

UNIVERSITÀ DI PISA



Facoltà di Ingegneria

Laurea Specialistica in Ingegneria dell'Automazione

Tesi di laurea

## **Stima della posizione di un veicolo subacqueo autonomo da misure acustiche di distanza**

Candidato:

*Chiara Burato* .....

Relatori:

*Ch.mo Prof. Ing. Andrea Caiti* .....

*Dott. Ing. Andrea Munafò* .....

*Dott. Ing. Vincenzo Calabrò* .....

Sessione di Laurea del 11/05/2012  
Anno accademico 2011/2012

## **Sommario**

Il lavoro di tesi affronta il problema di localizzazione assoluta di un veicolo autonomo subacqueo (AUV-Autonomous Underwater Vehicle) tramite misure acustiche di distanza.

In particolare vengono presentati due metodi di stima della posizione assoluta del veicolo, il primo vede l'utilizzo del filtro di Kalman esteso, il secondo considera il problema di stima come un problema di ottimizzazione vincolata.

Si presenta un esempio di utilizzo per entrambi i metodi. I dati a disposizione sono stati acquisiti durante una prova in mare svolta il giorno 22 Settembre 2010. La valutazione delle performance dei due metodi viene fatta attraverso simulazione ed analisi numerica dei dati in uscita.

## **Abstract**

This thesis faces the problem of the absolute localization of Autonomous Underwater Vehicle (AUV) based on ranging measurements.

We studied two algorithms of localization of vehicle, one is the Extended Kalman Filter and the second is a constrained optimization problem.

We explain also an illustrative example for both algorithms, using experimental data acquired during a survey at sea surface level on December 10, 2010. Finally the paper shows the way both algorithms works and compares their performance with a numerical validation of the analysis.

# Indice

<b>Introduzione</b>	<b>5</b>
<b>1 Descrizione, acquisizione ed elaborazione dei dati</b>	<b>8</b>
1.1 Descrizione e acquisizione dati . . . . .	9
1.2 Rielaborazione dei dati . . . . .	17
<b>2 Sistemi di riferimento e trasformazione di coordinate</b>	<b>23</b>
2.1 Introduzione . . . . .	23
2.2 Sistemi di riferimento e convenzioni . . . . .	25
2.2.1 Sistema geodetico . . . . .	25
2.2.2 Sistema di riferimento ECEF . . . . .	26
2.2.3 Sistema di riferimento NED . . . . .	26
2.2.4 Sistema di riferimento body . . . . .	27
2.3 Trasformazioni di coordinate . . . . .	28
2.3.1 Trasformazione di coordinate da geodetiche a ECEF . . . . .	28
2.3.2 Trasformazione di coordinate da ECEF a NED . . . . .	28
2.3.3 Trasformazione di coordinate dal sistema NED al sistema body . . . . .	29
<b>3 Modello cinematico e studio dell'osservabilità</b>	<b>30</b>
3.1 Modello cinematico del sistema . . . . .	30
3.2 Analisi dell'osservabilità . . . . .	31
<b>4 Il filtro di Kalman esteso (EKF)</b>	<b>33</b>
4.1 Il processo da stimare . . . . .	34
4.2 L'origine computazionale del filtro . . . . .	34
4.3 Applicazione al modello . . . . .	38
<b>5 Problema di ottimizzazione vincolata</b>	<b>43</b>
5.1 Gli algoritmi di ottimizzazione . . . . .	43
5.2 Problema di ottimo applicato al problema . . . . .	44

<b>6</b>	<b>Risultati e confronto</b>	<b>49</b>
<b>7</b>	<b>Conclusioni</b>	<b>51</b>
<b>A</b>	<b>Yalmip e SeDuMi</b>	<b>52</b>

# Elenco delle figure

1	Fòlaga, veicolo ibrido. . . . .	8
1.1	Area di test per la navigazione di ranging acustico della Fòlaga. . . . .	12
1.2	Traiettoria eseguita dalla Fòlaga. . . . .	13
1.3	La traiettoria della Fòlaga (da GPS) e l'andamento delle coordinate North e East . . . . .	16
1.4	L'andamento nel tempo del <i>range</i> e dell'angolo di bussola <i>theta</i> . . . . .	17
1.5	L'andamento nel tempo della velocità di <i>surge</i> e <i>sway</i> . . . . .	17
1.6	La traiettoria della Fòlaga (da GPS) e l'andamento delle coordinate North e East, per $Time \leq 7400s$ . . . . .	18
1.7	L'andamento nel tempo del <i>range</i> e dell'angolo di bussola <i>theta</i> per $Time \leq 7400s$ . . . . .	18
1.8	L'andamento nel tempo della velocità di <i>surge</i> e <i>sway</i> per $Time \leq 7400s$ . . . . .	19
1.9	Confronto fra il range rilevato dal master modem e il range generato dai dati GPS convertiti in NED; l'andamento dell'errore tra i due range, $Time \leq 7400s$ . . . . .	20
1.10	Verifica di traiettoria costante. . . . .	21
1.11	L'andamento dell'angolo di bussola <i>Theta</i> nel tempo. . . . .	21
1.12	L'andamento di <i>Theta</i> dopo aver riportato i $360^\circ$ sullo zero. . . . .	22
1.13	L'andamento di <i>Theta</i> filtrata con una media mobile. . . . .	23
1.14	L'andamento della velocità di <i>surge</i> e <i>sway</i> ricalcolate con l'angolo <i>theta</i> filtrato. . . . .	23
1.15	L'andamento della velocità di <i>surge</i> e <i>sway</i> filtrate con media mobile. . . . .	24
2.1	Le coordinate geodetiche $\lambda$ e $\varphi$ , rispettivamente longitudine e latitudine, le coordinate ECEF e le coordinate locali NED. . . . .	26
2.2	Coordinate body-frame e angoli RPY di un veicolo acquatico. . . . .	26
4.1	Confronto fra la traiettoria da GPS e quella stimata tramite l'EKF. . . . .	41
4.2	Andamento nel tempo dell'errore di stima (EKF). . . . .	42
4.3	Traiettoria stimata con l'EKF, durante una curva o una manovra a bassa velocità. . . . .	42

4.4	Andamento dell'errore di stima(EKF), durante una curva o una manovra a bassa velocità. . . . .	43
4.5	Traiettoria stimata con l'EKF, durante manovre a velocità più elevata. .	43
4.6	Andamento dell'errore di stima (EKF), durante manovre a velocità più elevata . . . . .	44
5.1	Confronto fra la traiettoria da GPS e quella stimata tramite il problema di ottimo. . . . .	49
5.2	Andamento dell'errore di stima (Problema di ottimo). . . . .	49
5.3	Andamento delle velocità di surge e sway calcolate dal problema di ottimo.	50
6.1	Confronto fra la traiettoria stimata dall'EKF e quella stimata tramite il problema di ottimo. . . . .	52

# Introduzione

L'utilizzo della robotica mobile nei diversi settori delle attività umane ha avuto un notevole sviluppo negli ultimi anni. Generalmente alcuni ambienti operativi possono risultare ostili all'intervento umano (i.e. ambienti spaziali, sottomarini, nucleari e militari), in questi particolari ambienti il robot è solo, in quanto non sono possibili interventi di operatori umani. Nasce quindi l'esigenza di un robot autonomo e affidabile nell'esecuzione dei task. Per poter essere definito autonomo, un robot mobile deve essere in grado di navigare nell'ambiente circostante, ricostruire un modello dell'ambiente ed eseguire il proprio task interagendo con l'ambiente.

Durante la navigazione quindi il robot autonomo deve potersi localizzare, ovvero deve conoscere la propria posizione in un ambiente noto, a partire da dati sensoriali.

I robot mobili utilizzati in ambito marino si possono classificare per tipologia e applicazioni. La principale distinzione fra i veicoli viene fatta a seconda della loro profondità di lavoro:

- veicoli per profondità *abissali*, in genere di grosse dimensioni, progettati per missioni a largo raggio (anche centinaia di km) e di lunga durata;
- veicoli per profondità *medio-basse* (profondità  $< 500$  m), di piccole dimensioni, progettati per missioni a corto raggio (fino a qualche decina di km) e di durata limitata (8 - 12 ore).

Di questi ultimi si può fare un'ulteriore distinzione:

- AUV tradizionali;
- glider oceanografici;
- sistemi ibridi.

Gli AUV "tradizionali" sono veicoli in genere siluriformi dotati di propulsori a elica, a volte orientabili, e di superfici idrodinamiche che ne consentono la navigazione e il controllo a velocità superiori di 1.5 nodi. Essi non hanno possibilità di hovering, ovvero di fermarsi su un punto ad una quota prestabilita.

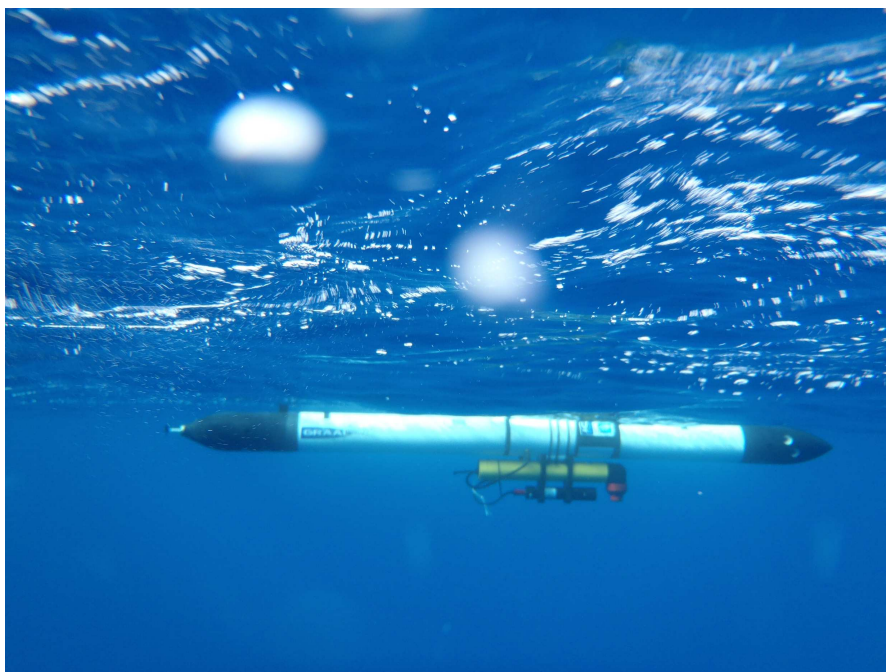


Figura 1: Fòlaga, veicolo ibrido.

I "glider" invece sono AUV particolari sviluppati per la ricerca oceanografica a lungo termine. Si tratta di veicoli privi di propulsione, che si muovono sfruttando il principio dell'aliante. Modificando opportunamente la galleggiabilità e l'assetto del veicolo, e sfruttando le proprie superfici idrodinamiche (ali e timone), riescono a spostarsi fra diverse profondità, riuscendo ad ottenere un movimento orizzontale netto nella direzione desiderata. Le modifiche di assetto e galleggiabilità richiedono un dispendio energetico molto minore di quello richiesto da un motore propulsivo, per cui a parità di batterie i glider hanno una autonomia di funzionamento molto maggiore di quella di qualsiasi AUV tradizionale. Ovviamente, i glider si muovono con velocità orizzontale molto limitata (0.2-0.4 nodi), e con stabilità della traiettoria fortemente influenzata dalle correnti.

Particolare interesse dal punto di vista della ricerca ma anche dell'oceanografia operativa hanno destato negli ultimi anni veicoli ad "attuazione ibrida", che presentano alcune caratteristiche simili a quelle dei glider ed alcune simili agli AUV standard. Un veicolo ibrido è appunto il veicolo subacqueo la "Fòlaga" utilizzato nei nostri esperimenti. Il veicolo si muove in superficie con dei propulsori standard e si immerge con un'attuazione simile a quella dei glider. Ha una lunghezza di ca. 200 cm, un diametro di ca. 140 mm e un peso in aria di ca. 30 Kg. La sua profondità operativa arriva fino ai 50m ed arriva a una velocità massima di superficie tra 2-4 nodi.



Un aspetto particolarmente interessante della Fòlaga è che non utilizza superfici idrodinamiche per il controllo di assetto e la propulsione: il veicolo è quindi composto unicamente da un corpo cilindrico, senza pinne o timoni. Inoltre, il veicolo ha la capacità di hovering, ed una manovrabilità a bassa velocità in genere molto maggiore di quella degli AUV standard. Il veicolo non è però dotato di appropriati sensori per la navigazione subacquea (DVL, sensori inerziali), che ne consentano la precisa georeferenziazione, essendo prevista la navigazione in superficie guidata da GPS.

Per far fronte a questo aspetto, nella seguente tesi è stato sviluppato un sistema di localizzazione assoluta per un robot autonomo. In generale i metodi di localizzazione assoluta riescono a fondere le informazioni di ranging acustico con i dati acquisiti dai sensori di navigazione.

Nello specifico vengono implementati due metodi di stima della posizione assoluta del veicolo, il filtro di Kalman esteso e un problema di ottimizzazione vincolata. I dati che si hanno a disposizione per le prove di simulazione sono stati acquisiti il giorno 22 settembre 2010 durante degli esperimenti svolti, dalle ca. *h*09 : 30 fino alle ca. *h*12 : 00 del mattino, attorno all'isola di Pianosa. Tutto il lavoro di tesi è stato svolto in ambiente Matlab, vediamone dunque in dettaglio i passaggi elaborati.

# Capitolo 1

## Descrizione, acquisizione ed elaborazione dei dati

Come abbiamo accennato nell'introduzione, il metodo che si vorrebbe provare è quello di localizzazione assoluta di un veicolo mobile sfruttando le misure di velocità e le misure di ranging acustico. Data la mancanza di un velocimetro a bordo veicolo, nel nostro caso le velocità vengono determinate tramite i dati di posizione GPS.

Le misure GPS inoltre vengono utilizzate per poter validare la stima della posizione effettuata con il ranging acustico.

Il sistema GPS però è soggetto a differenti tipi di errori che influenzano la distanza misurata e quindi la posizione stimata. Le principali cause di errore sono solitamente l'imperfezione delle orbite dei satelliti, l'errore degli orologi satellitari e gli effetti della ionosfera sul segnale.

È necessario quindi effettuare un processo di elaborazione dei dati cercando di:

- identificare e rimuovere i disturbi sui segnali,
- eliminare eventuali outlier nelle osservazioni.

Inoltre anche la comunicazione acustica subacquea è soggetta a rumore dovuto i.e. a effetti multi-path, ritardi di trasmissione ed altri fattori.

Viene quindi introdotta l'analisi off-line dei dati registrati il giorno in cui si sono svolte le prove sperimentali.

## 1.1 Descrizione e acquisizione dati

Le prove svolte durante il giorno 22 Settembre dell'anno 2010, sono state realizzate appositamente per valutare le capacità di localizzazione della Fòlaga. Le condizioni meteorologiche prevedevano mare calmo e assenza di vento per quasi tutta la giornata, con vento delicato nel pomeriggio.

Il master modem, posizionato a 10 m di profondità ed in posizione fissa

S1: Lat 42.5940 N, Lon 10.1060 E

ha comunicato per tutta la durata dell'esperimento con lo slave modem posizionato sulla Fòlaga, misurando la sua distanza da esso.

Inizialmente la Fòlaga, quasi sempre in superficie, ha seguito una traiettoria circolare lungo i punti attorno al master modem

A1: Lat 42.5930 N, Lon 10.1055 E

A2: Lat 42.5943 N, Lon 10.1039 E

A3: Lat 42.5951 N, Lon 10.1065 E

A4: Lat 42.5936 N, Lon 10.1081 E

per poi finire con una traiettoria a zig-zag, scendendo anche ad una profondità di 5 m (fig.1.1).

nell'ultimo tratto invece esegue un percorso a zig-zag, scendendo anche ad una profondità di 5 m (fig.1.2).

Le misure di range sono state registrate in un file di dati in formato excel. Nel dettaglio il file 'Folaga Range Test 22nd Sept.xls' ha la seguente forma

UTC Time hhmmss.ss	Channel	Range(m)	Time of flight(s)
071851.902	M03	214.50	0.144636
071854.904	M03	214.91	0.144915
⋮	⋮	⋮	⋮

Utilizzando la funzione *extract\_xls.m* (con un sistema operativo windows) si posso-

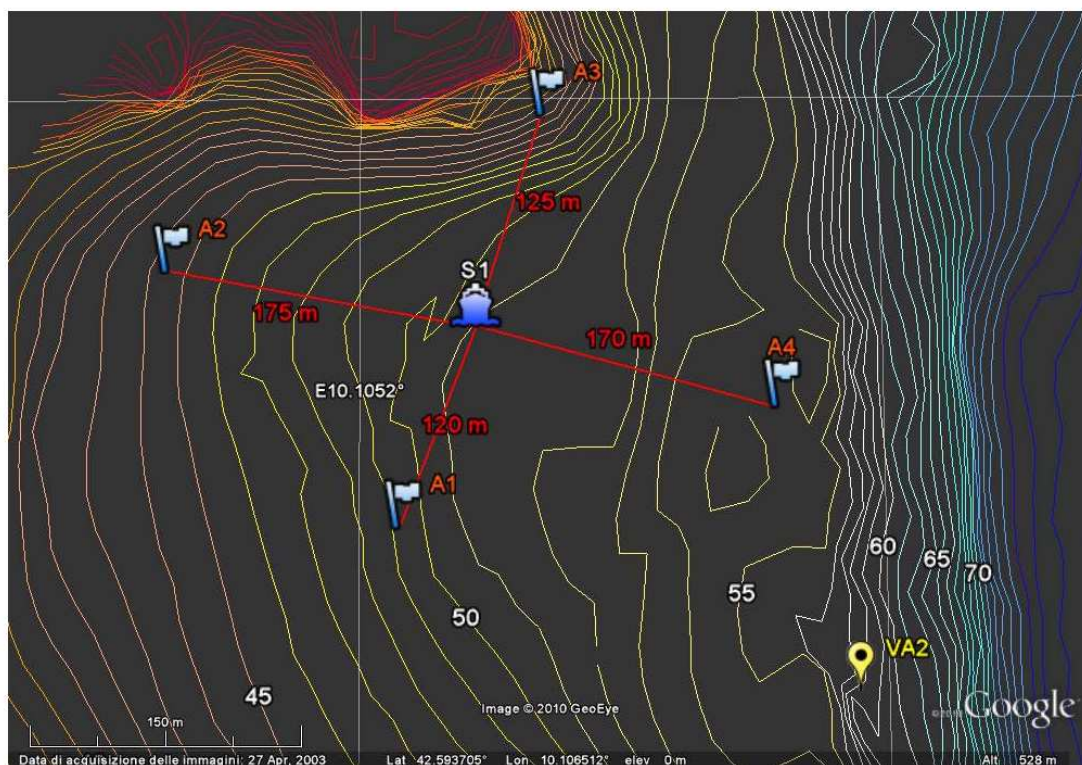


Figura 1.1: Area di test per la navigazione di ranging acustico della Fòlaga; A1, A2, A3 e A4 sono i punti lungo i quali si sposta la Fòlaga; S1 è la posizione del master modem.

no estrarre la prima e la seconda colonna della tabella, rispettivamente il vettore *time* e il vettore *r*. Una volta estratti i dati contenuti nel file di log del modem, vengono selezionati quelli compresi nell'intervallo di tempo d'interesse che va da un *timestart* = '09:18:51' fino a un *timeend* = '12:00:00', l'ora è in CEST (UTC +2h). La funzione *data\_modem.m* esegue l'estrazione dei dati scorrendo il vettore *time* e memorizzando gli indici dei valori all'interno dell'array coincidenti con gli orari di *timestart* e *timeend*, rispettivamente indice di inizio estrazione dati e indice di fine estrazione dati. A questo punto è possibile avere in uscita il vettore delle distanze *flg\_R* e il vettore delle ore in UTC *time\_utc* registrati nell'intervallo di tempo d'interesse.

Infine chiamando la funzione *sample\_time.m* si ricava il tempo di campionamento del modem.

I dati di navigazione GPS della Fòlaga sono stati registrati nel file 'folaga\_21e22\_navigation.log'. Tale file è stato suddiviso in più file a seconda del giorno e dell'orario per semplificare l'estrazione dei dati e renderla meno pesante. Il sotto-file di nostro in-

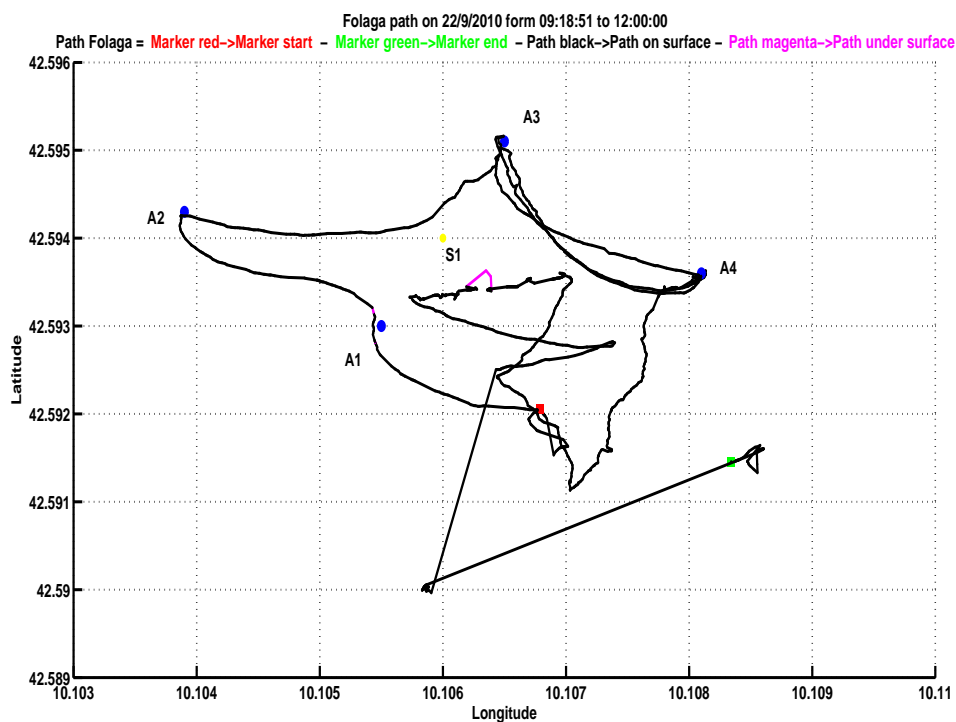


Figura 1.2: Traiettoria eseguita dalla Fòlaga; A1, A2, A3 e A4 sono i punti lungo i quali si sposta la Fòlaga; S1 è la posizione del master modem.

teresse è stato nominato 'Sept\_22\_a.log' e i dati al suo interno sono relativi alla prova eseguita durante la mattinata del 22 settembre. Nel seguito viene visualizzato un esempio di riga del file di log (in blu) con successiva tabella che identifica il significato di ogni variabile:

SYS: T= +00.000 C= +00.000 D=000 S=060 P=005 g=-397 b=549 B=038 I=023  
H=-100 V=-1.00 42.5933500 10.1051305 0.0000000 0.0000000 t=3.0000

<b>T</b>	temperatura[°C]
<b>C</b>	conduttività[mS/cm]
<b>D</b>	profondità[m]
<b>S</b>	angolo di bussola[°]
<b>P</b>	angolo di beccheggio[°]
<b>g</b>	quantità d'acqua incamerata dalla boa (rispetto al neutro) [gr]

<b>b</b>	pressione camera di allagamento [mbar]
<b>B</b>	posizione batterie rispetto al finecorsa (posteriore 0) [mm]
<b>I</b>	temperatura interna [°C]
<b>H</b>	umidità
<b>V</b>	voltaggio batteria [V]
<b>t</b>	secondi trascorsi dal 1970 [s]

dei quattro valori non assegnati a nessuna variabile i primi due identificano rispettivamente la latitudine e la longitudine. I dati vengono estratti chiamando la funzione *gps\_data.m*. In particolare vengono estratti l'angolo di bussola THETA, la latitudine e la longitudine LATLON, e la profondità DEPTH:

```
[status, THETA]=system(sprintf('cat %s | grep "t=%u\\.0" | ...
...grep "SYS:" | awk -F\\'\'S=\\'\'\\'\'{print $2}\\'\'|...
...awk -F\\'\' \\'\' \\'\'{print $1}\\'\'',filename,t));
[status, LATLON]=system(sprintf('cat %s | grep "t=%u\\.0" | ...
...grep "SYS:" | awk -F\\'\'V=\\'\'\\'\'{print $2}\\'\'|...
...awk -F\\'\'\\'\'\\'\'{print $2" "$3}\\'\'',filename,t));
[status, DEPTH]=system(sprintf('cat %s | grep "t=%u\\.0" | ...
...grep "SYS:" | awk -F\\'\'D=\\'\'\\'\'{print $2}\\'\'|...
...awk -F\\'\'\\'\'\\'\'{print $1}\\'\'',filename,t));
```

Viene poi verificato se sono presenti elementi vuoti o valori non significativi all'interno degli array estratti. Se esistono elementi vuoti per THETA, LATLON e DEPTH e se questi si trovano all'interno dell'array allora l'elemento *i*-esimo viene sostituito con il valore precedente, (*i* − 1)-esimo; se invece si trovano all'inizio dell'array, i vuoti vengono sostituiti dal primo valore disponibile trovato scorrendo l'indice dell'array. Allo stesso modo viene verificato se all'interno dell'array LATLON esistono dei valori non significativi dovuti ad errori del GPS.

Una volta estratti i dati, il passo successivo consiste nel trasformare le coordinate da geodetiche a NED (v. capitolo seguente). Tramite la funzione *geodtoned.m*, avente

come ingresso il vettore LATLON, si ottengono in uscita le coordinate nel frame NED. Si possono quindi calcolare le velocità north ed east tramite rapporto incrementale e quindi proiettare le stesse nel frame body mediante la matrice di rotazione.

Tutte le funzioni di cui si è discusso vengono lanciate attraverso una funzione main che è stata chiamata *DATASET.m*. In essa vengono inizializzati i valori del *timestart*, *timeend*, *DATE*, e il nome del file *filename* dal quale si vogliono prelevare i dati GPS.

I dati vengono salvati nella matrice *M\_data*.

```
M_data = [time' Tc' flg_R N E D theta surge sway];
```

É stata inoltre fatta un apposita funzione che mostra in uscita i dati raccolti nel dataset. Digidanto *ShowDataset* nel workspace di Matlab vengono mostrate le figure (1.3), (1.4) e (1.5).

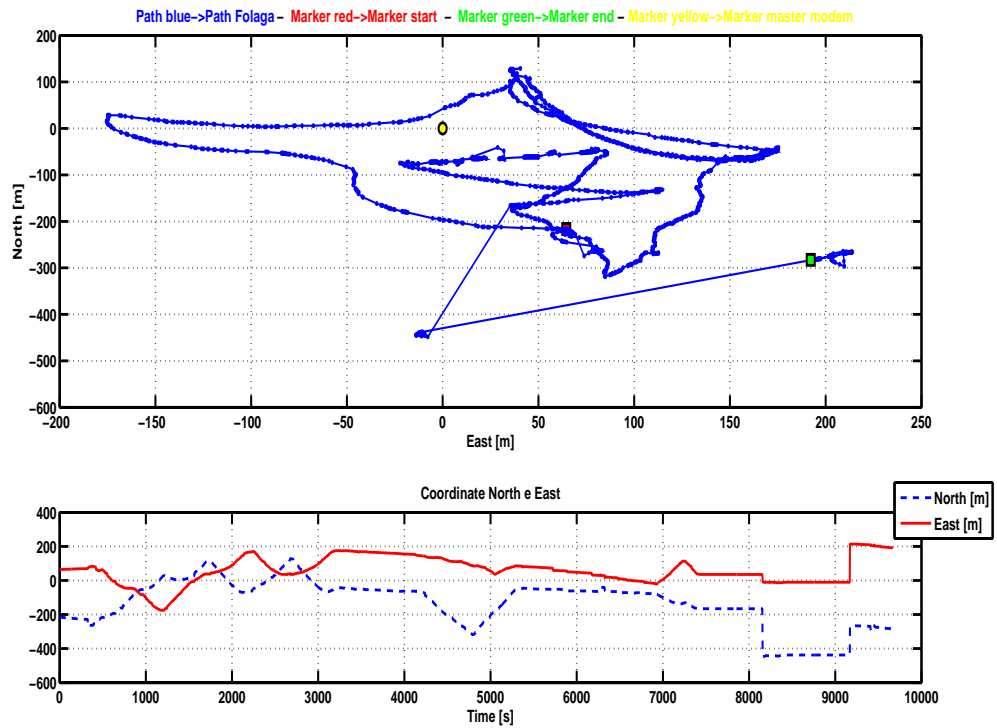


Figura 1.3: In alto, la traiettoria della Fòlaga (da GPS); in basso, l'andamento delle coordinate North e East nel tempo.

La mancanza di punti intermedi per un tempo  $Time \geq 7400s$  (fig.(1.3)), è dovuta al fatto che il veicolo, da quel momento in poi, si è immerso sotto i 10 m e quindi non vi è lettura GPS. Da qui in avanti verranno considerati i dati della matrice  $M\_data$  compresi nell'intervallo  $0 \leq Time \leq 7400s$ . Vengono quindi visualizzate le seguenti figure (1.6), (1.7) e (1.8).



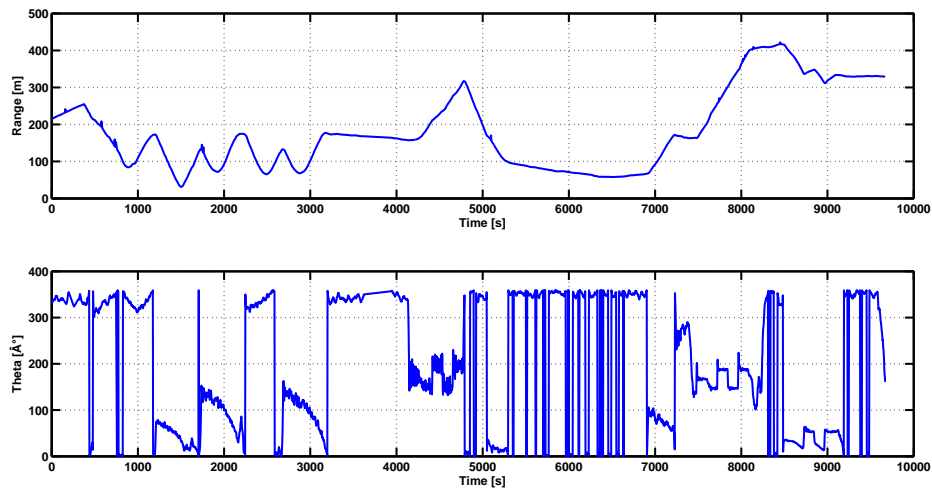


Figura 1.4: In alto, l'andamento della distanza della Fòlaga dal master modem (Range); in basso, l'andamento dell'angolo di orientamento della Fòlaga (Theta).

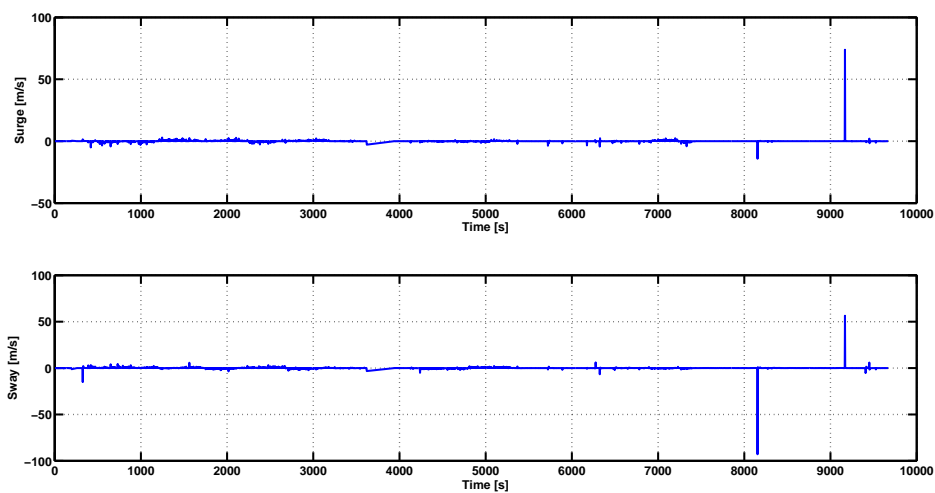


Figura 1.5: In alto, l'andamento della velocità di surge nel tempo, in basso l'andamento della velocità di sway.

Gli andamenti delle velocità (fig.(1.8)) non son reali in quanto presentano dei picchi che arrivano fino al valore di  $5m/s$ , mentre il veicolo non supera il  $1,5m/s$ . Nel capitolo seguente si cerca di identificare e rimuovere eventuali disturbi sui segnali.

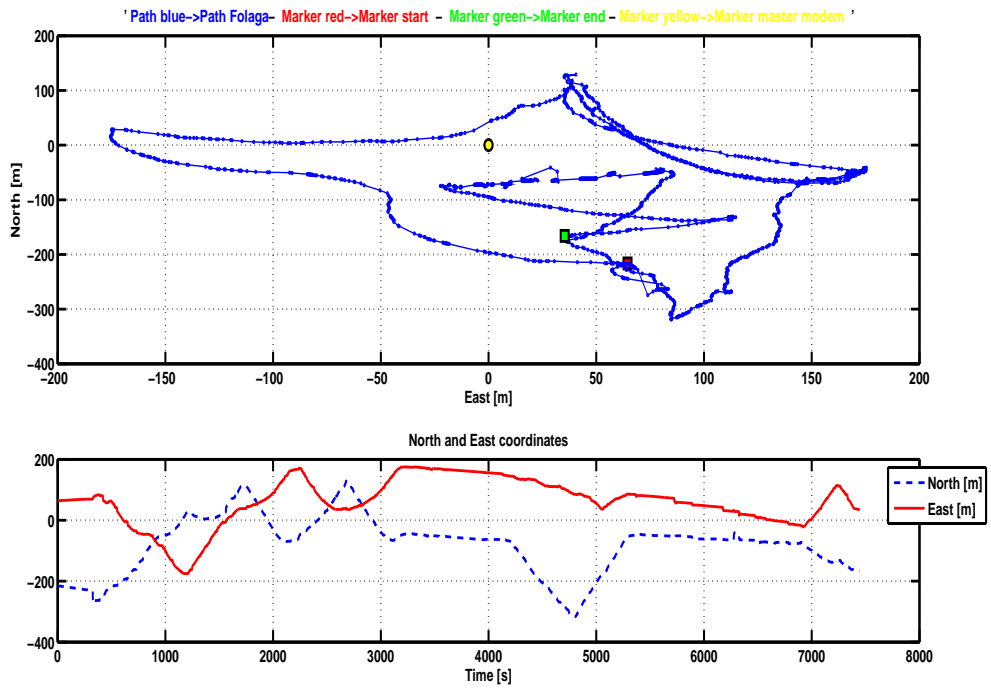


Figura 1.6: In alto, la traiettoria della Fòlaga (da GPS); in basso, l'andamento delle coordinate North e East nel tempo ( $Time \leq 7400s$ ).

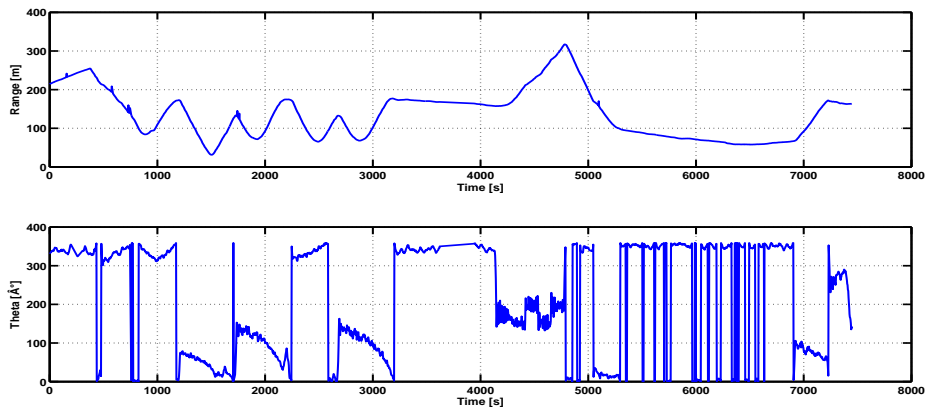


Figura 1.7: In alto, l'andamento della distanza della Fòlaga dal master modem (Range); in basso, l'andamento dell'angolo di orientamento della Fòlaga (Theta).

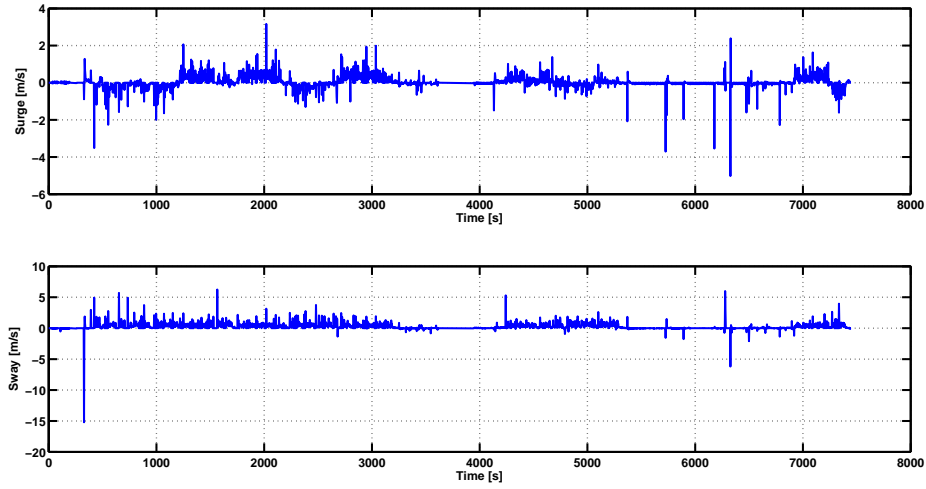


Figura 1.8: In alto l'andamento della velocità di surge nel tempo, in basso l'andamento della velocità di sway.

## 1.2 Rielaborazione dei dati

Talvolta la conversione delle coordinate da geodetiche a NED può generare bias. Questo perché trasformando le coordinate da geodetiche a ECEF viene utilizzato il raggio della terra, un valore molto alto ed anche molto incerto, quindi potrebbero amplificarsi i disturbi già presenti nei dati GPS. Si è quindi effettuato un test per verificare la validità del range ricavato mediante coordinate NED  $Range_{NED}$ , mettendolo a confronto con quello rilevato acusticamente  $Range$ , dove:

$$Range_{NED} = \sqrt{N^2 + E^2}$$

e l'errore tra i range è dato da:

$$Errore = Range - Range_{NED}$$

Di seguito viene mostrato l'andamento del range nei due casi e l'errore che viene commesso (fig.(1.9)).

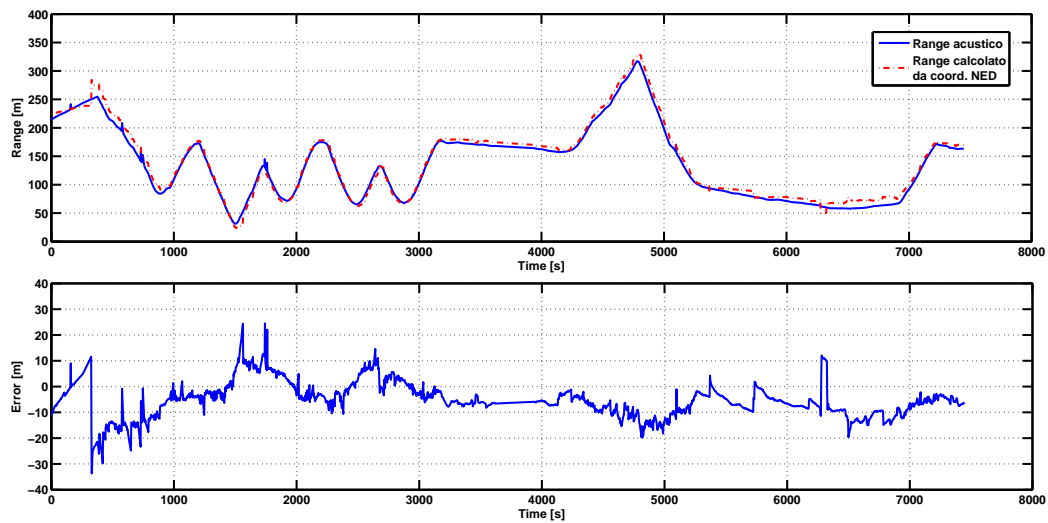


Figura 1.9: In alto, in blu l'andamento del range rilevato dal modem master, in rosso l'andamento del range generato dai dati GPS convertiti in NED; In basso, l'andamento dell'errore tra i due range.

Si può verificare che non vi è presenza di bias e l'errore rimane intorno i 10m (la precisione del GPS è di ca. 10m in navigazione), quindi è possibile continuare a utilizzare le coordinate nel sistema NED. In caso contrario si sarebbe utilizzata una conversione alternativa.

Osservando il grafico del range (fig.(1.9), in alto), questo sembrerebbe costante fra i 3000 e i 4000 secondi. Visualizzando nel dettaglio quel lasso di tempo si può effettivamente confermare che il veicolo rimane più o meno fermo nella stessa posizione e che quindi le misure son coerenti. Di seguito ne vengono mostrati i grafici.

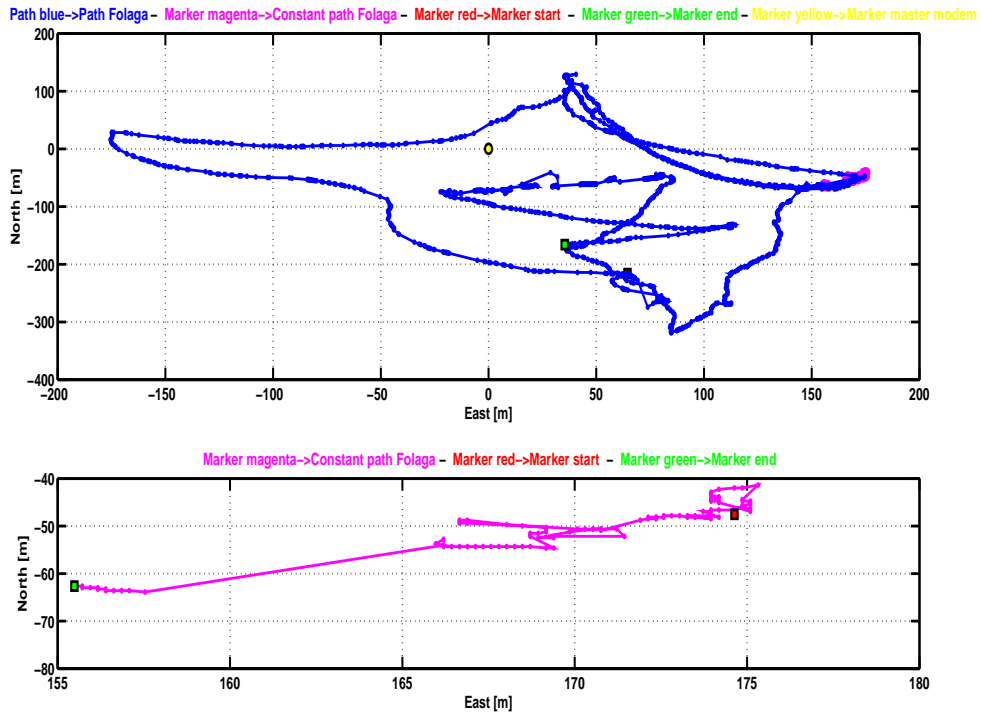


Figura 1.10: In alto, in blu la traiettoria del robot calcolata in NED e in rosso il tratto di traiettoria nell'intervallo di tempo [3200 4000]s; in basso in magenta il tratto di traiettoria nell'intervallo di tempo [3200 4000]s.

Esaminiamo ora l'andamento dell'angolo di bussola  $\theta$ . Ne viene mostrata di seguito la rappresentazione grafica nel tempo:

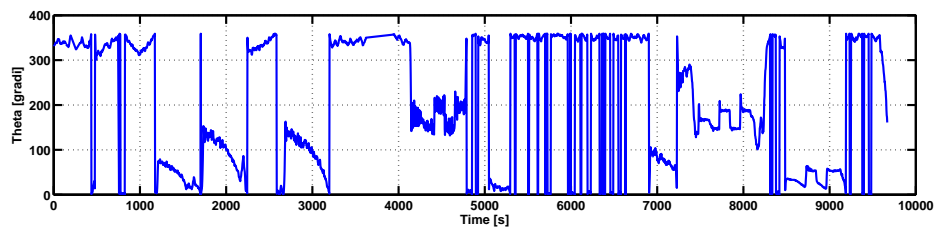


Figura 1.11: L'andamento dell'angolo di bussola  $\theta$  nel tempo.

Si vogliono eliminare i salti dovuti al passaggio da un intorno di  $0^\circ$  a un intorno di  $360^\circ$  (fig.1.11). Per un tempo  $Time \leq 4000s$  si impone un limite di  $\theta$  ( $\theta \leq 300^\circ$ ), per cui se maggiore di esso allora  $\theta_i = \theta_i - 2\pi$ .  $\theta$  diventerebbe una funzione

nel tempo ribaltata sullo zero, come infatti viene mostrato in fig.(1.12).

Analizzando e studiando bene la figura, si può notare come *Theta* sia fortemente ru-

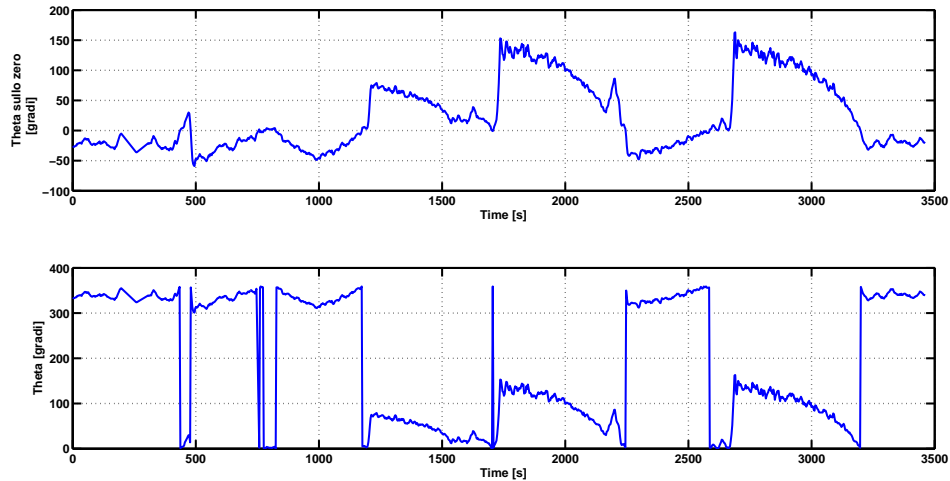


Figura 1.12: In alto, l'andamento di *Theta* dopo aver riportato i  $360^\circ$  sullo zero; in basso, l'andamento dell'angolo di bussola *Theta* nel tempo.

morosa. Applicando una media mobile semplice, con una finestra di 11 campioni alla volta, l'andamento di *theta* viene ammorbidito, eliminando quanto possibile il disturbo introdotto dagli errori di bussola. In fig.(1.13) se ne mostra l'andamento.

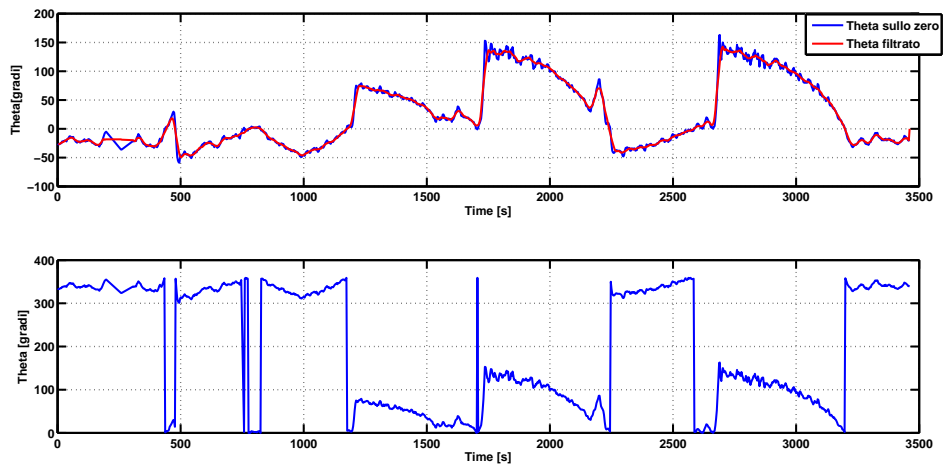


Figura 1.13: In alto, in rosso l'andamento di  $\theta$  dopo essere stata filtrata con una media mobile; in basso, l'andamento dell'angolo di bussola  $\theta$  nel tempo.

A questo punto sostituendo il nuovo angolo  $\theta_{filt}$  nella matrice di rotazione, si possono riproiettare la velocità north e east nel frame body, ottenendo quelle di surge e di sway. Limitandosi alla porzione di dati fra i 1000 e i 3500 secondi circa si ha:

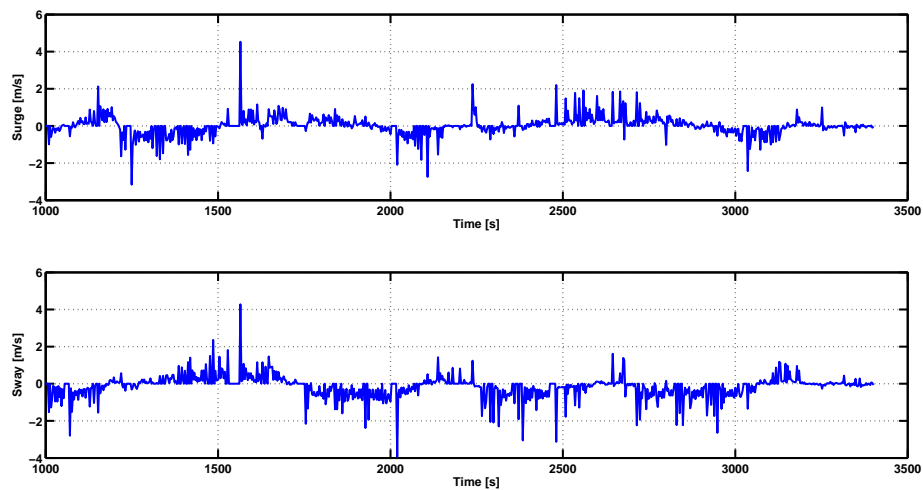


Figura 1.14: In alto, l'andamento della velocità di surge ricalcolata con il nuovo angolo di orientamento; in basso l'andamento della velocità di sway ricalcolata con il nuovo angolo di orientamento.

Come si è già accennato nella realtà il veicolo arriva ad una velocità non superiore al  $1,5m/s$ . Per limitare le velocità entro questo limite esse vengono filtrate con la media mobile semplice, in questo caso la finestra avrà 7 campioni (fig.(1.15)).

Questi dati rielaborati vengo salvati in una nuova matrice di log. In particolare vengono

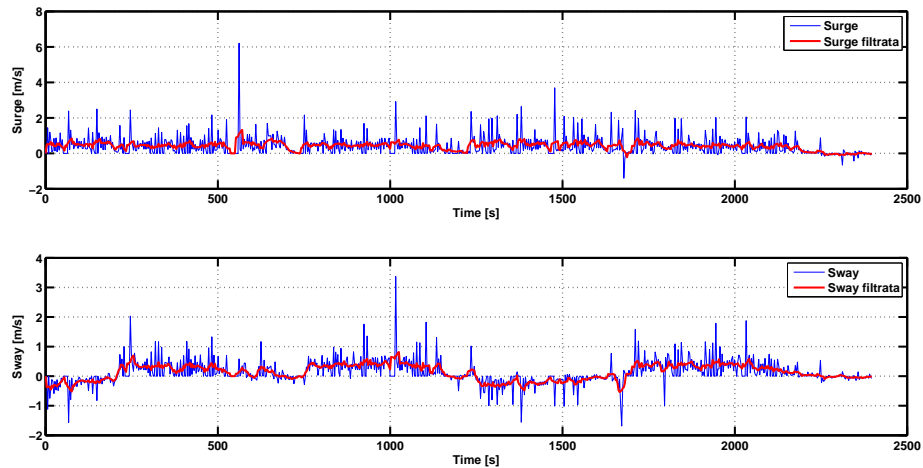


Figura 1.15: In alto, in blu l'andamento della velocità di surge ricalcolata con il nuovo angolo di orientamento, in rosso la velocità di surge filtrata con la media mobile ; in basso, in blu l'andamento della velocità di sway ricalcolata con il nuovo angolo di orientamento, in rosso la velocità di sway filtrata con la media mobile.

presi i dati compresi nell'intervallo di tempo fra 1000 e i 3500 secondi, poiché si ritiene siano quelli meno affetti da incertezza.



# Capitolo 2

## Sistemi di riferimento e trasformazione di coordinate

### 2.1 Introduzione

Esistono diversi tipi di sistemi di coordinate (o frame) usati nello studio e analisi della navigazione, della guida e del controllo dei veicoli autonomi [1]. In questo capitolo verranno introdotti i sistemi di coordinate adottati nel seguente lavoro, che comprendono:

- il sistema di coordinate geodetiche,
- il sistema di coordinate ECEF (earth-centered earth-fixed),
- il sistema di coordinate NED (north-east-down),
- il sistema body-frame del veicolo.

Inoltre vengono espone le relazioni che intercorrono tra questi sistemi di coordinate. In tale contesto è necessario sottolineare che i veicoli AUV sono normalmente utilizzati a bassa velocità e in regioni limitate, per questo motivo è possibile applicare la conversione fra sistema geodetico e NED. Per capire meglio l'importanza della precedente affermazione, tenendo conto che la terra viene approssimata ad un ellissoide, si pensi ad una regione limitata attorno ad un punto qualsiasi del globo terrestre, più questa regione si allargherà più la conversione fra sistemi di riferimento si allontanerà dalla realtà. Nel seguito vengono mostrati graficamente i sistemi di coordinate adottati (fig.(2.1) e fig.(2.2)).

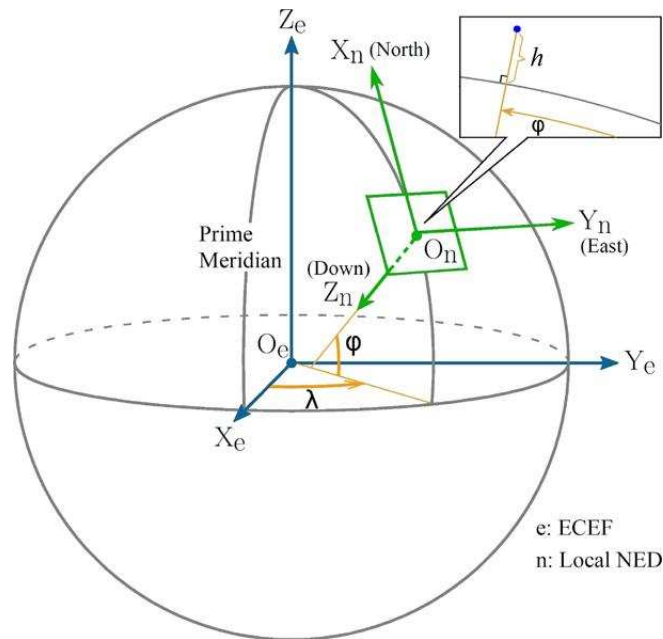


Figura 2.1: Le coordinate geodetiche  $\lambda$  e  $\phi$ , rispettivamente longitudine e latitudine, le coordinate ECEF e le coordinate locali NED.

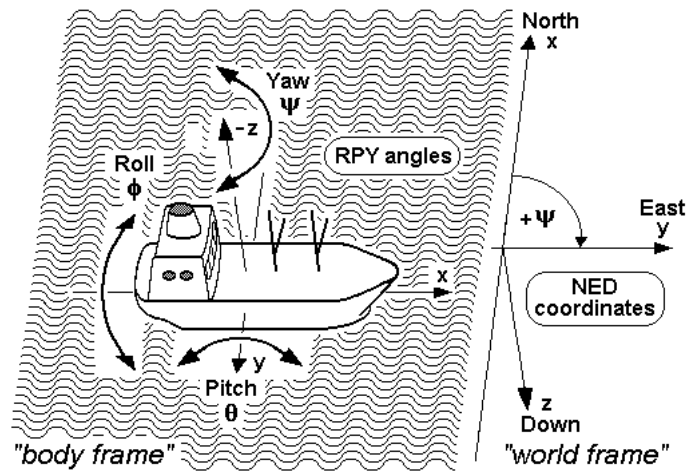


Figura 2.2: Coordinate body-frame e angoli RPY di un veicolo acquatico.

## 2.2 Sistemi di riferimento e convenzioni

### 2.2.1 Sistema geodetico

Il sistema di coordinate geodetico è ampiamente usato nella navigazione GPS. Si nota che non è un sistema di coordinate cartesiane usuale, ma un sistema che rappresenta un punto di coordinate vicino alla superficie terrestre in termini di longitudine, latitudine e altezza, che sono rispettivamente indicate con  $\lambda$ ,  $\varphi$ ,  $h$ . La longitudine misura l'angolo di rotazione (che varia da  $-180^\circ$  a  $180^\circ$ ) tra il Meridiano Principale e il punto d'interesse, la latitudine invece misura l'angolo (che varia da  $-90^\circ$  a  $90^\circ$ ) tra il piano equatoriale e la normale all'ellissoide di riferimento passante nel punto d'interesse, infine l'altezza è la distanza locale verticale tra il punto e l'ellissoide di riferimento. Il vettore posizione del punto d'interesse espresso nel frame geodetico viene denotato come segue:

$$P_g = \begin{pmatrix} \lambda \\ \varphi \\ h \end{pmatrix} \quad (2.1)$$

Assumendo un sistema di riferimento cartesiano, di assi X, Y, Z, e con origine O situata esattamente o approssimativamente nel centro della Terra, si può definire datum geodetico l'insieme delle informazioni che definiscono univocamente

- la forma,
- la posizione,
- l'assetto

della terra. Si adotta l'ellissoide come approssimazione matematica semplificata del geoide, definita univocamente da 8 parametri indipendenti:

- *2 parametri di forma* dell'ellissoide (dati generalmente dal semiasse maggiore  $a$  e dall'eccentricità  $e$  o, in alternativa, dal semiasse minore  $b$ ),
- *3 parametri di posizione* dell'ellissoide (coordinate del centro dell'ellissoide),
- *3 parametri di assetto* (angoli di direzione dell'ellissoide).

Tra parametri più importanti associati al frame geodetico si includono:

1. il semi-asse maggiore  $a$ ,
2. il semi-asse minore  $b$ ,
3. l'eccentricità  $e$ ,
4. il raggio di curvatura nel primo verticale  $N_E$ .

Questi parametri sono definiti o derivano sulle basi del modello dell'ellissoide WGS84 (World Geodetic System 1984, che è stato proposto originariamente nel 1984 e poi modificato successivamente nel 2004). Il WGS84 è un datum geodetico, geocentrico utilizzato per le misure GPS su tutta la superficie terrestre. I parametri associati a tale ellissoide vengono esplicitati come segue:

$$a = 6,378,137.0m, \quad (2.2)$$

$$b = 6,356,752.0m, \quad (2.3)$$

$$e = \frac{\sqrt{a^2 - b^2}}{a} = 0.08181919. \quad (2.4)$$

$$N_E = \frac{a}{\sqrt{1 - e^2 \sin^2 \varphi}} \quad (2.5)$$

### 2.2.2 Sistema di riferimento ECEF

Il sistema di coordinate ECEF ruota con la terra attorno al suo asse di rotazione z. Come tale, un punto fisso sulla superficie della terra avrà un set di coordinate fisse. L'origine e gli assi del sistema di coordinate ECEF (Fig.2.1) sono definite come segue:

1. L'origine (indicata con  $O_e$ ) si trova al centro della terra.
2. L'asse Z (indicato con  $Z_e$ ) è lungo l'asse di rotazione della terra, verso il polo nord.
3. L'asse X (indicato con  $X_e$ ) interseca la terra alla latitudine  $0^\circ$  e longitudine  $0^\circ$ .
4. L'asse Y (indicato con  $Y_e$ ) è ortogonale all'asse Z e l'asse X secondo la regola della mano destra.

Il vettore di coordinate espresso nel sistema di riferimento ECEF viene indicato con il pedice  $e$ , ed è dunque:

$$P_e = \begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix} \quad (2.6)$$

### 2.2.3 Sistema di riferimento NED

Il sistema di riferimento NED è quello in base al quale si calcola l'assetto del veicolo. È un sistema riferito alla geodesia, che può avere la sua origine coincidente con quella del sistema body. Basato sul modello dell'ellissoide WGS84, il sistema NED è costituito nel seguente modo:

1. L'origine (indicata con  $O_n$ ) è fissata arbitrariamente su un punto della superficie della terra.
2. L'asse Z (indicato con  $Z_n$ ) è ortogonale all'ellissoide di riferimento e diretto verso il centro della Terra;
3. L'asse X (indicato con  $X_n$ ) è diretto verso il nord geodetico;
4. L'asse Y (indicato con  $Y_n$ ) è diretto verso l'est geodetico ed ortogonale agli altri due.

Il sistema locale NED gioca un ruolo importante nel controllo di volo e di navigazione. Questo sistema viene spesso utilizzato nei rilievi GPS per esprimere posizioni tridimensionali o altre grandezze vettoriali (baselines, residui, spostamenti, ecc.) in un sistema che permette di scindere (anche se non rigorosamente) la componente altimetrica da quelle planimetriche. Le coordinate cartesiane locali, in un intorno dell'origine, corrispondono praticamente alle coordinate che si potrebbero misurare sul piano tangente, e quindi risultano particolarmente utili per tracciamenti in cantiere e altri problemi ingegneristici. Ovviamente, un riferimento euleriano può essere utilizzato solo in un limitato intorno dell'origine, quindi la navigazione su piccola scala viene normalmente effettuata entro questo frame. Il vettore di coordinate espresso nel sistema di coordinate locale NED viene indicato con il pedice  $n$ . Più specificatamente, il vettore posizione  $P_n$  e il vettore velocità  $V_n$ , sono definiti come

$$P_n = \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} \quad V_n = \begin{pmatrix} u_n \\ v_n \\ \omega_n \end{pmatrix} \quad (2.7)$$

#### 2.2.4 Sistema di riferimento body

Il sistema di riferimento body è quello solidale al veicolo. In un autoveicolo gli assi di riferimento del sistema body sono disposti nel seguente modo (come mostrato in fig.(2.2)):

1. L'origine (indicata con  $O_b$ ) è fissata nel baricentro del veicolo;
2. L'asse Z (indicato con  $Z_b$ ) è diretto verso l'alto (o il basso dipendentemente dalla convenzione adottata) ed è posizionato in modo da essere ortogonale agli altri due;
3. L'asse X (indicato con  $X_b$ ) è diretto longitudinalmente al veicolo ed è orientato nel verso di marcia normale del veicolo;

4. L'asse Y (indicato con  $Y_b$ ) è diretto trasversalmente al veicolo, ha direzione uscente dal lato destro ed è ortogonale agli altri due.

Il vettore di coordinate espresso nel sistema body viene indicato con il pedice  $b$ . Il vettore posizione  $P_b$  e il vettore velocità  $V_b$  vengono definiti come:

$$P_b = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad V_b = \begin{pmatrix} u \\ v \\ \omega \end{pmatrix} \quad (2.8)$$

## 2.3 Trasformazioni di coordinate

Vengono presentate nel seguito le trasformazioni necessarie fra i sistemi di coordinate adottati.

### 2.3.1 Trasformazione di coordinate da geodetiche a ECEF

La trasformazione del vettore posizione da sistema geodetico a sistema di coordinate ECEF è un passaggio intermedio per la conversione delle misure del GPS al sistema di coordinate NED. Dato un punto nel sistema geodetico:

$$P_g = \begin{pmatrix} \lambda \\ \varphi \\ h \end{pmatrix} \quad (2.9)$$

le sue coordinate nel sistema ECEF sono:

$$P_e = \begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix} = \begin{pmatrix} (N_E + h)\cos(\varphi)\cos(\lambda) \\ (N_E + h)\cos(\varphi)\sin(\lambda) \\ (N_E(1 - e^2) + h)\sin(\varphi) \end{pmatrix}, \quad (2.10)$$

dove  $N_E$  ed  $e$  sono dati rispettivamente dalle equazioni 2.5 e 2.4.

### 2.3.2 Trasformazione di coordinate da ECEF a NED

Dato il vettore posizione  $P_e$  in coordinate ECEF, le sue coordinate nel sistema locale NED sono:

$$P_n = R_e^n(P_e - P_{e,ref}) \quad (2.11)$$

dove,  $p_{e,ref}$  è la posizione dell'origine del sistema di riferimento NED (i.e.  $O_n$ , nel nostro caso è il punto in cui è posizionato il master modem) nel sistema di coordinate

ECEF, e  $R_e^n$  è la matrice di rotazione dal frame ECEF al frame NED. Nello specifico si hanno tre rotazioni:

1. Rotazione di  $\lambda$  attorno all'asse  $Z_e$ ,
2. Rotazione di  $-\varphi$  attorno all'asse  $Y_e$ ,
3. Rotazione di  $-90^\circ$  attorno al nuovo asse  $Y_n$

ovvero la matrice di rotazione è:

$$\begin{aligned}
 R_e^n &= R(-90^\circ)R(-\varphi)R(\lambda) \\
 &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos(\varphi_{ref}) & 0 & \sin(\varphi_{ref}) \\ 0 & 1 & 0 \\ -\sin(\varphi_{ref}) & 0 & \cos(\varphi_{ref}) \end{bmatrix} \begin{bmatrix} \cos(\lambda_{ref}) & \sin(\lambda_{ref}) & 0 \\ -\sin(\lambda_{ref}) & \cos(\lambda_{ref}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} -\sin(\varphi_{ref})\cos(\lambda_{ref}) & -\sin(\varphi_{ref})\sin(\lambda_{ref}) & \cos(\varphi_{ref}) \\ -\sin(\lambda_{ref}) & \cos(\lambda_{ref}) & 0 \\ -\cos(\varphi_{ref})\cos(\lambda_{ref}) & -\cos(\varphi_{ref})\sin(\lambda_{ref}) & -\sin(\varphi_{ref}) \end{bmatrix}
 \end{aligned} \tag{2.12}$$

dove  $\varphi$  e  $\lambda$  sono la longitudine e la latitudine corrispondenti al punto  $p_{e,ref}$ .

### 2.3.3 Trasformazione di coordinate dal sistema NED al sistema body

Le relazioni tra il sistema NED e il sistema body sono di tipo cinematico. La conversione da body a NED della velocità lineare è:

$$\dot{V}_n = R_b^n(\Theta)v_o^b \tag{2.13}$$

dove  $v_n$  è il vettore delle velocità nel sistema NED,  $v_o^b = [u \ v \ w]^T$  è il vettore delle velocità nel sistema body e  $\Theta = [\phi \ \vartheta \ \psi]$  è il vettore degli angoli di eulero. La matrice di rotazione body-NED è data dalle tre rotazioni:

$$\begin{aligned}
 R_b^n &= R_{z,\psi}R_{y,\vartheta}R_{x,\phi} \\
 &= \begin{bmatrix} c(\psi)c(\vartheta) & -s(\psi)c(\phi) + c(\psi)s(\vartheta)s(\phi) & s(\psi)s(\phi) + c(\psi)c(\phi)s(\vartheta) \\ s(\psi)c(\vartheta) & c(\psi)c(\phi) + s(\psi)s(\vartheta)s(\phi) & -c(\psi)s(\phi) + s(\psi)c(\phi)s(\vartheta) \\ -s(\vartheta) & c(\vartheta)s(\phi) & c(\vartheta)c(\phi) \end{bmatrix}
 \end{aligned} \tag{2.14}$$

Nel nostro caso viene considerata solo la rotazione intorno all'asse  $z$ , quindi la matrice di rotazione si riduce a:

$$R_b^n = R(\vartheta) = \begin{pmatrix} \cos(\vartheta) & -\sin(\vartheta) & 0 \\ \sin(\vartheta) & \cos(\vartheta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

## Capitolo 3

# Modello cinematico e studio dell'osservabilità

Si cerca un modello che abbia la forma

$$\begin{cases} \dot{p} = f(p, v) \\ z = h(p) \end{cases}$$

dove  $p$  è il vettore di stato,  $v$  è il vettore degli ingressi e  $z$  è il vettore delle uscite. Una volta identificato il modello verrà studiata la proprietà di osservabilità dello stesso.

### 3.1 Modello cinematico del sistema

L'equazione cinematica rappresentante il moto relativo del veicolo in 2D, è data da:

$$\dot{p} = R_b^n v \quad (3.1)$$

,dove  $R_b^n$  è la matrice di rotazione body-NED e  $v = [u \quad v]^T$  è il vettore delle velocità di surge  $u$  e di sway  $v$  del sistema body. Viene considerato il moto nel piano orizzontale del veicolo quindi il vettore di stato è  $p = [x \quad y]$ . L'angolo di rotazione nonché angolo di bussola del veicolo, che chiameremo  $\vartheta$ , è rispetto all'asse  $z$ , quindi la matrice di rotazione, come si è già mostrato nel capitolo precedente, è:

$$R_b^n = R(\vartheta) = \begin{pmatrix} \cos(\vartheta) & -\sin(\vartheta) \\ \sin(\vartheta) & \cos(\vartheta) \end{pmatrix}$$



L'uscita del sistema è la misura di range, quindi la distanza del veicolo dal master modem. Dato che quest'ultimo viene posizionato al centro del nostro sistema di riferimento NED, l'uscita è data da:

$$z = \frac{1}{2} p^T p \quad (3.2)$$

Il sistema dinamico non lineare a tempo continuo, dato dalle equazioni 3.1 e 3.2, è quindi:

$$\begin{cases} \dot{p} = R_b^n v \\ z = \frac{1}{2} p^T p \end{cases} \quad (3.3)$$

## 3.2 Analisi dell'osservabilità

Il modello non lineare definito mediante le equazioni (3.1) e (3.2) può essere facilmente ricondotto alla forma completa:

$$\begin{cases} p(t+1) = g_1 u + g_2 v \\ z(t) = h(p) = \frac{1}{2} p^T p \end{cases} \quad (3.4)$$

dove

$$g_1 = \begin{pmatrix} \cos(\vartheta) \\ \sin(\vartheta) \end{pmatrix} \quad g_2 = \begin{pmatrix} -\sin(\vartheta) \\ \cos(\vartheta) \end{pmatrix}$$

effettuando un'approssimazione lineare al sistema si potrebbe perdere l'osservabilità. Infatti effettuando la linearizzazione del sistema 3.4 nel punto di lavoro, quindi con ingresso nominale  $u = \bar{u}$  e stato nominale  $p = \bar{p}$ , questo diventa:

$$\begin{cases} \delta \dot{p} = B \delta v \\ \delta z = C \delta p \end{cases}$$

dove la matrice di uscita  $C \in \Re^{1 \times 2}$

$$C = \frac{\partial h}{\partial p} = p^T = [x \quad y]$$

Il sottospazio non osservabile ha dimensione 1:

$$\text{Ker} \{ [C^T \quad A^T C^T] \} \neq 0$$

$$\text{Ker} \{ [x \quad y]^T \} = [x \quad y]^T$$

Per evitare questa perdita di osservabilità si utilizzano degli strumenti diversi da quelli utilizzati per i sistemi lineari, che spesso ne sono una generalizzazione. Non si parla

più di osservabilità del sistema ma di *osservabilità locale* del sistema. Una panoramica sintetica dei concetti di osservabilità locale, degli strumenti matematici utilizzati, come le derivate di Lie, e della condizione di osservabilità viene data in un contesto di localizzazione relativa in [2].

Definendo le derivate di Lie dell'uscita scalare  $h_j$  come:

$$\begin{aligned} L_f^0 h_j &= h_j \\ L_f^1 h_j &= \nabla h_j f = \frac{\partial h_j}{\partial p} f = \sum_{i=1}^2 \frac{\partial h_j}{\partial p_i} f_i \\ L_f^2 h_j &= \frac{\partial}{\partial p} [L_f^1 h_j] f \\ &\vdots \\ L_f^n h_j &= \frac{\partial}{\partial p} [L_f^{n-1} h_j] f \end{aligned} \quad (3.5)$$

è possibile definire la matrice di osservabilità  $\mathcal{O}$ .

$$\mathcal{O} = \begin{bmatrix} \nabla L_f^0 h \\ \nabla L_f^1 h \\ \nabla L_f^2 h \\ \vdots \\ \nabla L_f^n h \end{bmatrix} \quad (3.6)$$

dove  $n \in \mathbb{N}$ . In particolare l'osservabilità locale del sistema non lineare è verificata dal pieno rango della matrice di osservabilità  $\mathcal{O}$ , quindi nel nostro caso si può scrivere:

$$\begin{aligned} h(x) &= \frac{1}{2} p^T p \\ L_{g_1} h &= \frac{\partial h_1}{\partial p} g_1 = x \cos(\vartheta) + y \sin(\vartheta) \\ L_{g_2} h &= \frac{\partial h_1}{\partial p} g_2 = -x \sin(\vartheta) + y \cos(\vartheta) \end{aligned}$$

e la matrice di osservabilità può essere:

$$\mathcal{O} = \begin{Bmatrix} x & y \\ x \cos(\vartheta) & y \sin(\vartheta) \end{Bmatrix}$$

La  $\dim(\mathcal{O}) = 2$  se il determinante della seguente matrice è diverso da zero:

$$D = \det \begin{bmatrix} x & y \\ x \cos(\vartheta) & y \sin(\vartheta) \end{bmatrix} = x y \sin(\vartheta) - y x \cos(\vartheta)$$

si verifica sempre che è diverso da zero, quindi il sistema può considerarsi localmente osservabile.

## Capitolo 4

### Il filtro di Kalman esteso (EKF)

Si vuole progettare un osservatore per lo stato del sistema  $p$ , la cui uscita è la stima della posizione assoluta, che indicheremo con  $\hat{p} \in \mathbb{R}^3$ . Un possibile metodo sistematico, per la progettazione dell'osservatore è dato mediante l'uso del Filtro di Kalman Esteso (EKF-Extended Kalman Filter, i.e. in [3] e [4]), tuttavia, sarebbe possibile l'utilizzo di altri tipi di filtri, come il *particle filter* [5] o l'*Unscented Kalman Filter* [6].

In particolare il Filtro di Kalman Esteso permette di stimare lo stato di un sistema con una incertezza sulla dinamica, partendo da misure affette da rumore (causato da frequenza di trasmissione, disturbi ambientali, disturbi nel moto dello strumento (o dell'imbarcazione che lo porta), beam pattern etc.). La forma generale del sistema al quale si applica il filtro di Kalman è:

$$\begin{cases} \dot{p} = f(p, u) + w \\ z = h(p) + v \end{cases} \quad (4.1)$$

dove  $p$  è il vettore di stato,  $w$  è il rumore di processo,  $z$  è il vettore delle misure e  $v$  il rumore di misura. Nel caso in cui  $f(\cdot)$  e  $h(\cdot)$  siano funzioni lineari e i rumori di processo e di misura siano Gaussiani, la soluzione ottima del problema di filtraggio ricorsivo è data dal filtro di Kalman. Nel caso non lineare non esiste in generale una soluzione ottima in forma chiusa. Tuttavia, assumendo che i rumori di processo e di misura siano Gaussiani e additivi, uno dei metodi classici per risolvere il problema di filtraggio non lineare è quello di linearizzare il sistema, nell'intorno della stima corrente, ed applicare il filtro di Kalman al sistema linearizzato; il filtro così ottenuto prende il nome di Filtro di Kalman Esteso (EKF, Extended Kalman Filter). Nei paragrafi successivi ne verranno spiegate le proprietà.

## 4.1 Il processo da stimare

Il Filtro di Kalman Esteso richiama il problema della stima dello stato  $p \in R^n$  di un processo non lineare che è governato da equazioni differenziali stocastiche

$$p_k = f(p_{k-1}, u_{k-1}, w_{k-1}) \quad (4.2)$$

La stima viene effettuata per mezzo di misurazioni  $z \in R^q$  tali che

$$z_k = h(p_k, v_k) \quad (4.3)$$

Le variabili casuali  $w_k$  e  $v_k$  rappresentano rispettivamente il rumore di processo e di misurazione. Essi sono assunti come indipendenti, bianchi e con distribuzione di probabilità normale

$$p(w) \sim N(0, Q) \quad (4.4)$$

$$p(v) \sim N(0, R) \quad (4.5)$$

La funzione non lineare  $f$  nell'equazione differenziale 4.22 lega lo stato al tempo  $k - 1$  allo stato corrente al tempo  $k$ , includendo funzioni di comando input  $u_{k-1}$  e di rumore di processo  $w_{k-1}$ . La funzione non lineare  $h$  nell'equazione della misurazione 4.3 lega lo stato  $p_k$  alla misurazione  $z_k$ .

In pratica non si conoscono i valori dei rumori  $w_k$  e  $v_k$  a ogni tempo di campionamento. Tuttavia si possono approssimare il vettore di stato e delle misure senza rumore, come

$$\tilde{p}_k = f(\hat{p}_{k-1}, u_{k-1}, 0) \quad (4.6)$$

e

$$\tilde{z}_k = h(\tilde{p}_k, 0) \quad (4.7)$$

dove  $\hat{p}_k$  è una certa stima a posteriori dello stato (al tempo  $k - 1$ ). È importante notare che un difetto fondamentale del EKF è che le distribuzioni (o densità nel caso continuo) delle diverse variabili casuali non sono più distribuite normalmente dopo aver subito le rispettive trasformazioni lineari. L'EKF è semplicemente uno stimatore di stato che approssima l'ottimalità della regola di Bayes con la linearizzazione.

## 4.2 L'origine computazionale del filtro

Per stimare un processo non lineare bisogna scrivere le nuove equazioni linearizzate nel punto di lavoro 4.6 e 4.7,

$$p_k \approx \tilde{p}_k + A(p_{k-1} - \hat{p}_{k-1}) + W w_{k-1}$$

$$z_k \approx \tilde{z}_k - H(p_k - \tilde{p}_k) + V v_{k-1}$$

dove:

- $p_k$  e  $z_k$  sono il vettore di stato e di misura allo stato attuale,
- $\tilde{p}_k$  e  $\tilde{z}_k$  sono il vettore di stato e di misura approssimati come in 4.6 e 4.7,
- $\hat{p}_k$  è la stima a posteriori dello stato al tempo  $k$ ,
- le variabili causali  $w_k$  e  $v_k$  rappresentano il rumore di processo e di misura rispettivamente alle 4.4 e 4.5.
- $A$  è la matrice Jacobiana delle derivate parziali della funzione  $f$  rispetto al vettore di stato  $x$ ,

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial p_{[j]}} \big|_{(\hat{p}_{k-1}, u_{k-1}, 0)}$$

- $W$  è la matrice Jacobiana delle derivate parziali della funzione  $f$  rispetto a  $w$ ,

$$W_{[i,j]} = \frac{\partial f_{[i]}}{\partial w_{[j]}} \big|_{(\hat{p}_{k-1}, u_{k-1}, 0)}$$

- $H$  è la matrice Jacobiana delle derivate parziali della funzione  $h$  rispetto a  $x$ ,

$$H_{[i,j]} = \frac{\partial h_{[i]}}{\partial p_{[j]}} \big|_{(\tilde{p}_k, 0)}$$

- $V$  è la matrice Jacobiana delle derivate parziali della funzione  $h$  rispetto a  $v$ ,

$$V_{[i,j]} = \frac{\partial h_{[i]}}{\partial v_{[j]}} \big|_{(\tilde{p}_k, 0)}$$

Si noti che per semplicità nella notazione delle matrici Jacobiane  $A$ ,  $W$ ,  $H$ ,  $V$ , non viene esplicitato il tempo  $k$  ma esse vengono comunque aggiornate a ogni tempo di campionamento. L'errore di stima a priori si può scrivere come:

$$\tilde{e}_{p_k} \equiv p_k - \tilde{p}_k \quad (4.8)$$

e l'errore di misura,

$$\tilde{e}_{z_k} \equiv z_k - \tilde{z}_k \quad (4.9)$$

Utilizzando le misure  $z_k$ , alle quali abbiamo accesso, si può stimare il vettore di stato  $p_k$ . Usando la 4.8 e 4.9 si possono scrivere le equazioni che governano l'andamento dell'errore:

$$\tilde{e}_{p_k} \approx A(p_{k-1} - \hat{p}_{k-1}) + \varepsilon_k \quad (4.10)$$

$$\tilde{e}_{z_k} \approx H\tilde{e}_{p_k} + \eta_k \quad (4.11)$$

dove  $\varepsilon_k$  e  $\eta_k$  rappresentano le variabili causali indipendenti con media nulla e matrici di covarianza  $VRV^T$  e  $WQW^T$ , con  $Q$  ed  $R$  rispettivamente come in 4.4 e 4.5.

La covarianza dell'errore stimata a priori è perciò

$$P_k^- = E[e_k^- e_k^{-T}] \quad (4.12)$$

e la covarianza dell'errore stimata a posteriori

$$P_k = E[e_k e_k^T] \quad (4.13)$$

Derivando le equazioni per il filtro di Kalman, si procede ricercando un'equazione che computi una stima a posteriori  $p_k$  come una combinazione lineare di un  $p_k^-$  stimato a priori e di una differenza pesata tra la misurazione attuale  $z_k$  ed una previsione per la misurazione  $H_k p_k^-$ . La stima a posteriori risulta pertanto

$$p_k = p_k^- + K(z_k - H_k p_k^-) \quad (4.14)$$

La differenza  $(z_k - H_k p_k^-)$  è detta innovazione di misurazione o residuo. Essa riflette la discrepanza tra la previsione di misurazione  $H_k p_k^-$  e la misurazione attuale  $z_k$ . Un residuo nullo implica che le due quantità sono in perfetto accordo. La matrice  $n \times m$   $K$  nella 4.14, detta guadagno di Kalman o blending factor, si determina in modo da minimizzare la covarianza dell'errore a posteriori 4.13. Questa minimizzazione può essere effettuata sostituendo la 4.14 nella precedente definizione di  $e_k$ , introducendo la relazione così ricavata nella 4.13, derivando rispetto a  $K$ , eguagliando a zero ed infine risolvendo rispetto a  $K$ . Una possibile forma del guadagno che minimizza la 4.13 è data dalla seguente espressione:

$$K_k = P_k H_k^T (H_k P_k H_k^T + R_k)^{-1} \quad (4.15)$$

Osservando la 4.15 si può vedere che al tendere a zero della covarianza dell'errore di misurazione  $R_k$ , il guadagno  $K$  pesa il residuo in maniera via via più ingente. Specificatamente

$$\lim_{R_k \rightarrow 0} K_k = H_k^{-1} \quad (4.16)$$

D'altra parte, al tendere a zero della covarianza dell'errore di stima a priori  $P_k^-$ , il guadagno  $K$  pesa il residuo in maniera via via meno ingente. Specificatamente

$$\lim_{P_k^- \rightarrow 0} K_k = 0 \quad (4.17)$$

Un altro modo di pensare l'azione di  $K$  è che quando la covarianza dell'errore  $R_k$  tende a zero, la misurazione  $z_k$  diviene via via più vera, mentre la previsione della misurazione sempre meno. D'altro canto, quando la covarianza dell'errore di stima a priori  $P_k^-$  tende a zero, la misurazione attuale  $z_k$  è via via meno vera, mentre la previsione della misurazione si avvicina sempre più al valore corretto.

Attraverso alcune semplici operazioni di calcolo matriciale è altrettanto possibile esprimere il guadagno di Kalman  $K$  nella seguente forma

$$K_k = P_k H_k R^{-1} \quad (4.18)$$

con

$$P_k = P_k^- (I + H_k^T R^{-1} H_k P_k^-)^{-1} \quad (4.19)$$

Se  $H_k = 1$ , è immediato che

$$K_k = P_k R^{-1} \quad (4.20)$$

e si può quindi osservare che  $K_k$  è direttamente proporzionale alla covarianza dell'errore ed inversamente proporzionale alla varianza dell'errore di misurazione:

- tanto maggiore è l'errore che si è commesso nella stima precedente e tanto maggiore è l'affinamento della stima attuale (guadagno) indotto dal filtraggio;
- tanto meno le misurazioni si discostano dal valore reale, tanto più sarà possibile ottenere stime precise.

La formulazione del filtro di Kalman esteso per sistemi di tipo tempo-continuo risulta essere quindi:

#### 1) Time update

$$\begin{aligned} \dot{\hat{p}} &= f(\hat{p}, u) \\ A &= \left. \frac{\partial f(p, u)}{\partial p} \right|_{p=\hat{p}, u=\hat{u}} \\ \dot{P} &= AP + PA^T + Q \end{aligned}$$

#### 2) Measurement update

$$\begin{aligned} C &= \left. \frac{\partial h(p)}{\partial p} \right|_{p=\hat{p}} \\ K &= PC(CPC^T + R)^{-1} \\ \hat{p}_k &= \hat{p}_{k-1} + K(z - h(\hat{p}_{k-1})) \\ P_k &= P_{k-1} - KCP_{k-1} \end{aligned} \quad (4.21)$$

L'integrazione della  $\hat{p} = f(\hat{p}, u)$  e della  $\dot{P}$  può essere effettuata adoperando il metodo numerico di Eulero in avanti.

### 4.3 Applicazione al modello

Lo scopo è quello di cercare di stimare il vettore delle variabili di stato  $p \in \mathbb{R}^2$ , tramite le misurazioni del *range*, le quali sono danneggiate da un rumore di misura gaussiano bianco. Il processo viene regolato dall'equazione non lineare:

$$\dot{p} = \begin{pmatrix} \cos(\vartheta) & -\sin(\vartheta) \\ \sin(\vartheta) & \cos(\vartheta) \end{pmatrix} u + w$$

Discretizzando con il metodo di Eulero, il processo si evolve come:

$$p_{k+1} = p_k + R_k u_k \tau + w_k \quad (4.22)$$

dove  $\tau$  è il tempo trascorso da un tempo  $k$  ad un tempo  $k+1$ . L'equazione di misura  $z \in \mathbb{R}^1$  è:

$$z_k = \|p_k\| + v_k \quad (4.23)$$

I segnali d'ingresso  $u_k$  sono le velocità del veicolo nel frame body, quindi le velocità di surge  $u$  e di sway  $v$  ricavate nel capitolo 1 (fig.(1.15)). Anche per la variabile  $\vartheta$  i valori che vengono usati sono quelli rielaborati che si visualizzano in figura (1.13).

Il filtro è stato implementato con linguaggio Matlab. All'interno della funzione di lancio vengono inizializzati i parametri del filtro e il vettore di stato. In quanto si conoscono le posizioni GPS  $p$  del veicolo si può supporre nota la posizione iniziale del veicolo:

$$p_0 = p + \varepsilon$$

dove con  $\varepsilon$  si introduce un piccolo errore.

Il valore iniziale della matrice di covarianza dell'errore di stima è:

$$P_0 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

I valori statici dei parametri del EKF, scelti per tentativi ed errori, sono:

$$Q = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \quad R = 1;$$

Applicando il filtro di Kalman Esteso al sistema di equazioni (4.1), si hanno le seguenti uscite del filtro:



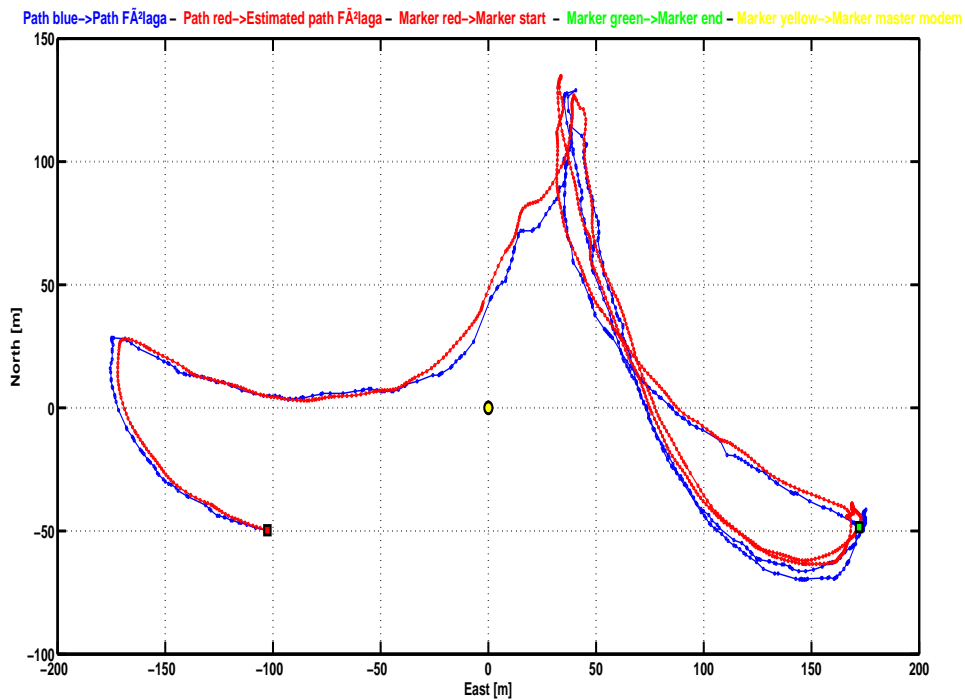


Figura 4.1: In blu la traiettoria da GPS del veicolo, in rosso la traiettoria stimata del veicolo.

Nonostante i rumori e le non linearità, l'errore di stima è inferiore a 10 m (come mostrato nella figura (4.2)). Inoltre vale la pena notare che i picchi di errore sono correlati alle basse velocità di manovra, quando il veicolo effettua delle curve di piccolo raggio (fig.(4.3) e (4.4)).

Nelle curve a piccolo e grande raggio, se il veicolo mantiene una velocità di rotta di ca.  $0.5m/s$ , l'errore è contenuto nell'ordine dei  $\pm 3m$  (fig.(4.5) e (4.6)).

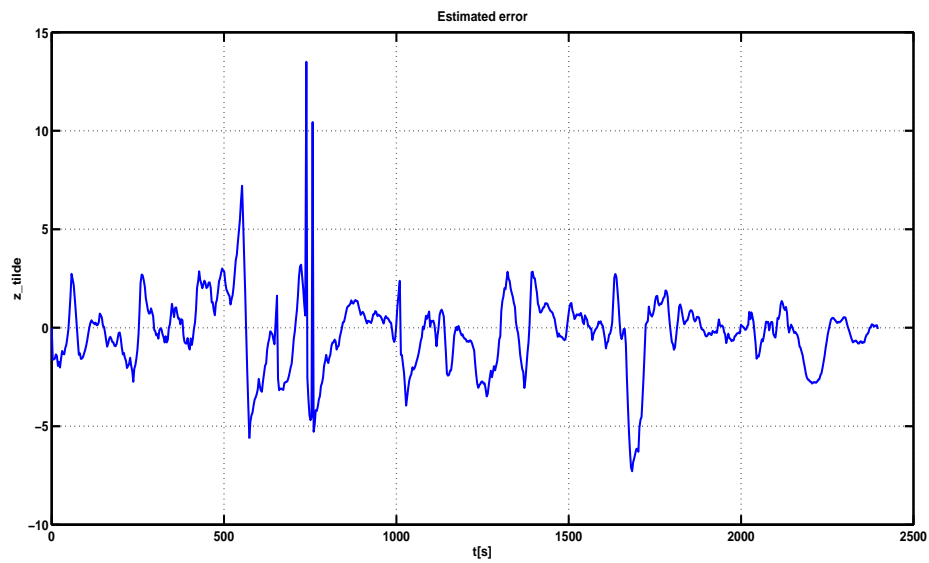


Figura 4.2: Andamento nel tempo dell'errore di stima.

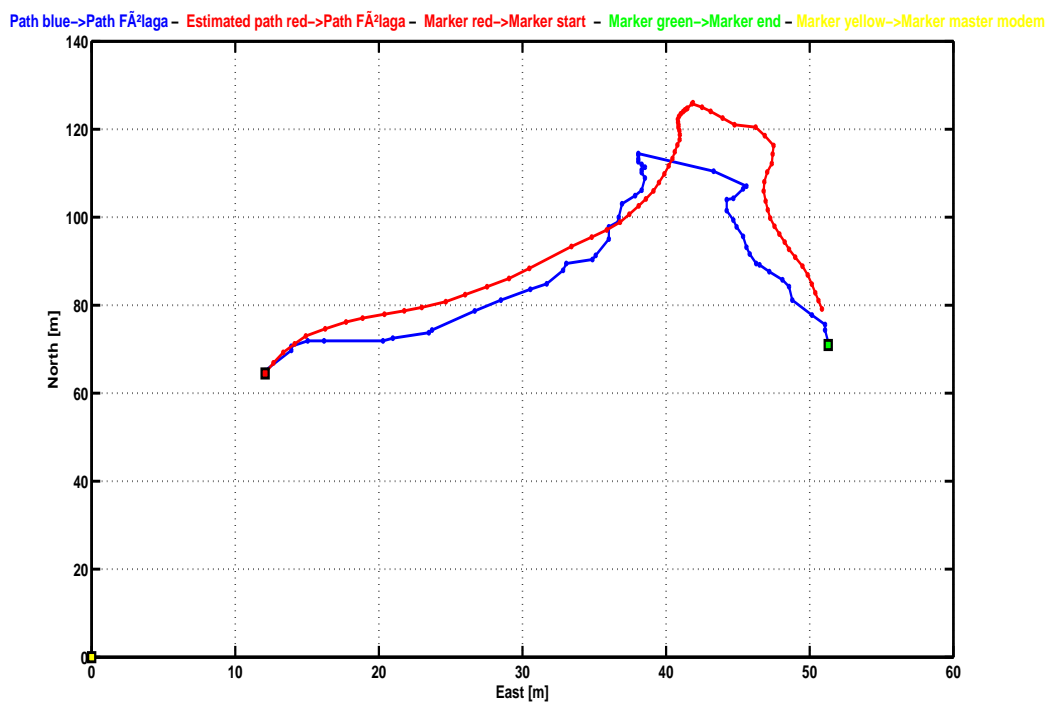


Figura 4.3: In blu la traiettoria da GPS del veicolo, in rosso la traiettoria stimata del veicolo, durante una curva o una manovra a bassa velocità.

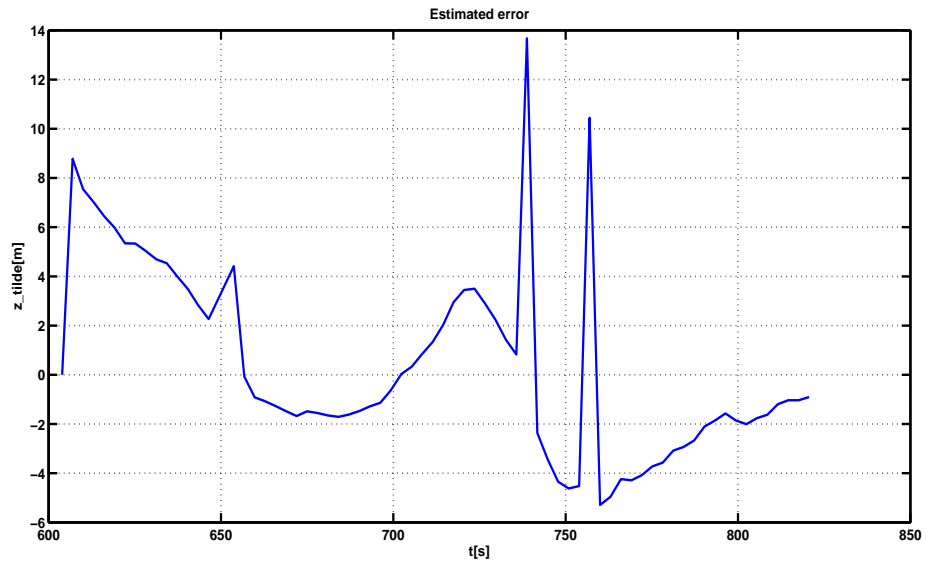


Figura 4.4: Andamento nel tempo dell'errore di stima quando il veicolo esegue una curva o una manovra a bassa velocità.

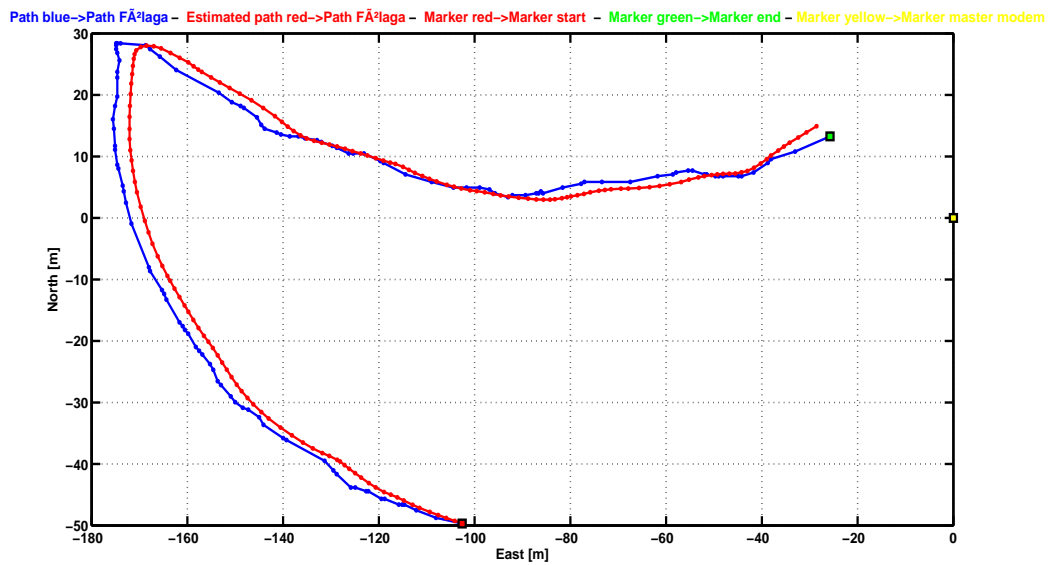


Figura 4.5: in blu la traiettoria da GPS del veicolo, in rosso la traiettoria stimata del veicolo, durante manovre a velocità più elevata.

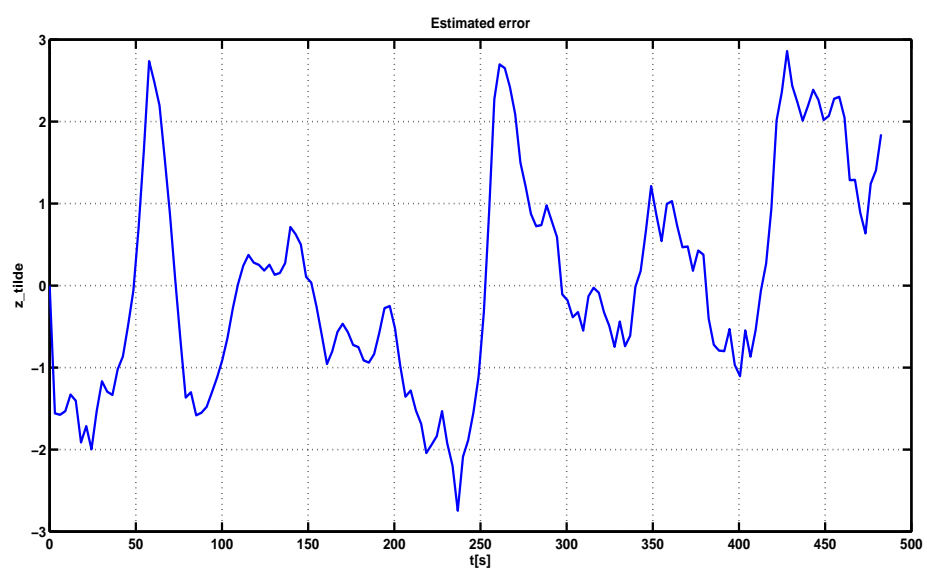


Figura 4.6: Andamento nel tempo dell'errore di stima quando il veicolo esegue manovre a velocità più elevata.

# Capitolo 5

## Problema di ottimizzazione vincolata

In questo capitolo si descrive un metodo alternativo a quello del filtro di Kalman per la stima simultanea della posizione e della velocità con i dati a disposizione.

In particolare, questo metodo considera il problema di stima come un problema di ottimizzazione vincolata. Il vincolo posto al problema di stima consiste nella rispondenza della stima ai dati GPS. Si noti che questo metodo non è implementabile in pratica su un veicolo, qualora tale veicolo disponesse di un sensore di velocità, ma è stato impostato per avere un algoritmo che generi una traiettoria ottima da poter confrontare con quella ricavata tramite filtro di Kalman.

### 5.1 Gli algoritmi di ottimizzazione

Si consideri un generico problema di ottimizzazione del tipo

$$\begin{cases} \min f(x) \\ x \in \mathbb{D} \end{cases} \quad oppure \quad \begin{cases} \max f(x) \\ x \in \mathbb{D} \end{cases}$$

dove la funzione  $f : \mathbb{D} \rightarrow \mathfrak{R}$  è detta *funzione obiettivo* e l'insieme  $\mathbb{D}$  è chiamato regione ammissibile del problema.

Un problema di ottimizzazione viene detto non vincolato se  $\mathbb{D} = \mathfrak{R}^n$ , mentre viene detto vincolato quando

$$\mathbb{D} = \{x \in \mathfrak{R} : g(x) \leq 0 \quad h(x) = 0\},$$

con  $g(x) = (g_1(x), \dots, g_m(x))$  e  $h(x) = (h_1(x), \dots, h_p(x))$ . In tal caso le funzioni  $g_i$ , per  $i = 1, 2, \dots, m$ , e  $h_j$ , per  $j = 1, 2, \dots, p$ , sono chiamate vincoli del problema. La ricerca

dei punti di minimo/massimo (locale o globale) procede cercando di trovare proprietà che caratterizzano tali punti. In particolare, cerca di stabilire condizioni necessarie o sufficienti di ottimalità. Le condizioni necessarie sono verificate dai punti di minimo/massimo, mentre le condizioni sufficienti garantiscono che il punto trovato sia quello di minimo/massimo. Purtroppo a volte succede che anche i punti nè di minimo nè di massimo verifichino tali condizioni di ottimo. Le condizioni di Kuhn-Tucker ad esempio, forniscono le condizioni necessarie per una soluzione ottimale [7]. Gli algoritmi di ottimizzazione devono quindi affrontare due esigenze spesso contrastanti:

- trovare soluzioni ammissibili;
- minimizzare la funzione obiettivo.

Molteplici sono gli algoritmi e i metodi per l'ottimizzazione vincolata, dal classico metodo del simplesso per i problemi di programmazione lineare, al metodo di Frank Wolfe per i problemi di programmazione non lineare vincolata e tanti altri [7] e [8].

## 5.2 Problema di ottimo applicato al problema

Si consideri il sistema dinamico del veicolo

$$\dot{p} = Ru \quad (5.1)$$

si cerca di trovare la stima del vettore posizione  $p$  e del vettore velocità  $u$  con i dati a disposizione.

In particolare le misure che vengono utilizzate sono l'angolo di bussola  $\vartheta$  e le misure di posizione GPS trasformate in coordinate NED, che chiameremo  $\eta$ . Le misure stimate sono la posizione del veicolo  $\hat{\eta}$  e le velocità di corpo  $\hat{V}_b$ .

La funzione di costo  $J$  che si vuole minimizzare è:

$$J = \gamma + \gamma_D + \varepsilon \gamma_V$$

dove:

- $\gamma$  rappresenta un upper bound della norma infinito dell'errore sulla cinematica del modello

$$\|\hat{\eta}(k) - \hat{\eta}(k-1) - T_c R \hat{V}_b\|_{\infty} \leq \gamma$$

- $\gamma_V$  rappresenta un upper bound sulla norma infinito dell'accelerazione del veicolo

$$\|\Delta \hat{V}_b\|_{\infty} \leq \gamma_V,$$

ed  $\varepsilon$  è un numero di tuning molto piccolo per eseguire uno smoothing sulle accelerazioni;

- $\gamma_D$  è un upper bound sulla norma infinito dell'errore di posizione

$$\|\hat{\eta} - \eta\|_{\infty} \leq \gamma_D.$$

Le equazioni sopra elencate sono i vincoli di modello, inoltre il problema deve soddisfare l'equazione cinematica del veicolo(5.1). Si ha quindi un problema di minimo vincolato.

Inoltre vengono definiti anche dei vincoli di coerenza dovuti ai limiti fisici del modello, questi sono:

- Vincolo sulla velocità di corpo del veicolo

$$\|\hat{V}_b\|_{\infty} \leq 1,5$$

- Vincolo sull'errore di posizione dello stato finale  $n$  del veicolo

$$\|\hat{\eta}_n - \eta_n\|_{\infty} \leq 10$$

- Vincolo sul valore del upper bound  $\gamma \geq 0$ ;
- Vincolo sul valore del upper bound  $\gamma_V \geq 0$ ;
- Vincolo sul valore del upper bound  $\underline{\gamma} \leq \gamma_D \leq 10$ , dove  $\underline{\gamma} \geq 0$ .

Per realizzare questo algoritmo di ottimo è stato utilizzato YALMIP [9], un toolbox per MATLAB che permette di modellizzare in modo efficiente dei problemi di ottimizzazione, utilizzando poi dei risolutori esterni per calcolare la soluzione.

Come risolutore è stato utilizzato SeDuMi [10], un pacchetto e software per risolvere problemi di programmazione semidefinita (v. anche Appendice A).

Dopo aver ricampionato i dati, con un tempo di campionamento  $T_c = 1sec$  viene lanciato il ciclo di for per la risoluzione del problema di ottimizzazione. Sotto vengono mostrate le righe di codice:

```

CON = set(gam>=0) + set(gamV>=0) + set(gamdists>=0) + ...
      set(gamdists<=10);

for i=2:N

Ri = rotnz(THN(i));

CON = CON + ...
      set(Pk(:,i-1)-NE(i-1,:)')>=-gamdists)+...
      set(Pk(:,i-1)-NE(i-1,:)')<= gamdists);

CON = CON + ...
      set(Vk(:,i-1)>=-1.5)+ set(Vk(:,i-1)<=1.5);

CON = CON + ...
      set(Vk(:,i-1)-Vk(:,i))>=-gamV)+ set(Vk(:,i-1)-Vk(:,i)<=gamV);

CON = CON + ...
      set((Pk(:,i)-Pk(:,i-1)-Tc(i)*Ri(1:2,1:2)*Vk(:,i-1))<=gam)+...
      set((Pk(:,i)-Pk(:,i-1)-Tc(i)*Ri(1:2,1:2)*Vk(:,i-1))>=-gam);

end

CON = CON + ...
      set(Pk(:,N)-NE(N,:)')>=-10)+ set(Pk(:,N)-NE(N,:)')<=10);

opt = sdpsettings('solver','sedumi');
sol = solvesdp(CON, gam + gamdists + 1e-10*gamV ,opt);

```

Di seguito viene mostrata la traiettoria ottima calcolata dall'algoritmo di ottimizzazione (5.1), l'errore che viene commesso (5.2) e le velocità di surge e di sway (5.3).



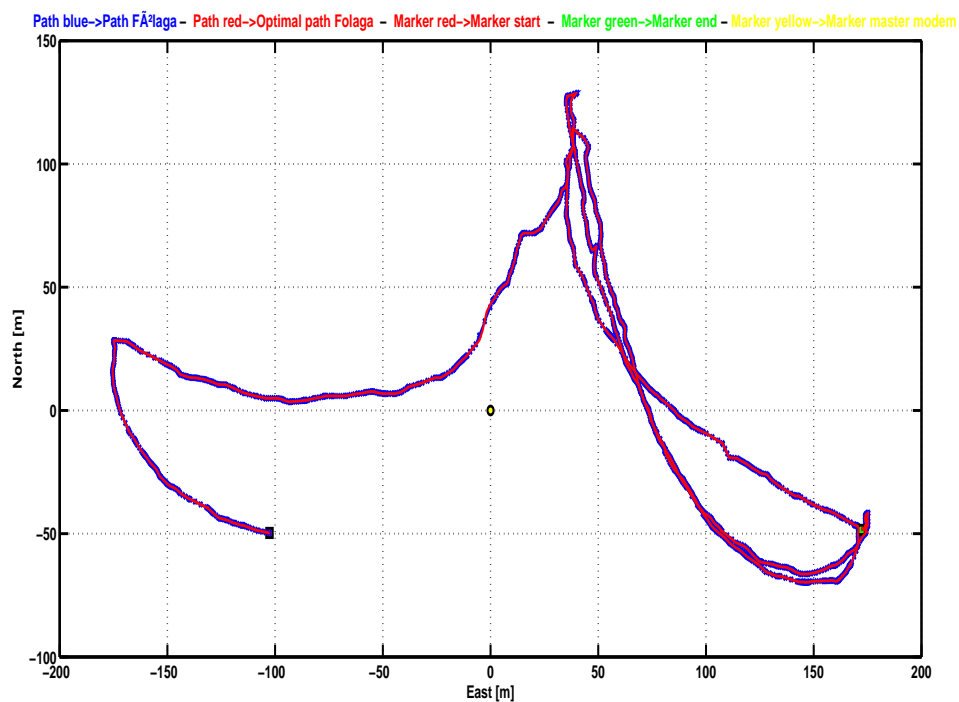


Figura 5.1: In blu la traiettoria da GPS del veicolo, in rosso la traiettoria ottima del veicolo.

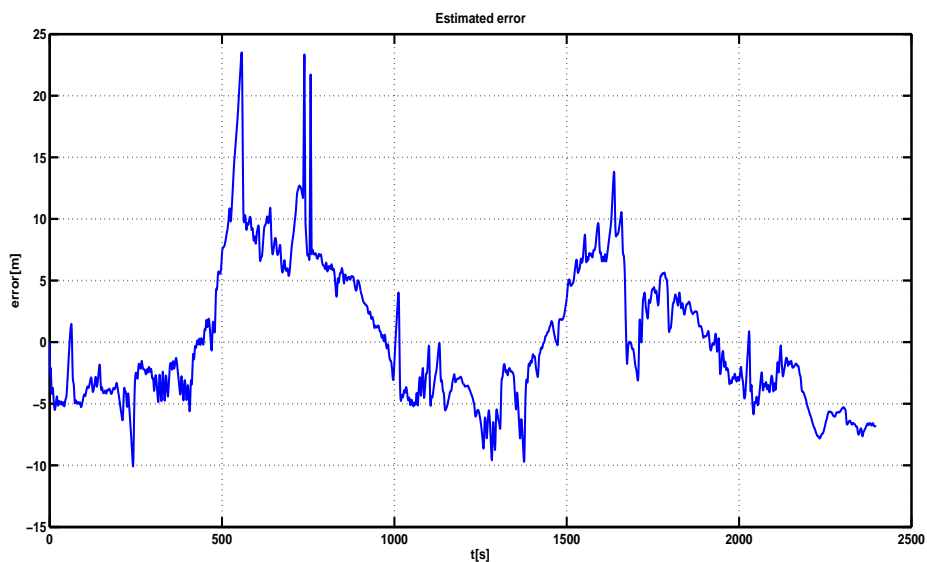


Figura 5.2: Andamento nel tempo dell'errore di stima.

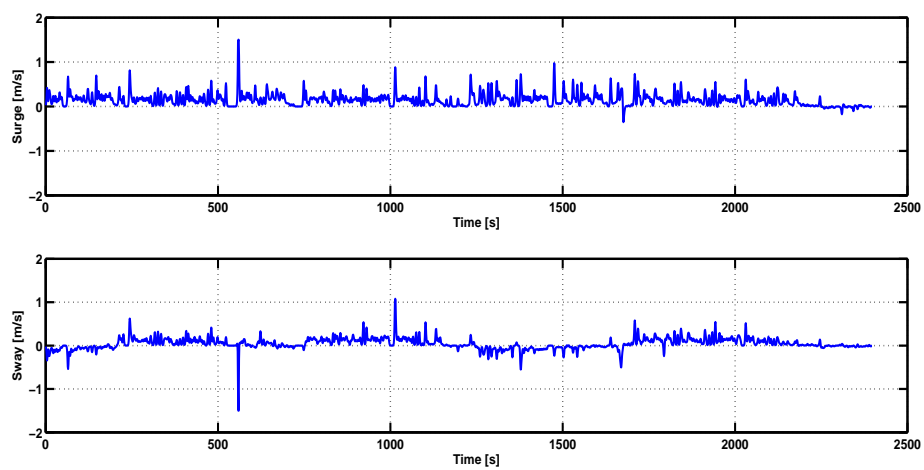


Figura 5.3: Andamento nel tempo delle velocità di surge e sway.

## Capitolo 6

### Risultati e confronto

Dopo aver implementato i due algoritmi per la stima della posizione assoluta del veicolo e studiato singolarmente le loro performance, in questo capitolo viene mostrato un confronto fra i due. Ricordiamo che l'algoritmo di ottimizzazione vincolata e' stato impostato per avere un algoritmo alternativo al filtro di Kalman.

Dai risultati ottenuti sembra evidente che le prestazioni del filtro di Kalman siano elevate. Analizzando nella figura (6.1) le traiettorie generate dai due algoritmi di stima, si evince che il percorso stimato dal filtro di Kalman in alcuni tratti si sovrappone alla traiettoria ottimale. Si può affermare dunque l'efficacia del Filtro di Kalman Esteso nei problemi di stima della posizione assoluta di un veicolo autonomo.

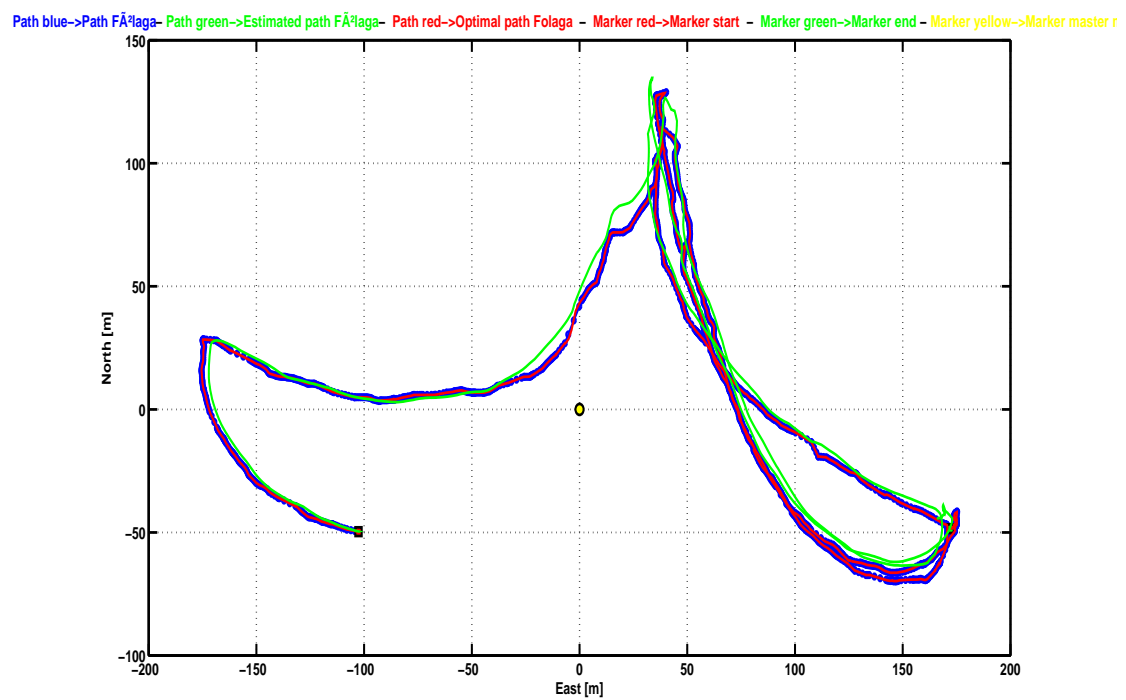


Figura 6.1: In blu la traiettoria da GPS del veicolo, in verde la traiettoria stimata dal Filtro di Kalman, in rosso la traiettoria stimata dal problema di ottimo.

# Capitolo 7

## Conclusioni

In questo lavoro di tesi si è trattato il problema di stima della posizione assoluta di un veicolo AUV attraverso le sole misure acustiche di distanza. I dati trattati sono stati rilevati durante una prova in mare il giorno 22 settembre 2010. Durante gli esperimenti il master modem (in posizione fissa) ha comunicato per tutto il tempo con lo slave modem posizionato sopra la Fòlaga, registrando la sua distanza da esso.

Il veicolo a bordo è dotato di bussola, quindi è stato possibile determinare il suo orientamento, ma non ha sensori di velocità. Per questo motivo, conoscendo le coordinate di posizione GPS, è stato necessario ricavare le velocità di navigazione del veicolo mediante rapporto incrementale. Queste stesse poi sono state proiettate nel sistema body. Dopo un'attenta analisi off-line dei dati a disposizione, si è studiata la proprietà di osservabilità del sistema non lineare del robot. Si è innanzitutto studiata l'osservabilità del sistema linearizzato e poi l'osservabilità locale del sistema usando le derivate di Lie. Quindi, è stato sviluppato il Filtro di Kalman Esteso e si sono testate numericamente le sue prestazioni, nel caso 2D, mandandolo in simulazione con i dati acquisiti durante la prova in mare. I risultati dimostrano l'efficacia del lavoro proposto.

Viene inoltre implementato anche un semplice problema di ottimo per computare la posizione assoluta del veicolo. La traiettoria ottimale ottenuta è stata messa a confronto con la traiettoria stimata dal filtro di Kalman confermando l'efficienza dello stesso.

Oltre alla dotazione di opportuni sensori di navigazione per la Fòlaga, i lavori futuri potrebbero trattare lo studio sull'integrazione del filtro sul robot, per aumentare il suo livello di performance. Inoltre, si potrebbe analizzare nel dettaglio l'impatto degli effetti multi-path, i ritardi di trasmissione e i valori anomali a cui è soggetta una comunicazione acustica subacquea.

# Appendice A

## Yalmip e SeDuMi

Yalmip è un linguaggio di modellazione per definire e risolvere avanzati problemi di ottimizzazione. È implementato come toolbox gratuito per MATLAB ed è stato ideato da Johan Löfberg.

La principale motivazione per usare Yalmip è il rapido sviluppo di algoritmi. Il linguaggio è coerente con la sintassi standard di MATLAB e risulta quindi estremamente semplice per chi ha già familiarità con l'ambiente.

Un altro vantaggio di Yalmip consiste nell'implementazione di innumerevoli accorgimenti per la modellazione, che permettono all'utente di concentrarsi sul problema di alto livello lasciando al software il compito di ottimizzare numericamente il modello.

**Classi di problemi** Il linguaggio di modellazione supporta un gran numero di classi di problemi di ottimizzazioni, come quelli lineari, quadratici, geometrici, multiparametrici ed ovviamente permette di risolvere sistemi di disequazioni matriciali (LMI) in maniera efficiente.

**Risolutori** Una delle idee centrali in Yalmip è quella di focalizzarsi sul linguaggio e su algoritmi di alto livello, rimandando a risolutori esterni il compito di computare il problema. In ogni caso, anche Yalmip stesso implementa algoritmi interni per l'ottimizzazione, che tipicamente risolvono sotto-problemi a partire da quello originale usando i risolutori esterni. Uno dei possibili risolutori di cui Yalmip costituisce di fatto l'interfaccia è SeDuMi.

SeDuMi è un pacchetto software completamente gratuito per risolvere problemi di ottimizzazione, rilasciato sotto licenza GNU/GPL. L'interfaccia più usata da SeDuMi è appunto Yalmip. SeDuMi è scritto nel linguaggio di MATLAB ed in C ed è stato svi-

luppato per funzionare sotto MATLAB stesso. SeDuMi funziona quindi su Windows, Linux, Unix, AIX, Solaris e MacOS.

# Bibliografia

- [1] Prof. Lorenzo Pollini, Dispense del corso Sistemi di guida e navigazione, Università di Pisa.
- [2] X.S. Zhou and S.I. Roumeliotis (Dicembre 2008). Robot-to-robot relative pose estimation from range measurements. *IEEE Transactions on Robotics*.
- [3] Gianluca Antonelli, Filippo Arrichiello, Stefano Chiaverini, Gaurav S. Sukhatme (2010), Observability analysis of relative localization for AUVs based on ranging and depth measurements, *IEEE International Conference on Robotics and Automation Anchorage Convention District*.
- [4] Gao Rui, Mandar Chitre, Cooperative positioning using range-only measurements between two AUVs, *Acoustic Research Laboratory, Tropical Marine Science Institute, National University of Singapore*.
- [5] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, e Per-Johan Nordlund (Febbraio 2002), Particle Filters for Positioning, Navigation, and Tracking, *IEEE transaction on signal processing*.
- [6] Simo Särkkä, Member, IEEE (Settembre 2007), On Unscented Kalman Filtering for State Estimation of Continuous-Time Nonlinear Systems, *IEEE transactions on automatic control*.
- [7] Prof. Massimo Pappalardo, Lezioni di ricerca operativa, Università di Pisa.
- [8] Prof. Andrea Caiti, Dispense del corso di controllo e identificazione dei sistemi incerti, Università di Pisa.
- [9] J. Löfberg (2004), YALMIP : a toolbox for modeling and optimization Computer Aided Control Systems Design, *IEEE International Symposium on*.
- [10] Florian Potra, Cornelis Roos, and Tamas Terlaky (1999), Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. Optimization Methods and Software.