



UNIVERSITÀ DI PISA
Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in
INGEGNERIA ROBOTICA E DELL'AUTOMAZIONE

Loosely coupled GPS/INS integration for UAV Applications

DOCENTE:
Lorenzo Pollini

STUDENTE:
Alessio Tumminello

Indice

Introduzione	3
EKF	3
Loosely coupled integration	3
Classificazione dei sistemi di riferimento	4
• Sistema assi corpo (b-frame):	4
• Sistema ECI - Earth Centered Inertial (i-frame):.....	4
• Sistema ECEF – Earth Centered Earth Fixed (e-frame):.....	4
• Sistema Geodetico (g-frame):	5
• Sistema di navigazione NED North – Est – Down (n-frame):.....	5
Simulazione dei dati e dei sensori.....	6
Simulazione dei sensori IMU e GPS.....	7
Stima dell’orientamento e del bias del giroscopio	10
Stima a priori	11
Predizione	12
Correzione	12
Stima della posizione, della velocità e del bias dell’accelerometro.....	13
Stima a priori	16
Predizione	17
Correzione	17
Risultati ottenuti.....	18
Bibliografia	23

Introduzione

La progettazione di un sistema di navigazione autonomo per un UAV si basa sui dati forniti da una suite di sensori che solitamente è costituita da un sistema di navigazione inerziale (INS) che fornisce accelerazioni, velocità angolari e assetto del robot, un GPS, il quale provvede alle misure di posizione e velocità al suolo e infine un magnetometro.

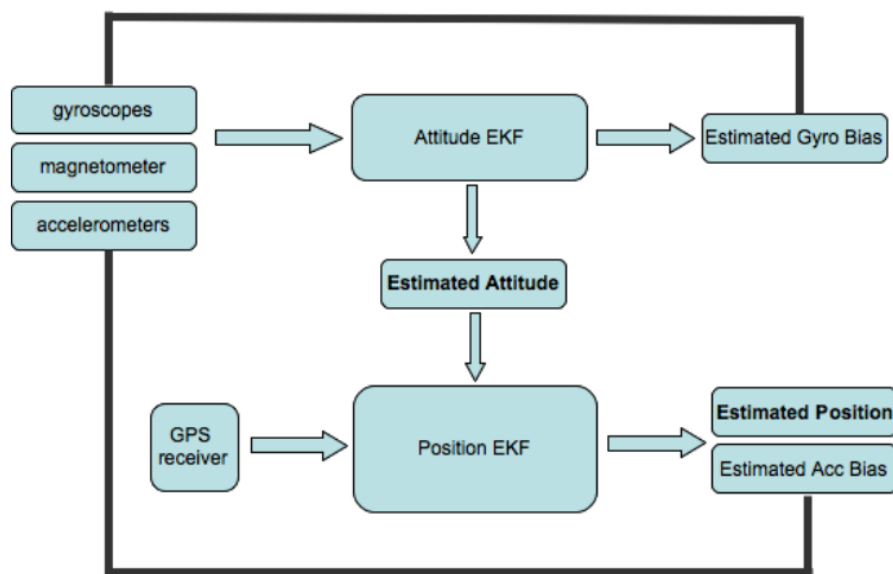
EKF

La perdita di precisione dovuta all'utilizzo di sensori 'low cost' viene superata tramite un'integrazione tra i dati inerziali e GNSS (global navigation satellite system) mediante un algoritmo di fusione basato su una tecnica di Extended Kalman Filtering (EKF), la quale provvede ad una stima dell'assetto, della velocità e della posizione del robot. L'utilizzo di questa tecnica ha il vantaggio di essere semplice da implementare ma di contro, dato che sfrutta una linearizzazione, potrei non avere una buona approssimazione dettata dal fatto che lo stato iniziale che fornisco è diverso da quello reale.

Loosely coupled integration

In questa applicazione è stata utilizzata la cosiddetta 'Loosely coupled integration', essa sfrutta le misurazioni del GPS anche per calibrare l'unità inerziale; infatti, vengono progettati due EKF, il primo determina l'orientamento stimato e il bias stimato del giroscopio sulla base dei dati forniti dall'accelerometro, dal giroscopio e dal magnetometro, il secondo fornisce la posizione stimata e il bias stimato dell'accelerometro sulla base di dati forniti dal GPS. Le misure di bias stimato sia del giroscopio che dell'accelerometro vengono utilizzate per ridurre l'errore presente in questi ultimi sensori.

Schema a blocchi



Capitolo 1

Classificazione dei sistemi di riferimento

I sistemi di riferimento analizzati sono:

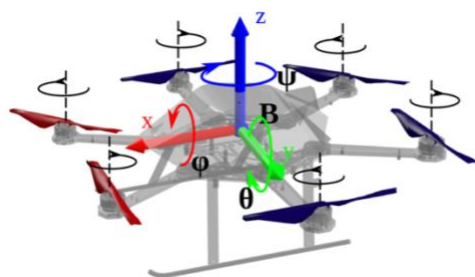
- **Sistema assi corpo (*b-frame*):**

È il sistema di riferimento del robot, con origine nel suo baricentro (punto O). Convenzionalmente gli assi sono allineati con la direzione di moto (X), la verticale locale (Z) e la direzione data dal completamento della terna destra (Y).

Le misure di accelerometri e giroscopi sono indicate in frame body.

Si assumerà quindi:

- $\omega_{i/b}^b$: vettore di velocità angolare, misurata dai giroscopi, della piattaforma rispetto al frame inerziale;
- a_b : vettore di accelerazione della piattaforma misurata dagli accelerometri.

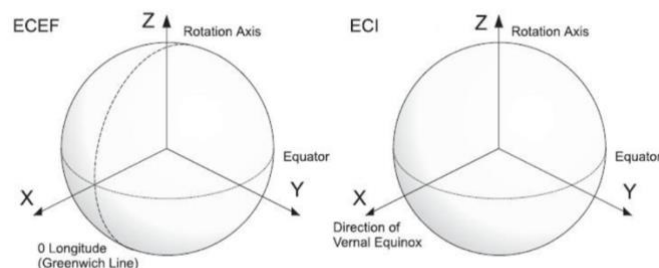


- **Sistema ECI - Earth Centered Inertial (*i-frame*):**

È un sistema con origine al centro della Terra, è coincidente con il riferimento inerziale ed è fisso rispetto al moto di rotazione terrestre.

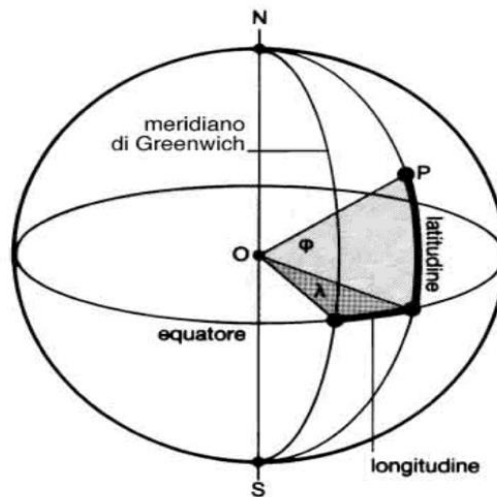
- **Sistema ECEF – Earth Centered Earth Fixed (*e-frame*):**

È un sistema non inerziale con origine al centro della Terra, solidale con essa e con il suo moto di rotazione.



- **Sistema Geodetico (*g-frame*):**

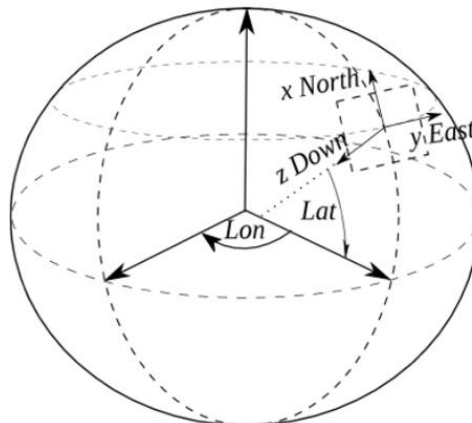
Le coordinate di un punto P in un sistema ECEF possono essere individuate con 3 parametri: Latitudine (Φ), Longitudine (λ) e altitudine (h).



- **Sistema di navigazione NED North – Est – Down (*n-frame*):**

È un frame locale con origine nel sistema di navigazione considerato, punto P, ed assi allineati con la verticale locale (D - down) e le direzioni nord (N) ed est (E).

La verticale locale D è definita dalla normale rispetto all'elissoide di riferimento. Le direzioni N ed E si riferiscono rispettivamente alla direzione di proiezione del vettore di velocità angolare terrestre lungo il piano orizzontale di navigazione ed alla direzione definita per il completamento della terna destra.

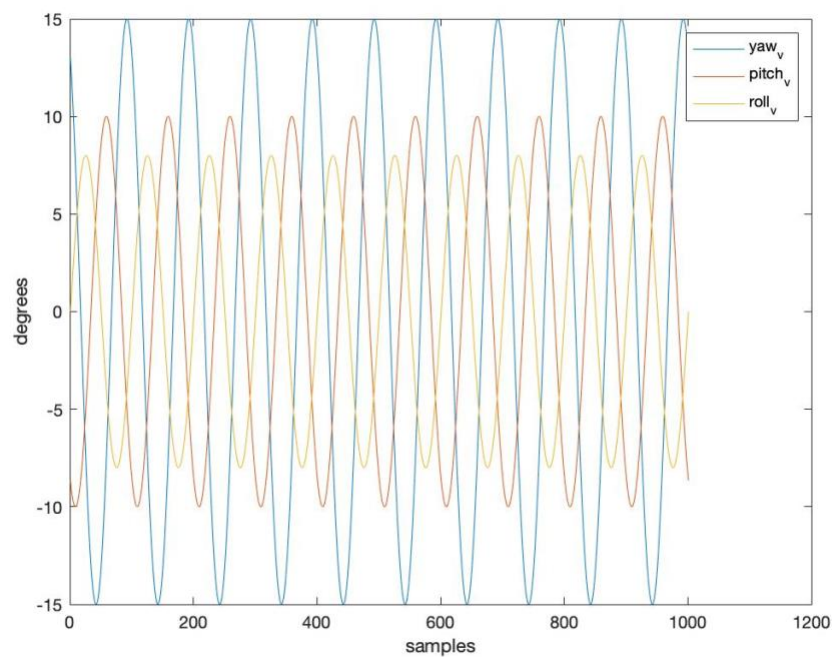
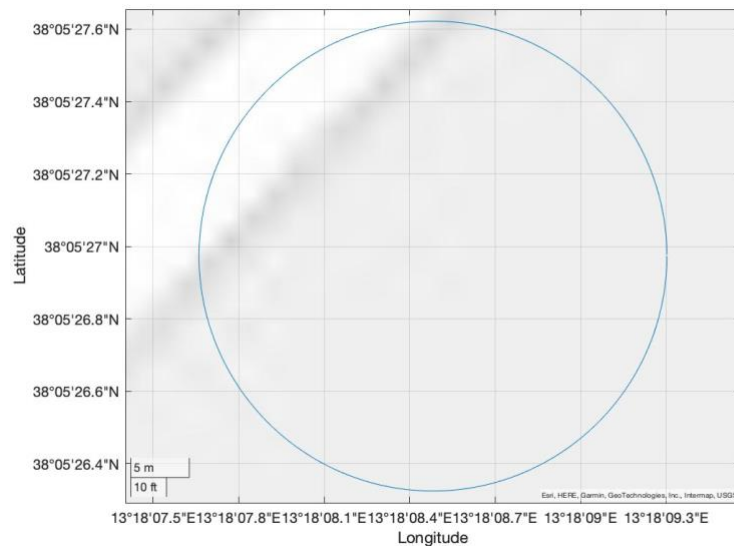


Capitolo 2

Simulazione dei dati e dei sensori

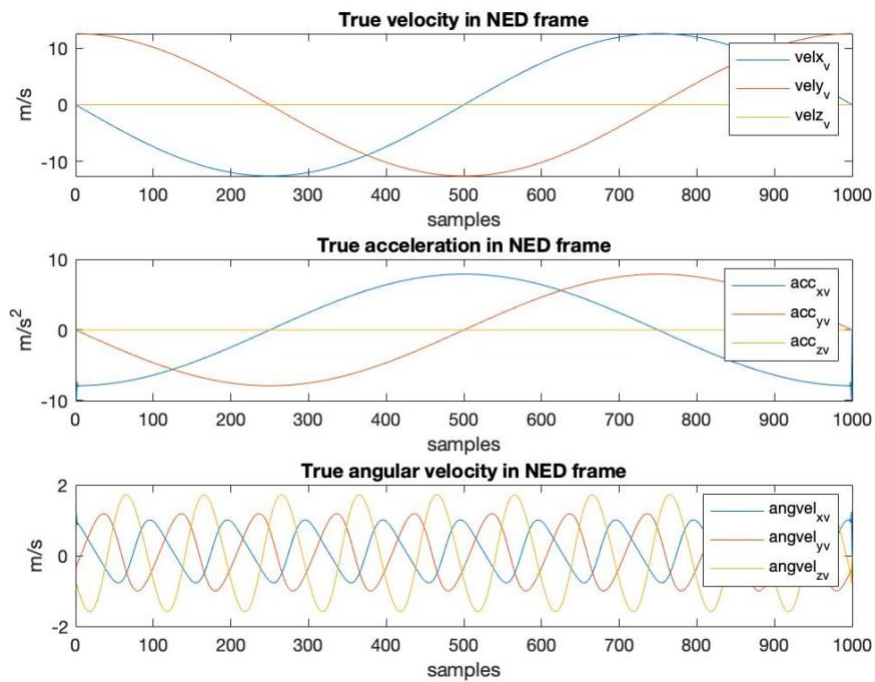
Simulazione dei dati

Inizialmente si è scelto di fissare una data traiettoria circolare di raggio ' r ', centrata in delle coordinate di riferimento e percorsa in ' t_{max} ' secondi, inoltre, durante questo percorso, si è supposto che il drone abbia un orientamento sinusoidale lungo *yaw*, *pitch* e *roll*.



A partire dalla posizione nota si sono ottenuti, per derivazione, i dati di *velocità* ed *accelerazione*, mentre conoscendo l'orientamento si è ricavata la *velocità angolare*.

Queste operazioni vengono eseguite in modo automatico attraverso la funzione *waypointTrajectory* di MATLAB, ad essa vengono forniti una serie di waypoints (definiti dalla posizione prescelta) con le relative informazioni di tempo di arrivo ed orientamento (prescelto).



Simulazione dei sensori IMU e GPS

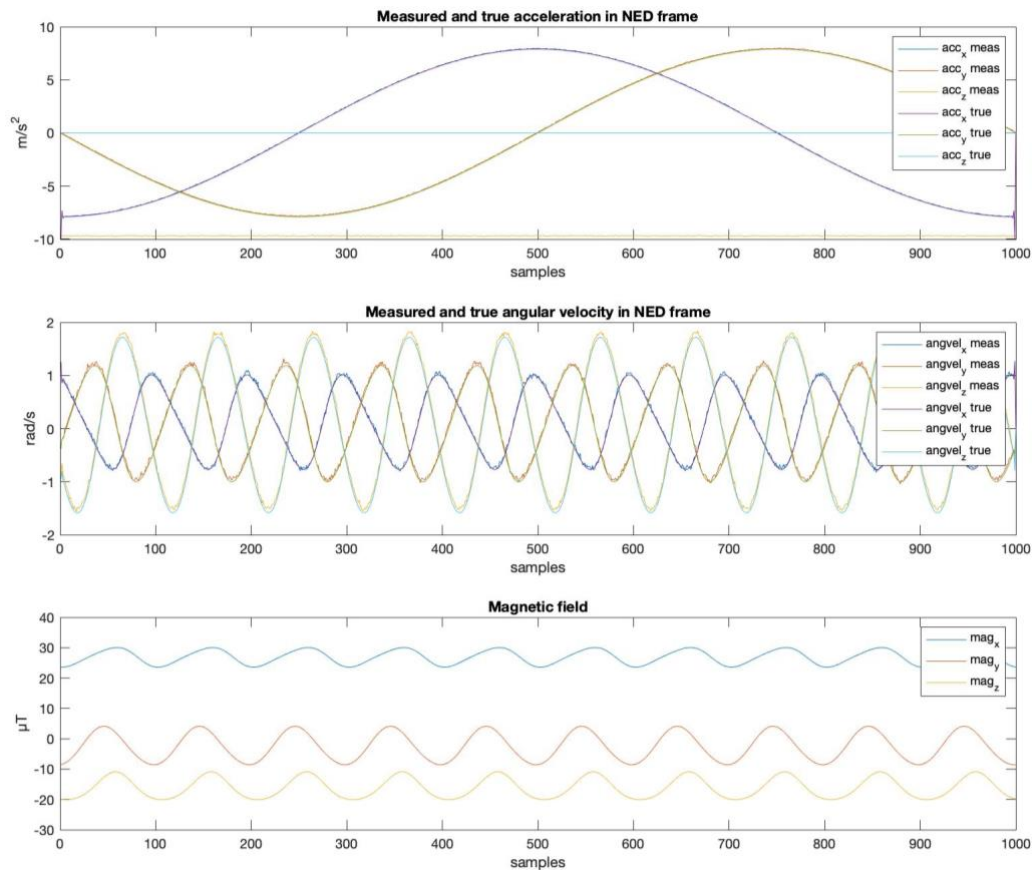
IMU

La simulazione dei sensori utilizzati, ovvero IMU e GPS, è stata effettuata sfruttando le funzioni *imuSensor* e *gpsSensor* di MATLAB. Quest'ultime permettono di simulare del bias e del rumore in modo immediato, da aggiungere al set di dati veri che gli viene fornito in ingresso.

```
%IMU settings
imu = imuSensor('accel-gyro-mag', 'SampleRate', imuFs);

imu.Gyroscope.ConstantBias = [0.02 0.04 0.08];
imu.Gyroscope.NoiseDensity = [0.004 0.004 0.004];

imu.Accelerometer.ConstantBias = [0.03 0.06 0.09];
imu.Accelerometer.NoiseDensity = [0.004 0.004 0.004];
```



Lo svantaggio principale nell'utilizzo della funzione *imuSensor* è stata l'impossibilità di impostare lo standard *wgs84* per la definizione del modello di gravità.

Dunque, è stato necessario definire quest'ultimo a priori ed in seguito è stato fornito alla funzione '*imuSensor*' producendo così una misura realistica di accelerazione soggetta a rumore e bias.

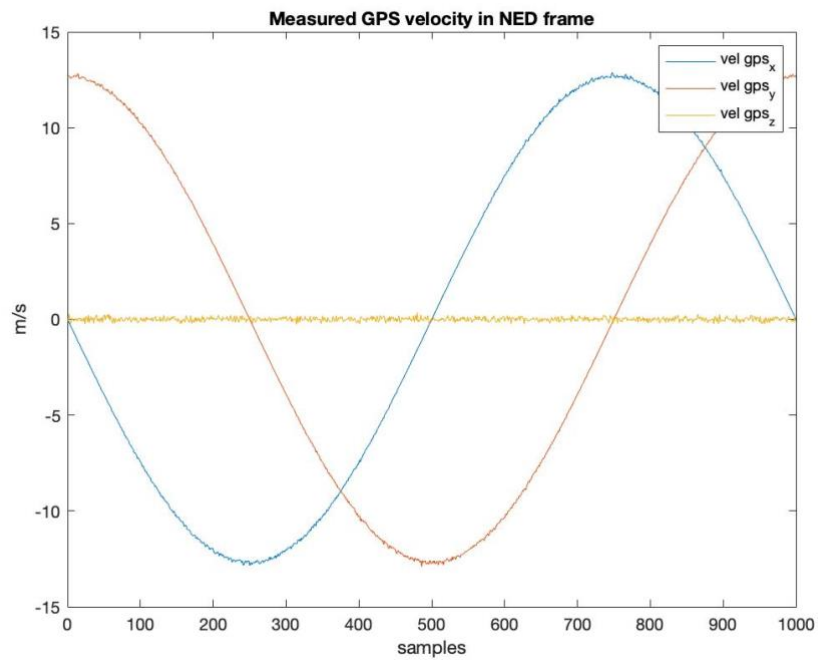
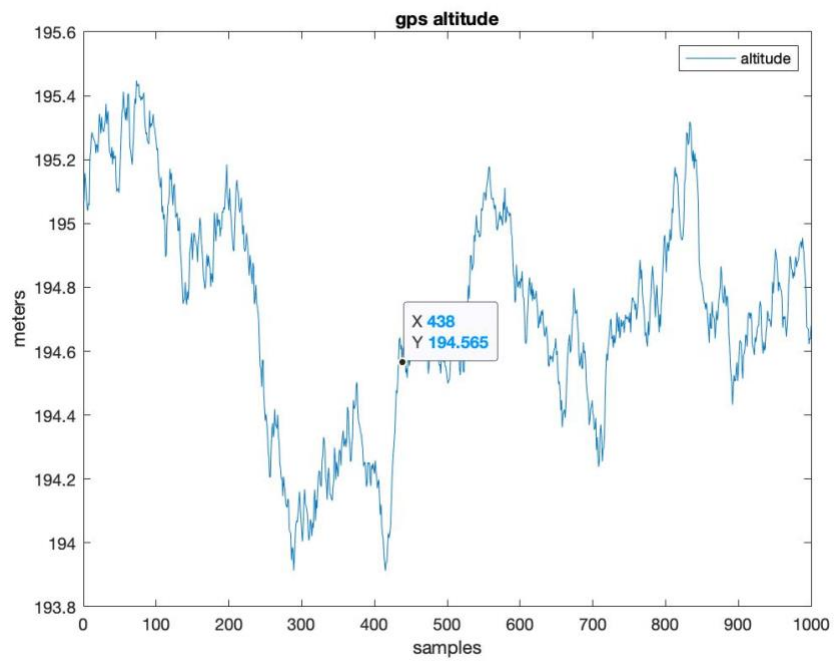
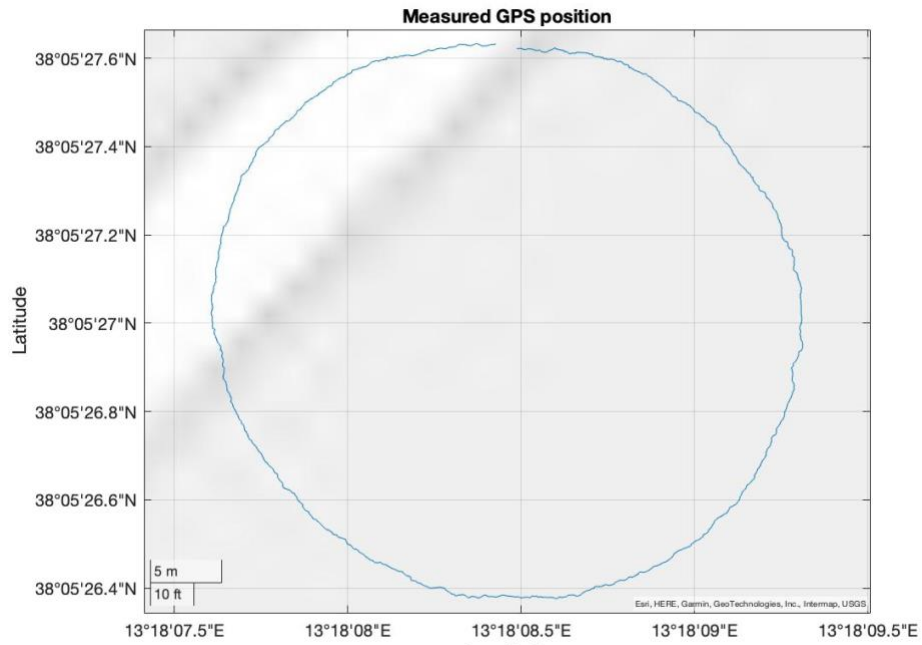
GPS

Per quanto riguarda la simulazione del GPS è stata utilizzata la funzione '*gpsSensor*', la quale, sulla base di una posizione di riferimento restituisce dei dati in termini di posizione e velocità soggetti anch'essi a rumore.

La posizione di riferimento scelta è: *Latitudine* 38.090826°
Longitudine 13.302356°
Altitudine 195.035°

%%GPS settings

```
gps = gpsSensor('SampleRate',gpsFs,'PositionInputFormat','Local','ReferenceFrame','NED');
gps.ReferenceLocation = localOrigin;
gps.HorizontalPositionAccuracy = 1.0;
gps.VerticalPositionAccuracy = 1.0;
gps.VelocityAccuracy = 0.1;
```

Capitolo 3

Stima dell'orientamento e del bias del giroscopio

Il sistema non lineare che descrive la dinamica dell'orientamento e del bias del giroscopio è:

$$\begin{cases} \dot{x}_a = f_a(x_a, \omega) + w_a \\ y_a = h_a(x_a) + v_a \end{cases}$$

in cui:

$$\bullet \quad x_a = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ b_{\omega_x} \\ b_{\omega_y} \\ b_{\omega_z} \end{bmatrix} \quad \text{vettore di stato costituito dal quaternione e dal bias del giroscopio}$$

$$\bullet \quad y_a = \begin{bmatrix} a_x \\ a_y \\ a_z \\ m_x \\ m_y \\ m_z \end{bmatrix} \quad \text{vettore delle misure di accelerometro e magnetometro in terna body}$$

- w_a rumore di processo che tiene conto dei fattori ambientali che interferiscono col sistema
- v_a rumore di misura che tiene conto delle imprecisioni dei sensori

Dalle **equazioni dell'orientamento** che descrivono come varia il quaternione:

$$\dot{q} = \frac{1}{2} \Xi(q) * \omega = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} * \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

si ricava la seguente dinamica non lineare, la quale, essendo una loosely coupled integration, tiene conto dell'errore del giroscopio dovuta al bias:

$$f_a(x_a, \omega) = \begin{bmatrix} \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} * \begin{bmatrix} \omega_x - b_{\omega_x} \\ \omega_y - b_{\omega_y} \\ \omega_z - b_{\omega_z} \end{bmatrix} \\ 0^{3 \times 1} \end{bmatrix}$$

mentre: $h_a(x_a) = [I^{4 \times 4} : 0^{4 \times 3}] * x_a$

La stima dell'orientamento avviene attraverso un primo EKF, esso inizialmente linearizza la dinamica intorno ad una stima che conosco a priori e poi esegue iterativamente una fase di *predizione* e una di *correzione*.

Stima a priori

Il vettore di stato iniziale è stato così scelto:

$$\hat{x}_{a_k} = [q_0 \quad q_1 \quad q_2 \quad q_3 \quad 0 \quad 0 \quad 0]^T$$

La matrice dei rumori di processo è una matrice 7x7 diagonale, i cui parametri sono stati tarati sulla base del comportamento della stima:

$$Q_{att} = \begin{bmatrix} 1^{-17} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1^{-17} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1^{-17} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1^{-17} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1^{-15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1^{-15} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1^{-15} \end{bmatrix}$$

Il vettore di misure utilizzato per la fase di correzione è ottenuto attraverso la tecnica Q-Triad, questa, sulla base della rotazione delle misure di accelerometro e magnetometro determina una misura di orientamento.

Dunque, valutando la varianza dell'errore tra un quaternioni ricavato dalle misure inerziali soggette a rumore e bias, e un quaternioni ricavato da misure inerziali non soggette a nessun incertezza si sono ottenute 4 varianze che poste lungo la diagonale di una matrice 4x4 hanno determinato la matrice dei rumori di misura.

$$R_{att} = \begin{bmatrix} \sigma_{q_0} & 0 & 0 & 0 \\ 0 & \sigma_{q_1} & 0 & 0 \\ 0 & 0 & \sigma_{q_2} & 0 \\ 0 & 0 & 0 & \sigma_{q_3} \end{bmatrix}$$

Mentre la matrice di covarianza è stata definita sulla base della scelta del vettore di stato iniziale:

$$P_k = P_{att} = \begin{bmatrix} 1^{-7} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1^{-7} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1^{-7} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1^{-7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1^{-6} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1^{-6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1^{-6} \end{bmatrix}$$

Predizione

Nella fase di predizione si predice il valore assunto dal vettore di stato all'istante **k+1** sulla base della conoscenza a priori del vettore stato all'istante attuale **k**:

$$\begin{aligned} \hat{x}_{a_{k+1}} &= \Phi_k \cdot \hat{x}_{a_k} \\ P_{k+1} &= \Phi_k \cdot P_k \cdot \Phi_k^T + P_k \end{aligned}$$

in cui: $\Phi_k \approx \left. \frac{df_a}{dx_a} \right|_k \cdot \Delta t$ è la matrice di transizione ottenuta dalla linearizzazione della dinamica.

Correzione

Durante la fase di correzione aggiornano le misure valutando il vettore di stato all'istante attuale **k** sulla base della conoscenza del vettore di stato all'istante precedente **k-1**.

Anzitutto determino il guadagno di Kalman:

$$K_k = P_{k-1} \cdot H_k^T \cdot (H_k \cdot P_{k-1} \cdot H_k^T + R)^{-1}$$

poi valuto l'innovazione: $e_k = y_k - H_k \cdot \hat{x}_{a_{k-1}}$

dove y_k è una misura 'sintetica' del quaternioni ottenuta attraverso la tecnica Q-Triad.

Quindi:

$$\begin{aligned} \hat{x}_{a_k} &= \hat{x}_{a_{k-1}} + K_k \cdot e_k \\ P_k &= (I - K_k \cdot H_k) \cdot P_{k-1} \end{aligned}$$

in cui: $H_k = \left. \frac{dh_a}{dx_a} \right|_k$ è la matrice delle misure.

Eseguendo in modo iterativo le fasi di predizione e correzione ottengo il bias stimato del giroscopio e l'orientamento stimato, il primo lo uso per correggere le misure del giroscopio in ingresso al filtro mentre il secondo viene fornito al secondo filtro di Kalman che stimerà la posizione.

Capitolo 4

Stima della posizione, della velocità e del bias dell'accelerometro

Il sistema non lineare che descrive la dinamica della posizione, della velocità (espressa nel frame di navigazione) e del bias dell'accelerometro è:

$$\begin{cases} \dot{x}_p = f_p(x_p, a^n) + w_p \\ y_p = h_p(x_p) + v_p \end{cases}$$

in cui:

$$\bullet \quad x_p = \begin{bmatrix} \Phi \\ \lambda \\ h \\ v_N \\ v_E \\ v_D \\ b_{a_x} \\ b_{a_y} \\ b_{a_z} \end{bmatrix} \text{ vettore di stato costituito dal vettore posizione}$$

(latitudine, longitudine e quota), dal vettore velocità in terna NED e dal bias dell'accelerometro.

$$\bullet \quad y_p = \begin{bmatrix} \Phi_{GPS} \\ \lambda_{GPS} \\ h_{GPS} \\ v_{N_{GPS}} \\ v_{E_{GPS}} \\ v_{D_{GPS}} \end{bmatrix} \text{ vettore delle misure di posizione e velocità fornite dal GPS}$$

• w_p rumore di processo

• v_p rumore di misura

Le equazioni della dinamica che descrivono come varia la posizione e la velocità sono:

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\lambda} \\ \dot{h} \\ \dot{v}_N \\ \dot{v}_E \\ \dot{v}_D \end{bmatrix} = \begin{bmatrix} D^{-1} \cdot V^n \\ C_b^n \cdot a_b - (2\omega_{ie} + \omega_{en}) \times V^n + g^n \end{bmatrix}$$

$$\text{in cui: } D^{-1} = \begin{bmatrix} \frac{1}{M+h} & 0 & 0 \\ 0 & \frac{1}{(N+h) \cdot \cos \Phi} & 0 \\ 0 & 0 & -1 \end{bmatrix},$$

mentre:

$$\dot{V}^n = C_b^n \cdot a_b - (2\omega_{ie}^n + \omega_{en}^n) \times V^n + g^n$$

è l'**equazione di navigazione** che descrive l'evoluzione della velocità nel frame di navigazione.

Questa equazione è costituita da:

- $a^n = C_b^n \cdot a_b$ è il vettore delle accelerazioni espresso in terna NED. Dato che le equazioni di navigazione sono definite nel frame di navigazione, è necessario introdurre una matrice di rotazione che permetta il cambio di coordinate dal frame body nel quale sono espresse le misure sensoriali. Questa matrice è indicata come C_b^n ed è ottenuta con le stime del quaternione ricavate dal precedente filtro:

$$C_b^n = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

- $-2\omega_{ie}^n \times V^n$ è la accelerazione di Coriolis, in cui:

$$\omega_{ie}^n = C_e^n \cdot \omega_{ie}^e = \begin{bmatrix} -s_\Phi c_\lambda & -s_\Phi s_\lambda & c_\Phi \\ -s_\lambda & c_\lambda & 0 \\ -c_\Phi c_\lambda & -c_\Phi s_\lambda & -s_\Phi \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ \omega_{ie} \end{bmatrix} = \text{velocità}$$

angolare della terna ECEF rispetto all'ECI espressa in terna NED.

$$\begin{aligned}
C_e^n &= C_y^T(-90^\circ) C_y^T(-\Phi) C_z(\lambda) = \\
&= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_\Phi & 0 & s_\Phi \\ 0 & 1 & 0 \\ -s_\Phi & 0 & c_\Phi \end{bmatrix} \cdot \begin{bmatrix} c_\lambda & s_\lambda & 0 \\ -s_\lambda & c_\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} = \\
&= \begin{bmatrix} -s_\Phi c_\lambda & -s_\Phi s_\lambda & c_\Phi \\ -s_\lambda & c_\lambda & 0 \\ -c_\Phi c_\lambda & -c_\Phi s_\lambda & -s_\Phi \end{bmatrix}
\end{aligned}$$

- $\omega_{en}^n \times V^n$ in cui:

$\omega_{en}^n = \text{transport rate}$ = velocità angolare della terna NED rispetto all'ECEF espressa in terna NED; ovvero quella velocità angolare, dovuta allo spostamento, imposta dalla condizione di mantenere la terna NED parallela alla superficie della Terra. Per valutare il transport rate, scelgo il frame di navigazione (NED), e definisco un frame intermedio n' che è ottenuto da una rotazione di λ e di Φ , a partire dal frame ECEF, seguita da un'altra rotazione di 90° intorno all'asse y :

$$\text{quindi: } \omega_{en'}^{n'} = C_y^T(-\Phi) \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\lambda} \end{bmatrix} + \begin{bmatrix} 0 \\ -\dot{\Phi} \\ 0 \end{bmatrix}$$

$$\text{allora: } \omega_{en}^n = C_y^T(-90^\circ) \cdot \omega_{en'}^{n'} = \begin{bmatrix} \dot{\lambda} \cos \Phi \\ -\dot{\Phi} \\ -\dot{\lambda} \sin \Phi \end{bmatrix} = \dots$$

dalle equazioni della dinamica, sapendo che: $\dot{\Phi} = \frac{v_E}{M+h}$, $\dot{\lambda} = \frac{v_N}{(N+h) \cos \Phi}$

$$\dots = \begin{bmatrix} \frac{v_E}{N+h} \\ \frac{-v_N}{M+h} \\ \frac{v_E}{(M+h)} \tan \Phi \end{bmatrix}$$

I parametri M e N corrispondono al raggio meridiano e al raggio normale che valgono: $N = \frac{r_E}{\sqrt{1-e^2}(\sin \Phi)^2}$, $M = N \cdot \frac{1-e^2}{(1-e^2(\sin \Phi)^2)^2}$

dove: $r_E = \text{raggio equatoriale} = 6378137 \text{ m}$

$r_P = \text{raggio polare} = 6356752.3142 \text{ m}$

$$e = \text{eccentricità} = \sqrt{1 - \frac{r_P^2}{r_E^2}}$$

- g^n è il modello di gravità che tiene conto della quota:

$$g^n = 9.8 \left(\frac{R}{R+h} \right)^2$$

dove $R = \sqrt{MN}$

Le precedenti notazioni seguono tutte lo standard WGS-84.

Dunque, dalla dinamica della posizione e della velocità ricavo il seguente modello non lineare:

$$f_p(x_p, a^n) = \begin{bmatrix} D^{-1} \cdot V^n \\ a^n - (2\omega_{ie} + \omega_{en}) \times V^n + g^n - b_a \\ 0^{3 \times 1} \end{bmatrix}$$

mentre: $h_p(x_p) = [I^{6 \times 6} : 0^{6 \times 3}] \cdot x_p$

Stima a priori

Per quanto riguarda la stima a priori del vettore di stato sono stati fatti gli stessi ragionamenti visti per il precedente filtro di Kalman:

$$\hat{x}_{p_k} = [\Phi \quad \lambda \quad h \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$$

La matrice dei rumori di processo scelta è:

$$Q_{pos} = \begin{bmatrix} 1^{-10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1^{-10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1^{-10} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1^{-10} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1^{-10} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1^{-10} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1^{-6} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1^{-6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1^{-6} \end{bmatrix}$$

La matrice dei rumori di misura è ottenuta dalla varianza dell'errore di posizione e velocità:

$$R_{pos} = \begin{bmatrix} \sigma_{GPS_{pos}} & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{GPS_{pos}} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{GPS_{pos}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{GPS_{vel}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{GPS_{vel}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{GPS_{vel}} \end{bmatrix}$$

Mentre la matrice di covarianza è:

$$P_k = P_{pos} == \begin{bmatrix} 1^{-9} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1^{-9} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1^{-9} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1^{-2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1^{-2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1^{-2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1^{-2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1^{-2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1^{-2} \end{bmatrix}$$

Predizione

$$\hat{x}_{p_{k+1}} = \Phi_k \cdot \hat{x}_{p_k}$$

$$P_{k+1} = \Phi_k \cdot P_k \cdot \Phi_k^T + P_k$$

in cui: $\Phi_k \approx \left. \frac{df_p}{dx_p} \right|_k \cdot \Delta t$

Correzione

Anzitutto determino il guadagno di Kalman:

$$K_k = P_{k-1} \cdot H_k^T \cdot (H_k \cdot P_{k-1} \cdot H_k^T + R)^{-1}$$

poi valuto l'innovazione: $e_k = y_k - H_k \cdot \hat{x}_{p_{k-1}}$

dove y_k è il vettore di misure del GPS: $y_k = \begin{bmatrix} \Phi_{GPS} \\ \lambda_{GPS} \\ h_{GPS} \\ v_{N_{GPS}} \\ v_{E_{GPS}} \\ v_{D_{GPS}} \end{bmatrix}$

Quindi:

$$\hat{x}_{p_k} = \hat{x}_{p_{k-1}} + K_k \cdot e_k$$

$$P_k = (I - K_k \cdot H_k) \cdot P_{k-1}$$

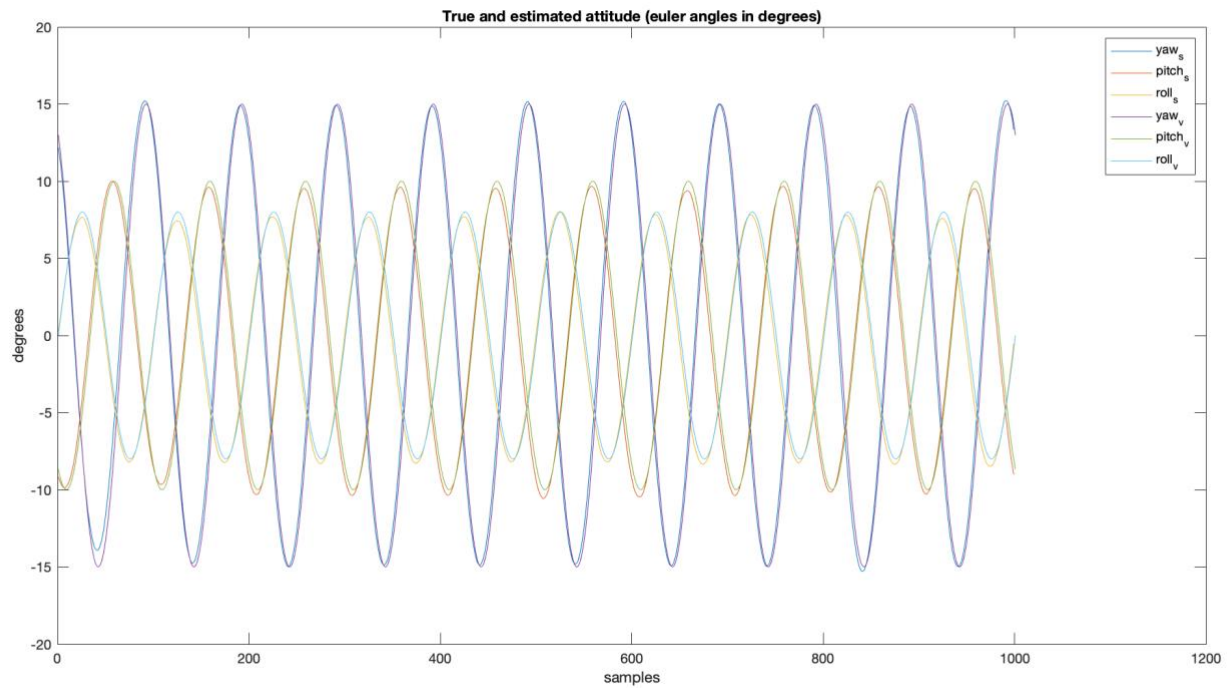
in cui: $H_k = \left. \frac{dh_p}{dx_p} \right|_k$ è la matrice delle misure.

Eseguendo in modo iterativo le fasi di predizione e correzione ottengo la posizione stimata e il bias stimato, quest'ultimo corregge le misure di accelerazione.

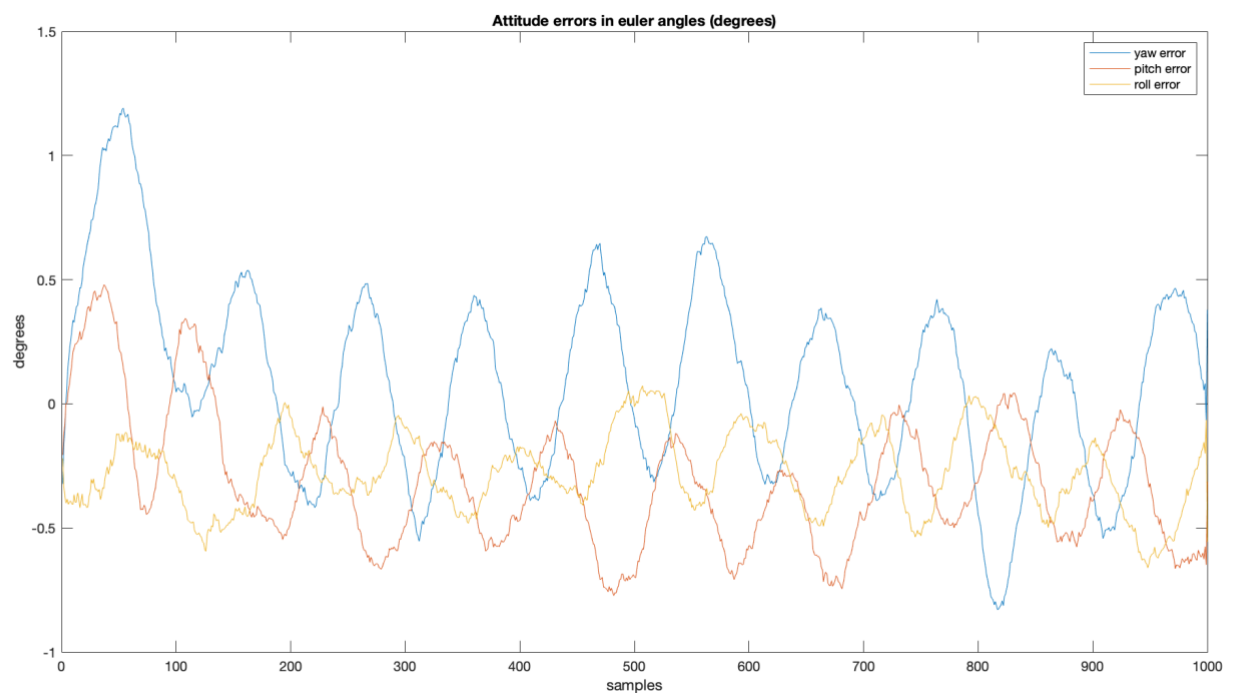
Capitolo 5

Risultati ottenuti

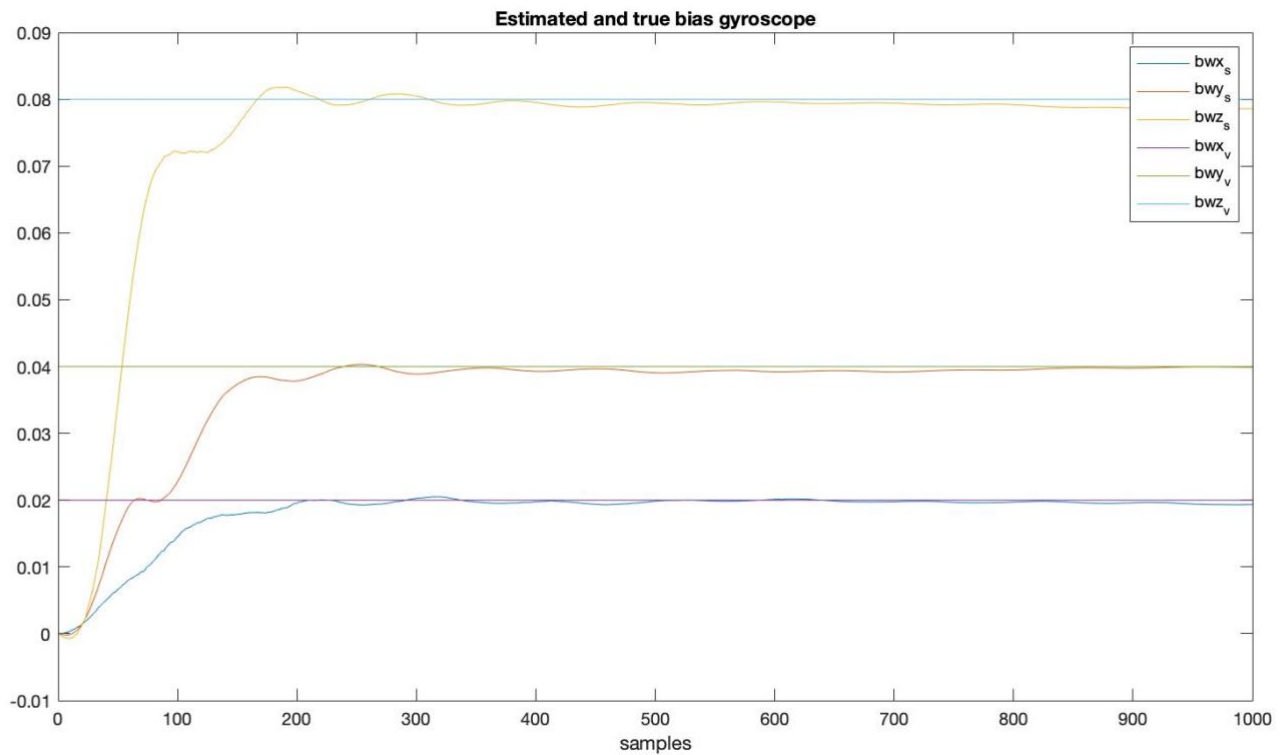
Confronto tra orientamento vero e stimato in angoli di Eulero



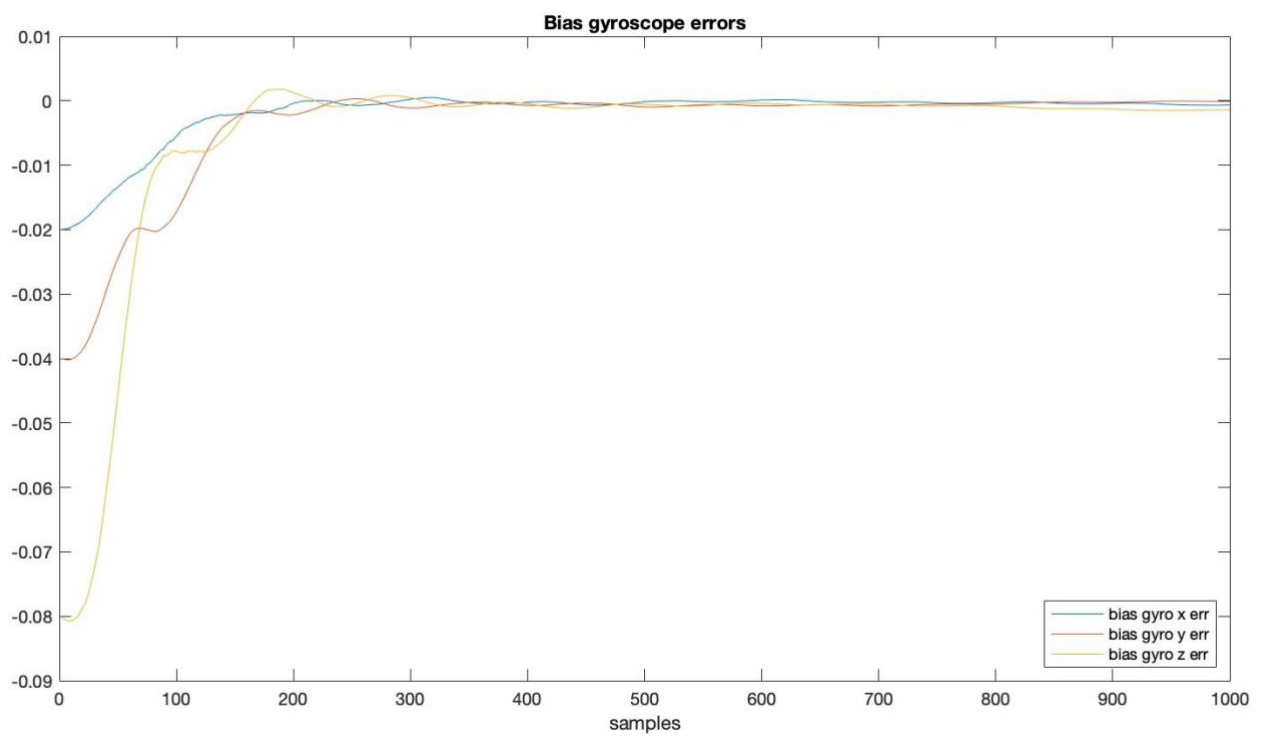
Errore di stima di orientamento



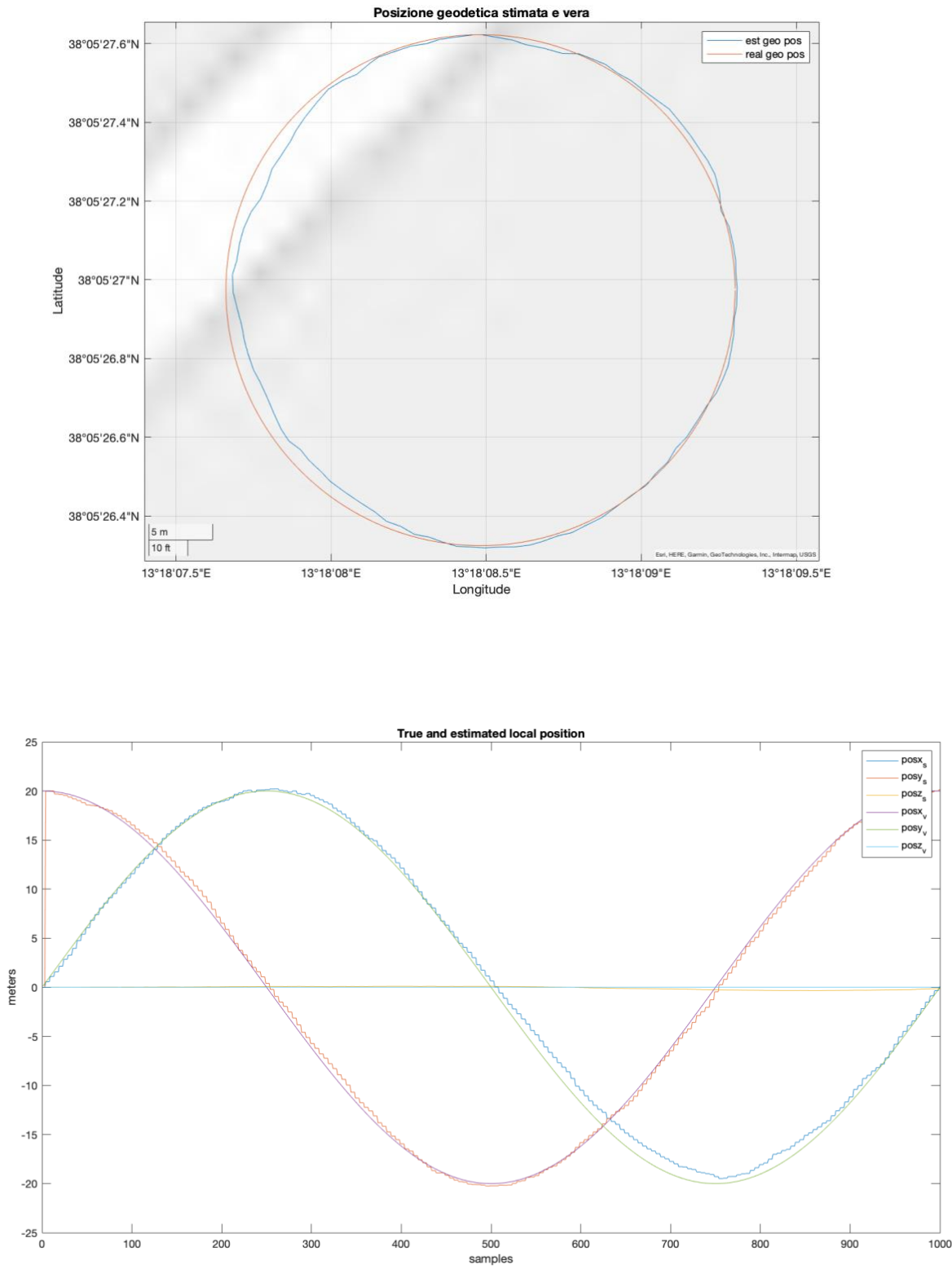
Confronto tra bias del giroscopio vero e stimato



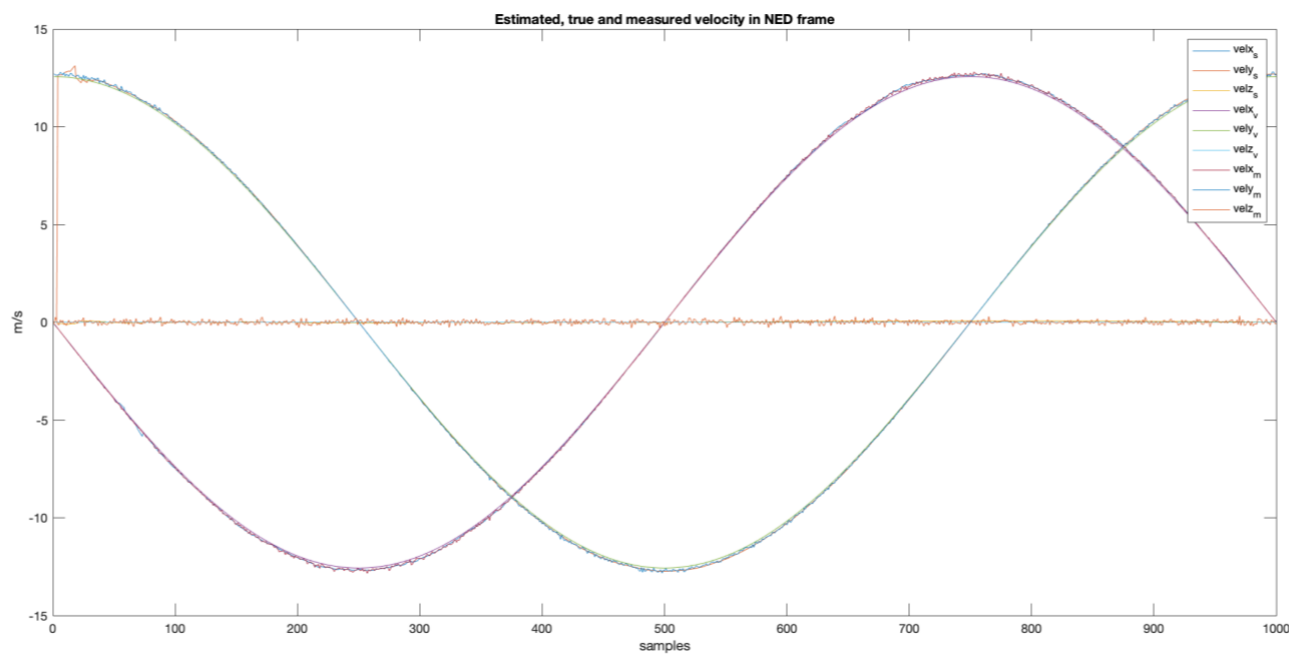
Errore di stima di bias del giroscopio



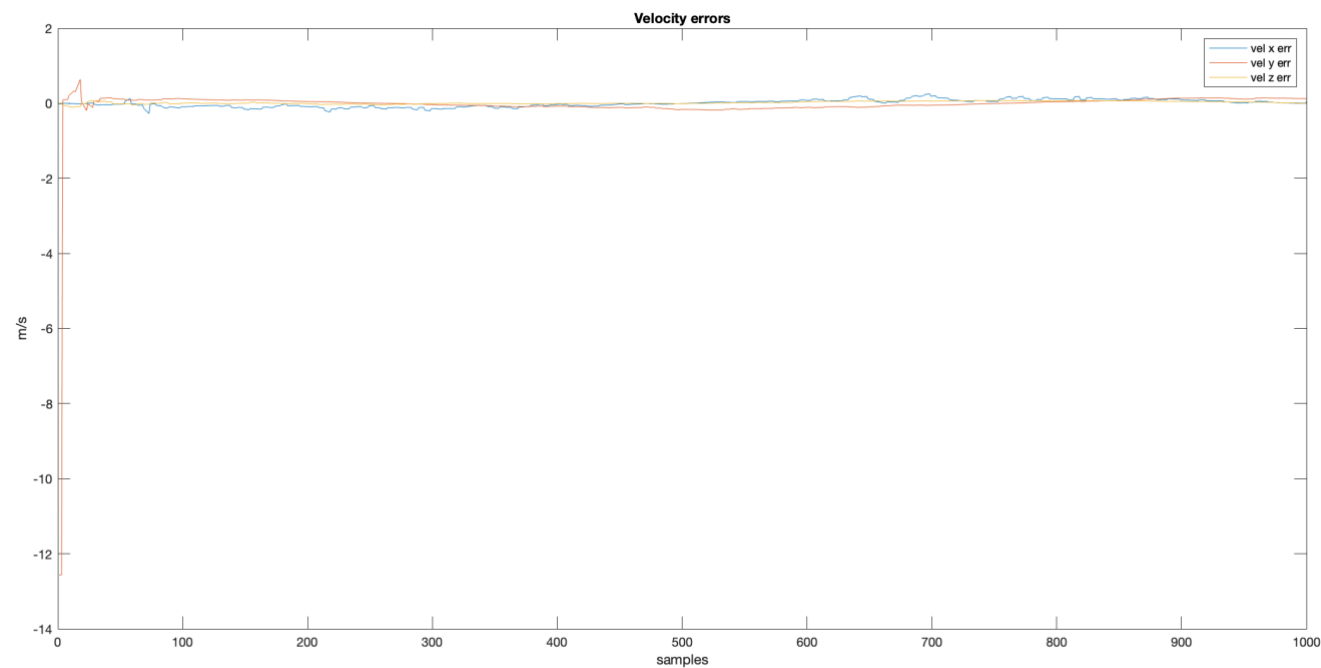
Confronto tra posizione vera e stimata

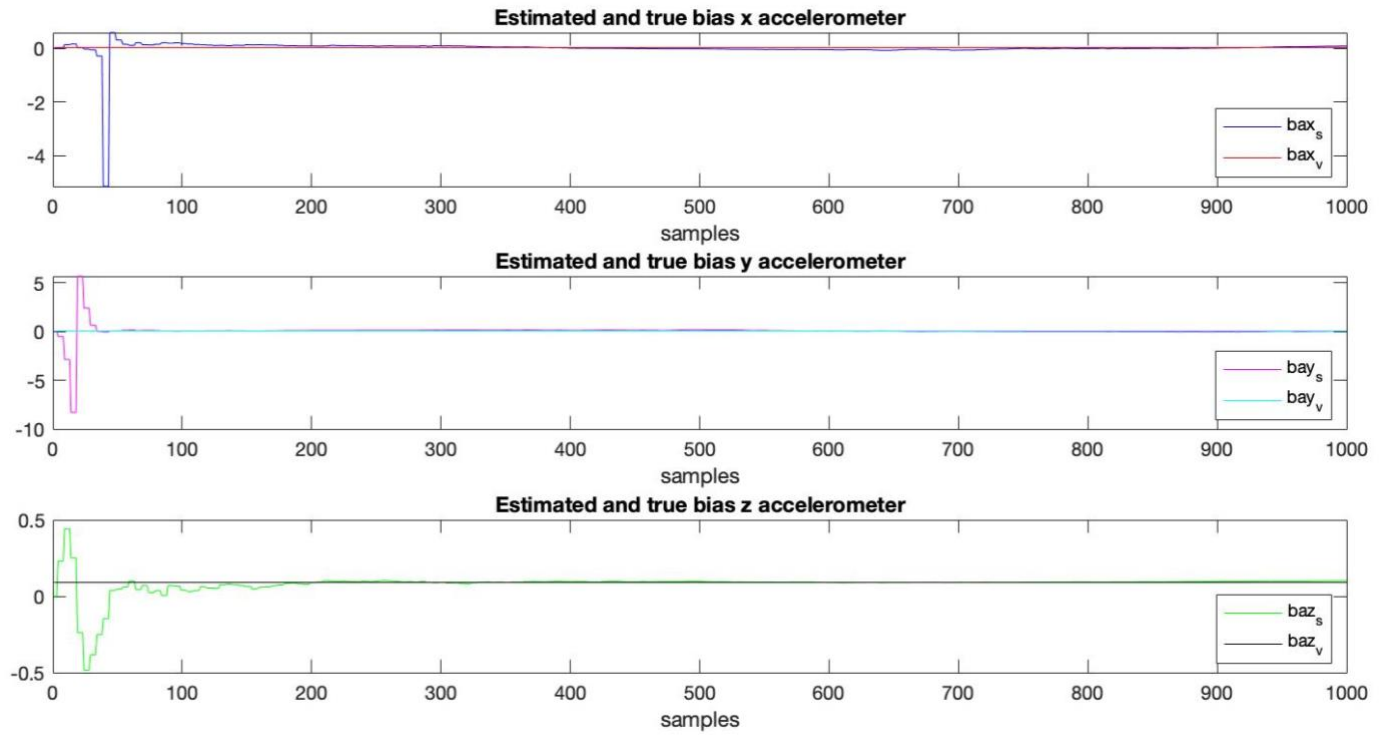


Confronto tra velocità vera, misurata e stimata



Errore di stima di velocità



Confronto tra bias dell'accelerometro stimato e vero

Bibliografia

- *Slide del corso 'Sistemi di guida e navigazione' – Lorenzo Pollini, Università di Pisa*
- Giulietti F., Turci M., *Coupled GPS/INS integration for Unmanned Aerial Vehicle*

Applications

- Titterton D.H., *Strapdown Inertial Navigation Technology. Second Edition*