



UNIVERSITA' POLITECNICA DELLE MARCHE

FACOLTA' DI INGEGNERIA

---

Corso di Laurea triennale: Ingegneria Informatica e dell'Automazione

**Sviluppo di algoritmi di data fusion per piattaforme inerziali  
basati sul filtro di Kalman**

**Development of data fusion algorithms for inertial platforms  
based on the Kalman filter**

Relatore: Chiar.mo  
Prof. **Paolo Crippa**

Tesi di Laurea di:  
**Giuseppe Rinaldi**

Correlatore: Chiar.mo  
Prof. **Giorgio Biagetti**

**A.A. 2018/2019**



# INDICE

<b>Introduzione .....</b>	<b>6</b>
<b>Capitolo 1: Il Software .....</b>	<b>8</b>
1.1 Matlab ed il suo utilizzo .....	8
1.2 I grafici in Matlab .....	9
<b>Capitolo 2: Il database .....</b>	<b>11</b>
2.1 Rotazione 360 gradi .....	12
2.2 Acquisizione 30 minuti.....	14
<b>Capitolo 3: Le piattaforme inerziali .....</b>	<b>18</b>
3.1 Angoli di Eulero.....	18
3.2 Accelerometro .....	19
3.3 Giroscopio.....	21
3.4 Magnetometro .....	23
3.5 Come funziona una piattaforma inerziale.....	24
<b>Capitolo 4: Il filtro di Kalman .....</b>	<b>26</b>
4.1 Cenni storici ed applicazioni .....	26
4.2 Funzionamento.....	27
4.3 Modello matematico e algoritmo .....	29
4.4 Il filtro di Kalman esteso .....	33
<b>Capitolo 5: Ahrfilter basato sul filtro di Kalman .....</b>	<b>34</b>
5.1 Introduzione .....	34
5.2 Funzionamento.....	35
5.3 Proprietà del filtro .....	36
5.4 Implementazione in Matlab.....	37
<b>Capitolo 6: Risultati .....</b>	<b>39</b>
6.1 Output .....	39
<b>Conclusioni .....</b>	<b>41</b>

**Bibliografia..... 42**  
**Sitografia ..... 42**



# Introduzione

Lo scopo della tesi è quello di filtrare dati raccolti da sensori quali accelerometro, giroscopio e magnetometro, per riuscire a determinare la posizione della persona che indossa il dispositivo. Lo scopo finale, infatti sarebbe quello di sviluppare un dispositivo indossabile per pazienti anziani ricoverati in una struttura ospedaliera e poter monitorare i loro movimenti per la loro sicurezza.

Alcuni miei colleghi hanno realizzato la scheda elettronica con i sensori utilizzati, altri si sono occupati della fase di test, quindi hanno registrato dati su movimenti elementari come una rotazione di 90 gradi o un aumento e diminuzione della quota.

Il test più complesso eseguito è stato quello di una passeggiata di circa 30 minuti e quindi con diversi movimenti registrati dai sensori. Il database a disposizione è risultato quindi essere molto vasto ed utile.

I dati raccolti sono risultati essere molto rumorosi, perciò, io e altri miei colleghi abbiamo filtrato i dati con diversi algoritmi.

Io ho utilizzato l'ahrsfilter basato sul filtro di Kalman, notando che i dati così filtrati risultano meglio comprensibili grazie ad una riduzione significativa del rumore.

La tesi è stata suddivisa in diverse sezioni:

1. Nel capitolo 1 viene fatta un'analisi del software utilizzato e le funzioni che sono state più utilizzate.
2. Nel capitolo 2 si procede con la presentazione e la descrizione del database con relativi esempi.
3. Nel capitolo 3 vengono descritti gli elementi che costituiscono una piattaforma inerziale, la loro spiegazione fisica e le diverse implementazioni.
4. Nel capitolo 4 viene spiegato il filtro di Kalman, partendo dai cenni storici, per poi passare ad illustrare il suo funzionamento ed il modello matematico. Si vede infine il filtro di Kalman esteso.

5. Nel capitolo 5 si presenta l'ahrsfilter applicato sul filtro di Kalman, si spiegano il suo utilizzo, le sue proprietà e com'è stato implementato in Matlab.
6. Nel capitolo 6 vengono esposti i risultati dell'algoritmo da me utilizzato, confrontandoli con i dati iniziali. Si conclude la tesi con un'idea su un possibile sviluppo futuro.

# Capitolo 1: Il software

## 1.1 Matlab ed il suo utilizzo

Il software che è stato utilizzato per l'elaborazione dei dati è Matlab (Matrix Laboratory), un potente ambiente interattivo per il calcolo numerico che integra in modo efficiente la computazione, la visualizzazione e la programmazione. L'ambiente Matlab consente di gestire variabili, importare ed esportare dati, svolgere calcoli, disegnare grafici, programmare istruzioni e sviluppare applicazioni. Queste caratteristiche rendono questo software uno strumento utilizzabile in molteplici situazioni.

Il software utilizza tre tipi di file:

- M-file,
- MAT-file
- file dati

Gli M-file hanno estensione m e sono utilizzati per scrivere sequenze di comandi (script e funzioni); essendo file ASCII (American Standard Code for Information Interchange) possono essere scritti utilizzando qualsiasi editor di testi. Matlab, tuttavia, prevede un editor molto utile che include funzioni di correzione (debug). I MAT-file, permettono di salvare i nomi ed i valori delle variabili generate durante una sessione di lavoro. Sono file binari e pertanto possono essere letti soltanto dal software che li ha creati. Inoltre, Matlab è in grado di utilizzare file dati ASCII creati da altri tipi di programma.

Matlab permette di trattare grandi insiemi di dati come singole variabili chiamate array e di effettuare calcoli complessi utilizzando poche righe di codice.

La programmazione in Matlab è caratterizzata dall'utilizzo di variabili predefinite (con un valore prefissato, come ad esempio pi greco), variabili definite dall'utente, vettori, matrici e grafici.





Figura 1: Logo di Matlab

L'interfaccia principale di MATLAB è composta da diverse finestre che è possibile affiancare, spostare, ridurre a icona, ridimensionare e così via. Le finestre principali, le più usate, sono quattro:

- *Command Window*
- *Workspace*
- *Current directory*
- *Command history*

La *Command Window* è una finestra dell'interfaccia principale di MATLAB, nella quale è possibile digitare comandi supportati e visualizzare a schermo in tempo reale i risultati.

Il *Workspace* è lo spazio di lavoro contenente le variabili dichiarate, mentre la finestra *Current Directory* permette di esplorare il contenuto delle cartelle sul proprio hard disk.

Nella finestra *Command History* sono elencati tutti i comandi digitati di recente, divisi per ora e data.

## 1.2 Grafici in Matlab

Come precedentemente detto, Matlab consente di rappresentare graficamente funzioni e dati memorizzati in vettori e matrici.

Per tracciare in modo corretto un grafico è necessario applicare una serie di regole:

1. Innanzitutto, è necessario che ogni asse abbia un titolo o etichetta con il nome di ciò che rappresenta e l'unità di misura adottata.
2. È inoltre consigliabile che ogni asse abbia dei segni di graduazione regolarmente distanziati.
3. Se il grafico visualizza più curve bisogna adottare una legenda per caratterizzarle.
4. Se invece il grafico visualizza diversi insiemi di dati sperimentali è conveniente adottare dei simboli diversi come marcatore del dato.

Utilizzando le indicazioni appena descritte impareremo facilmente a tracciare i grafici in Matlab. Di seguito sono presentati i principali comandi per descrivere graficamente curve bidimensionali. Per rappresentare graficamente una funzione si utilizza il comando plot; la sintassi del comando è:

```
plot (vettorex, vettorey, 'opzioni')
```

dove:

- vettorex e vettorey sono i vettori di dati (rispettivamente ascisse e ordinate dei punti);
- opzioni è una stringa opzionale che definisce il tipo di colore, di simbolo e di linea che si vogliono usare nel grafico.

Si possono creare grafici multipli con una singola chiamata; Matlab traccia automaticamente i diversi grafici attraverso un predefinito (ma settabile dall'utente) elenco di colori che permette di distinguere ciascuna funzione.

Ad esempio, è possibile specificare colore, stile della linea e marcatori dei punti con il seguente comando:

```
plot (vettorex, vettorey, 'colore-stile-marcatore')
```

dove il colore-stile-marcatore è costituito da una sequenza di caratteri che rappresentano un colore, uno stile di linea ed un tipo di marcatore.

## Capitolo 2: Il database

Il database utilizzato è stato realizzato da alcuni colleghi che hanno eseguito diversi movimenti elementari con il dispositivo, come l'aumento e la diminuzione della quota, un'inclinazione avanti e indietro del dispositivo, una rotazione di 90° e così via.

Ogni movimento è stato registrato con i sensori di accelerometro, giroscopio e magnetometro. Per studiare più movimenti contemporaneamente è stata effettuata una passeggiata di 30 minuti attorno al complesso universitario di Monte Dago (Ancona).

Come vedremo, tutti i dati raccolti presentavano un rumore molto significativo, che rendeva difficile la lettura dei dati.

I dati raccolti si presentano suddivisi in colonne:

- Time stamp
- Orario
- Rilevazione lungo l'asse x
- Rilevazione lungo l'asse y
- Rilevazione lungo l'asse z

21567	084327.977706	0.775390625	0.1123046875	0.599609375
21573	084328.023409	0.771484375	0.1044921875	0.615234375
21579	084328.082595	0.7783203125	0.1103515625	0.6064453125
21586	084328.127728	0.771484375	0.11328125	0.6064453125
21592	084328.173422	0.7763671875	0.1044921875	0.6162109375
21598	084328.232617	0.775390625	0.1162109375	0.5986328125
21604	084328.277708	0.7734375	0.1083984375	0.619140625
21611	084328.323410	0.7763671875	0.109375	0.6005859375

Figura 2: Screen di una parte del database

## 2.1 Rotazione di 360 gradi

Questo test rientra tra gli esperimenti per registrare movimenti base, il dispositivo infatti è stato ruotato semplicemente di 360 gradi e sono state registrate le variazioni subite dall'oggetto con i sensori sopra citati.

Sono stati effettuati altri test del genere, come ad esempio una traslazione verso destra ed il ritorno nella posizione di partenza o anche un'inclinazione in avanti e indietro e sempre il ritorno nella posizione iniziale.

Di seguito è mostrato il codice per il caricamento dei dati e per la rappresentazione del loro andamento sullo schermo tramite un grafico:

```

clc
close all
clear all
format long
acc_matrix = importdata ("Accelerometro (Rotazione 360).txt");
timestamp = acc_matrix (:,1);           %uguale in tutti i 3 file
orario = acc_matrix (:,2);             %uguale in tutti i 3 file
acc_x_axis = acc_matrix (:,3);
acc_y_axis = acc_matrix (:,4);
acc_z_axis = acc_matrix (:,5);
deltaT = [0 ; diff(orario)];           %differenza tra le notifiche ricevute in microsecondi
figure('Name','Accelerometro','NumberTitle','off');
plot (orario, acc_x_axis, 'black', orario, acc_y_axis, 'red', orario, acc_z_axis, 'blue')
xlabel('Tempo [hhmmss.micros]')
ylabel('Forza di gravità [g]')
title('Accelerometro')

gyr_matrix = importdata ("Giroscopio (Rotazione 360).txt");
gyr_x_axis = gyr_matrix (:,3);
gyr_y_axis = gyr_matrix (:,4);
gyr_z_axis = gyr_matrix (:,5);
integrale_x = cumtrapz (gyr_x_axis);
integrale_y = cumtrapz (gyr_y_axis);
integrale_z = cumtrapz (gyr_z_axis);
gradi_x_axis = integrale_x .* deltaT;
gradi_y_axis = integrale_y .* deltaT;
gradi_z_axis = integrale_z .* deltaT;
figure('Name','Giroscopio','NumberTitle','off');
plot (orario, gradi_x_axis, 'black', orario, gradi_y_axis, 'red', orario, gradi_z_axis, 'blue')
xlabel('Tempo [hhmmss.micros]')
ylabel('Gradi [°]')
title('Giroscopio')

```

```

magn_matrix = importdata ("Magnetometro (Rotazione 360).txt");
magn_x_axis = magn_matrix (:,3);
magn_y_axis = magn_matrix (:,4);
magn_z_axis = magn_matrix (:,5);
figure('Name','Magnetometro','NumberTitle','off');
plot (orario, magn_x_axis, 'black', orario, magn_y_axis, 'red', orario, magn_z_axis, 'blue')
xlabel('Tempo [hhmmss.micross]')
ylabel('Induzione magnetica [ÅuT]')
title('Magnetometro')

##### confronto tra i 3 sensori #####
figure('Name','Confronto tra i 3 sensori','NumberTitle','off');
ax1 = subplot(3,1,1);
plot (ax1, orario, acc_x_axis, 'black', orario, acc_y_axis, 'red', orario, acc_z_axis, 'blue')
title(ax1,'Accelerometro')
ylabel(ax1,'Forza di gravità [g]')

```

In figura 3 viene mostrato l'output generato dal codice:

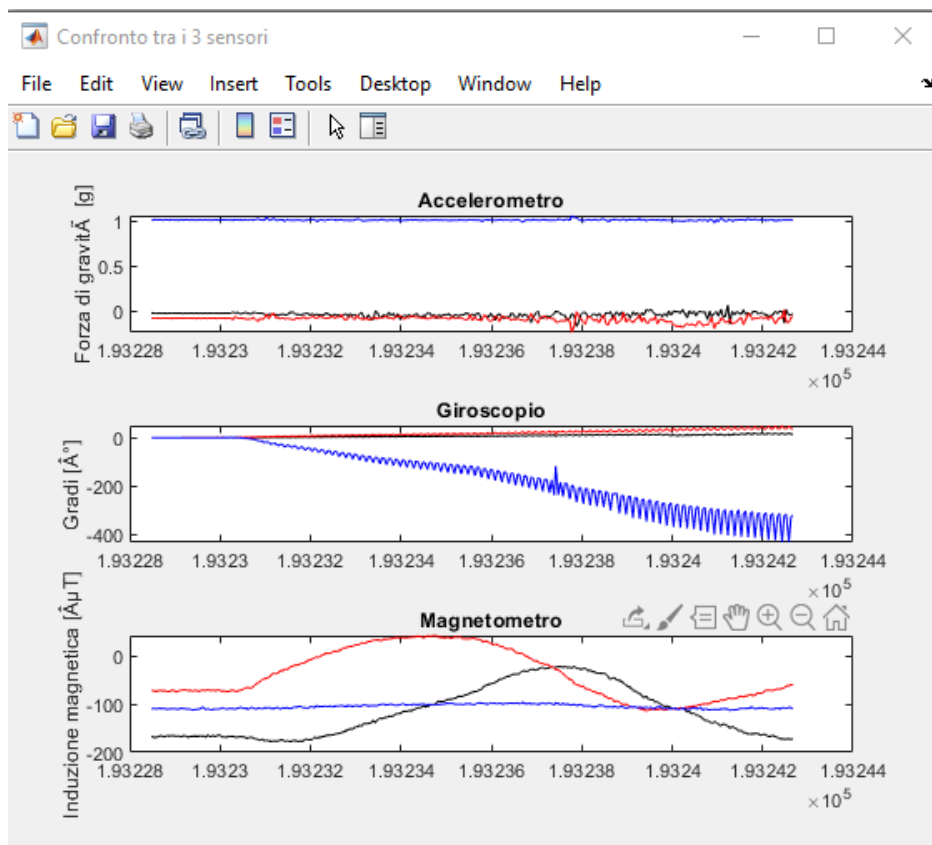


Figura 3: I risultati del test rotazione di 360 gradi

I grafici rappresentano le misure registrate dai singoli sensori sui 3 assi. L'andamento descritto dalla curva di color nero rappresenta ciò che avviene sull'asse X, il rosso sull'asse Y, il blu sull'asse Z.

## 2.2 Acquisizione 30 minuti

Questo è stato il test più complesso effettuato, poiché comprende i movimenti fondamentali del quotidiano e quindi una grande quantità di informazioni sottoforma di dati. Per effettuare questa raccolta dati un mio collega ha indossato uno zaino con al suo interno il sensore per l'acquisizione dei dati relativi all'accelerometro, al giroscopio e al magnetometro ed ha camminato per circa mezzora attorno al polo universitario. I sensori hanno registrato perciò diverse accelerazioni, cambiamenti di quota e rotazioni.

In figura 4 viene segnato in rosso il percorso fatto a piedi.



Figura 4: Il percorso effettuato nella passeggiata di circa 30 minuti

Di seguito viene riportato il codice usato in Matlab per l'elaborazione dei dati:

```

clc
close all
clear all
format long
acc_matrix = importdata ("Accelerometro.txt");
timestamp = acc_matrix (:,1);           %uguale in tutti i 3 file
orario = acc_matrix (:,2);             %uguale in tutti i 3 file
acc_x_axis = acc_matrix (:,3);
acc_y_axis = acc_matrix (:,4);
acc_z_axis = acc_matrix (:,5);
deltaT = [0 ; diff(orario)];           %differenza "temporale" tra le notifiche
%il vettore "orario" Ã¨ espresso in hhmmss.micros quindi ad ogni minuto
%successivo vi Ã¨ un aumento fittizio di 40 secondi e ad ogni ora successiva
%vi Ã¨ un aumento fittizio di 40 minuti
for i = 1:length(deltaT)
    if (deltaT(i)>1 && deltaT(i)<41)
        deltaT(i) = deltaT(i) - 40;
    end
    if (deltaT (i)>41)
        deltaT(i) = deltaT(i) - 4040;
    end
end

for i = 1:length(gradi_x_axis)
    while (abs(gradi_x_axis(i))>360)
        if (gradi_x_axis(i)>0)
            gradi_x_axis (i) = gradi_x_axis(i) - 360;
        end
        if (gradi_x_axis(i)<=-360)
            gradi_x_axis (i) = gradi_x_axis(i) + 360;
        end
    end
end
while (abs(gradi_y_axis(i))>360)
    if (gradi_y_axis(i)>0)
        gradi_y_axis(i) = gradi_y_axis(i) - 360;
    end
    if (gradi_y_axis(i)<=-360)
        gradi_y_axis(i) = gradi_y_axis(i) + 360;
    end
end
while (abs(gradi_z_axis(i))>360)
    if (gradi_z_axis(i)>0)
        gradi_z_axis(i) = gradi_z_axis(i) - 360;
    end
    if (gradi_z_axis(i)<=-360)
        gradi_z_axis(i) = gradi_z_axis(i) + 360;
    end
end

```



```

tempo = cumsum (deltaT);    %gli elementi di "tempo" sono espressi in secondi.microsecondi
figure('Name','Accelerometro','NumberTitle','off');
plot (tempo, acc_x_axis, 'black', tempo, acc_y_axis, 'red', tempo, acc_z_axis, 'blue')
xlabel('Tempo [s]')
ylabel('Forza di gravità [g]')
legend ('Asse x', 'Asse y', 'Asse z')
title('Accelerometro')

gyr_matrix = importdata ("Giroscopio.txt");
gyr_x_axis = gyr_matrix (:,3);
gyr_y_axis = gyr_matrix (:,4);
gyr_z_axis = gyr_matrix (:,5);
integrale_x = cumtrapz (gyr_x_axis);
integrale_y = cumtrapz (gyr_y_axis);
integrale_z = cumtrapz (gyr_z_axis);
gradi_x_axis = integrale_x .* deltaT;
gradi_y_axis = integrale_y .* deltaT;
gradi_z_axis = integrale_z .* deltaT;

    end
end
figure('Name','Giroscopio','NumberTitle','off');
plot (tempo, gradi_x_axis, 'black', tempo, gradi_y_axis, 'red', tempo, gradi_z_axis, 'blue')
xlabel('Tempo [s]')
ylabel('Gradi [°]')
legend ('Asse x', 'Asse y', 'Asse z')
title('Giroscopio')

magn_matrix = importdata ("Magnetometro.txt");
magn_x_axis = magn_matrix (:,3);
magn_y_axis = magn_matrix (:,4);
magn_z_axis = magn_matrix (:,5);
figure('Name','Magnetometro','NumberTitle','off');
plot (tempo, magn_x_axis, 'black', tempo, magn_y_axis, 'red', tempo, magn_z_axis, 'blue')
xlabel('Tempo [s]')
ylabel('Induzione magnetica [µT]')
legend ('Asse x', 'Asse y', 'Asse z')
title('Magnetometro')

figure('Name','Confronto tra i 3 sensori','NumberTitle','off');
ax1 = subplot(3,1,1);
plot (ax1, tempo, acc_x_axis, 'black', tempo, acc_y_axis, 'red', tempo, acc_z_axis, 'blue')
legend ('Asse x', 'Asse y', 'Asse z')
title(ax1,'Accelerometro')
ylabel(ax1,'Forza di gravità [g]')

ax2 = subplot(3,1,2);
plot (ax2, tempo, gradi_x_axis, 'black', tempo, gradi_y_axis, 'red', tempo, gradi_z_axis, 'blue')
legend ('Asse x', 'Asse y', 'Asse z')
title(ax2,'Giroscopio')
ylabel(ax2,'Gradi [°]')

ax3 = subplot(3,1,3);
plot(ax3, tempo, magn_x_axis, 'black', tempo, magn_y_axis, 'red', tempo, magn_z_axis, 'blue')
legend ('Asse x', 'Asse y', 'Asse z')
title(ax3,'Magnetometro')
ylabel(ax3,'Induzione magnetica [µT]')

```



Come viene mostrato in figura 5, i dati raccolti risultano essere molto rumorosi e quindi di scarsa utilità per gli scopi finali.

Il giroscopio ed in particolar modo la misura sull'asse z è risultato essere molto instabile e rumoroso, probabilmente condizionato anche dal fatto che il dispositivo si trovasse all'interno di uno zaino. Anche l'accelerometro, sebbene sia più stabile del giroscopio, risulta essere abbastanza rumoroso lungo la terna degli assi.

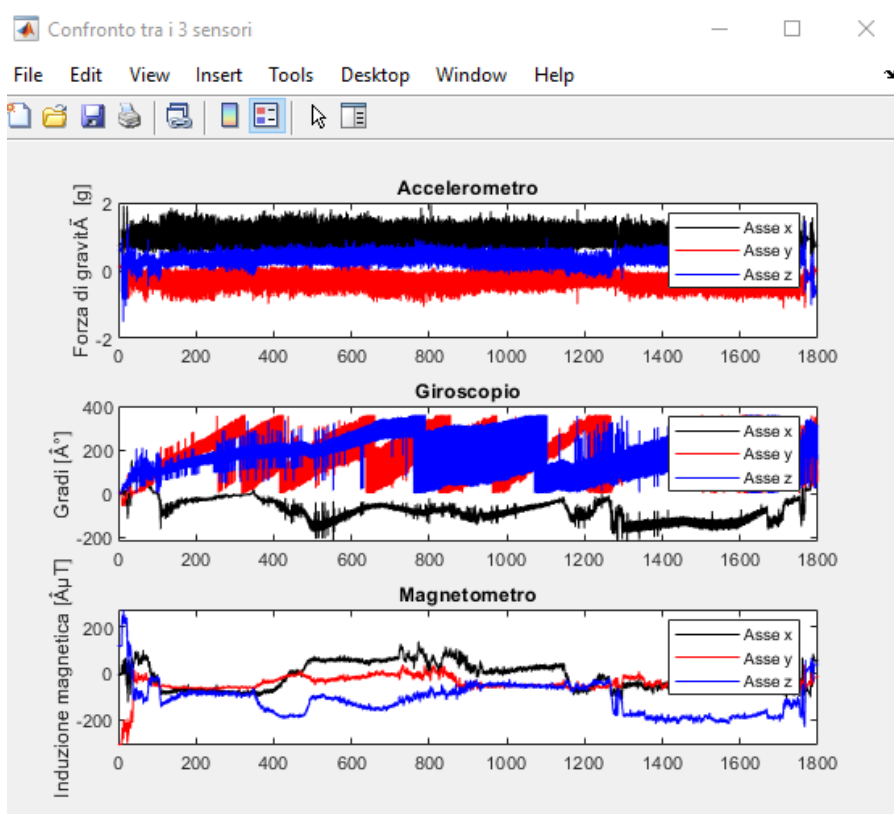


Figura 5: andamento dei dati raccolti durante la passeggiata di circa 30 minuti

## Capitolo 3: Le piattaforme inerziali

In questo capitolo andremo a studiare le componenti di una piattaforma inerziale e le spiegazioni fisiche che permettono la stima dell'assetto e della posizione tramite l'utilizzo di sensori. Un sensore, quindi acquisisce dati che verranno trasformati in informazioni che il calcolatore utilizzerà a seconda dell'utilizzo di cui l'utente necessita.

I sensori che analizzeremo sono l'accelerometro, il giroscopio ed il magnetometro.

### 3.1 Angoli di Eulero

Gli angoli di Eulero descrivono la posizione di un sistema di riferimento XYZ solidale con un corpo rigido attraverso una serie di rotazioni a partire da un sistema di riferimento fisso xyz. I due sistemi di riferimento coincidono nell'origine.

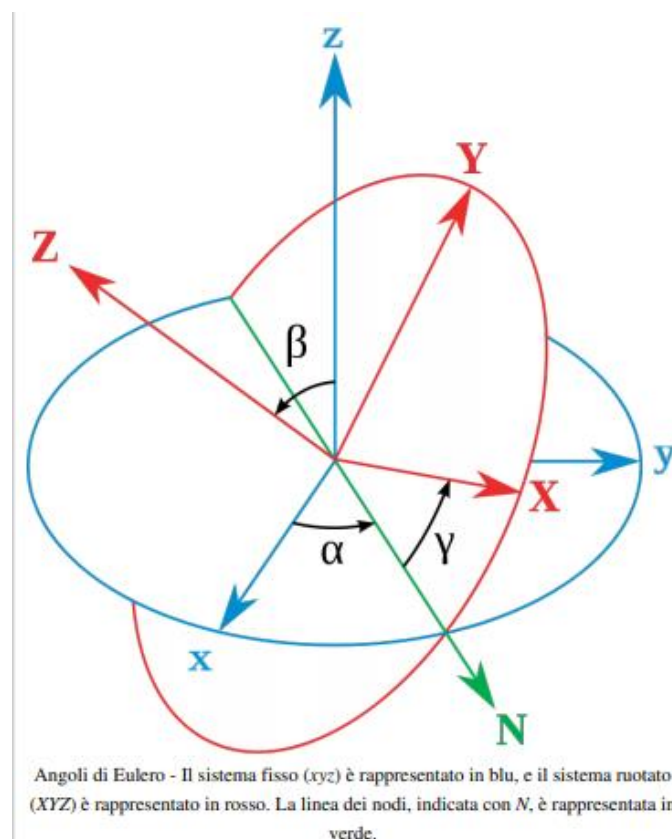


Figura 6: Rappresentazione degli angoli di Eulero

Se i piani  $xy$  e  $XY$  sono distinti, si intersecano in una retta (passante per l'origine) detta linea dei nodi (N). Se i piani coincidono, si definisce la linea dei nodi N come l'asse X. Gli angoli di Eulero sono i tre angoli seguenti:

- $\alpha$  è l'angolo tra l'asse  $x$  e la linea dei nodi. Detto angolo di precessione, o rollio, è definito in  $[0, 2\pi[$  oppure in  $[-\pi, \pi[$  ;
- $\beta$  è l'angolo tra gli assi  $z$  e  $Z$ . Detto angolo di nutazione, o beccheggio, è definito in  $[0, -\pi]$  oppure in  $[-\frac{\pi}{2}, \frac{\pi}{2}]$ ;
- $\gamma$  è l'angolo tra la linea dei nodi e l'asse X. Detto angolo di rotazione propria, o imbardata, è definito in  $[0, 2\pi[$  oppure in  $[-\pi, \pi[$  ;

Dal punto di vista dell'algebra lineare il passaggio dal sistema di riferimento fisso  $xyz$  a quello ruotato  $XYZ$  equivale ad operare un cambiamento di base, ovvero passare da una base  $(\hat{e}_1, \hat{e}_2, \hat{e}_3)$  (con  $\hat{e}_1, \hat{e}_2$  ed  $\hat{e}_3$ , rispettivamente i versori degli assi  $x, y$  e  $z$ ) ad una  $(\hat{i}, \hat{j}, \hat{k})$  (ovvero i versori degli assi  $X, Y$  e  $Z$ ) tramite una matrice di cambiamento di base  $A$ . La matrice  $A$  è una matrice quadrata  $3 \times 3$  ortogonale che rappresenta una rotazione nello spazio. Gli angoli di Eulero permettono di rappresentare  $A$  in forma relativamente semplice come moltiplicazione di 3 matrici di rotazione lungo i tre assi  $x, y$  e  $z$ .

## 3.2 Accelerometro

L'accelerometro è un dispositivo in grado di rilevare o misurare un'accelerazione, ovvero una variazione di velocità nel tempo e convertirla in un segnale elettrico. Nella maggior parte degli accelerometri, il principio di funzionamento è il medesimo: si basa sulla rilevazione dell'inerzia di una massa quando viene sottoposta ad un'accelerazione.

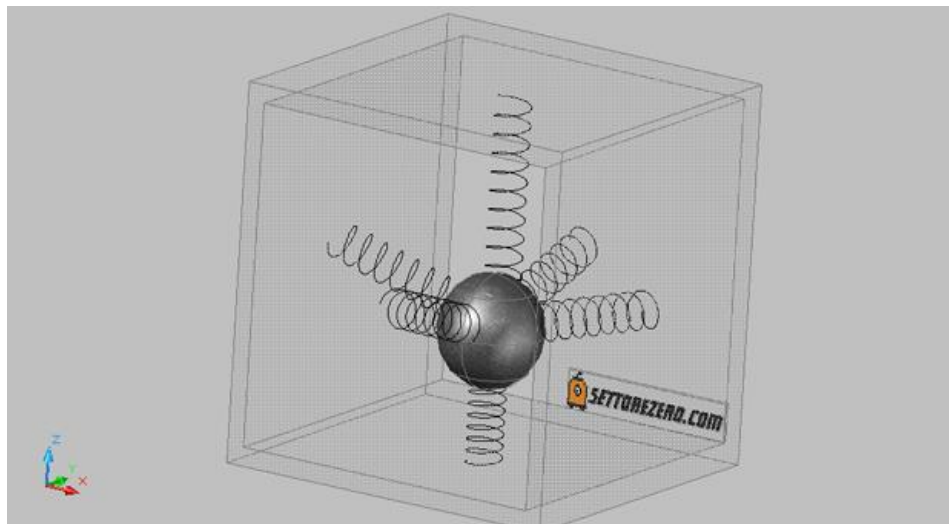


Figura 7: principio di funzionamento dell'accelerometro

Possiamo immaginare un accelerometro come un cubo con al suo interno una sfera sospesa da molle agganciate alle facce del cubo.

La sfera non è perfettamente al centro ma orientata verso il basso a causa della forza di gravità. L'accelerometro per come è realizzato, è in grado di misurare sia l'accelerazione statica (ad esempio quando lo ruotiamo sul posto) che l'accelerazione dinamica (ad esempio: lo fissiamo su un'automobile in corsa).

L'uso dell'accelerometro è aumentato notevolmente negli ultimi anni poiché accanto alle tradizionali applicazioni in ambito scientifico ed aerospaziale, è stato adottato nel campo domestico, industriale e professionale. Con il moltiplicarsi delle applicazioni, si sono diversificate anche le tipologie di questi strumenti ed oggi se ne possono contare decine di tipi, ognuno con caratteristiche funzionali e costruttive differenti.

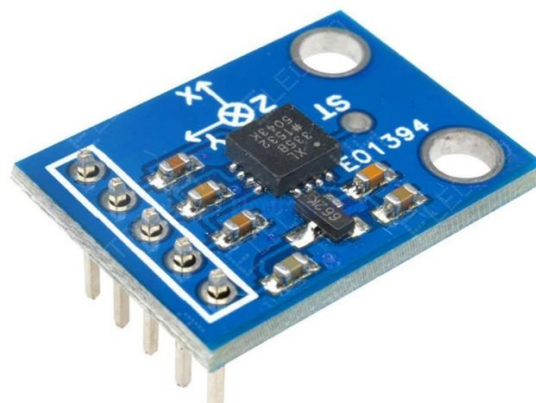


Figura 8: Esempio di sensore accelerometro

### 3.3 Giroscopio

Il giroscopio è un dispositivo fisico in grado di muoversi in modo peculiare sul proprio asse per misurare la velocità di rotazione sugli assi  $x$ ,  $y$ ,  $z$ . Viene utilizzato per misurare il moto rotazionale di un corpo. Il principio su cui si basa il giroscopio è quello della legge di conservazione del momento angolare, secondo cui il momento angolare di un sistema rimane costante a meno che agisca su di esso una forza esterna. Quando questo accade il corpo tende a mantenere un orientamento fisso, con l'asse che punta nella direzione della rotazione.

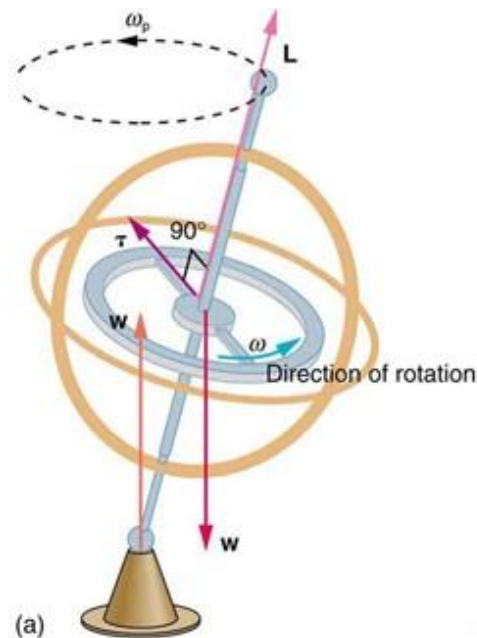


Figura 9: Modello di giroscopio

Essenzialmente è costituito da un rotore a forma di toroide che ruota intorno al suo asse, quando il rotore è in rotazione il suo asse tende a mantenersi parallelo a sé stesso e ad opporsi ad ogni tentativo di cambiare il suo orientamento. L'equazione fondamentale che descrive un qualunque sistema rigido in rotazione è:

$$M = \frac{dL}{dt} = \frac{d(I\omega)}{dt} = I\alpha$$

dove:

- $M$  è il vettore del momento meccanico
- $L$  è il vettore del momento angolare
- $I$  è il momento di inerzia
- $\omega$  è il vettore della velocità angolare
- $\alpha$  è il vettore dell'accelerazione angolare

Dall'equazione deriva che se viene applicato un momento meccanico  $M$  perpendicolarmente all'asse di rotazione, quindi perpendicolare anche ad  $L$ , si sviluppa una forza perpendicolare sia a  $M$  che ad  $L$ . Ne deriva un moto chiamato di precessione. La velocità angolare ( $\Omega_p$ ) di tale moto è data da:

$$M = \Omega_p \times L$$

Il giroscopio è stato utilizzato nella progettazione e realizzazione di molti oggetti, dai più semplici come la bicicletta, a dispositivi molto più complessi come il telescopio spaziale Hubble. Gli effetti giroscopici si possono riscontrare anche nei cuscinetti di un'automobile durante una sterzata o nei rapidi movimenti dell'hard disk di un computer.

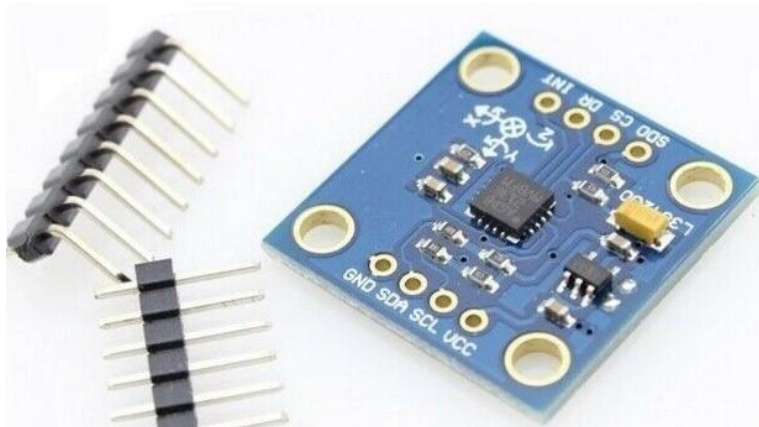


Figura 10: Esempio sensore giroscopio

## 3.4 Magnetometro

Il magnetometro è uno strumento che misura il magnetismo, quindi sia la magnetizzazione di un materiale come un ferromagnete, ma anche la direzione, la forza o relativa variazione di un campo magnetico in una determinata posizione.

Il vettore induzione magnetica è rappresentato con:

$$\vec{B} = \mu_0 \mu_r \vec{H}$$

Dove:

- H = campo magnetico
- $\mu_r$  = permeabilità magnetica relativa del mezzo
- $\mu_0$  = permeabilità magnetica nel vuoto

I campi magnetici sono vettori quantitativi caratterizzati da forza e direzione. La forza di un campo magnetico si misura in tesla (T).

Grazie alla misura del campo magnetico terrestre, comparata al riferimento gravitazionale misurato da un accelerometro tri assiale incorporato, questi sensori intelligenti consentono di calcolare con estrema accuratezza gli angoli di orientamento sui tre assi: rollio, beccheggio e imbardata.



Figura 11: La bussola è un semplice esempio di magnetometro



Esistono vari tipo di magnetometri, in base all'utilizzo che necessitiamo fare. Questi dispositivi, infatti, trovano impiego in una grande vastità di applicazioni, come ad esempio per indagini magnetiche nell'archeologia, infatti l'argilla dei mattoni e del vasellame acquista, per effetto della cottura, una propria magnetizzazione. Inoltre, i magnetometri vengono utilizzati per valutare il pericolo di estrazione del carbone dalle miniere o per scopi militari. Questi dispositivi sono installati anche negli smartphone e sulle navicelle spaziali.

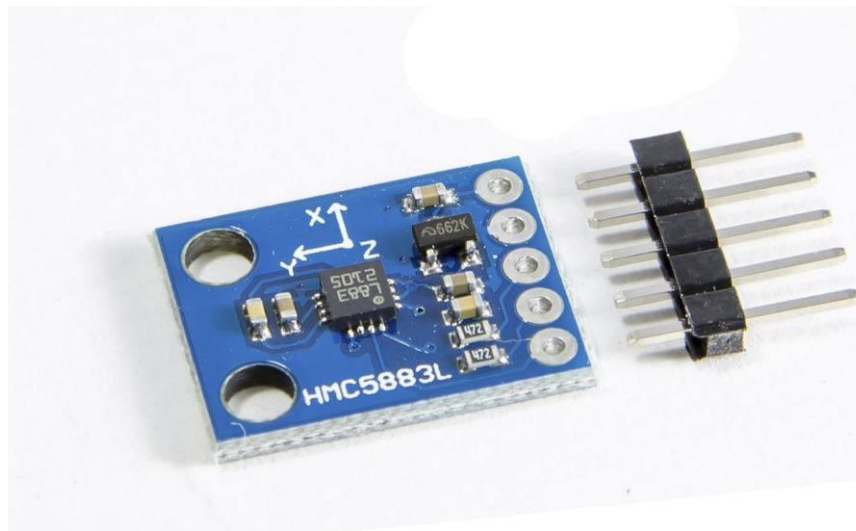


Figura 12: Esempio sensore magnetometro

### 3.5 Le piattaforme inerziali

La piattaforma inerziale, nota anche come IMU (inertial measurement unit), è un sistema elettronico per stimare la posizione, la velocità, le accelerazioni e l'orientamento di un veicolo senza la necessità di riferimenti esterni. La navigazione inerziale viene utilizzata in alternativa ai sistemi GPS, per tutti quei casi in cui il segnale GPS potrebbe essere non disponibile (come in contesti indoor) o non garantito (come nel caso di scenari militari, dove il segnale può essere oscurato da contromisure elettroniche). I sistemi inerziali vengono anche utilizzati in cooperazione con il GPS per migliorare l'accuratezza di breve periodo.

I sensori necessari sono quelli che permettono di descrivere il moto di un corpo rigido nello spazio, quindi accelerometro, giroscopio e



magnetometro. Conoscendo istantaneamente questi valori e il punto di partenza si può sapere esattamente dove è il veicolo e cosa sta facendo. Infatti, integrando due volte l'accelerazione sono in grado di risalire allo spostamento, mentre integrando una sola volta la velocità angolare riescano a capire l'orientazione del sistema.



Figura 13: Esempio di piattaforma inerziale.

Inizialmente furono ideate per essere montate sui missili per monitorare se il bersaglio venisse centrato. Ad oggi il loro impiego è più vasto ed ancora in fase di sviluppo per altre applicazioni sia militari che civili. Sulle moto, ad esempio, è installata una piattaforma inerziale che permette così alla centralina di sapere costantemente quale sia lo stato dinamico del veicolo.

# Capitolo 4: Il filtro di Kalman

In questo capitolo andremo ad analizzare il filtro di Kalman, argomento principale della tesi.

## 4.1 Cenni storici ed applicazioni

Il filtro di Kalman deve il suo nome a Rudolf Emil Kálmán, ingegnere e matematico statunitense. Nel 1967, durante una visita al Centro di Ricerche Ames della NASA presentò le sue idee a Stanley Schmidt, ingegnere aerospaziale che intuì le potenzialità del filtro. Egli fu il primo a sviluppare una realizzazione pratica del filtro di Kalman, introducendo così questo strumento nel computer di bordo del programma Apollo per la stima delle traiettorie. Il problema che si risolse con il filtro di Kalman fu quello del recupero della capsula al rientro. Il quesito era il seguente: data una serie di punti come tracce radar e dato il modello dinamico del rientro della capsula, cercare di individuare il punto di splash down più probabile e l'area di dispersione dello splash down stesso. Il suo contributo risultò quindi fondamentale per la riuscita della missione che portò l'uomo sulla luna nel 1969.

Nel 2009, grazie al significativo contributo dato alla scienza Rudolf Emil Kalman è stato fregiato della medaglia National Medal of Science dal presidente americano in carica Barak Obama.

Sono molteplici le applicazioni del filtro di Kalman, una comune è quella per la guida, la navigazione e il controllo di veicoli, in particolare sugli aeromobili e veicoli spaziali. Inoltre, è utilizzato in una vasta gamma di applicazioni: dai radar alla computer vision, ma anche per la stima dei modelli macroeconomici strutturali. Infine, è un concetto ampiamente applicato nell'elaborazione dei segnali e nell'econometria.

## 4.2 Funzionamento

Il filtro di Kalman è strumento ricorsivo per stimare lo stato di un sistema dinamico lineare (la variante del filtro di Kalman esteso si applica anche a sistemi non lineari), sulla base di misure corrotte da rumore. Da menzionare il concetto di filtro ottimo che definisce il filtro di Kalman in senso statistico ed equivale a dire che, se applicassimo dei filtri di prova molte volte per la stessa applicazione, allora la media dei risultati del filtro di Kalman sarebbe migliore della media dei risultati di qualsiasi filtro di prova.

Il filtro di Kalman consente di costruire un osservatore che fornisce la stima ottima dello stato bilanciando l'informazione estratta dal modello e quella estratta dalla misura a seconda dell'affidabilità delle informazioni. Per raggiungere questo obiettivo il filtro processa tutte le informazioni possibili, senza alcun riguardo alla loro precisione utilizzando:

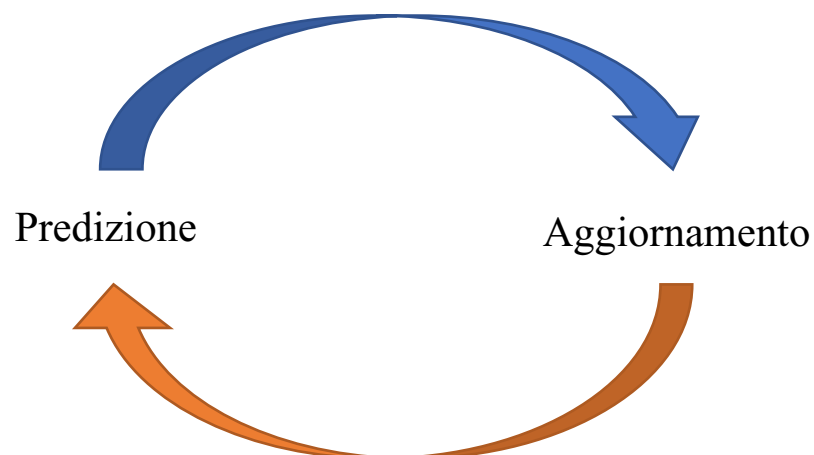
- la conoscenza del sistema e la sua dinamica di misura;
- la descrizione statistica del rumore associato al sistema, dell'errore di misura e l'incertezza dei modelli dinamici usati;
- ogni possibile informazione attorno alle condizioni iniziali delle variabili di interesse.

Il rumore sulla misura e l'incertezza sul modello possono essere modellati come dei processi stocastici, ovvero come variabili casuali.

Il filtro di Kalman utilizza processi Gaussiani per modellare le incertezze agenti sul sistema. Possiamo considerare il filtro essenzialmente come un insieme di equazioni matematiche che implementano uno stimatore di tipo predittore - correttore che è ottimale dal momento che minimizza la covarianza dell'errore stimata quando la condizione di linearità gaussiana del sistema è soddisfatta.

Possiamo descrivere il funzionamento del filtro di Kalman con alcuni passaggi chiave:

- Se non si ha nessuna stima dello stato iniziale, si inizializzi il filtro con un valore casuale e una matrice di covarianza grande (quanto dipende dall'applicazione, in modo da segnalare al filtro che la stima attuale è assolutamente inaccurata).
- Il filtro stima lo stato del processo in certi istanti di tempo e quindi realizza un feedback sulla base delle misure rumorose.
- Le equazioni del filtro di Kalman appartengono a due gruppi: predizione dello stato e aggiornamento basato sulle misure.
- Le equazioni di predizione dello stato proiettano in avanti lo stato corrente e la covarianza dell'errore di stima al fine di ottenere una stima a priori per il successivo istante temporale.
- Le equazioni di aggiornamento dello stato realizzano il meccanismo in retroazione, cioè incorporano le nuove misure nella stima a priori per ottenere una stima a posteriori migliorata.



## 4.3 Modello matematico e algoritmo

Dato un sistema lineare come quello in figura 10:

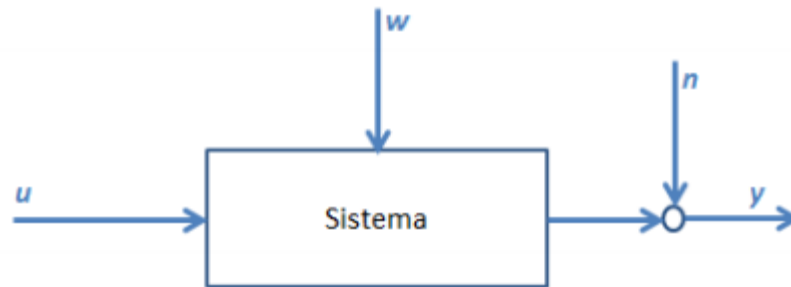


Figura 14: Esempio di sistema.

Le equazioni che lo caratterizzano sono:

$$\begin{cases} \dot{x} = Ax + Bu + \omega \\ y = Cx + n \end{cases}$$

dove:

- $x = nx1$  vettore di stato;
- $y = mx1$  vettore di misura;
- $w = nx1$  rumore di processo (disturbo del sistema);
- $n = mx1$  rumore di misura;

Da sottolineare che  $w$  e  $n$  sono rumori gaussiani bianchi.

L'attributo "bianco" per un rumore è riferito alle sue caratteristiche temporali (o di frequenza), mentre l'attributo "Gaussiano" è riferito alla sua ampiezza. Possiamo quindi dire che per ogni singolo istante temporale, la densità di probabilità dell'ampiezza di un rumore Gaussiano ha la forma di una campana. Questa assunzione è giustificata in senso fisico dal fatto che, tipicamente, vi è un gran numero di piccole sorgenti che contribuiscono a creare il rumore di misura. Matematicamente questo fenomeno è descritto dal teorema del limite centrale:

all'aumentare del numero di variabili casuali indipendenti che si sommano tra loro, qualunque sia la distribuzione di probabilità di ciascuna, la distribuzione di probabilità della somma tende ad essere quella Gaussiana. Oltre a questa giustificazione se ne può aggiungere una di ordine pratico: di solito di una variabile affetta da rumore si conoscono solo le statistiche del primo e del secondo ordine, rispettivamente la media e la varianza (o la deviazione standard).

In tal caso, cioè quando non si conoscono altre statistiche di ordine superiore al secondo, non vi è altra distribuzione da assumere migliore di quella Gaussiana.

In generale, infatti, il rumore di fondo sovrapposto ad un segnale misurato è spesso assimilabile ad una distribuzione di questo tipo in virtù delle seguenti caratteristiche:

- Ha media nulla;
- È più probabile che il rumore abbia un valore assoluto prossimo a zero piuttosto che valori grandi;
- È simmetrico rispetto allo zero;

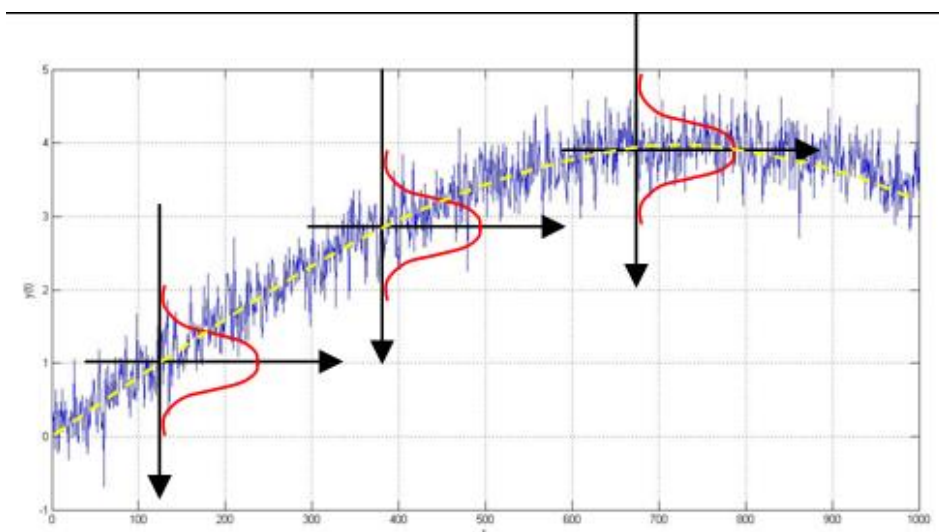


Figura 15: rumore gaussiano a media nulla

Torniamo al sistema di figura 10, lo scopo ora è quello di trovare una stima  $\hat{x}$  del vettore di stato  $x$  a partire da misure incomplete e rumorose. L'equazione del filtro di Kalman è la seguente:

$$\dot{\hat{x}} = Ax + Bu + L(y - C\hat{x})$$

$L$  è una matrice  $n \times m$  e rappresenta il guadagno del filtro. Esso ci garantisce che la stima dell'errore converga a zero.

$$e = x - \hat{x}$$

$$\lim_{t \rightarrow \infty} e(t) \rightarrow 0$$

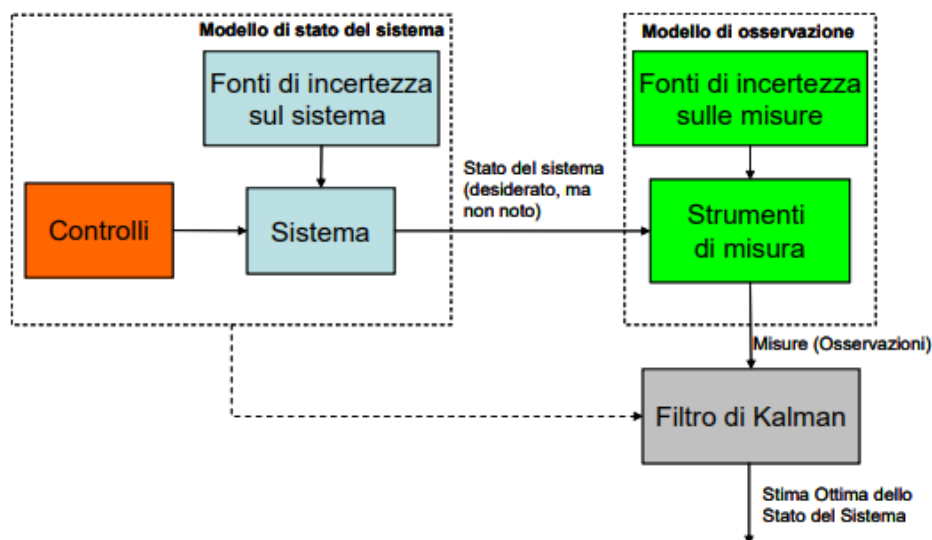


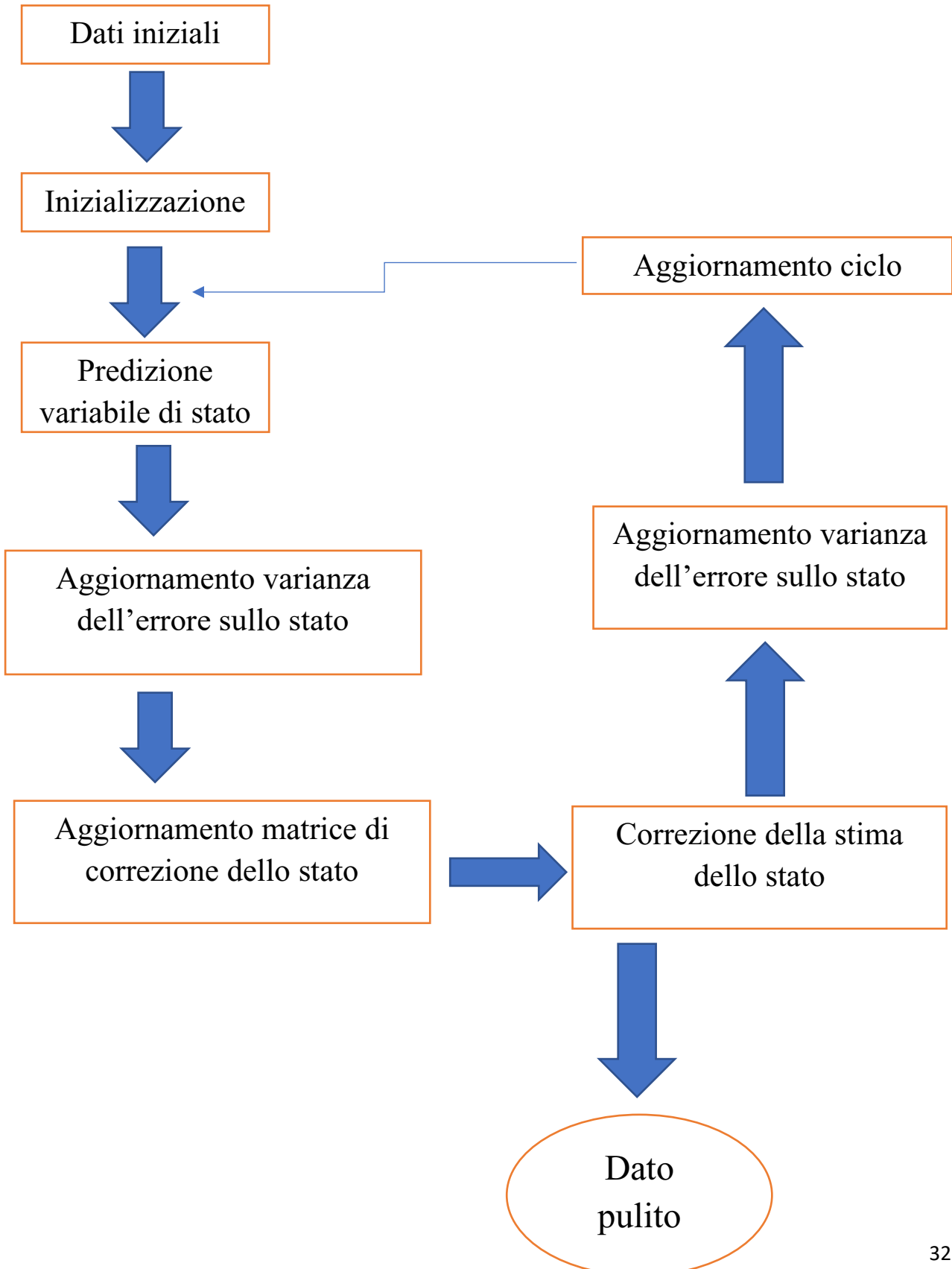
Figura 16: Schema a blocchi del filtro di Kalman

Il filtro di Kalman è un algoritmo ricorsivo, cioè per calcolare un nuovo stato richiede solo l'ultima "ipotesi migliore", piuttosto che l'intera storia dello stato di un sistema. Questo è fondamentale per un'implementazione al calcolatore del filtro di Kalman, esso indica infatti che l'algoritmo non richiede la memorizzazione e la rielaborazione di tutta la storia dei dati ogni volta che è disponibile una nuova misura, ma sono necessari solo lo stato stimato all'istante precedente e la misura attuale per calcolare la stima dello stato corrente.

Il filtro deve essere inizializzato e poi cicla, come già visto tra due stadi:

- Predizione
- Aggiornamento

Possiamo schematizzare l'algoritmo del filtro di Kalman in questo modo:





## 4.4 Filtro di Kalman Esteso

Il filtro di Kalman esteso *Extended Kalman Filter* (EKF) è una versione non-lineare del filtro di Kalman usata quando l'evoluzione o l'osservazione dello stato del sistema sono non-lineari.

Un sistema non lineare a tempo discreto, formato dall'evoluzione dello stato e dalla sua osservazione, può essere scritto in forma generalizzata come:

$$\begin{cases} x_{k+1} = f(x_k, u_k, w_k) \\ z_k = h(x_k, v_k) \end{cases}$$

dove, oltre allo stato  $x_k$  e agli ingressi  $u_k$ , anche gli errori di processo  $w_k$  e di osservazione  $v_k$  possono influire in maniera non lineare nell'evoluzione dello stato  $f$  e nell'osservazione  $h$ .

Per poter essere applicato, EKF richiede il calcolo degli Jacobiani sia di  $f$  che di  $h$ . Attraverso le matrici delle derivate, infatti, è possibile sfruttare le stesse formulazioni matematiche fatte per il caso di Kalman lineare su funzioni non-lineari usando come matrici:

$$\begin{aligned} A_k &= \left. \frac{\partial f(x, u_k, \bar{w})}{\partial x} \right|_{\hat{x}_{\bar{k}}} & W_k &= \left. \frac{\partial f(\hat{x}_{\bar{k}}, u_k, w)}{\partial w} \right|_{\bar{w}} \\ H_k &= \left. \frac{\partial h(x, \bar{v})}{\partial x} \right|_{\hat{x}_{\bar{k}}} & V_k &= \left. \frac{\partial h(x, v)}{\partial v} \right|_{\bar{v}} \end{aligned}$$

e usando come equazione di aggiornamento:

$$\hat{x}_k = \hat{x}_{\bar{k}} + K_k(z_k - h(\hat{x}_{\bar{k}}))$$

La linearizzazione viene effettuata rispetto ad una “traiettoria” di riferimento, intendendo genericamente con “traiettoria” l'andamento nel tempo degli stati del sistema. Ci sono due possibilità per la scelta della traiettoria di riferimento: ci si può basare su una traiettoria predefinita a priori oppure si possono prendere come riferimento le stesse stime calcolate dal filtro.

# Capitolo 5: Ahrsfilter basato sul filtro di Kalman

## 5.1 Introduzione

L'Attitude Heading Reference System (AHRS) è un sistema capace di mantenere una stima accurata dell'assetto del corpo mentre esso si muove. A differenza degli algoritmi INS (Inertial Navigation System) dove sono disponibili anche le informazioni di velocità, in uno stimatore dell'assetto le uniche misure considerate disponibili sono quelle derivanti dalla IMU, e quindi le misure di velocità angolare date dai giroscopi, quelle di forza specifica date dagli accelerometri, e la misura del campo magnetico terrestre dato dai magnetometri.

In linea di principio, se si dispone di informazioni esatte sull'assetto iniziale e di giroscopi di elevata accuratezza, l'integrazione delle equazioni differenziali che descrivono l'evoluzione nel tempo dell'assetto in funzione delle velocità angolari è sufficiente a fornire gli angoli di assetto del velivolo con bassi errori per lunghi periodi di funzionamento. A causa degli errori sull'assetto iniziale, degli errori di integrazione numerica delle equazioni differenziali e soprattutto a causa delle imperfezioni dei giroscopi, questo tipo di soluzione comporta dei crescenti errori di assetto che limitano il tempo di funzionamento entro il quale i dati forniti dal sistema sono sufficientemente accurati.

Il problema principale nella realizzazione dell'AHRS secondo questa soluzione concettualmente elementare è legato alla necessità di usare dei giroscopi dai costi e dagli ingombri improponibili per la maggior parte delle applicazioni a cui il sistema è destinato. La soluzione di base per la realizzazione dell'AHRS, che consente di abbassare i costi e gli ingombri del sistema, impiega giroscopi di qualità inferiore a quelli necessari nel caso precedente e si avvale di sensori aggiuntivi con i quali limitare gli errori di assetto. Un sistema di questo tipo è costituito da giroscopi, accelerometri e magnetometri, tutti triassiali ed installati su piattaforma inerziale.

## 5.2 Funzionamento

Il principio di funzionamento è il seguente:

- Partendo da valori iniziali ottenuti dagli accelerometri, gli angoli di assetto vengono calcolati ed aggiornati ad alta frequenza mediante l'integrazione delle velocità angolari. A causa dei rilevanti errori di misura dei giroscopi, si sviluppano degli errori di assetto che crescono rapidamente nel tempo. Gli angoli di beccheggio e di rollio così calcolati vengono inviati al filtro di Kalman.
- Ad opportuni intervalli di tempo, dalle misure degli accelerometri si effettua una rilevazione del vettore gravità, mediante il quale si ricava la misura degli angoli di beccheggio e di rollio in modo indipendente dall'integrazione delle velocità angolari. Questi angoli di assetto, affetti anch'essi da errori, vengono inviati a loro volta al filtro di Kalman.
- Il filtro elabora i dati provenienti dal calcolo e dalla misura dell'assetto per ottenere degli angoli di beccheggio e di rollio aggiornati ad alta frequenza ed affetti da errori limitati. Le misure dell'assetto vengono impiegate dal filtro per stimare e correggere gli errori che si sviluppano nell'integrazione delle velocità angolari.

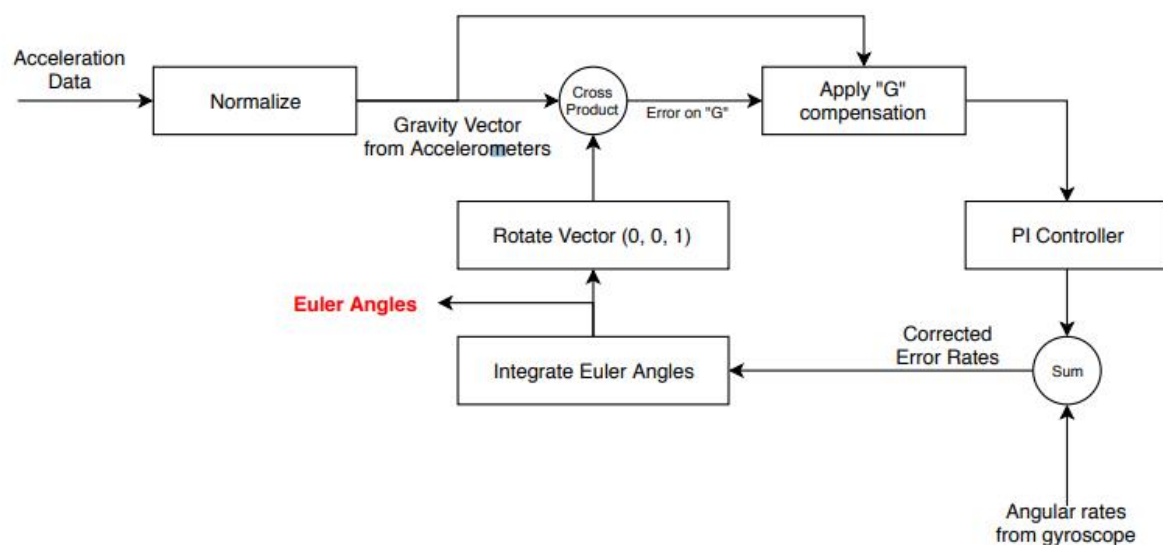


Figura 17: Schema funzionamento ahrsfilter

## 5.2 Proprietà del filtro

La differenza chiave tra un'unità di misura inerziale (IMU) e un AHRS è l'aggiunta di un sistema di elaborazione di bordo che fornisce informazioni sull'assetto e la direzione dell'oggetto in esame restituendo le informazioni relative al beccheggio, all'imbardata e al rollio. Con il data fusion, la deriva dall'integrazione dei giroscopi è compensata da vettori di riferimento, vale a dire la gravità e il campo magnetico terrestre.



Figura 18: Rappresentazione grafica di rollio, beccheggio e imbardata

Il filtro utilizza un vettore di stato a 12 elementi per tenere traccia dell'errore di stima per l'orientamento, la polarizzazione del giroscopio, l'accelerazione lineare e il disturbo magnetico.

Le principali proprietà impostate di default nel filtro sono:

- Accelerometro:

Varianza del rumore del segnale =  $0.00019247 \text{ (m/s}^2\text{)}^2$

Varianza del rumore di accelerazione lineare =  $0.0096236 \text{ (m/s}^2\text{)}^2$

- Magnetometro:

Varianza del rumore del segnale =  $0.1 \mu\text{T}^2$

Varianza del rumore di disturbo magnetico =  $0.5 \mu\text{T}^2$

- Giroscopio:

Varianza del rumore del segnale =  $9.1385e^{-5} (\text{rad/s})^2$

Varianza della deriva di offset del giroscopio =  $3.0462e^{-13} (\text{rad/s})^2$

## 5.3 Implementazione in Matlab

Il codice in Matlab per la realizzazione di un ahrsfilter è il seguente:

```
accelerometerReadings = accelerometerReading;
gyroscopeReadings = gyroscopeReading;
magnetometerReadings = magnetometerReading;
decim = 2;
fuse = ahrsfilter('SampleRate',Fs,'DecimationFactor',decim);
q = fuse(accelerometerReadings,gyroscopeReadings,magnetometerReadings);
time = (0:decim:size(accelerometerReadings,1)-1)/Fs;
figure(1)
plot(time,eulerd(q,'ZYX','frame'))
title('Orientation Estimate')
legend('z-axis', 'y-axis', 'x-axis')
ylabel('Rotation (degrees)')
```

L'algoritmo presuppone che il dispositivo sia fermo prima della prima chiamata. Si inizia creando le matrici e caricandole nell'ambiente di lavoro. Successivamente, con la funzione Fuse, come detto in precedenza, si fondono i dati con la frequenza di campionamento desiderata ( $F_s$ ). In questo modo viene fornita una stima dell'orientamento del corpo nel tempo.

Infine, l'orientamento è definito dallo spostamento angolare necessario per ruotare un sistema di coordinate padre, in un sistema di coordinate

figlio così da tracciare l'orientamento negli angoli di Eulero in gradi nel tempo. È indispensabile disporre di relazioni che leghino i quaternioni e gli angoli di Eulero perché questi ultimi costituiscono una descrizione più pratica dell'assetto.

I dati relativi all'accelerometro, al giroscopio e al magnetometro sono raccolti in matrici  $N \times 3$ . Le colonne di tale matrice rappresentano rispettivamente gli assi x-y-z, mentre le righe corrispondono ai singoli campioni

La funzione chiave del filtro è la seguente:

FUSE = ahrsfilter

Tale funzione restituisce un oggetto indiretto del filtro Kalman per fondere i dati dell'accelerometro, del giroscopio e del magnetometro e per stimare l'orientamento del dispositivo e la velocità angolare. Questa funzione rappresenta perciò il data fusion dell'algoritmo.

# Capitolo 6: Risultati

L'ahrsfilter ha generato buoni risultati e grazie ai grafici prodotti in output i dati sono diventati più leggibili e comprensibili.

## 6.1 Output

Di seguito sono mostrati i risultati prodotti dall'ahrsfilter. I grafici vengono descritti dall'asse X (sul quale c'è il tempo) e dall'asse Y (dove troviamo i gradi relativi alla rotazione dell'oggetto in esame).

Dal filtraggio dei dati riguardanti la rotazione di 360 gradi è stato generato il seguente l'output:

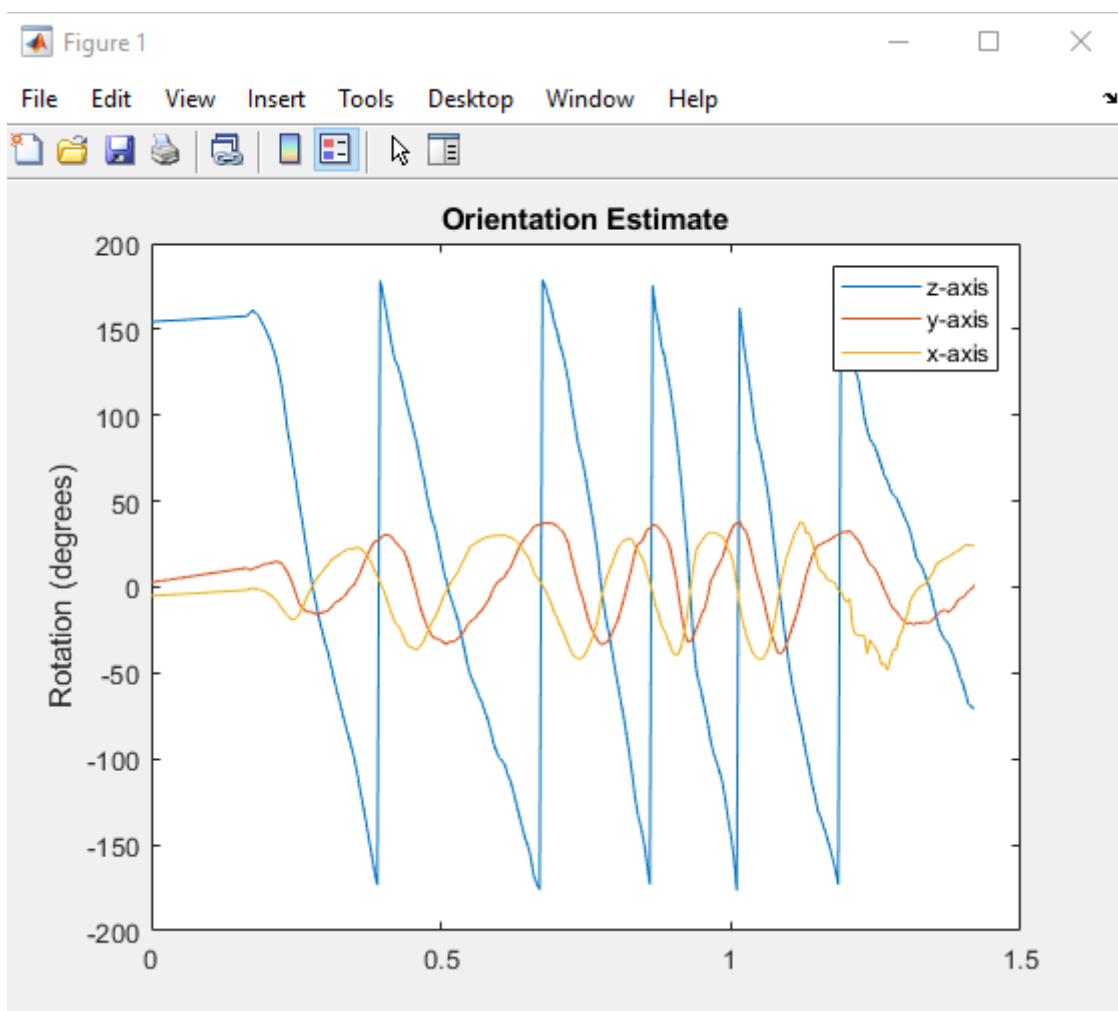


Figura 19: Rotazione 360 gradi filtrata

Ponendo a confronto i grafici in figura 3 e figura 19 possiamo notare come il segnale, una volta filtrato, risulta essere molto più leggibile e il rumore è decisamente attenuato.

Filtrando i dati riguardanti la passeggiata di 30 minuti l'output generato è mostrato in figura 20:

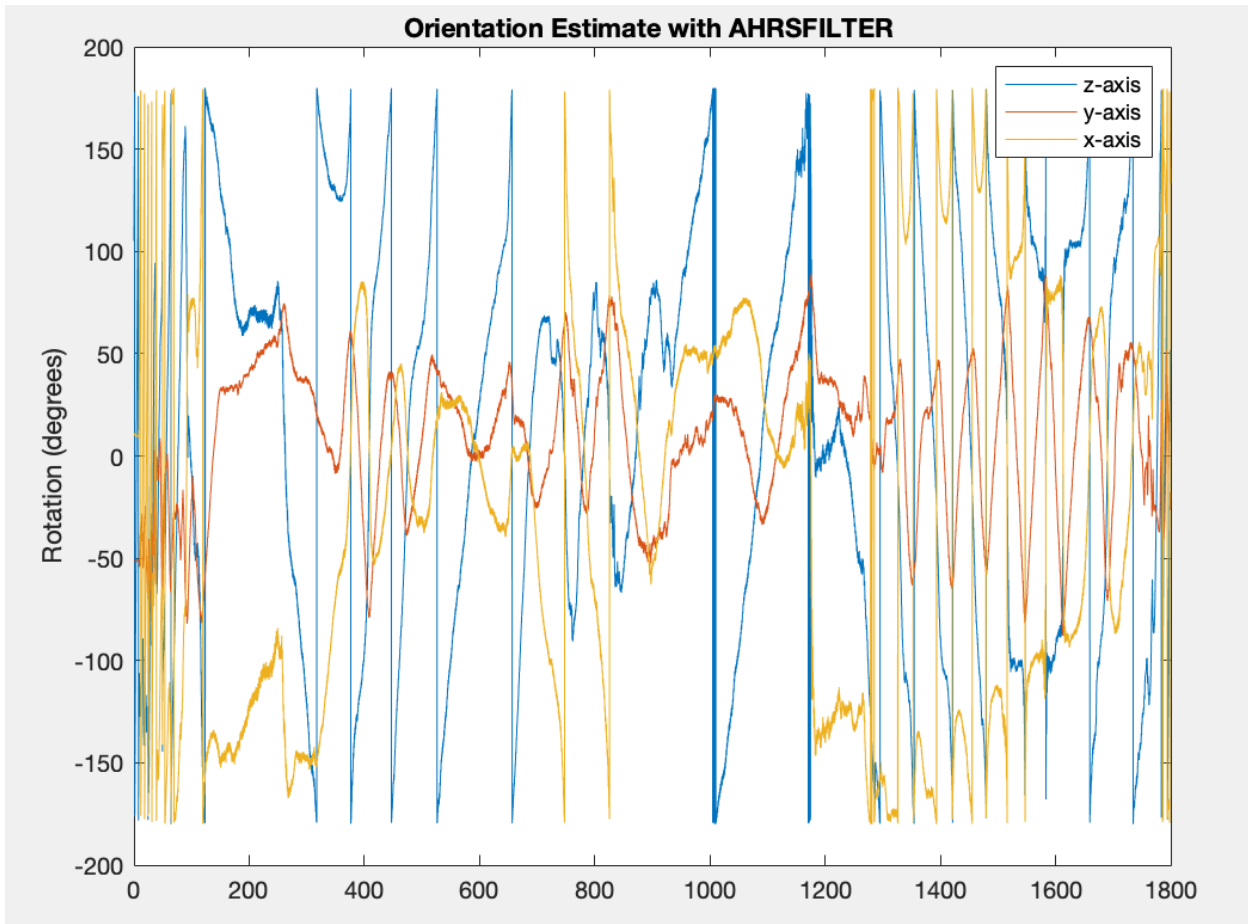


Figura 20: Passeggiata di 30 minuti filtrata

Anche in questo caso possiamo porre a confronto l'output iniziale del nostro test (figura 5) non filtrato con l'output generato dall'ahrsfilter basato sul filtro di Kalman (figura 20). Questo filtraggio è stato effettuato con una frequenza di campionamento di 20 Hz. Nonostante non sia un segnale totalmente pulito dal rumore, possiamo comunque notare una significativa riduzione del disturbo e di conseguenza una più facile lettura dei dati. L'ahrsfilter si è dimostrato quindi essere molto utile per questo tipo di dati da filtrare grazie anche alla sua facile implementazione.



# Conclusioni

In questa attività di tesi è stata introdotta un'analisi di segnali presi da sensori IMU, questi dati sono stati trattati nel modo migliore possibile cercando di stimare nel modo più corretto e preciso l'orientazione dell'oggetto in esame sia nel tempo che nello spazio. Nei limiti delle simulazioni svolte, il filtro utilizzato si è dimostrato in grado di conseguire delle prestazioni di livello elevato. Sono stati studiati algoritmi di tipo stocastico basati sul filtro di Kalman presenti in letteratura e l'esplorazione algoritmica svolta durante l'attività di tirocinio ha confermato l'ottima qualità del filtro di Kalman e dei filtri costruiti partendo da esso. Nel capitolo 4 si è studiato con accuratezza il filtro e nel capitolo 5 è stato introdotto l'uso dell'ahrsfilter, un metodo veloce e ad alte prestazioni che permette un utilizzo più vasto del filtro di Kalman semplificandolo. Grazie al sistema di riferimento NED (North, East, Down) l'ahrsfilter risulta essere più preciso poiché l'asse  $y$  dell'oggetto punta sempre alla sua destra, mentre il suo asse  $z$  positivo punta sempre verso il basso.

Una possibile implementazione futura per aumentare l'accuratezza e la precisione del filtro potrebbe essere quella di applicare un filtro passa basso ai segnali acquisiti dal giroscopio. Le misure di tale sensore, infatti, risultano essere le più influenzate dal rumore, inoltre, il giroscopio è il dispositivo dove il filtraggio di Kalman risulta essere meno efficace. Successivamente si dovrebbe procedere creando un altro algoritmo di data fusion per la stima dell'orientamento del corpo. Così facendo il risultato finale risulterebbe sicuramente più accurato e la precisione della stima aumenterebbe.

## Bibliografia

[https://www.sci.unich.it/~acciaro/guida\\_a\\_matlab.pdf](https://www.sci.unich.it/~acciaro/guida_a_matlab.pdf)

Guzzi R. (2012) I filtri di Kalman. In: Introduzione ai metodi inversi. UNITEXT - Collana di Fisica e Astronomia, vol 32. Springer, Milano

<http://www.dii.unimore.it/~lbiagiotti/MaterialeTSC1516/TSC-03-KalmanFilter.pdf>

<http://www.automazione.ingre.unimore.it/pages/corsi/materiale didattico/CRI1112/CRI07-II%20Filtro%20di%20Kalman.pdf>

<http://www.miro.ing.unitn.it/download/Didactics/RoboticPerceptionAction/SensorFusion/Dispensa%20SENSOR%20FUSION%20def.pdf>

<https://core.ac.uk/download/pdf/14696257.pdf>

<https://www.math.unipd.it/~michela/brevimatlabplot.pdf>

[https://dipmat.univpm.it/~demeio/public/Meccanica%20Razionale%20-%20Materiale/Angoli\\_Eulero\\_Wiki\\_1.pdf](https://dipmat.univpm.it/~demeio/public/Meccanica%20Razionale%20-%20Materiale/Angoli_Eulero_Wiki_1.pdf)

## Sitografia

<https://www.settorezero.com/wordpress/cosa-sono-come-funzionano-e-a-cosa-servono-gli-accelerometri/>

<https://www.thedifferentgroup.com/2017/02/10/giroscopio-come-funziona/>