dai sensori IR per la lettura del numero di tic sulle ruote.

La fase di aggiornamento, o correzione, consiste in più step: la prima è di osservazione ed è finalizzata a collezionare i dati dai sensori installati, la seconda è di predizione della misura in funzione della precedente posizione nello spazio, la terza è di confronto

dei risultati della misurazione con quelli dell'osservazione, da cui ricava una sintesi fondendo il ricavato.

La posa del robot è descritta da

$$X_r = [x \quad y \quad \theta]' \quad (4.2.1)$$

Ipotizzando di essere in un sistema ideale, esente da errori sistematici di ogni tipo, il sistema viene descritto da 3.1.19 dipendente dall'orientamento del robot e dall'ingresso dato ad ogni ruota.

Definendo

$$J_{inv}(\theta(t)) = \frac{r}{4} \cdot \begin{bmatrix} \sqrt{2} \sin(\theta + \frac{\pi}{4}) & \sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) & \sqrt{2} \cos(\theta + \frac{\pi}{4}) \\ -\sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) & -\sqrt{2} \cos(\theta + \frac{\pi}{4}) & \sqrt{2} \sin(\theta + \frac{\pi}{4}) \\ -\frac{1}{l_1+l_2} & \frac{1}{l_1+l_2} & \frac{1}{l_1+l_2} & -\frac{1}{l_1+l_2} \end{bmatrix} \quad (4.2.2)$$

Scegliendo un tempo di campionamento T sufficiente, possiamo discretizzare il sistema

$$\begin{bmatrix} x((k+1)T) \\ y((k+1)T) \\ \theta((k+1)T) \end{bmatrix} - \begin{bmatrix} x((k)T) \\ y((k)T) \\ \theta((k)T) \end{bmatrix} = T \cdot J_{inv}(\theta(kT)) \cdot \begin{bmatrix} \omega_1(kT) \\ \omega_2(kT) \\ \omega_3(kT) \\ \omega_4(kT) \end{bmatrix} \quad (4.2.3)$$

Il robot potrebbe presentare degli errori dovuti a slittamenti, discontinuità e ondulazioni del suolo; per questo motivo, nelle equazioni che regolano il moto del sistema, vengono inseriti i disturbi gaussiani a media nulla definiti come $v(k) = [v_x \quad v_y \quad v_\theta]'$ tali che

$$E\{v(k)\} = 0$$

$$E\{v(k)v'(h)\} = Q(k, \Delta s) \quad h = k$$

$$E\{v(k)v'(h)\} = 0 \quad h \neq k$$

con $Q(k, \Delta s)$ simmetrica e semidefinita positiva, dipendente dalla distanza fra due step consecutivi, tale da essere una matrice diagonale dipendente dall'incertezza della stima della posa. Le incertezze dell'odometria, basata sulle misurazioni degli encoder, può essere affinata utilizzando le informazioni provenienti dai sensori esterni di distanza. In particolare, sono in uso quattro sensori ultrasuoni gestibili tramite il vettore di dati

$$Y(k) = [y_1 \quad \dots \quad y_4]'$$

$$Y(k) = h(k, X_r) + d(k) \quad (4.2.4)$$

in cui $h(k, X_r)$ indica la relazione fra posizione del robot e rilevamento dei sensori esterni[1]; $d(k)$ indica il rumore gaussiano a media nulla con una certa matrice di covarianza diagonale $R(k)$ per modellare gli errori di misura. Avendo nota la posa del robot e le misure degli encoder, è possibile ipotizzare la nuova posizione $X_{r,ipotesi}(k+1)$. A questo punto si possono calcolare le risposte dei sensori esterni e confrontarle con le misure effettive non appena il robot sarà nella posa $X_r(k+1)$; in questo modo vengono ridotti gli eventuali errori di stima della posizione.

Avendo un sistema non lineare, si necessita di linearizzare l'equazione 4.2.3

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} x(k) + T \frac{r\sqrt{2}}{4} (\sin(\theta + \frac{\pi}{4})\omega_1 + \cos(\theta + \frac{\pi}{4})\omega_2 + \sin(\theta + \frac{\pi}{4})\omega_3 + \cos(\theta + \frac{\pi}{4})\omega_4) \\ y(k) + T \frac{r\sqrt{2}}{4} (-\cos(\theta + \frac{\pi}{4})\omega_1 + \sin(\theta + \frac{\pi}{4})\omega_2 - \cos(\theta + \frac{\pi}{4})\omega_3 + \sin(\theta + \frac{\pi}{4})\omega_4) \\ \theta(k) + T \frac{r\sqrt{2}}{4} (-\frac{\omega_1}{l_1+l_2} + \frac{\omega_2}{l_1+l_2} + \frac{\omega_3}{l_1+l_2} - \frac{\omega_4}{l_1+l_2}) \end{bmatrix}$$

ed il sistema linearizzato diventa

$$\begin{aligned} F(k) &\triangleq \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial \theta} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial \theta} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial \theta} \end{bmatrix} = \\ &= \begin{bmatrix} 1 & 0 & \frac{r\sqrt{2}}{4} (\cos(\theta + \frac{\pi}{4})\omega_1 - \sin(\theta + \frac{\pi}{4})\omega_2 + \cos(\theta + \frac{\pi}{4})\omega_3 - \sin(\theta + \frac{\pi}{4})\omega_4) \\ 0 & 1 & \frac{r\sqrt{2}}{4} (\sin(\theta + \frac{\pi}{4})\omega_1 + \cos(\theta + \frac{\pi}{4})\omega_2 + \sin(\theta + \frac{\pi}{4})\omega_3 + \cos(\theta + \frac{\pi}{4})\omega_4) \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (4.2.5)$$

In modo analogo è possibile scrivere la relazione della matrice W , jacobiana della dinamica rispetto ai disturbi.

A questo punto sono riportate le equazioni in uso per l'implementazione del filtro di Kalman esteso.

$$\widehat{x}_{k+1}^- = f(\widehat{x}_k, u_k, 0) \quad (4.2.6)$$

$$P_{k+1}^- = F_k P_k F_k^T + W_k Q_k W_k^T \quad (4.2.7)$$

$$\widehat{x}_{k+1} = \widehat{x}_{k+1}^- + K_{k+1}(z_{k+1} - h(\widehat{x}_{k+1}^-, 0)) \quad (4.2.8)$$

$$K_{k+1} = P_{k+1}^- H_{k+1}^T (H_{k+1} P_{k+1}^- H_{k+1}^T + L_{k+1} R_{k+1} L_{k+1}^T)^{-1} \quad (4.2.9)$$

$$P_{k+1} = (I_n - K_{k+1} H_{k+1}) P_{k+1}^- \quad (4.2.10)$$

Capitolo 5

Risultati

In questo capitolo saranno illustrati i risultati dell'implementazione del filtro di Kalman esteso sviluppato in ambiente MATLAB sul robot descritto precedentemente e dotato a livello sensoriale di:

- encoder IR posti sulle ruote;
- 4 sensori ad ultrasuoni posti nel punto medio dei quattro lati del robot con un range di misura da $[0.02; 4]$ m con una incertezza ¹ pari a ± 0.003 m;
- misurazione della distanza ogni 5 cm percorsi dal robot.

Le misure che seguono sono state rilevate ponendo due sensori IR ad una distanza fissa e utilizzando il tempo intercorso fra i due segnali.

Per quanto riguarda il movimento del robot,

- velocità di traslazione sul versore $\vec{E}_x$ pari a 1.21 m/s
- velocità di traslazione sul versore $\vec{E}_y$ pari a 0.02 m/s
- velocità di rotazione pari a 0.43 rad/s

¹L'incertezza del sensore HC-SR04 è letta dal datasheet del produttore.

Il tempo di campionamento T è pari a 0.05 s tenendo conto del tempo massimo impiegato dalla lettura dei sensori ultrasuoni e infrarossi.

5.1 Risultati simulazione

In questa fase è stato studiato il comportamento del robot modellando la dinamica esplicita precedentemente. In particolare, una volta implementato il filtro di Kalman, il robot ha percorso una traiettoria complessa in uno spazio esteso indoor in modo da testare le capacità effettive del sistema.

Lo stato iniziale effettivo è pari a (27; 23) metri, mentre lo stato iniziale stimato ha una differenza di ± 5 cm; al robot viene fatta percorrere una distanza pari a 5 m procedendo secondo la direzione $\vec{E}_x$; successivamente in direzione $\vec{E}_y$, obliqua, rotazione pari a $-\pi/2$ ed in ultimo in direzione $-\vec{E}_y$.

Certamente la traiettoria appena elencata comporta una certa complessità; la motivazione a tutto ciò è riconducibile alla necessità di testare il più possibile la capacità dell'algoritmo di adattarsi a scenari differenti, vari e alcune volte per testare il filtro su un robot con un movimento ologonico, quindi del tutto differente dalle solite applicazioni.

In questa sezione verranno inoltre riportate le simulazioni al variare della matrice di covarianza P , della posa iniziale della stima e simulando, per una ruota, un differente diametro.

5.1.1 Risultati del sistema

Una volta impostata la traiettoria da far eseguire al robot, gli errori odometrici di varianza e gli errori del sensore in base al suo datasheet, possiamo studiare gli effetti del filtro applicato.

Facendo percorrere al robot la traiettoria in figura 5.1, possiamo notare come, nonostante sia presente un errore proporzionale allo spostamento delle ruote per gli encoder e un errore di lettura in base alla risoluzione del sensore, il filtro di Kalman risponda perfettamente e stimi il sistema senza sforzo alcuno.

È infatti in grado di raggiungere ogni punto intermedio ed è privo di errori non trascurabili sul punto di arrivo.

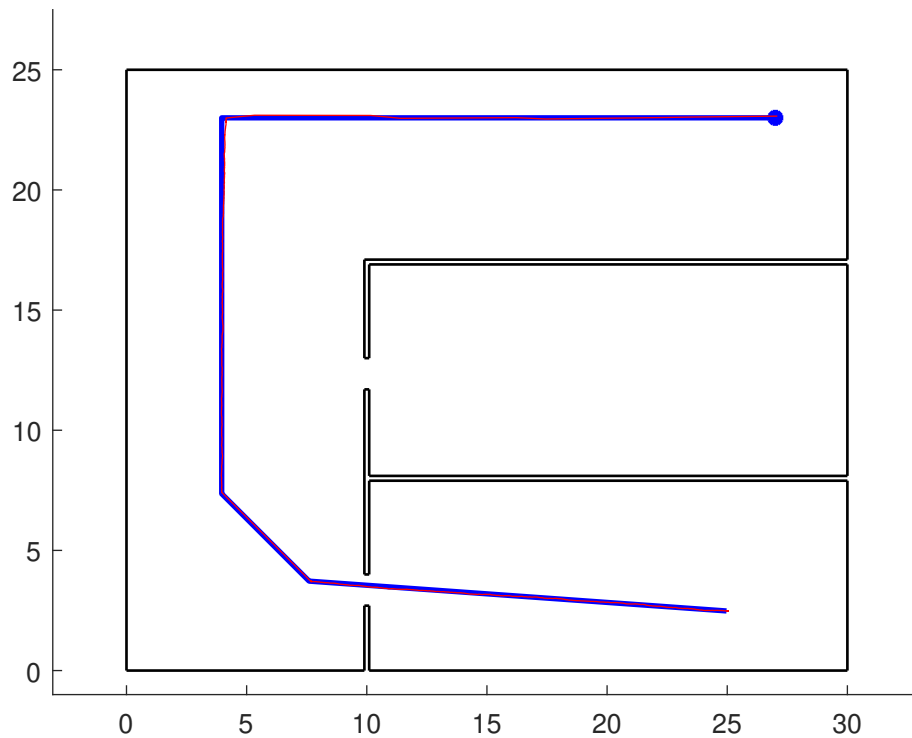


Figura 5.1: simulazione robot; in blu il percorso effettivo, in rosso quello stimato

Nella figura 5.2 possiamo vedere l'andamento descritto per ogni componente della

posa X_r stimata ed effettiva nel tempo.

Anche in questo caso, non notiamo alcuna differenza sostanziale nella stima attuata dal filtro.

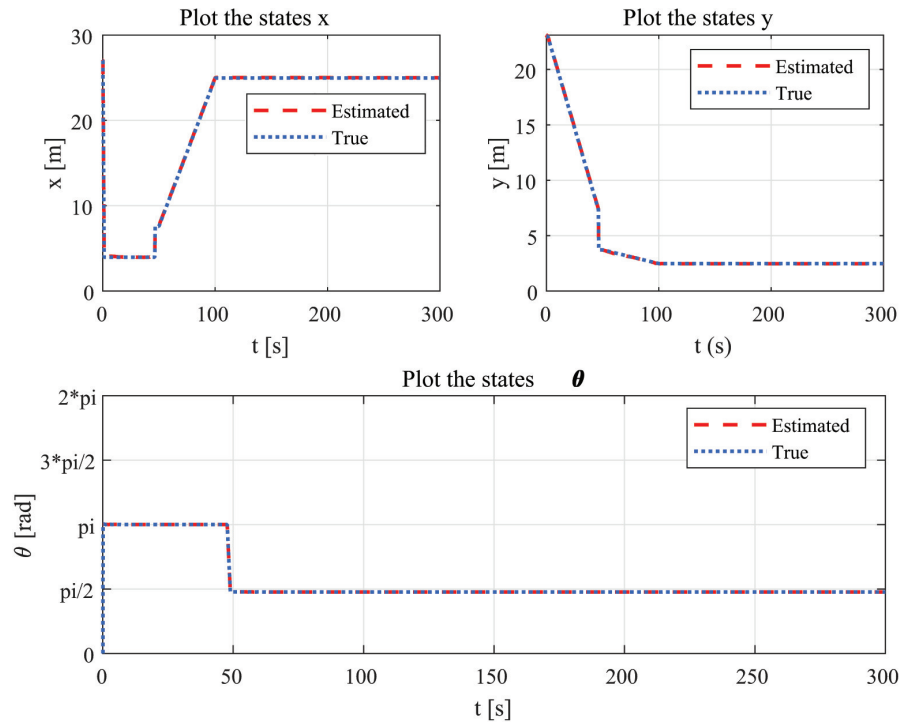


Figura 5.2: andamento del robot: grafico posa stimata e posa effettiva

Possiamo notare l'andamento dell'errore logaritmico della differenza fra la posa stimata e quella effettiva 5.3; per tutte e tre le componenti della posa è nullo.

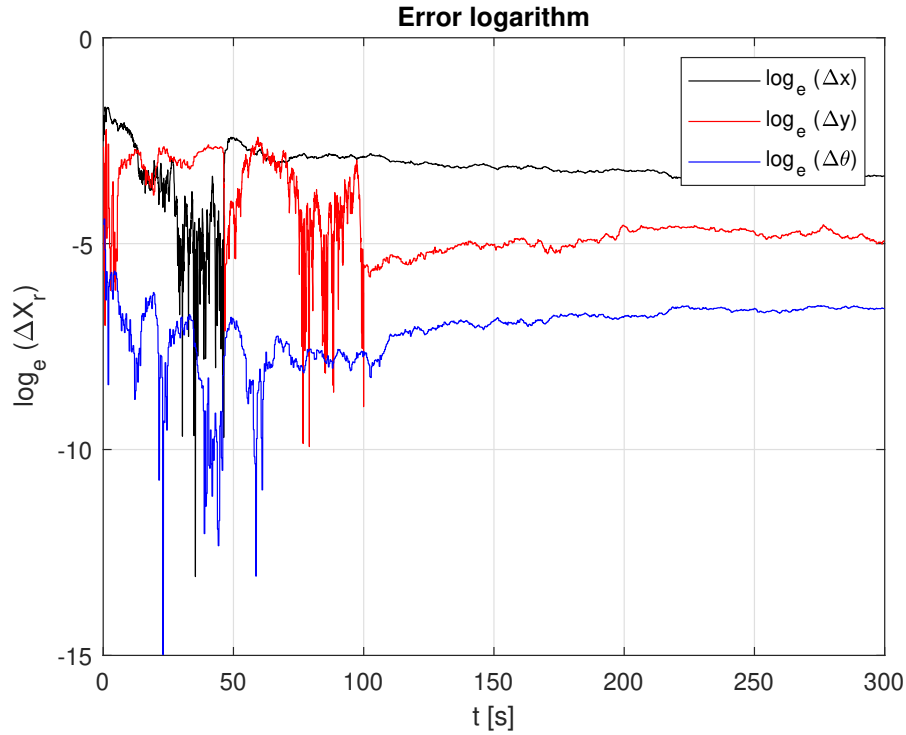


Figura 5.3: errore logaritmico della posa stimata e della posa effettiva.

5.1.2 Risultati con diverse condizioni iniziali

In questa sezione vedremo i risultati andando a variare lo stato iniziale della stima. In particolare, aumentando entrambe le coordinate di partenza di più di 0.5 m, possiamo notare (figura 5.4) come il filtro di Kalman riesca dopo il primo step a tendere alla traiettoria effettiva finchè non viene bruscamente cambiata direzione: in questo tratto, avendo una velocità laterale decisamente minore di quella di traslazione in avanti, impiega più tempo, e quindi più step, a stimare lo stato correttamente per rientrare dall'errore iniziale.

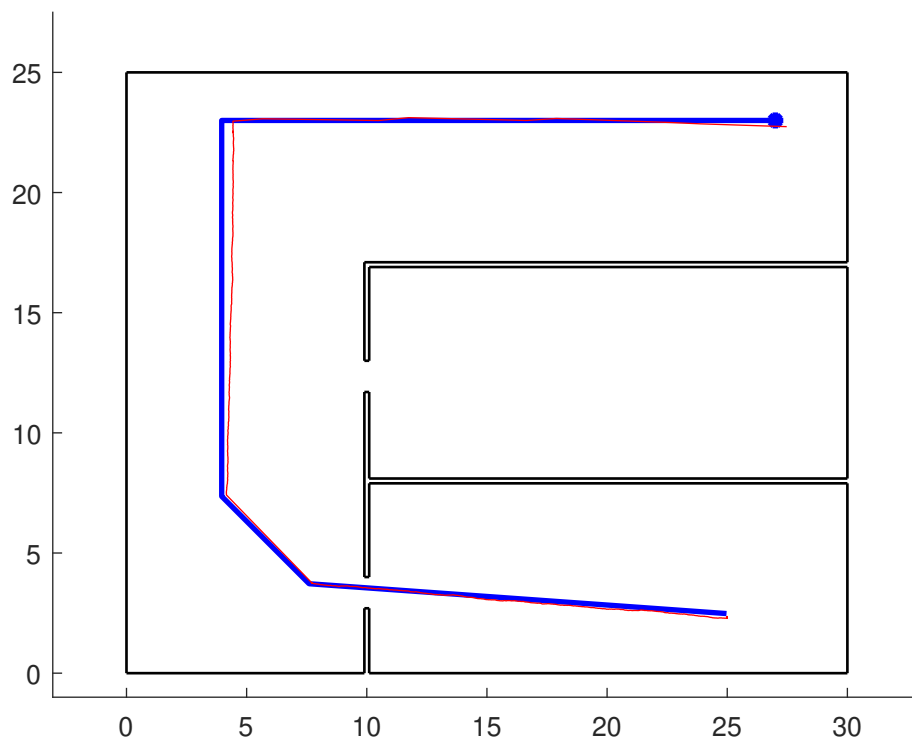


Figura 5.4: simulazione robot; in blu il percorso effettivo, in rosso quello stimato con diverso uno stato iniziale.

Allo stesso modo possiamo notare anche dai grafici riportati (figura 5.5) l'andamento delle componenti della posa, il leggero discostamento fra la stima e la componente effettiva in quel punto.

Nel grafico 5.6 è rappresentato l'errore logaritmico della differenza delle componenti fra quelle stimate e quelle effettive; ancor di più possiamo notare che non sussistono errori notevoli per quanto riguarda la stima della posizione.

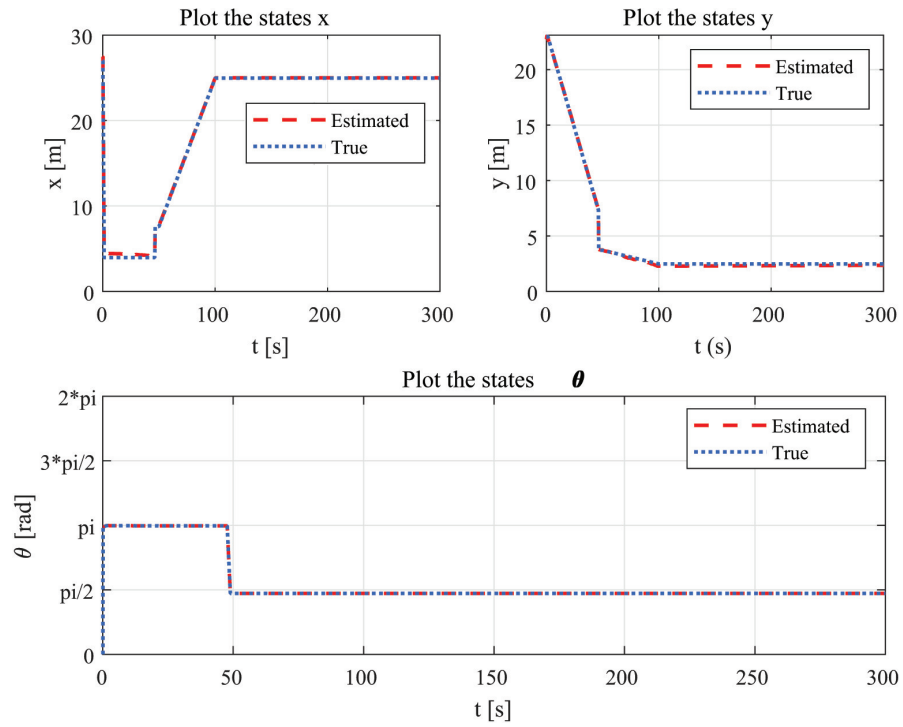


Figura 5.5: andamento del robot: posa stimata ed effettiva con diverso stato iniziale.

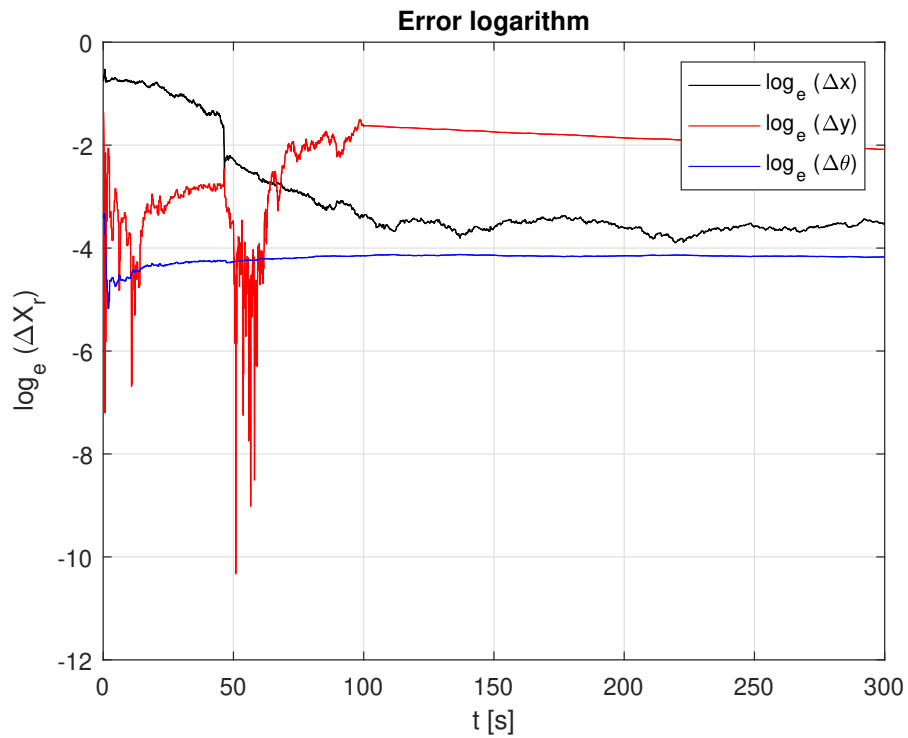


Figura 5.6: errore logaritmico della posa stimata e effettiva con diverso stato iniziale.

5.1.3 Risultati al variare della matrice di covarianza

La matrice di covarianza veniva, precedentemente, inizializzata diagonale utilizzando il quadrato della deviazione standard dalla posizione iniziale pari a 0.01 per x e y , invece pari a 0.02 per θ .

La variazione effettuata in questa sezione riguarda esattamente i valori di deviazione standard, pari a 1 per x e y e pari a 2 per θ e quindi una differenza sostanziale nella inizializzazione.

Possiamo notare come la stima sia diversa dal percorso effettivo nel primo breve tratto ma che, già dopo pochi step iniziali, si avvicini e vada a tendere alla traiettoria in blu e la matrice si aggiorni sensibilmente (figura 5.7).

Tenendo conto del fatto che, partendo da una matrice di covarianza con valori sulla diagonale nettamente maggiori di ben quattro ordini di grandezza, l'algoritmo vada a diminuire la P per adattarla al sistema; si tratta di un buon andamento, anche per quanto riguarda i disturbi minimi presenti nel primo tratto fino al cambio di direzione.

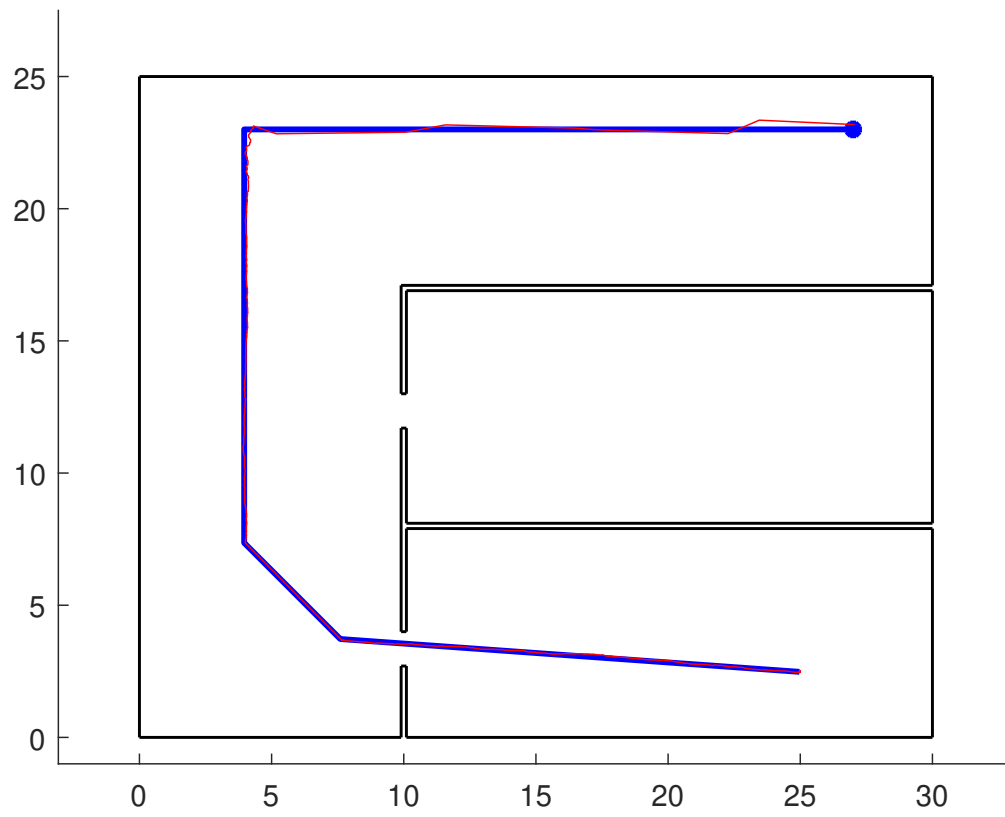


Figura 5.7: simulazione robot; in blu il percorso effettivo, in rosso quello stimato con matrice di covarianza aumentata.

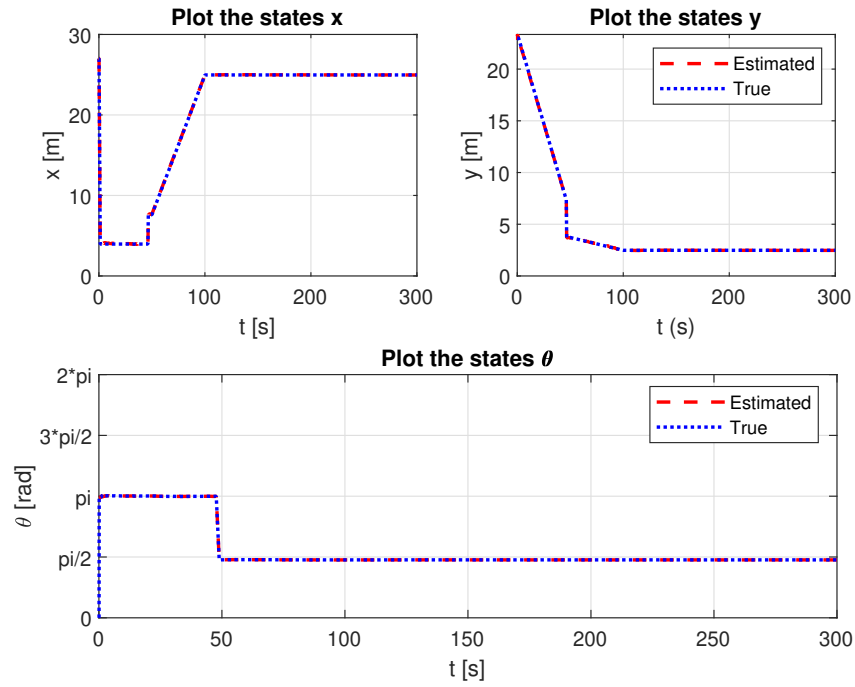


Figura 5.8: andamento del robot: posa stimata ed effettiva al variare della matrice di covarianza.

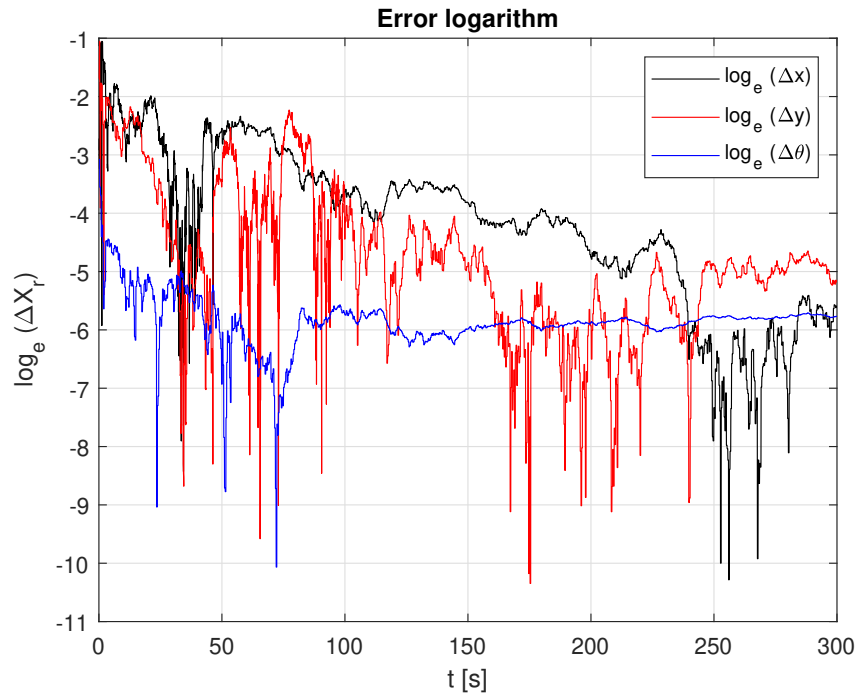


Figura 5.9: errore logaritmico della posa stimata e posa effettiva con matrice di covarianza aumentata.

5.1.4 Risultati al variare del raggio di una ruota

Se aumentassimo il raggio di una ruota del robot, avremmo l'occasione di osservare come l'odometria sia falsata e produca errori in funzione del tempo.

In particolare, nonostante gli encoder vadano a leggere un certo numero di tic, il robot avrà percorso una traiettoria diversa; il filtro di Kalman non riesce a stimare a lungo termine la traiettoria corretta e gli errori odometrici si accumulano con il passare del tempo.

È possibile osservare il comportamento assunto dal filtro andando ad aumentare del 5% il raggio di una ruota (figura 5.10); da notare il comportamento della traiettoria in rosso che in alcuni tratti diverge dalla traiettoria in blu ma che, dopo alcuni step, si avvicina nuovamente al percorso effettivo.

In figura 5.11 è possibile notare come la stima dell'orientamento sia coerente all'orientamento effettivo, mentre le altre due componenti della posa diventino diverse in alcuni piccoli tratti.

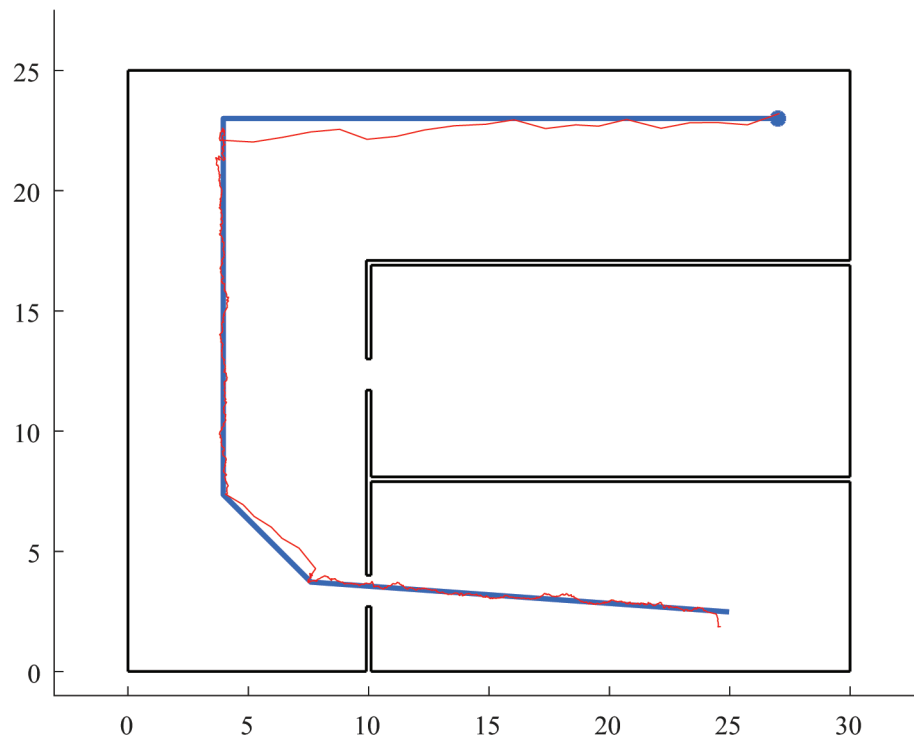


Figura 5.10: simulazione robot; in blu il percorso effettivo, in rosso quello stimato con una ruota più grande.

Viene ora riportato in figura 5.11 l'andamento delle componenti della posa del robot per mostrare la differenza fra la stima e la posa effettiva.

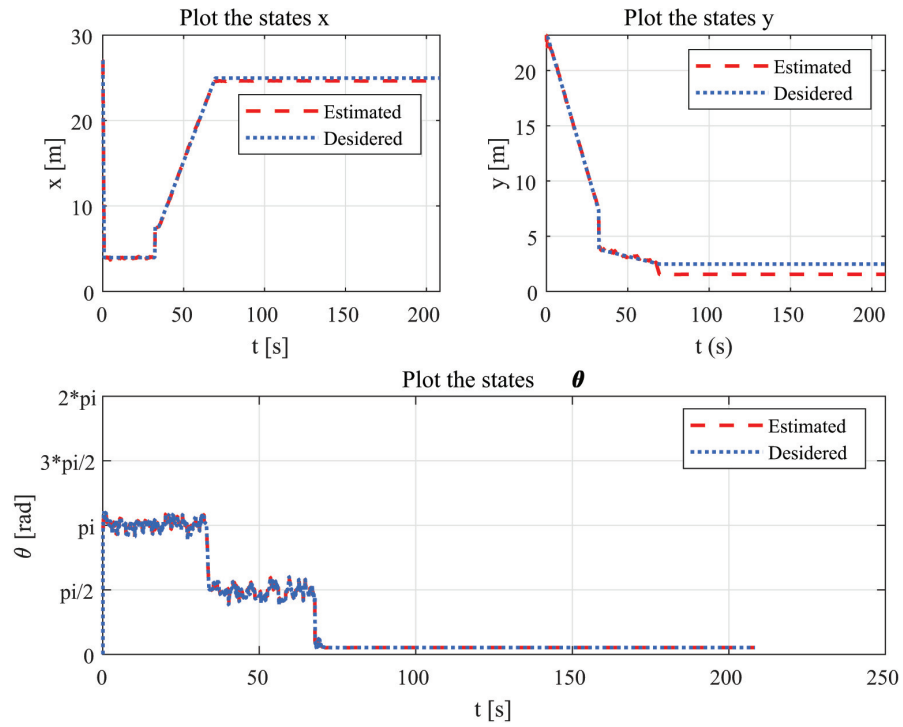


Figura 5.11: andamento del robot: grafico posa stimata e posa effettiva con il raggio di una ruota maggiore.

In ultimo (figura 5.12) è riportato il logaritmo dell'errore nel tempo delle componenti della posa stimata e della posa effettiva.

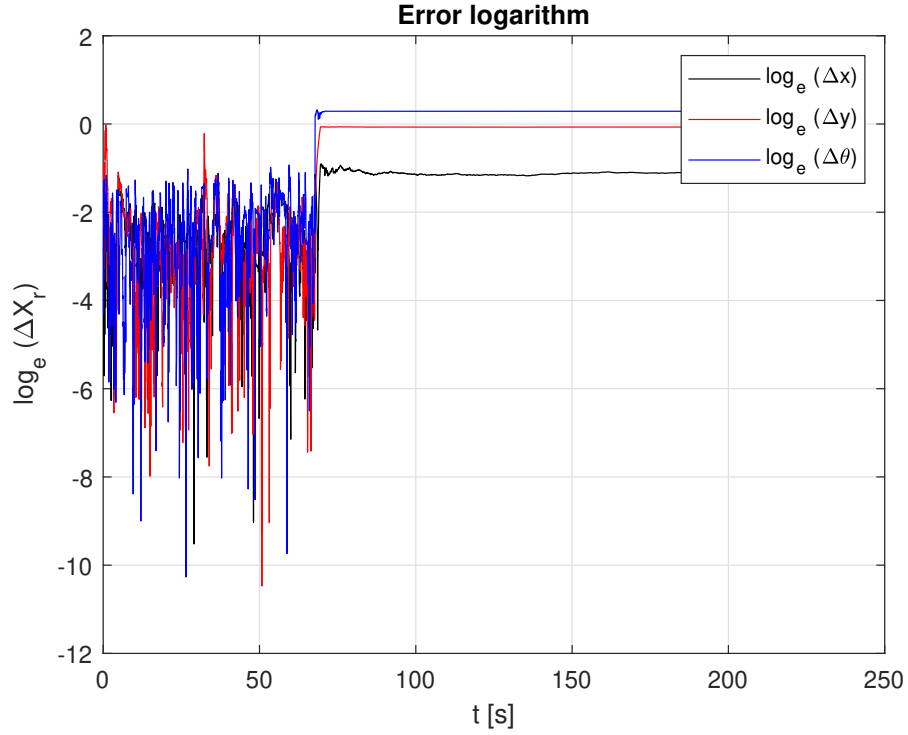


Figura 5.12: errore logaritmico della posa stimata e posa effettiva con una ruota dal raggio maggiore.

5.2 Risultati sperimentali

In quest'ultima fase è stato studiato il comportamento del prototipo su una certa traiettoria stabilita in precedenza in uno spazio ridotto, rispetto alla fase simulativa, sempre indoor.

Dopo aver ricreato l'ambiente in figura, la traiettoria del robot desiderata ha condizioni di stato iniziali in (6;8); non avendo a disposizione strumentazione ad alta risoluzione, l'imprecisione è di ± 5 cm .

Viene fatto percorrere al robot una distanza di 5 m procedendo sulla direzione $\vec{E}_x$ per

poi procedere su $\vec{E}_y$ per 2 m.

La traiettoria appena esplicitata, rispetto alla precedente utilizzata in fase di simulazione, è meno varia ma permette di poter comunque studiare il comportamento del robot. Il filtro di Kalman viene applicato al movimento del robot in modo offline: una volta fatto svolgere al robot il percorso, Arduino salva i dati odometrici e le misurazioni dei sensori esterni in modo tale da utilizzarli in MATLAB ed osservare se il percorso generato è vicino a quello proposto dal robot.

Anche in questa sezione verranno riportate le varianti sulla simulazione per quanto riguarda lo stato iniziale e la matrice di covarianza.

5.2.1 Risultati sperimentali del sistema

Per poter misurare lo spostamento del robot è stato ideato un sistema di sensori IR posizionati lungo tutta la stanza in modo da poter capire la velocità del robot in ogni tratto percorso. Infatti, da questa misurazione, è emersa la velocità di traslazione in direzione $\vec{E}_x$ e $\vec{E}_y$. Una volta fatto ciò, è stato dato il compito di descrivere la traiettoria.

Nella figura 5.13 sono riportati in blu l'andamento desiderato del robot, in nero quello effettivamente percorso dal prototipo e in rosso la correzione effettuata dal filtro.

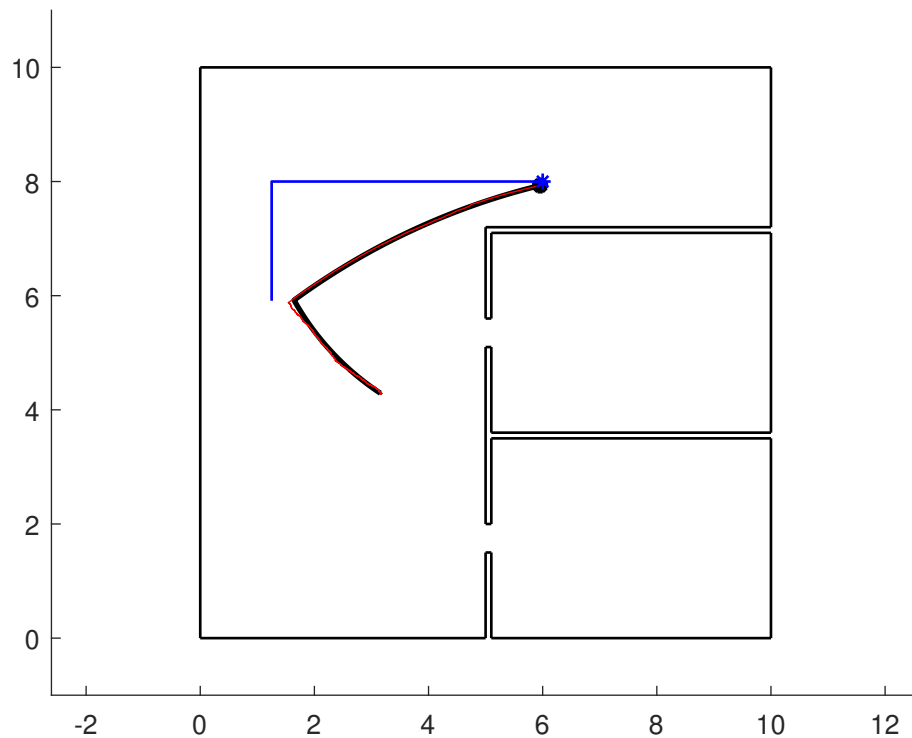


Figura 5.13: movimento del robot nello spazio indoor.

La differenza sostanziale fra la traiettoria blu e nera è data da un errore meccanico presente sullo chassis che fa in modo che l'asse di una ruota non sia parallelo agli altri assi, ma leggermente ruotato: questo fa in modo che il robot compia una traiettoria curva.

L'errore meccanico deriva da un'incorretta misurazione della distanza fra due centri di due viti responsabili del sostegno dell'intero motore.

Il filtro stima perfettamente la traiettoria del prototipo; per questo motivo l'odometria non sembra avere alcuna problematica. Anche osservando le figure 5.14 e 5.15 possiamo notare l'andamento delle componenti della posa del robot e l'errore logaritmico.

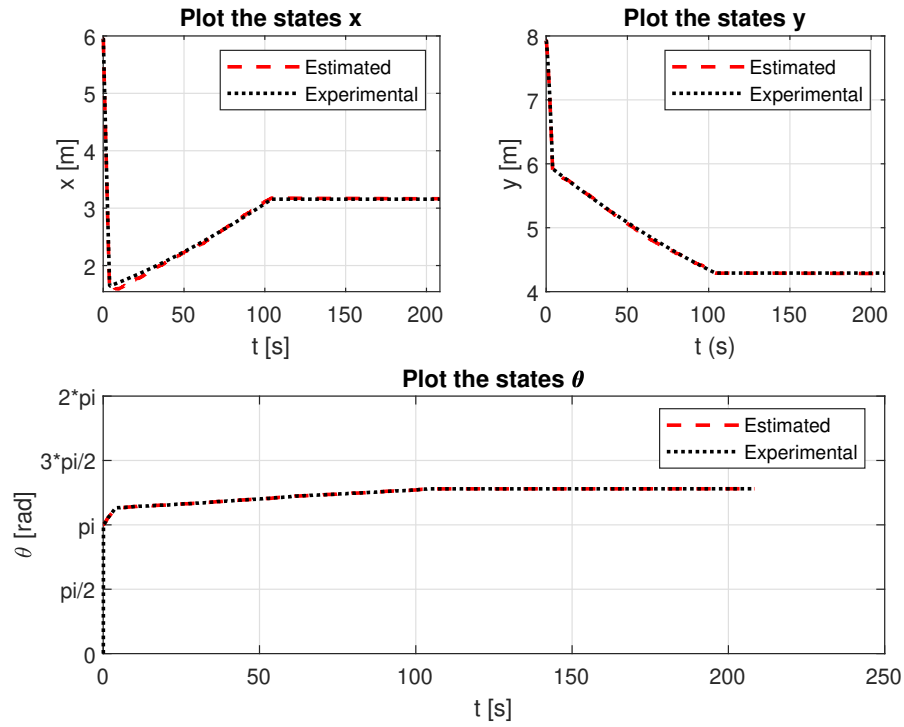


Figura 5.14: andamento del robot. Grafico posa stimata e posa sperimentale.

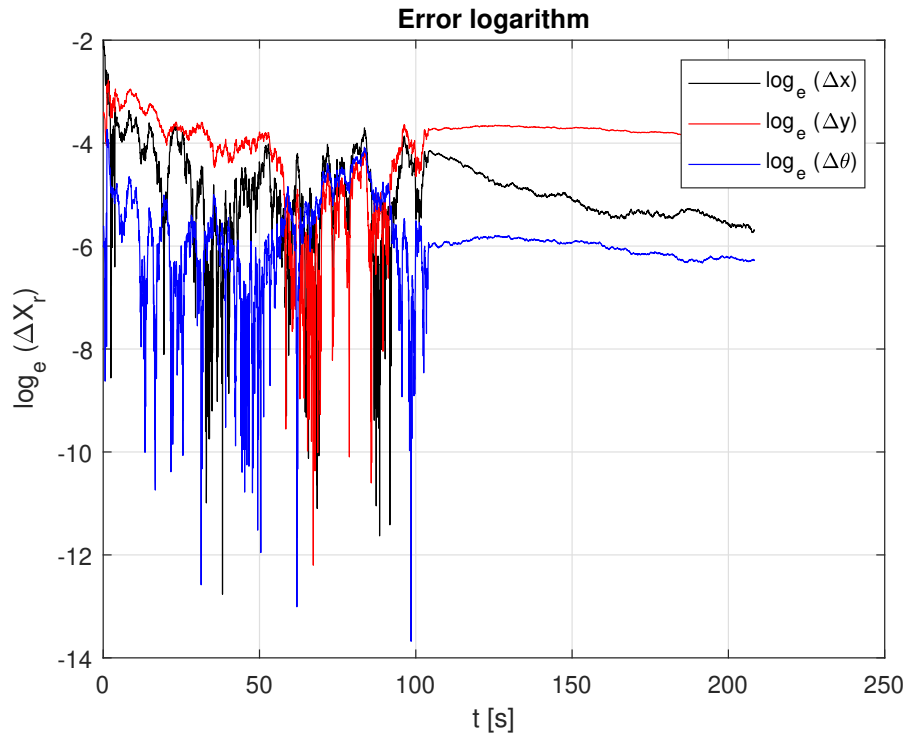


Figura 5.15: errore logaritmico della posa stimata e sperimentale.

5.2.2 Risultati con diverse condizioni iniziali

Andando a variare le condizioni iniziali di circa 0.5 m, possiamo notare (figura 5.16) come la stima sia fedele alla traiettoria del prototipo se non per dei piccoli scostamenti. Il filtro riesce a stimare la posa con un piccolo margine di errore (figura 5.17 e 5.18).

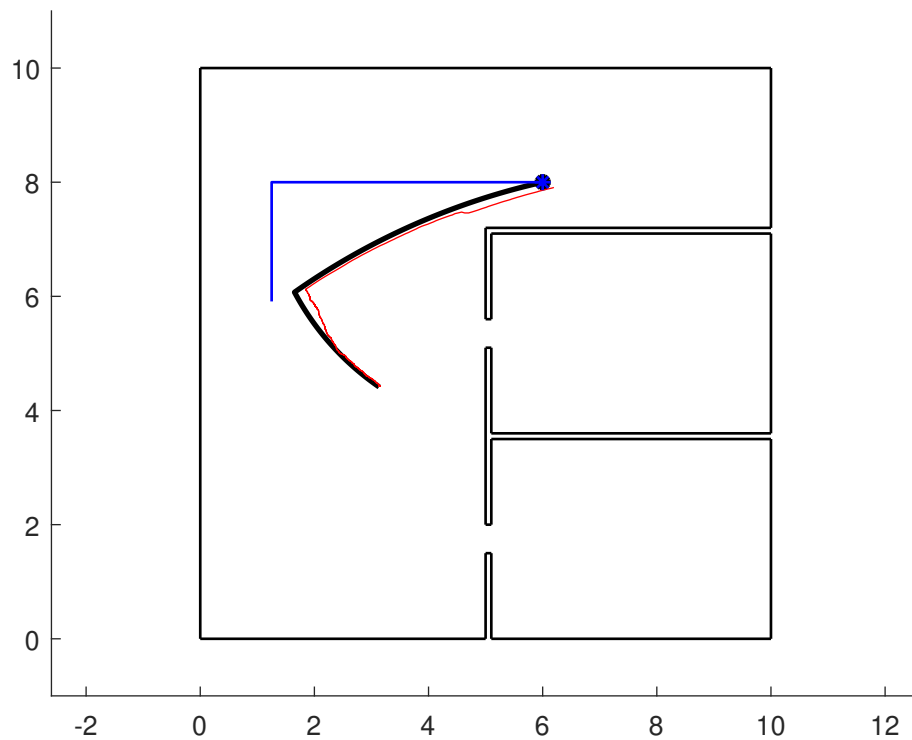


Figura 5.16: movimento del robot nello spazio indoor con diverse condizioni iniziali.

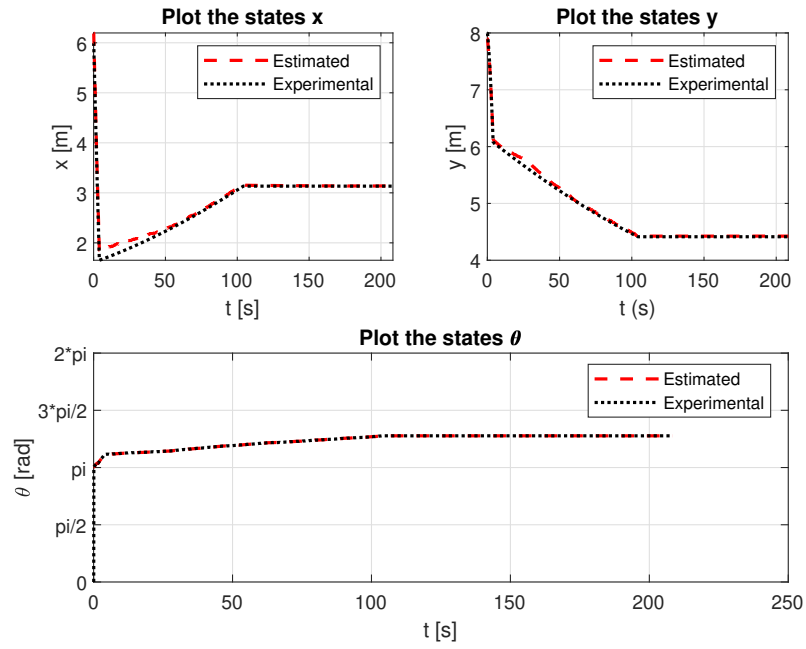


Figura 5.17: andamento del robot. Grafico posa stimata e posa sperimentale con diverse condizioni iniziali.

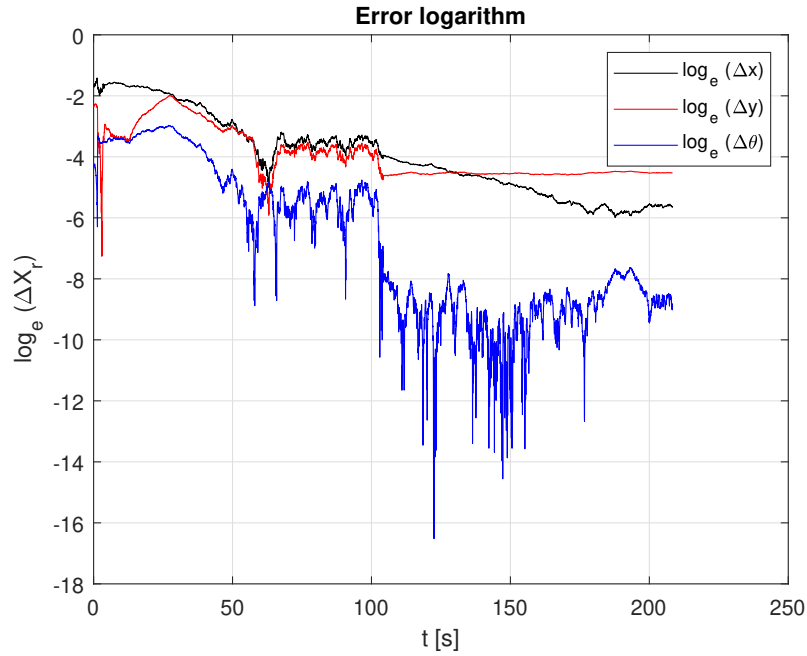


Figura 5.18: errore logaritmico della posa stimata e sperimentale con diverse condizioni iniziali.

5.2.3 Risultati al variare della matrice di covarianza

La matrice di covarianza veniva, precedentemente, inizializzata diagonale utilizzando il quadrato della deviazione standard dalla posizione iniziale, pari a 0.01 per x e y e pari a 0.02 per θ .

La variazione effettuata in questa sezione riguarda esattamente i valori di deviazione standard, pari a 1 per x e y e pari a 2 per θ .

Possiamo notare come la stima (in rosso) nella prima porzione di tratto sia lontana dalla traiettoria del prototipo (in nero), ma che dopo alcuni step riesca a stimare correttamente l'andamento figura (5.19).

Discorso ben diverso per la figura 5.21 in cui, nel primo tratto, è visibile il risultato del logaritmo dell'errore.

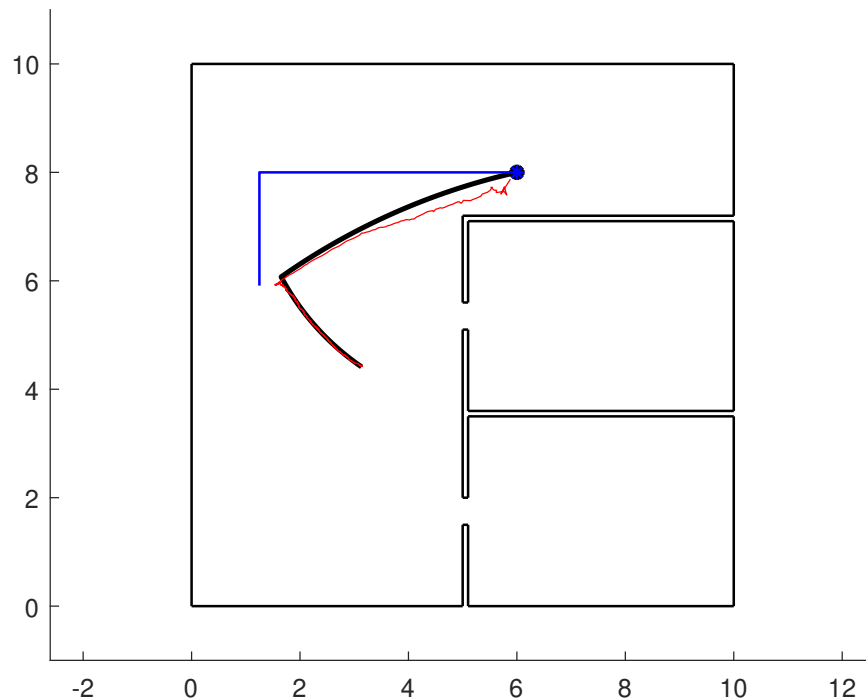


Figura 5.19: movimento del robot nello spazio indoor con matrice di covarianza maggiore.

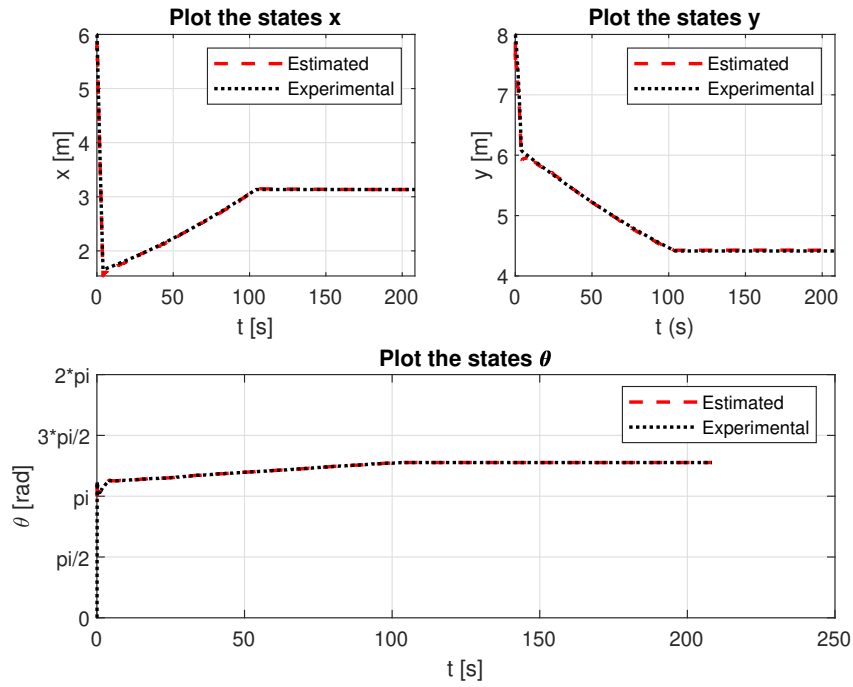


Figura 5.20: andamento del robot. Grafico posa stimata e posa sperimentale con matrice di covarianza maggiore.

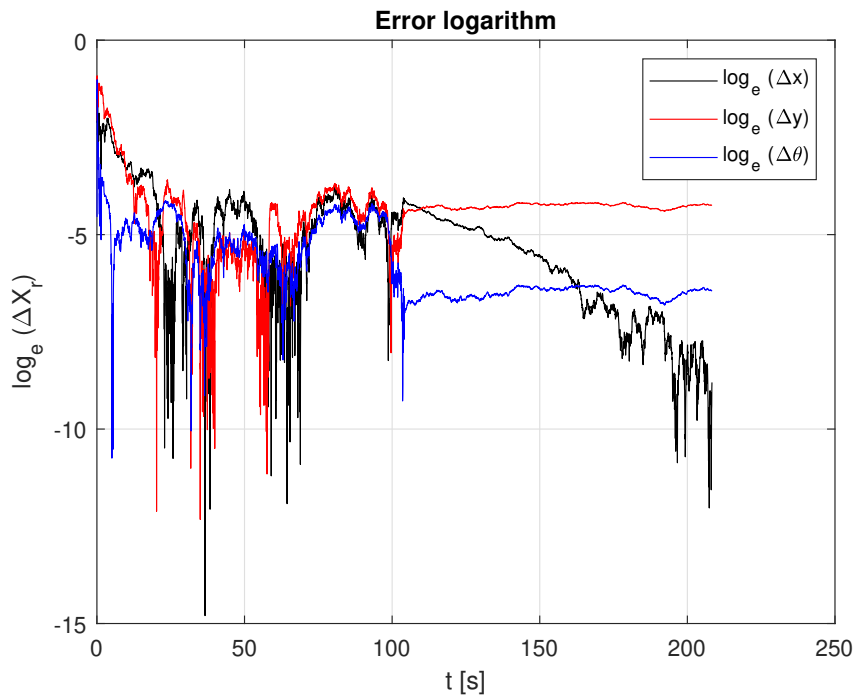


Figura 5.21: errore logaritmico della posa stimata e della posa sperimentale con matrice di covarianza maggiore.

Capitolo 6

Conclusioni e sviluppi futuri

La fusione delle informazioni provenienti dai sensori interni ed esterni attraverso il filtro di Kalman esteso risolve il problema della localizzazione del robot con ottimi risultati che vedono la stima dello stato tendente allo stato del robot. Nonostante le perturbazioni presenti sulle misure odometriche e su quelle dei sensori ultrasonici che misurano la distanza dalle pareti dell'ambiente, il filtro di Kalman riesce a stimare correttamente la posa del robot.

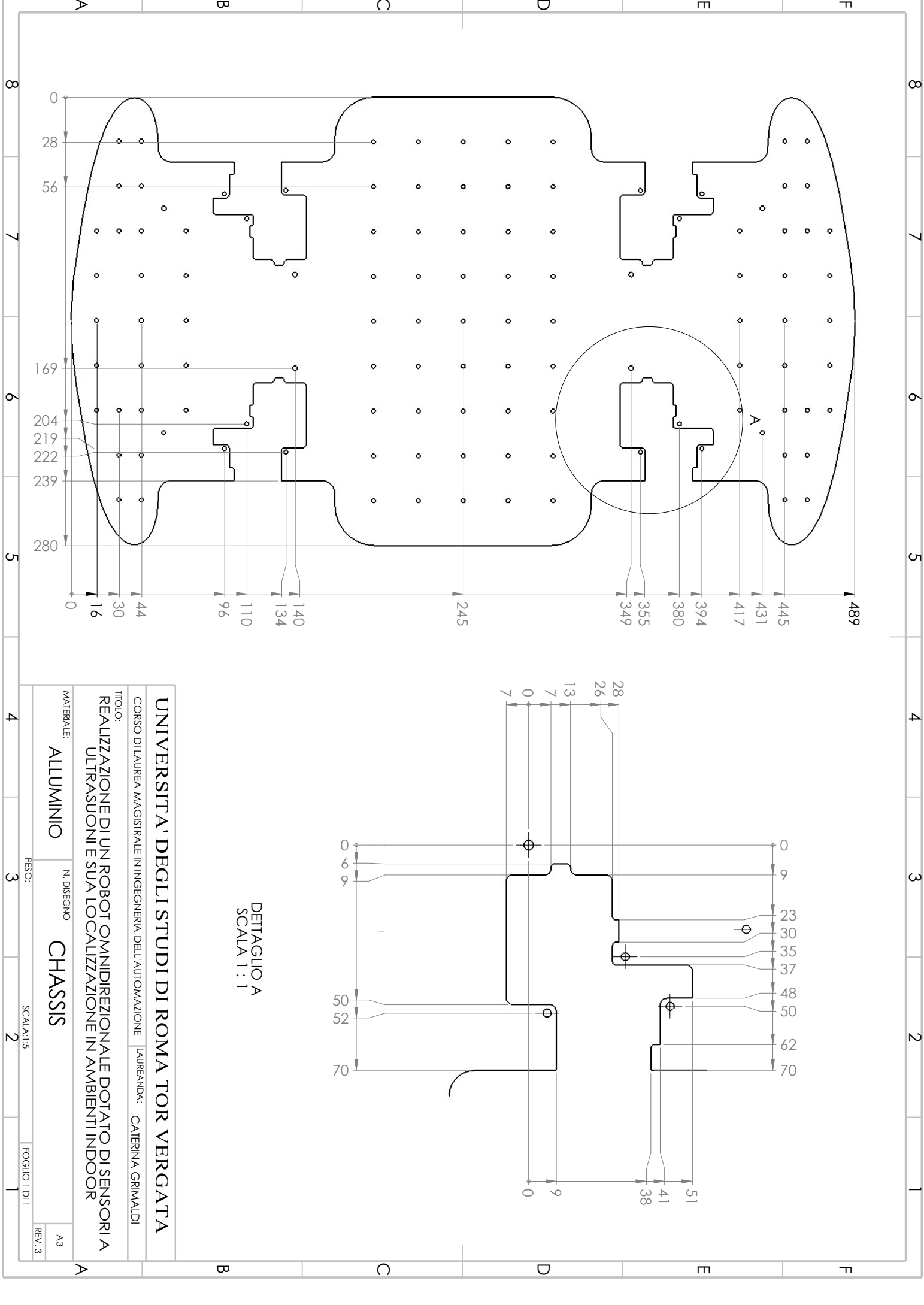
Il filtro viene utilizzato su un sistema non lineare, approssimato con la linearizzazione; tutto ciò comporta delle approssimazioni, responsabili anche delle minime rumorosità della risposta. Da un punto di vista hardware, il robot potrebbe essere migliorato in modo tale da modificare il comportamento del prototipo, e cioè:

- registrando lo chassis in alluminio per evitare che la ruota con l'asse di rotazione non parallelo agli altri produca errore;
- utilizzare uno spessore poco maggiore per lo chassis in modo da evitare che fletta;
- ricorrendo a encoder a forcella in modo da ridurre l'errore odometrico;

- utilizzando sensori con maggiore risoluzione per poter rilevare al meglio le misure.

Attualmente il robot è impiegato per poter raggiungere una certa posizione compensando gli errori odometrici; per i suoi sviluppi in ambiente indoor è possibile dotarlo di un sistema di carico in modo da utilizzarlo come carrello mobile autonomo. Se si aumentasse il raggio delle ruote, è possibile che trasporti carichi sempre più pesanti rispetto ad un normale robot dotato di ruote semplici.

L'aggiunta di un braccio con due gradi di libertà renderebbe il sistema ancora più completo ed utilizzabile in molti contesti industriali. L'aggiunta di una stazione autoricaricante affinché il robot possa utilizzarla non appena la batteria raggiunge un livello minimo di soglia, renderebbe il prototipo più autonomo.



UNIVERSITA' DEGLI STUDI DI ROMA TOR VERGATA

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA DELL'AUTOMAZIONE LAUREANDA: CATERINA GRIMALDI

TITOLO:
REALIZZAZIONE DI UN ROBOT OMNIDIREZIONALE DOTATO DI SENSORI A ULTRASUONI E SUA LOCALIZZAZIONE IN AMBIENTI INDOOR

MATERIALE: ALLUMINIO

N. DISEGNO

CHASSIS

PESO:

SCALA:1:5

FOGLIO 1 DI 1

A3
REV. 3

Elenco delle figure

2.1	procedimento di realizzazione <i>bottom up</i>	7
2.2	vista del prototipo finale poggiato su un piano.	8
2.3	vista dall'alto del prototipo.	8
2.4	specifiche di una ruota mecanum.	11
2.5	progettazione dello <i>chassis</i> in lega di alluminio.	13
2.6	schema elettrico di collegamento.	15
2.7	microcontrollore Arduino ATmega 2560.	16
2.8	batteria GTF 26650 li.ion	17
2.9	collegamento batteria.	18
2.10	sensore di prossimità ad infrarossi FC-51.	19
2.11	sensore ultrasuoni HC-SR04.	21
2.12	schema base di un ponte H.	23
2.13	ponte H L298N.	24
2.14	convertitore DC/DC LM2596.	25
2.15	motore a corrente continua.	26
2.16	motore passo-passo.	26
2.17	schema del motoriduttore.	27

3.1	forze e sistemi di riferimento di una ruota in grado di rotolare senza strisciare.	29
3.2	ruota omnidirezionale con l'asse dei rulli a 90 gradi rispetto all'asse di rotazione dell'intera ruota.	30
3.3	ruota mecanum con l'asse dei rulli posti a 45 gradi rispetto all'asse di rotazione dell'intera ruota.	31
3.4	descrizione delle forze presenti su una ruota omnidirezionale.	31
3.5	definizione del sistema di riferimento e delle grandezze di un robot omnidirezionale.	33
3.6	definizione del verso di rotazione di ogni singola ruota in base al movimento da effettuare.	37
4.1	posa iniziale e posa predetta tramite filtro di Kalman esteso.	41
5.1	simulazione robot; in blu il percorso effettivo, in rosso quello stimato	47
5.2	andamento del robot: grafico posa stimata e posa effettiva	48
5.3	errore logaritmico della posa stimata e della posa effettiva.	49
5.4	simulazione robot; in blu il percorso effettivo, in rosso quello stimato con diverso uno stato iniziale.	50
5.5	andamento del robot: posa stimata ed effettiva con diverso stato iniziale.	51
5.6	errore logaritmico della posa stimata e effettiva con diverso stato iniziale.	51
5.7	simulazione robot; in blu il percorso effettivo, in rosso quello stimato con matrice di covarianza aumentata.	53
5.8	andamento del robot: posa stimata ed effettiva al variare della matrice di covarianza.	54

5.9	errore logaritmico della posa stimata e posa effettiva con matrice di covarianza aumentata.	54
5.10	simulazione robot; in blu il percorso effettivo, in rosso quello stimato con una ruota più grande.	56
5.11	andamento del robot: grafico posa stimata e posa effettiva con il raggio di una ruota maggiore.	57
5.12	errore logaritmico della posa stimata e posa effettiva con una ruota dal raggio maggiore.	58
5.13	movimento del robot nello spazio indoor.	60
5.14	andamento del robot. Grafico posa stimata e posa sperimentale. . . .	61
5.15	errore logaritmico della posa stimata e sperimentale.	61
5.16	movimento del robot nello spazio indoor con diverse condizioni iniziali. .	62
5.17	andamento del robot. Grafico posa stimata e posa sperimentale con diverse condizioni iniziali.	63
5.18	errore logaritmico della posa stimata e sperimentale con diverse condizioni iniziali.	63
5.19	movimento del robot nello spazio indoor con matrice di covarianza maggiore.	64
5.20	andamento del robot. Grafico posa stimata e posa sperimentale con matrice di covarianza maggiore.	65
5.21	errore logaritmico della posa stimata e della posa sperimentale con matrice di covarianza maggiore.	65

Bibliografia

- [1] FORNI F., *Filtro di Kalman esteso per la localizzazione di robot mobili*, Tesi di Laurea in Ingegneria Informatica presso l'Università di Tor Vergata 2004.
- [2] WADA M., *Holonomic and omnidirectional wheelchairs with synchronized 4WD mechanism*, International Conference on Intelligent Robots and Systems, San Diego USA, 2007.
<https://ieeexplore.ieee.org/document/4399585>
- [3] PODNAR G., Blackwell M., *Uranus*, NASA.
<https://www.cs.cmu.edu/~gwp/robots/Uranus.html>
- [4] VEHICLE TECHNOLOGIES INC, *Sidewinder omnidirectional lift track*, Vehicle Technologies, 2010.
<http://www.airtrax.com/>
- [5] TAHERI H., QIAO B., GHAEMINEZHAD N., *Kinematic Model of a Four Mecanum Wheeled Mobile Robot*, International Journal of Computer Applications, Volume 113 - No. 3, 2015
- [6] WON YOON S., PARK S., JONG SHIK K., *Kalman Filter Sensor Fusion for Mecanum Wheeled Automated Guided Vehicle Localization*, Journal of Sensors,

- vol. 2015, Article ID 347379, pp. 1-7, 2015.
<https://doi.org/10.1155/2015/347379>
- [7] LI Y., DAI S., ZHENG Y., TIAN F., YAN X., *Modeling and Kinematics Simulation of a Mecanum WheelPlatform in RecurDyn*, Journal of Robotics, vol. 2018, pp. 1-7, 2018.
- [8] ADASCALITEI F., DOROFTEI I., *Practical applications for mobile robots based on Mecanum wheels - a system survey*, Romanian Review Precision Mechanics, Optics and Mechatronics, pp. 21-29, 2011.
- [9] VAN HAENDEL R., *Omnidirectional mobile robot - design and implementation, bioinspiration and robotics walking and climbing robots*, ISBN 978-3-902613-15-8, pp. 511-524, Vienna, 2007.
- [10] MOHD NASIR N. Z., ZAKARIA M. A., RAZALI S., BIN ABU M. Y., *Autonomous mobile robot localization using Kalman filter*, MATEC Web of Conferences, Vol. 90, 2017.