

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA
Sede di Forlì

Corso di Laurea in
INGEGNERIA AEROSPAZIALE
Classe L-9 (codice 8263)

ELABORATO FINALE DI LAUREA
In Elaborazione Dati per la Navigazione

**Sviluppo di un ambiente software di test e analisi delle
prestazioni di INS di livello Navigation e Tactical Grade**

Relatore
Prof. Matteo Zanzi

Presentata da
Valerio Fogliano

Anno Accademico 2014/2015

Sessione III

Sommario

Indice delle figure	V
Ringraziamenti	VII
1. Introduzione	1
2. Sistemi di riferimento	3
2.1. <i>Classificazione sistemi di riferimento</i>	3
2.2. <i>Trasformazione di coordinate tra due sistemi di riferimento</i>	10
2.3. <i>Matrici di rotazione</i>	11
2.4. <i>Equazioni di propagazione dell'assetto e della matrice di rotazione</i>	16
3. Sistemi di navigazione inerziale INS	21
3.1. <i>Principio di funzionamento base</i>	21
3.2. <i>Equazioni di navigazione standard</i>	23
3.3. <i>Sistemi di tipo Strapdown per la navigazione inerziale</i>	27
4. Sensori inerziali	31
4.1. <i>Definizioni</i>	31
4.2. <i>Accelerometri</i>	32
<i>Principio di funzionamento</i>	32
<i>Classificazione e tipologie di accelerometri</i>	36
4.3. <i>Giroscopi</i>	40
<i>Principio di funzionamento</i>	40
<i>Classificazione e tipologie di giroscopi</i>	42
4.4. <i>Errori sensori</i>	46
<i>Fattore di scala (SF - Scale Factor)</i>	48
<i>Errore di allineamento (Misalignment)</i>	48
<i>Deriva (Bias)</i>	49
<i>Rumore bianco (White Noise)</i>	50
5. MATLAB	55
5.1. <i>Introduzione al potenziale di MATLAB</i>	55
5.2. <i>Simulink</i>	56
6. Simulatore	59
6.1. <i>Input</i>	60
<i>Definizione della traiettoria</i>	60
<i>Modellizzazione matematica del velivolo come punto materiale 3D</i>	63
<i>Implementazione angolo di banco</i>	64
6.2. INS⁻¹	66

<i>Calcolo output accelerometri</i>	67
<i>Calcolo matrici di rotazione</i>	68
6.3. IMU e rumori dei sensori	69
6.4. INS	71
<i>Accelerazioni ed assetto</i>	72
<i>Vettore posizione e coordinate geodetiche</i>	73
6.5. Output simulatore	74
7. Risultati dei test	75
7.1. <i>Grafici e valori significativi</i>	75
Conclusioni	87
Riferimenti	89

Indice delle figure

Figura 1: Sistemi di riferimento ECEF & ECI.....	4
Figura 2: Coordinate geodetiche di un punto sulla superficie terrestre	5
Figura 3: Sezione Equatoriale.....	6
Figura 4: Sezione verticale della Terra in corrispondenza di un generico asse equatoriale w	6
Figura 5: Coordinate NED - ECEF.....	8
Figura 6: Assi Body (1,2,3) e NED.....	9
Figura 7: Angoli di Eulero.....	11
Figura 8: Prima rotazione Eulero.....	12
Figura 9: Seconda rotazione Eulero.....	13
Figura 10: Terza rotazione Eulero.....	14
Figura 11: Componenti della velocità rotazionale velivolo	16
Figura 12: Schema generale di un INS	22
Figura 13: Piattaforma stabilizzata (a sinistra) e Strapdown (a destra)	28
Figura 14: Modello generale di un sistema Strapdown.....	30
Figura 15: Schema generale di un accelerometro con sistema massa-molla-smorzatore	32
Figura 16: Variazione uscita al variare del coefficiente di smorzamento con ingresso a gradino.	35
Figura 17: Accelerometro MEMS	36
Figura 18: Materiali piezoelettrici	37
Figura 19: Misurazione deformazione materiale mediante misura della tensione.....	37
Figura 20: Trasformatore ad E.....	38
Figura 21: Accelerometro a pendolo.....	38
Figura 22: Accelerometro laser.....	39
Figura 23: Semplificazione dinamica dell'accelerometro laser	39
Figura 24: Rotazione tra due sistemi di riferimento.....	40
Figura 25: Giroscopio MEMS a forchetta.....	42
Figura 26: Giroscopio meccanico	43
Figura 27: FOG - Fiber Optical Gyro	44
Figura 28: Giroscopio Laser.....	46
Figura 29: Errori dei sensori	47
Figura 30: Distribuzione Gaussiana.....	51
Figura 31: Ambiente di programmazione Simulink.....	56
Figura 32: Generica modellizzazione su Simulink di un accelerometro MEMS.....	57
Figura 33: Layout Simulatore.....	59
Figura 34: Holding Pattern	60
Figura 35: Calcolo variazione angolo di prua	62
Figura 36: Rappresentazione velocità in assi NED.....	63
Figura 37: Implementazione angolo di banco su Simulink	65
Figura 38: Andamento della φ	65
Figura 39: Andamento della φ	65
Figura 40: Blocco <i>INS</i> - 1.....	66
Figura 41: Implementazione errori sui giroscopi	69
Figura 42: Andamento tipico di un White Noise	70
Figura 43: Andamento Bias relativo a tre giroscopi.....	70
Figura 44: Blocco <i>INS</i>	71

Figura 45: Calcolo assetto INS	72
Figura 46: Accelerometro MEMS - Giroscopio MEMS – Piano orizzontale.....	76
Figura 47: Accelerometro MEMS - Giroscopio MEMS - quota.....	76
Figura 48: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell’errore di posizione sul canale orizzontale [m]	77
Figura 49: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell’errore di posizione sul canale verticale [m]	77
Figura 50: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell’errore sulle componenti di velocità traslazionale [m/s]	78
Figura 51: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell’errore sugli angoli d’assetto [deg]	78
Figura 52: Accelerometro MEMS - Giroscopio FOG - quota.....	79
Figura 53: Accelerometro MEMS - Giroscopio FOG – Piano orizzontale	79
Figura 54: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell’errore di posizione sul canale orizzontale [m]	80
Figura 55: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell’errore di posizione sul canale verticale [m]	80
Figura 56: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell’errore sulle componenti di velocità traslazionale [m/s].....	81
Figura 57: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell’errore sugli angoli d’assetto [deg]	81
Figura 58: Accelerometro MEMS - Giroscopio Laser - quota	82
Figura 59: Accelerometro MEMS - Giroscopio Laser – Piano orizzontale.....	82
Figura 60: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell’errore di posizione sul canale verticale [m]	83
Figura 61: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell’errore di posizione sul canale orizzontale [m]	83
Figura 62: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell’errore sulle componenti di velocità traslazionale [m/s].....	84
Figura 63: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell’errore sugli angoli d’assetto [deg]	84



Ringraziamenti

Prima di raccontare il mio lavoro, ci terrei ad elargire dei ringraziamenti diretti a chi lo ha reso possibile e chi mi ha aiutato, direttamente o indirettamente, nello svolgimento di quest'ultimo.

Per queste ragioni ringrazio l'Università di Bologna che, educandomi attraverso il percorso di studi, ha contribuito alla formazione di un mio metodo di studio ed ha migliorato le mie capacità logiche legate ai problemi di natura ingegneristica.

Ringrazio poi il professor Matteo Zanzi, il quale mi ha fatto appassionare alla materia, dandomi anche dell'ottimo materiale da cui attingere per i miei studi, e mi ha seguito pazientemente, passo dopo passo, mettendomi a disposizione la sua esperienza nel settore per far fronte alle varie problematiche che si sono presentate lungo questo cammino.

Infine, vorrei dedicare un grande ringraziamento anche alla mia famiglia, che ha sempre creduto in me e mi ha supportato fino al completamento di questa mia prima fase di studi, e a tutti coloro che durante questi anni mi sono stati vicini, aiutandomi nel mio cammino.



1. Introduzione

Sin dalla sua nascita, il metodo di studio scientifico di un fenomeno, si è sempre basato, nelle sue fasi iniziali, sull'osservazione dello stesso.

Quando gli occhi, in concerto con gli altri sensi, non riuscivano più a vedere ciò che la mente riusciva a pensare, si è dovuti ricorrere a nuove idee per riuscire ad osservare più lontano e più a fondo, e dunque, in questo quadro, nascono i primi *sensori*.

A seconda del tipo di osservazione e della loro composizione, essi vengono categorizzati ed utilizzati.

Nel nostro caso, l'attività svolta e raccontata in questa tesi è basata in particolare su due tipi di sensori, di cui tratteremo successivamente all'interno di questa disamina: *giroscopi* ed *accelerometri*.

Il mio progetto si è basato sulla simulazione, tramite la piattaforma MATLAB/Simulink, di alcuni dati presenti all'interno dei vari datasheets, ovvero i così detti di livello *navigation* e *tactical grade*, che rappresentano possibili errori (come poi vedremo ci occuperemo principalmente degli errori derivanti da processi stocastici) presenti all'interno dei sensori. Tutto ciò è risultato poi utile per poter mettere a confronto diversi sensori e misurarne la loro bontà durante una traiettoria fittizia che verrà descritta nel proseguo.

Inoltre, per introdurre le equazioni che governano il simulatore, nei primi capitoli saranno presenti delle brevi nozioni sulla meccanica/cinematica di un velivolo e sui sistemi di riferimento maggiormente utilizzati nell'ambito dell'elaborazione dati per la navigazione, una discreta modellizzazione dei sensori che verranno successivamente confrontati e una breve descrizione degli ambienti di programmazione utilizzati.

Concludendo questa introduzione, vorrei far notare al lettore che, per brevità, si assumerà che esso abbia già familiarità con il linguaggio di programmazione MATLAB, e che abbia delle basi matematiche che permettano la comprensione di modelli matematici in cui sono presenti derivate ed integrali di quantità sia scalari che vettoriali o matriciali.



2. Sistemi di riferimento

Per poter introdurre il lettore a comprendere al meglio il mio lavoro, faremo una breve sunto, all'interno di questo capitolo, riguardante, come detto precedentemente, i sistemi di riferimento che saranno utilizzati all'interno del simulatore e la loro modellizzazione matematica.

2.1. Classificazione sistemi di riferimento

- **Sistema ECI (Earth Centered Inertial)**

Sistema cartesiano ortogonale OXYZ con:

- origine O posta nel centro della Terra;
- asse Z coincidente con l'asse di rotazione media della Terra;
- assi X e Y giacenti sul piano equatoriale terrestre, con asse X diretto verso una stella (punto γ della costellazione dell'Ariete o punto Vernale);
- asse Y perpendicolare agli altri due in modo da formare una terna destrorsa.

- **Sistema ECEF (Earth Centered Earth Fixed)**

Sistema cartesiano ortogonale OXYZ, fisso rispetto alla Terra, con:

- origine O posta nel centro della Terra;
- asse Z coincidente con l'asse di rotazione media della Terra;
- assi X e Y giacenti sul piano equatoriale terrestre, con asse X passante per l'intersezione tra equatore e meridiano di Greenwich;
- asse Y perpendicolare agli altri due in modo da formare una terna destrorsa.

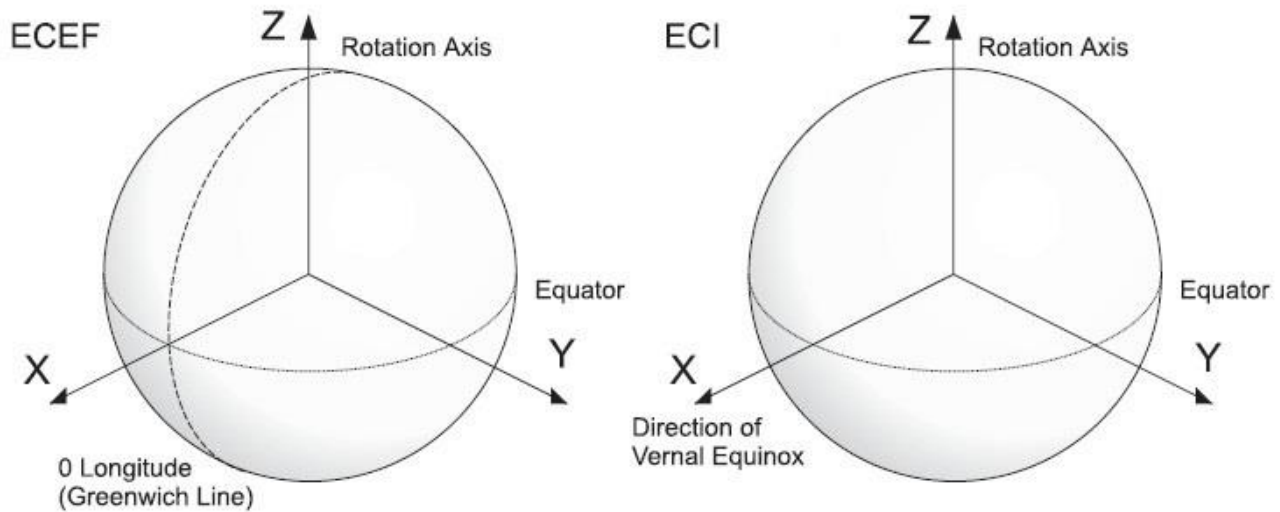


Figura 1: Sistemi di riferimento ECEF & ECI

Come si evince dalla descrizione anzidetta dei due sistemi, e dalla loro rappresentazione nella Figura 1, essi presentano la stessa posizione del polo Nord, ovvero il punto di intersezione del meridiano di Greenwich (o qualsiasi altro meridiano) e l'asse Z, ed inoltre sono entrambi geocentrici.

Le coordinate di un punto P in un sistema ECEF possono essere individuate o dalle coordinate cartesiane (X, Y, Z) o dalle coordinate geocentriche o dalle coordinate geodetiche¹.

➤ *Coordinate geocentriche* (terra modellata come una sfera):

- longitudine: angolo compreso tra il piano contenente il meridiano di Greenwich ed il piano contenente il meridiano passante per il punto considerato;
- latitudine: angolo compresa tra il piano equatoriale e la retta congiungente il punto P considerato con il centro della Terra;
- quota: distanza tra il punto P e l'intersezione con la superficie della Terra della retta congiungente il punto P e il centro della Terra.

¹ Questi ultimi due valutano entrambi l'entità della longitudine (λ), latitudine (Φ) e quota (h), ma varia la forma di riferimento assunta per la terra

➤ *Coordinate geodetiche* (terra modellata come un ellissoide):

- longitudine: angolo compreso tra il piano contenente il meridiano di Greenwich ed il piano contenente il meridiano passante per il punto considerato;
- latitudine: angolo compresa tra il piano equatoriale e la verticale locale, ovvero la retta passante per il punto P considerato e la proiezione P' di tale punto sulla superficie terrestre di riferimento tale che il segmento PP' è perpendicolare al piano tangente alla superficie terrestre passante per P';
- quota: distanza tra il punto P e la sua proiezione P' sulla superficie della Terra.

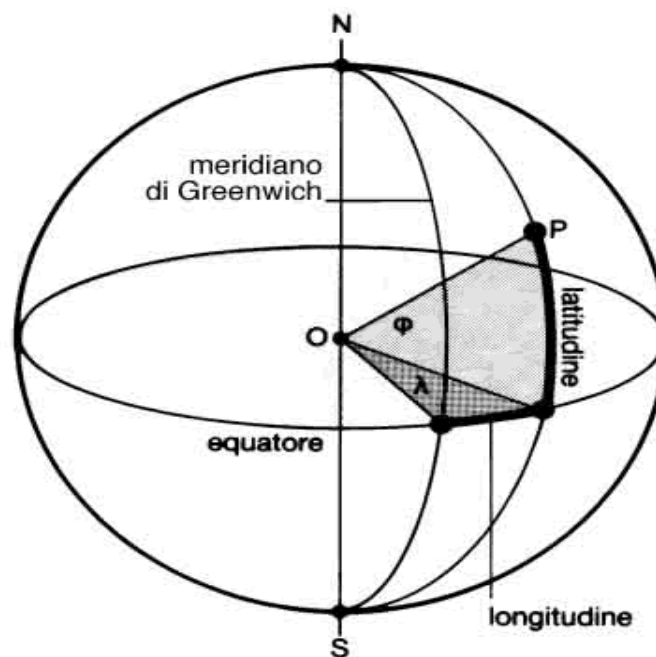


Figura 2: *Coordinate geodetiche di un punto sulla superficie terrestre*

Nota: descrizione del WGS84 (World Geodetic System 1984)

WGS84 (sigla di World Geodetic System 1984) è un sistema di coordinate geografiche geodetiche, mondiale, basato su un ellissoide di riferimento elaborato nel 1984. Esso costituisce un modello matematico della Terra da un punto di vista geometrico (forma e dimensione), geodetico e gravitazionale, costruito sulla base delle misure e delle conoscenze scientifiche e tecnologiche disponibili al 1984.

Avente la seguente terna:

- o origine O nel centro di massa della Terra;
- o asse Z ha direzione del polo di riferimento IERS²; questa direzione corrisponde alla direzione del polo convenzionale terrestre BIH, con un'incertezza di 0.005'';
- o asse X passante per il meridiano di riferimento IERS, il quale è coincidente con il meridiano zero (meridiano di Greenwich) identificato dal BIH, con un'incertezza di 0.005'';
- o asse Y scelto in modo da forma una terna destrorsa con i precedenti e poggiato sul piano equatoriale [1].

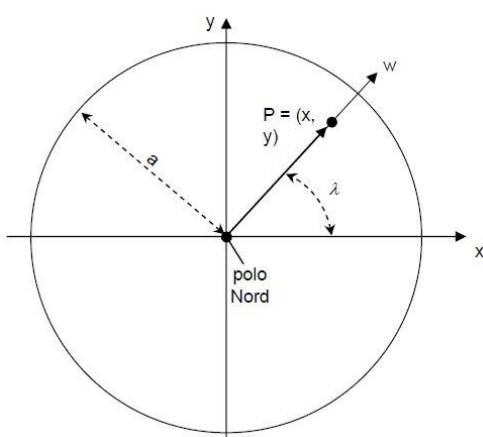


Figura 3: Sezione Equatoriale

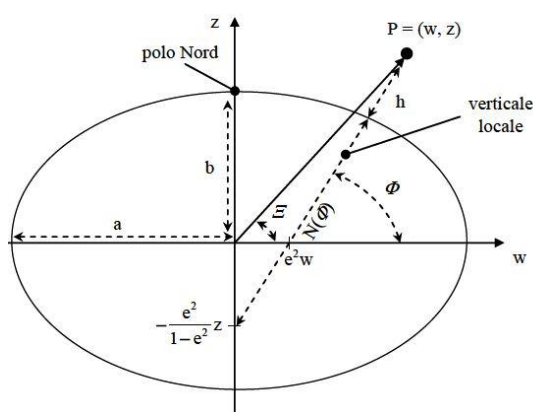


Figura 4: Sezione verticale della Terra in corrispondenza di un generico asse equatoriale w.

Nel caso in figura 4, la coordinata orizzontale w vale $\bar{w} = \sqrt{\bar{x} + \bar{y}}$.

Andiamo ora a vedere come possiamo trasformare le coordinate geodetiche³ WGS-84 in coordinate cartesiane ECEF $(\lambda, \Phi, h) \rightarrow (x, y, z)$.

Allora, considerando le seguenti:

λ	Longitudine geodetica
Φ	Latitudine geodetica
Ξ	Latitudine geocentrica
h	Quota
a	Semiasse maggiore terrestre = 6378137 [m]
b	Semiasse minore terrestre = 6356752.3142 [m]

² International Earth Rotation Service.

³ Come si evince anche dalla figura 4, bisogna distinguere la latitudine geocentrica Ξ dalla latitudine geodetica Φ .

$$e \quad \text{Eccentricità terrestre} = \sqrt{1 - \left(\frac{b}{a}\right)^2} \approx 0.08$$

Geometricamente, si ha:

$$N(\Phi) = \frac{a}{\sqrt{1 - e^2 \sin^2 \Phi}} \quad (2.1)$$

$$\rho(\Phi) = \frac{a(1 - e^2)}{(1 - e^2 \sin^2 \Phi)^{3/2}} \quad (2.2)$$

$$\cos \Xi = \frac{N(\Phi) + h}{P} \cos \Phi \quad (2.3)$$

Dove $N(\Phi)$ distanza tra la superficie dell'ellissoide e l'asse di rotazione z calcolata lungo la verticale locale mentre, $\rho(\Phi)$ è il raggio di curvatura locale dell'ellisse (nel piano verticale).

Infine, la trasformazione di coordinate sarà la seguente:

$$\mathbf{P}^e = \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{ECEF} = \begin{pmatrix} (N(\phi) + h) \cos \lambda \cos \phi \\ (N(\phi) + h) \sin \lambda \cos \phi \\ (N(\phi)(1 - e^2) + h) \sin \phi \end{pmatrix} \quad (2.4)$$

Fine nota

- **Sistema NED (North East Down)**

Sistema mobile definito in riferimento al piano orizzontale locale, cioè il piano tangente alla superficie terrestre in qualsiasi punto (vedi Figura 3).

In particolare, considerando un punto P in una certa posizione, il sistema NED corrispondente a quella posizione avrà:

- origine posta in P;
- asse Down (D), rivolto verso il basso, è diretto lungo la verticale locale, cioè la retta passante per il punto considerato P e la proiezione P' di tale punto sulla superficie terrestre assunta come riferimento e tale da essere perpendicolare al piano tangente alla superficie terrestre passante per P';
- assi Nord (N) e Est (E) giacenti sul piano orizzontale⁴, ovvero quello passante per P e perpendicolare alla verticale locale;
- l'asse N coincidente con l'intersezione tra il piano orizzontale ed il piano contenente l'asse di rotazione terrestre e P⁵. Il verso positivo dell'asse Nord è quello diretto verso il polo Nord geografico terrestre;
- asse E perpendicolare all'asse Nord ed all'asse Down in modo che la terna (N, E, D) sia una terna destrorsa. Il verso positivo di E è quello diretto verso Est;
- l'azimuth⁶ è l'angolo compresa tra l'asse N e la proiezione dell'asse longitudinale del velivolo sul piano N-E o piano orizzontale.

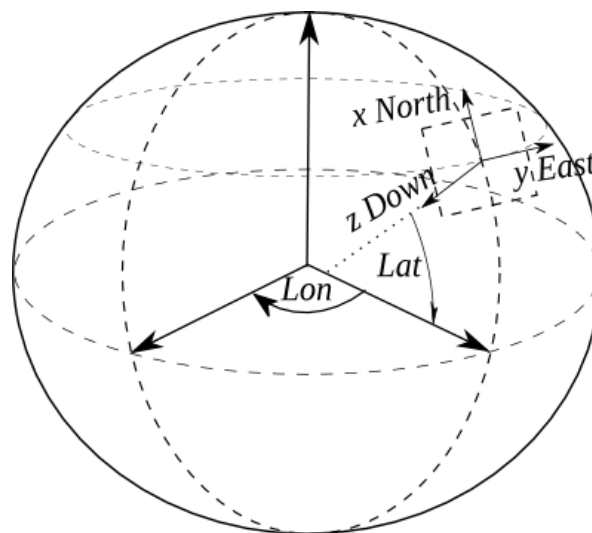


Figura 5: Coordinate NED - ECEF

⁴ Il piano orizzontale è parallelo al piano tangente alla superficie terrestre e passante per la proiezione P'

⁵ Questo piano contiene anche il meridiano passante per P'.

⁶ Detto anche prua o heading.

- **Sistema Assi Corpo (Body)**

Sistema di riferimento solidale al velivolo, con origine sul velivolo (vedi Figura 4), solitamente nel baricentro, e tale che:

- asse 1 (x) lungo l'asse longitudinale del velivolo;
- asse 3 (z) giace sul piano di simmetria verticale del velivolo ed è perpendicolare all'asse 1 e punta verso il basso.
- asse 2 (y) è quello trasversale al velivolo⁷ e perpendicolare agli assi 1 e 3, in modo da ottenere una terna destrorsa.

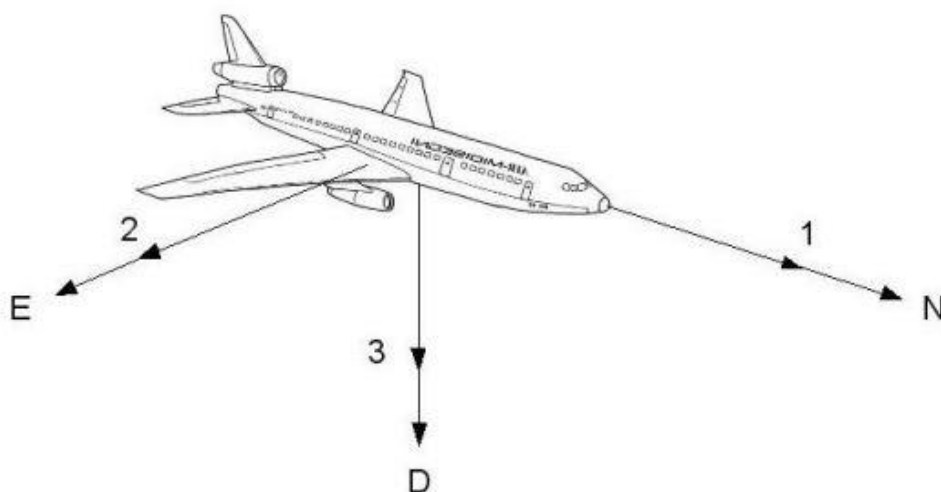


Figura 6: Assi Body (1,2,3) e NED

⁷ L'asse 2 è diretto verso l'ala destra, se si guarda il velivolo dall'alto.

2.2. *Trasformazione di coordinate tra due sistemi di riferimento*

Sia $\mathbf{v} \in \mathbb{R}^3$ tale che \mathbf{v}^a sia la terna che rappresenta le componenti del vettore \mathbf{v} rispetto ad un sistema di riferimento a, supposto fisso, e \mathbf{v}^b una terna che rappresenta le componenti del vettore \mathbf{v} rispetto ad un sistema di riferimento b, supposto in movimento rispetto ad a e con l'origine di b coincidente con quella di a.

Sia poi \mathbf{C}_a^b la matrice di trasformazione dal sistema di riferimento a verso b.

Allora, vale che:

$$\mathbf{v}^b = \mathbf{C}_a^b \mathbf{v}^a \quad (2.5)$$

Inoltre, per la proprietà di *ortonormalità*, si può dimostrare che l'inversa della matrice di rotazione coincide con la sua trasposta, e dunque avremo:

$$\mathbf{v}^a = (\mathbf{C}_a^b)^{-1} \mathbf{v}^b = (\mathbf{C}_a^b)^T \mathbf{v}^b = \mathbf{C}_b^a \mathbf{v}^b \quad (2.6)$$

Assumeremo durante tutta la trattazione questo tipo di notazione ed, in particolare, avremo che l'apice indicherà il sistema di riferimento rispetto al quale sono espresse le coordinate di un vettore, mentre il pedice indicherà il sistema di riferimento rispetto al quale è definito lo stesso vettore.

Per concludere questa prima trattazione, nella seconda parte del capitolo andremo a studiare come si ricava una matrice di rotazione, le principali matrici di rotazione che sono state utilizzate all'interno del simulatore ed infine studieremo le equazioni di propagazione dell'assetto, nonché della stessa matrice di rotazione, in quanto sarà uno dei punti focali in cui sarà concentrata l'attenzione, in fase preliminare, per lo sviluppo del simulatore stesso.

2.3. *Matrici di rotazione*

Le equazioni cinematiche specificano l'evoluzione nel tempo di un insieme di parametri che definiscono l'assetto del velivolo a un fissato riferimento inerziale.

In generale, per assetto, si intende l'orientamento di un sistema di riferimento qualsiasi rispetto ad un altro, preso come riferimento. In pratica, spesso si considera l'assetto di un corpo come l'orientamento del sistema Body rispetto al NED.

Esistono diversi parametri che ci permettono di esprimere l'assetto di un generico corpo rigido. All'interno della nostra trattazione mostreremo uno di questi metodi matematici che va sotto il nome di "teorema di Eulero" o anche "angoli di Eulero".

Il teorema di Eulero afferma che il più generico spostamento di un corpo rigido, che lascia fisso un punto, è una rotazione attorno ad un certo asse, l'asse di Eulero, passante per il punto fisso [2]. In particolare, nella nostra trattazione, considereremo problemi in \mathbb{R}^3 e dunque avremo tre rotazioni.

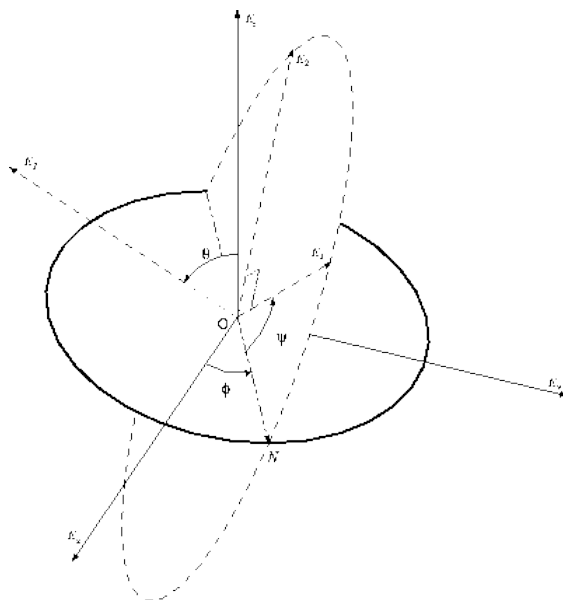


Figura 7: Angoli di Eulero

Per facilità di trattazione considereremo una generica rotazione per passare da un sistema di riferimento a, che assumeremo inerziale (possiamo ad esempio assumerlo in prima approssimazione come il sistema di riferimento NED), ad un sistema di riferimento b (esso può essere ricondotto al sistema di riferimento body), che assumeremo ruotato rispetto ad a ma con stessa origine.

Inoltre assumeremo che entrambi i nostri sistemi di riferimento siano costituiti da una terna cartesiana destrorsa di assi ortogonali tra loro.

Denoteremo con x , y e z gli assi di a e con 1 , 2 e 3 gli assi di b .

Supponiamo inizialmente che b coincida con a ($x=1, y=2, z=3$) e che subisca, dal suo orientamento iniziale, una rotazione attorno all'asse 3 in senso antiorario⁸ di un angolo ψ , detto angolo di prua. Dopo la rotazione il nuovo asse $3'$ continuerà a coincidere con l'asse 3 , mentre gli altri assi avranno la posizione indicata in Figura 8.

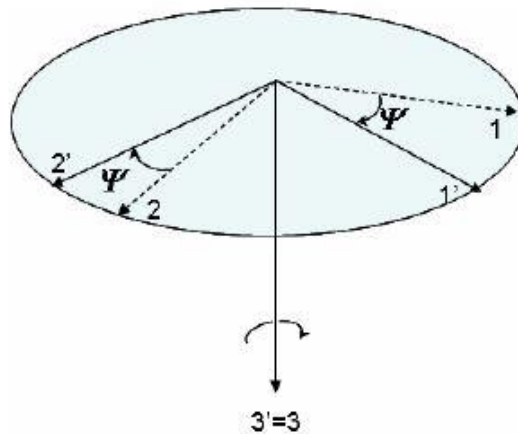


Figura 8: Prima rotazione Eulero

Dunque, la matrice di rotazione, che ci permette di passare dagli assi 1 , 2 e 3 agli assi $1'$, $2'$ e $3'$, sarà:

$$\mathbf{C}_{1,2,3}^{1',2',3'} = \begin{pmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.7)$$

La seconda rotazione che andremo a compiere sarà attorno all'asse $2'$ di un angolo che diremo θ , detto angolo di elevazione, così da avere che il nuovo asse $2''$ coinciderà con l'asse $2'$; mentre $1''$ e $3''$ saranno ruotati di un angolo θ rispetto a $1'$ e $3'$.

⁸ Per convenzione, si assume che la rotazione antioraria sia positiva, mentre quella oraria sia negativa.

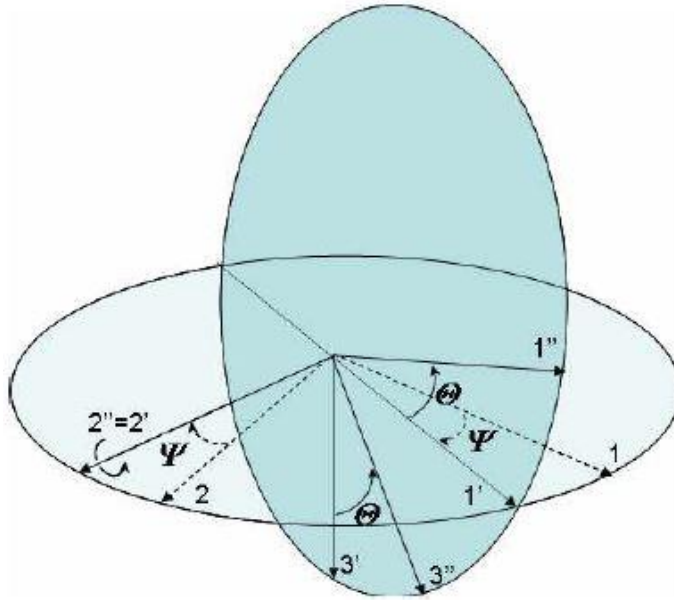


Figura 9: Seconda rotazione Eulero

Dunque, la matrice di rotazione che ci permette di passare dagli assi $1', 2'$ e $3'$ agli assi $1'', 2''$ e $3''$ sarà:

$$\mathbf{C}_{1',2',3'}^{1'',2'',3''} = \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix} \quad (2.8)$$

Infine, l'ultima rotazione che andremo a compiere sarà attorno all'asse $1''$ di un angolo che diremo φ , detto angolo di inclinazione o di banco⁹⁹ o di rollio, così da avere che il nuovo asse x coinciderà con l'asse $1''$; mentre y e z saranno ruotati di un angolo φ rispetto a $2''$ e $3''$.

⁹⁹ Bank Angle in inglese.

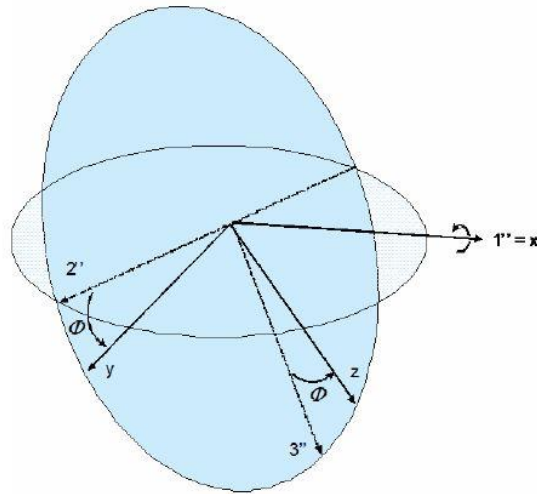


Figura 10: Terza rotazione Eulero

Dunque, la matrice di rotazione che ci permette di passare dagli assi 1'', 2'' e 3'' agli assi x, y e z sarà:

$$\mathbf{C}_{1'',2'',3''}^{x,y,z} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & \sin \varphi \\ 0 & -\sin \varphi & \cos \varphi \end{pmatrix} \quad (2.9)$$

Concludendo, la matrice di rotazione che ci permetterà di trasformare le coordinate di un vettore espresse nel sistema di riferimento avente coordinate 1, 2 e 3 in coordinate x, y e z di un altro sistema di riferimento, il cui orientamento è specificato dagli angoli di Eulero (considerati nella sequenza ψ, θ, φ), sarà data dai contributi di tutte e tre le rotazioni anzidette, rispettando l'ordine in cui sono state effettuate le rotazioni:

$$\mathbf{C}_{1,2,3}^{x,y,z} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & \sin \varphi \\ 0 & -\sin \varphi & \cos \varphi \end{pmatrix} \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix} \begin{pmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.10)$$

E dunque, considerando la rotazione dal sistema di riferimento NED al sistema body, ed eseguendo i calcoli, avremo:

$$\mathbf{C}_{NED}^{body}(\psi, \theta, \varphi) = \begin{pmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \theta \sin \varphi \cos \psi - \cos \varphi \sin \psi & \sin \theta \sin \varphi \sin \psi + \cos \varphi \cos \psi & \sin \varphi \cos \theta \\ \sin \theta \cos \varphi \cos \psi + \sin \varphi \sin \psi & \sin \theta \cos \varphi \sin \psi - \sin \varphi \cos \psi & \cos \varphi \cos \theta \end{pmatrix} \quad (2.11)$$

Applicando poi la proprietà di ortonormalità della matrice di rotazione, come detto nel precedente paragrafo, si ha:

$$\mathbf{C}_{body}^{NED}(\psi, \theta, \varphi) = (\mathbf{C}_{NED}^{body}(\psi, \theta, \varphi))^{-1} = (\mathbf{C}_{NED}^{body}(\psi, \theta, \varphi))^T = \begin{pmatrix} \cos \theta \cos \psi & \sin \theta \sin \varphi \cos \psi - \cos \varphi \sin \psi & \sin \theta \cos \varphi \cos \psi + \sin \varphi \sin \psi \\ \cos \theta \sin \psi & \sin \theta \sin \varphi \sin \psi + \cos \varphi \cos \psi & \sin \theta \cos \varphi \sin \psi - \sin \varphi \cos \psi \\ -\sin \theta & \sin \varphi \cos \theta & \cos \varphi \cos \theta \end{pmatrix} \quad (2.12)$$

Infine, per completare il quadro, riportiamo la matrice di rotazione per trasformare le coordinate di un vettore espresse in un sistema ECEF in coordinate NED:

$$\mathbf{C}_{ECEF}^{NED}(\Phi, \lambda) = \begin{pmatrix} -\sin \Phi \cos \lambda & -\sin \Phi \sin \lambda & \cos \Phi \\ -\sin \lambda & \cos \lambda & 0 \\ -\cos \Phi \cos \lambda & -\cos \Phi \sin \lambda & -\sin \Phi \end{pmatrix} \quad (2.13)$$

Dove ricordiamo che Φ e λ sono rispettivamente: la latitudine e la longitudine geodetica [3].

N.B.: il dominio di definizione degli angoli di Eulero è il seguente:

$$\begin{cases} \varphi \in]-\pi, \pi [\\ \theta \in]-\pi/2, \pi/2 [\\ \psi \in]-\pi, \pi [\end{cases} \quad (2.14)$$

2.4. Equazioni di propagazione dell'assetto e della matrice di rotazione

Come detto in precedenza, questo paragrafo risulterà strategico ai fini della scrittura del simulatore per poter evitare, o perlomeno diminuire, errori di calcolo dovuti all'utilizzo di derivate o dovuti all'eccessivo uso di integrazioni.

Infatti, com'è noto, il calcolo della derivata di una funzione in un punto implica un processo al limite che può solo essere approssimato da un calcolatore con conseguente nascita di errori di calcolo non indifferenti.

In particolare, senza scendere troppo nel merito, sappiamo che l'integrazione porta ad errori di calcolo numerici minori, ed accettabili se il numero di integrazioni presenti nel simulatore non è eccessivo, rispetto all'operatore di derivazione.

Allora, di conseguenza, dobbiamo fare in modo che le derivate siano svolte prima di essere implementate all'interno del simulatore e gli integrali siano in minor numero possibile.

Andiamo ora a studiare come gli angoli di Eulero evolvono nel tempo e il legame che sussiste fra loro e le velocità angolari di un velivolo.

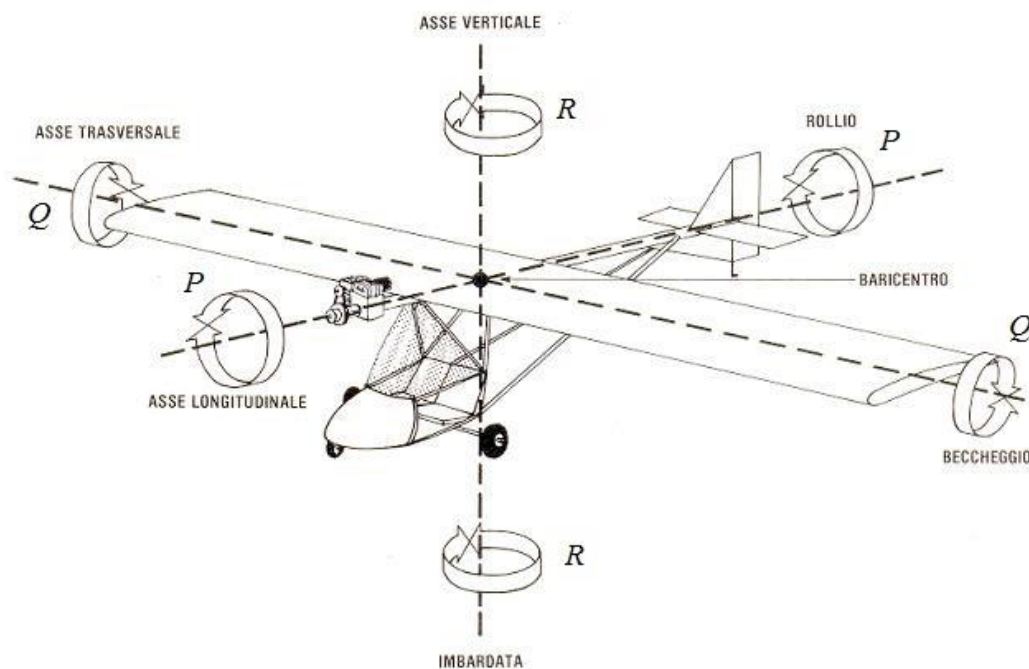


Figura 11: Componenti della velocità rotazionale velivolo

Assumeremo, da ora in poi, che la componenti della velocità traslazionale del corpo rigido di cui parleremo, espresse nel sistema body, sono:

$$\mathbf{v}^b = [U, V, W]^T$$

Mentre, le componenti della velocità rotazionale, come si vede anche dalla Figura 11, espresse in assi body, sono:

$$\boldsymbol{\omega}^b = [P, Q, R]^T$$

Dove P è la velocità angolare di rollio (roll rate), Q è la velocità angolare di beccheggio (pitch rate) e R è la velocità angolare di imbardata (yaw rate).

Per trovare il legame che sussiste fra le velocità angolare succitate e gli angoli di Eulero, possiamo esprimere il vettore $\boldsymbol{\omega}^b$ come somma dei vettori di rotazione dei sistemi corrispondenti, esprimendoli in assi body:

$$\begin{pmatrix} P \\ Q \\ R \end{pmatrix} = \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} + \mathbf{C}_{1,2,3}^{body} \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + \mathbf{C}_{1,2,3}^{body} \mathbf{C}_{1,2,3}^{body} \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \quad (2.15)$$

Dove, le matrici di rotazione che appaiono, sono:

$$\mathbf{C}_{1,2,3}^{body} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & \sin \varphi \\ 0 & -\sin \varphi & \cos \varphi \end{pmatrix} \quad (2.16)$$

$$\mathbf{C}_{1,2,3}^{body} = \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix} \quad (2.17)$$

Eseguendo i calcoli, si ha che:

$$\begin{cases} P = \dot{\psi} \sin \theta + \dot{\varphi} \\ Q = \dot{\psi} \sin \varphi \cos \theta + \dot{\theta} \cos \varphi \\ R = \dot{\psi} \cos \varphi \cos \theta - \dot{\theta} \sin \varphi \end{cases} \quad (2.18)$$

E, mediante alcuni passaggi algebrici, si ottengono le tre equazioni differenziali ordinarie, non lineari, dette equazioni cinematiche di propagazione degli angoli di Eulero:

$$\begin{cases} \dot{\varphi} = P + Q \sin \varphi \tan \theta + R \cos \varphi \tan \theta \\ \dot{\theta} = Q \cos \varphi - R \sin \varphi \\ \dot{\psi} = Q \sin \varphi \sec \theta + R \cos \varphi \sec \theta \end{cases} \quad (2.19)$$

È da notare che, per piccoli angoli di elevazione ed inclinazione, gli angoli di Eulero coincidono con gli angoli di beccheggio, rollio ed imbardata [3].

Come sarà spiegato successivamente, il simulatore utilizzerà un sistema di navigazione inerziale denominato Strap-down, il quale utilizza la derivata della matrice di rotazione da body a NED per riportare, istante per istante, le accelerazioni, fornite dagli accelerometri, in assi NED.

È dunque necessario ora capire in che modo possiamo calcolare la derivata ordinaria di una matrice di rotazione e dunque la sua propagazione nel tempo.

Assumiamo di avere due sistemi di riferimento, A e B, con il secondo che ruota rispetto al primo con velocità di rotazione $\boldsymbol{\omega}_{B/A}$.

Per definizione di derivata, si ha:

$$\frac{d}{dt} \mathbf{C}_A^{B(t)} = \dot{\mathbf{C}}_A^{B(t)} = \lim_{dt \rightarrow 0} \frac{\mathbf{C}_A^{B(t+dt)} - \mathbf{C}_A^{B(t)}}{dt} \quad (2.20)$$

Dunque, se assumiamo che B è il sistema body ed il vettore velocità angolare ha componenti in assi body, allora il vettore velocità angolare possiamo assumerlo pari a:

$$\boldsymbol{\omega}_{B/A}^B = \boldsymbol{\omega}_{b/i}^{body} = \begin{pmatrix} P \\ Q \\ R \end{pmatrix} \quad (2.21)$$

Se poi scomponiamo la rotazione in due rotazioni $\mathbf{C}_A^{B(t)}$ e $\mathbf{C}_B^{B(t+dt)}$, dove:

$$\mathbf{C}_A^{B(t+dt)} = \mathbf{C}_{B(t)}^{B(t+dt)} \mathbf{C}_A^{B(t)} \quad (2.22)$$

Allora possiamo scrivere:

$$\dot{\mathbf{C}}_A^{B(t)} = \lim_{dt \rightarrow 0} \frac{\mathbf{C}_{B(t)}^{B(t+dt)} \mathbf{C}_A^{B(t)} - \mathbf{C}_A^{B(t)}}{dt} = \lim_{dt \rightarrow 0} \frac{(\mathbf{C}_{B(t)}^{B(t+dt)} - \mathbf{I})}{dt} \mathbf{C}_A^{B(t)} \quad (2.23)$$

Dove:

$$\mathbf{C}_{B(t)}^{B(t+dt)} - \mathbf{I} = \begin{pmatrix} 1 & d\psi & -d\theta \\ -d\psi & 1 & d\varphi \\ d\theta & -d\varphi & 1 \end{pmatrix} - \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & d\psi & -d\theta \\ -d\psi & 0 & d\varphi \\ d\theta & -d\varphi & 0 \end{pmatrix}$$

$$\lim_{dt \rightarrow 0} \frac{\mathbf{C}_{B(t)}^{B(t+dt)} - \mathbf{I}}{dt} \mathbf{C}_A^{B(t)} = \begin{pmatrix} 0 & \dot{\psi} & -\dot{\theta} \\ -\dot{\psi} & 0 & \dot{\varphi} \\ \dot{\theta} & -\dot{\varphi} & 0 \end{pmatrix} \mathbf{C}_A^{B(t)} \quad (2.24)$$

Infine, supponendo piccoli angoli di imbardata, rollio e beccheggio:

$$\begin{pmatrix} 0 & \dot{\psi} & -\dot{\theta} \\ -\dot{\psi} & 0 & \dot{\varphi} \\ \dot{\theta} & -\dot{\varphi} & 0 \end{pmatrix} = \begin{pmatrix} 0 & R & -Q \\ -R & 0 & P \\ Q & -P & 0 \end{pmatrix} \quad (2.25)$$

Nota: Prodotto vettoriale

Siano \mathbf{v} e \mathbf{w} due vettori in \mathbb{R}^3 , con $\mathbf{v} = (v_1, v_2, v_3)^T$ e $\mathbf{w} = (w_1, w_2, w_3)^T$ allora, eseguendone il prodotto vettoriale, avremo:

$$\mathbf{v} \times \mathbf{w} = (\mathbf{v} \times) \mathbf{w}$$

Dove:

$$(\mathbf{v} \times) = \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} \quad (2.26)$$

Inoltre, prendendo due sistemi di riferimento, a e b, avremo che:

$$\mathbf{v}^a \times \mathbf{w}^a = (\mathbf{v}^a \times) \mathbf{w}^a = \mathbf{C}_b^a (\mathbf{v}^b \times) \mathbf{w}^b = \mathbf{C}_b^a (\mathbf{v}^b \times) \mathbf{C}_a^b \mathbf{w}^a$$

E dunque:

$$\begin{aligned} (\mathbf{v}^a \times) \mathbf{w}^a &= \mathbf{C}_b^a (\mathbf{v}^b \times) \mathbf{C}_a^b \mathbf{w}^a \\ (\mathbf{v}^a \times) &= \mathbf{C}_b^a (\mathbf{v}^b \times) \mathbf{C}_a^b \end{aligned} \quad (2.27)$$

Fine nota

Allora, possiamo dire che, in base alla nota precedente, e ricordando la (2.21):

$$\dot{\mathbf{C}}_A^{B(t)} = \begin{pmatrix} 0 & R & -Q \\ -R & 0 & P \\ Q & -P & 0 \end{pmatrix} \mathbf{C}_A^{B(t)} = -(\boldsymbol{\omega}_{B/A}^B \times) \mathbf{C}_A^{B(t)} \quad (2.28)$$

E dunque:

$$\dot{\mathbf{C}}_A^{B(t)} = (\boldsymbol{\omega}_{A/B}^B \times) \mathbf{C}_A^{B(t)} \quad (2.29)$$

Abbiamo ora concluso questo capitolo, avendo dettato tutte le basi matematiche, riguardanti un primo approccio alla cinematica del nostro corpo rigido, che risulteranno utili al lettore al fine di comprendere al meglio il lavoro svolto e per introdurlo ai sistemi di navigazione inerziale, di cui parleremo nel prossimo capitolo.



3. Sistemi di navigazione inerziale INS

3.1. Principio di funzionamento base

I sistemi di navigazione inerziale (Inertial Navigation System – INS) si basano su leggi della meccanica classica formulate da Sir Isaac Newton.

Le leggi di Newton ci dicono che il moto di un corpo continuerà uniformemente lungo una linea retta finché non sarà disturbato da una forza esterna agente sul corpo. Le leggi ci dicono anche che questa forza produrrà un'accelerazione del corpo, direttamente proporzionale alla forza stessa, tramite un fattore di proporzionalità m che rappresenta proprio la massa del corpo.

Allora, avendo la possibilità di misurare l'accelerazione, tramite gli accelerometri, sarebbe dunque possibile calcolare il cambiamento di velocità e di posizione utilizzando delle successive integrazioni matematiche dell'accelerazione nel tempo.

Un sistema di navigazione inerziale, solitamente, utilizza tre accelerometri, ognuno dei quali calcola l'accelerazione in una singola direzione (gli accelerometri vengono comunemente montati con i loro assi sensitivi mutuamente perpendicolari).

È necessario inoltre mantenere traccia della direzione in cui gli accelerometri stanno puntando. Il moto rotazionale del corpo, rispetto al sistema di riferimento inerziale, può essere “sentito” mediante l'utilizzo di sensori giroscopici, utilizzati per determinare l'orientamento degli accelerometri in ogni istante; con quest'ultima informazione, è allora possibile ricavare le accelerazioni nel sistema di riferimento utile, prima del processo di integrazioni che prenderà posto.

Dunque, la navigazione inerziale è il processo per cui le misure fornite dai giroscopi e dagli accelerometri, contenuti nella IMU (Inertial Measurement Unit), sono utilizzate per determinare la posizione del velivolo in cui sono installati.

Combinando due set di misure, è possibile definire il moto traslazionale del velivolo all'interno del sistema di riferimento inerziale e così calcolare la sua posizione all'interno dello stesso.

Diversamente da altri tipi di sistemi di navigazione, i sistemi inerziali sono completamente auto-contenuti all'interno del velivolo, nel senso che non dipendono dalla trasmissione di segnali da velivoli o la ricezione da una sorgente esterna e dunque sono immuni da disturbi esterni, quali quelli che perturbano, ad esempio, un segnale GPS. [4]

Lo svantaggio principale dei sistemi INS è che la loro precisione, come vedremo poi attraverso i grafici degli andamenti, tende a peggiorare nel tempo, per via del fenomeno della deriva, dovuto alla natura intrinseca del concetto di integrale nel tempo applicato ad un errore di misura, anche piccolo.

Per questa ragione, essi devono periodicamente essere re-inizializzati mediante altri sistemi quali ad esempio il GNSS o essere integrati mediante la data-fusion con altri sistemi quali ad esempio altimetri barometrici.

Nel prossimo paragrafo, all'interno di questo capitolo, vedremo come, tramite le equazioni di navigazione standard, è possibile stabilire un legame tra le misure inerziali e la soluzione di navigazione.

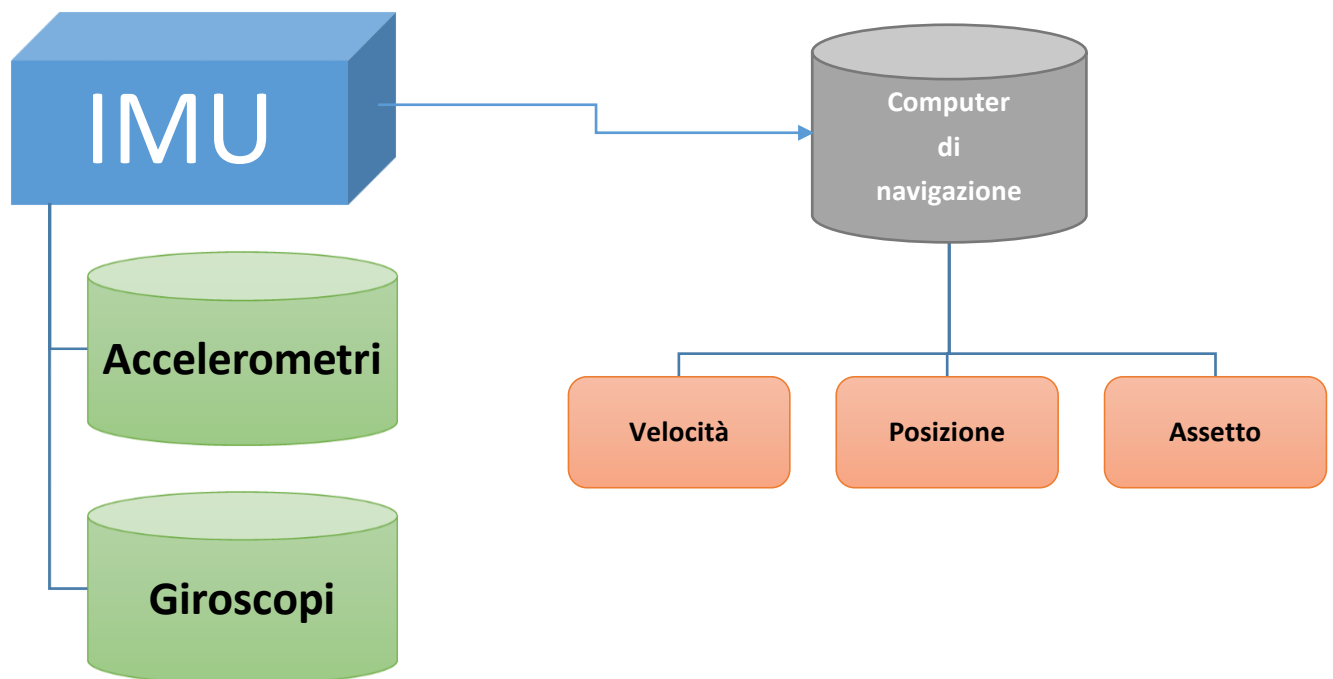


Figura 12: Schema generale di un INS

3.2. *Equazioni di navigazione standard*

Le equazioni di navigazione standard sono equazioni differenziali ordinarie, non lineari, del primo ordine che, in base alla conoscenza delle condizioni iniziali di velocità, posizione ed assetto del velivolo, dettate in un dato istante iniziale t_0 , forniscono l'evoluzione nel tempo della cosiddetta soluzione di navigazione, ovvero la posizione e la velocità, in funzione delle uscite degli accelerometri e dei giroscopi (accelerazione e rotazioni).

Notevole importanza assume il sistema di riferimento in cui verranno scritte le coordinate dei vettori, che può facilitare, o render più ostiche, la risoluzione e la deduzione delle stesse.

La nostra scelta è caduta sul sistema di riferimento NED, detto anche di navigazione; naturalmente, le uscite dei sensori non saranno direttamente in assi NED e dunque in seguito dovremo studiare come possiamo trasformare queste uscite portandole nel nostro sistema di navigazione.

Il principio base su cui si basano le equazioni di navigazione è l'integrazione; infatti, la velocità è ottenuta come l'integrale dell'accelerazione e la posizione come l'integrale della velocità. Come già anticipato all'interno del capitolo precedente, nella risoluzione delle equazioni all'interno del simulatore, dovremo trovare delle vie che ci permettono di ridurre il numero di integrali presenti per evitare eventuali problemi connessi al calcolo numerico.

Cominciamo col definire tutte le possibili variabili e costanti di cui faremo uso durante la determinazione delle equazioni suddette:

- \mathbf{a}_i *Accelerazione inerziale*
- \mathbf{g} *Accelerazione gravitazionale*
- i *Sistema di riferimento inerziale ECI*
- e *Sistema di riferimento ECEF*
- n *Sistema di riferimento NED*
- b *Sistema di riferimento body*
- \mathbf{P} *Vettore di posizione del baricentro del velivolo rispetto ad ECEF*
- $\boldsymbol{\omega}_{a/s}$ *Velocità di rotazione del sistema di riferimento generico a rispetto al sistema di riferimento generico s*
- $\boldsymbol{\Omega} = \boldsymbol{\omega}_{e/i}$ *Velocità di rotazione terrestre rispetto al sistema ECI*
- \mathbf{V}_i *Velocità traslazionale rispetto al sistema inerziale ECI*
- \mathbf{V}_e *Velocità traslazionale rispetto al sistema ECEF*
- t *Tempo*

N.B.: Ricordiamo, come già detto in precedenza, l'apice di un vettore indica il sistema di riferimento rispetto a cui sono scritte le coordinate del vettore, mentre il pedice ci dice in quale sistema di riferimento è definito il vettore.

Come punto di partenza per la nostra dimostrazione, ricaviamo il legame che sussiste tra il vettore posizione definito in ECEF e con coordinate nel sistema inerziale e il vettore posizione definito in ECEF e con coordinate espresse in ECEF:

$$\mathbf{P}^i = \mathbf{C}_e^i \mathbf{P}^e \quad (3.1)$$

Da cui, derivando rispetto al tempo si ha:

$$\mathbf{V}_i^i = \frac{d\mathbf{P}^i}{dt} = \dot{\mathbf{C}}_e^i \mathbf{P}^e + \mathbf{C}_e^i \dot{\mathbf{P}}^e \quad (3.2)$$

Ricordando poi l'espressione della derivata temporale di una matrice di rotazione, descritta dalla (2.29), si ha:

$$\mathbf{V}_i^i = \mathbf{C}_e^i (\boldsymbol{\omega}_{e/i}^e \times) \mathbf{P}^e + \mathbf{C}_e^i \mathbf{V}_e^e \quad (3.3)$$

Derivando poi ambo i membri della precedente otteniamo:

$$\dot{\mathbf{V}}_i^i = \dot{\mathbf{C}}_e^i (\boldsymbol{\Omega}^e \times) \mathbf{P}^e + \mathbf{C}_e^i (\dot{\boldsymbol{\Omega}}^e \times) \mathbf{P}^e + \mathbf{C}_e^i (\boldsymbol{\Omega}^e \times) \dot{\mathbf{P}}^e + \dot{\mathbf{C}}_e^i \mathbf{V}_e^e + \mathbf{C}_e^i \dot{\mathbf{V}}_e^e \quad (3.4)$$

Assumendo che la velocità terrestre è costante, $\dot{\boldsymbol{\Omega}}^e = 0$, notando che \mathbf{a}_i^i è il vettore $\dot{\mathbf{V}}_i^i$, ovvero l'accelerazione inerziale, abbiamo:

$$\mathbf{a}_i^i = \mathbf{C}_e^i (\boldsymbol{\Omega}^e \times) (\boldsymbol{\Omega}^e \times) \mathbf{P}^e + \mathbf{C}_e^i (\boldsymbol{\Omega}^e \times) \mathbf{V}_e^e + \mathbf{C}_e^i (\boldsymbol{\Omega}^e \times) \mathbf{V}_e^e + \mathbf{C}_e^i \dot{\mathbf{V}}_e^e \quad (3.5)$$

Moltiplicando ambo i membri a sinistra per la matrice di rotazione \mathbf{C}_i^e e considerando che:

$$\mathbf{V}_e^e = \mathbf{C}_n^e \mathbf{V}_e^n$$

La quale, derivata, diviene:

$$\dot{\mathbf{V}}_e^e = \mathbf{C}_n^e (\boldsymbol{\omega}_{n/e}^n \times) \mathbf{V}_e^n + \mathbf{C}_n^e \dot{\mathbf{V}}_e^n \quad (3.6)$$

Andando ora a sostituire il tutto all'interno dell'equazione (3.5), avremo:

$$\mathbf{a}_i^e = (\boldsymbol{\Omega}^e \times)(\boldsymbol{\Omega}^e \times) \mathbf{P}^e + (\boldsymbol{\Omega}^e \times) \mathbf{V}_e^e + (\boldsymbol{\Omega}^e \times) \mathbf{V}_e^e + \mathbf{C}_n^e (\boldsymbol{\omega}_{n/e}^n \times) \mathbf{V}_e^n + \mathbf{C}_n^e \dot{\mathbf{V}}_e^n \quad (3.7)$$

Infine, moltiplicando a sinistra entrambi i membri dell'equazione appena ricavata per \mathbf{C}_n^e e riordinando il tutto, avremo la forma finale dell'equazione di navigazione standard:

$$\dot{\mathbf{V}}_e^n = \mathbf{a}_i^n - \boldsymbol{\Omega}^n \times (\boldsymbol{\Omega}^n \times \mathbf{P}^n) - (2\boldsymbol{\Omega}^n + \boldsymbol{\omega}_{n/e}^n) \times \mathbf{V}_e^n \quad (3.8)$$

Andiamo ora ad analizzare ogni singolo termine presente nell'equazione per scriverlo in componenti espresse in assi NED¹⁰.

$$\dot{\mathbf{V}}_e^n = \begin{pmatrix} \dot{V}_N \\ \dot{V}_E \\ \dot{V}_D \end{pmatrix}, \quad \mathbf{V}_e^n = \begin{pmatrix} V_N \\ V_E \\ V_D \end{pmatrix}, \quad \text{Velocità rotazionale terrestre} \quad \boldsymbol{\Omega}^n = \begin{pmatrix} \Omega \cos \Phi \\ 0 \\ -\Omega \sin \Phi \end{pmatrix},$$

$$\text{Accelerazione inerziale} \quad \mathbf{a}_i^n = \begin{pmatrix} a_N \\ a_E \\ a_D \end{pmatrix}, \quad \text{Accelerazione indotta dalla rotazione terrestre} \quad \overbrace{\boldsymbol{\Omega}^n \times (\boldsymbol{\Omega}^n \times \mathbf{P}^n)} = \Omega^2 \|\mathbf{P}^n\| \cos \Xi \begin{pmatrix} \sin \Phi \\ 0 \\ \cos \Phi \end{pmatrix}$$

¹⁰ In questa trattazione si assume che la terra abbia la forma descritta nel modello WGS84 con i relativi dati descritti nel capitolo precedente riguardo $N(\Phi)$, $\rho(\Phi)$ con Φ latitudine geodetica.

$$\boldsymbol{\omega}_{n/e}^n = \begin{pmatrix} \omega_N \\ \omega_E \\ \omega_D \end{pmatrix} = \begin{pmatrix} \frac{V_E}{N(\Phi) + h} \\ -\frac{V_N}{\rho(\Phi) + h} \\ -\frac{V_E}{N(\Phi) + h} \tan \Phi \end{pmatrix}, \quad (3.9)$$

Dove:

$$\|\mathbf{P}^n\| = \sqrt{(N(\Phi) + h)^2 \cos^2 \Phi + (N(\Phi)(1 - e^2) + h)^2 \sin^2 \Phi} \quad (3.10)$$

Dalle precedenti si possono determinare anche le relazioni che forniscono la posizione in termini di coordinate geodetiche:

$$\begin{cases} \dot{\lambda} = \frac{V_E}{(N(\phi) + h) \cos \phi} = \frac{\omega_N^n}{\cos \Phi} \\ \dot{\phi} = \frac{V_N}{\rho(\phi) + h} = -\omega_E^n \\ \dot{h} = -V_D \end{cases} \quad (3.11)$$

Da cui, conoscendo le condizioni iniziali, $\lambda(t_0) = \lambda_0$, $\Phi(t_0) = \Phi_0$, $h(t_0) = h_0$, si ha:

$$\begin{cases} \lambda(t) = \int_{t_0}^t \dot{\lambda}(\tau) d\tau + \lambda_0 \\ \Phi(t) = \int_{t_0}^t \dot{\Phi}(\tau) d\tau + \Phi_0 \\ h(t) = \int_{t_0}^t \dot{h}(\tau) d\tau + h_0 \end{cases}$$

N.B.: per la risoluzione della (3.8) è necessario conoscere le condizioni iniziali di velocità e posizione di tutte le loro componenti e dunque saranno necessarie sei condizioni iniziali.

Come abbiamo già detto più volte, le accelerazioni ci vengono fornite dagli accelerometri, i quali non ci forniscono direttamente le accelerazioni inerziali \mathbf{a}_i , ma ci forniscono quest'ultime decurtate dalla componente dell'accelerazione gravitazionale presente sull'asse di ingresso del sensore; dunque avremo che:

$$\mathbf{a}_i = \mathbf{a}_m + \mathbf{g}$$

Dove con \mathbf{a}_m abbiamo indicato le accelerazioni misurate dagli accelerometri.

Allora, la (3.8), diviene:

$$\dot{\mathbf{V}}_e^n = \mathbf{a}_m^n + \mathbf{g}^n - \boldsymbol{\Omega}^n \times (\boldsymbol{\Omega}^n \times \mathbf{P}^n) - (2\boldsymbol{\Omega}^n + \boldsymbol{\omega}_{n/e}^n) \times \mathbf{V}_e^n \quad (3.12)$$

Dove:

$$\mathbf{g}^n = \begin{pmatrix} g \sin(\Phi - \Xi) \\ 0 \\ g \cos(\Phi - \Xi) \end{pmatrix} \quad (3.13)$$

Abbiamo in questa sezione visto le equazioni che il computer di bordo deve elaborare per riuscire ad estrapolare la soluzione di navigazione ad ogni istante di tempo t .

Andiamo ora a studiare un particolare tipo di IMU che è stata implementata all'interno del nostro simulatore, ovvero il sistema così detto Strapdown [3].

3.3. Sistemi di tipo Strapdown per la navigazione inerziale

Con la loro evoluzione, i sistemi di navigazione inerziale a piattaforma stabilizzata, necessariamente divengono basati su design meccanici sempre più intricati. Negli ultimi anni, la tecnologia è andata avanti in molte discipline dell'elettronica e della meccanica. Tutto ciò si riflette nell'evoluzione del sistema di navigazione inerziale Strapdown.

Con l'ausilio di meno parti e meccanismi mobili rispetto ai sistemi stabilizzati, il sistema Strapdown ha elevati benefici che ne derivano dall'uso di computer tecnologicamente avanzati, i quali sono costruiti su tecnologie basate sull'elettronica, sull'ottica e sullo stato solido.

I sistemi di navigazione inerziale Strapdown sono fissati rigidamente al velivolo, dunque, essi si muovono con il corpo; di conseguenza, i loro giroscopi sentono e misurano le stesse velocità angolari del corpo in movimento. [5]

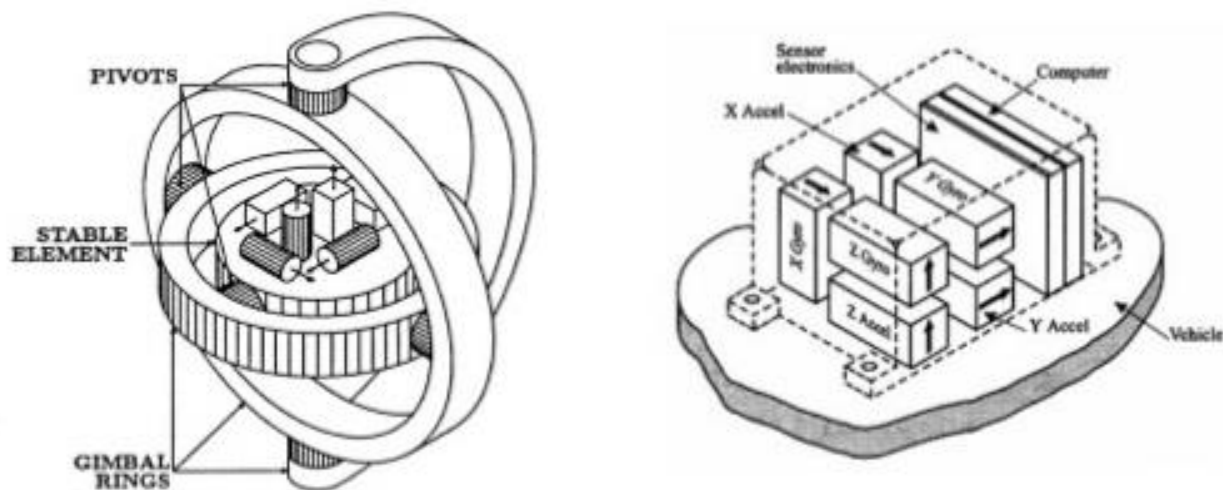


Figura 13: Piattaforma stabilizzata (a sinistra) e Strapdown (a destra)

Di conseguenza, per fornire le accelerazioni in assi NED, e dunque nel sistema di navigazione che abbiamo utilizzato per la descrizione delle nostre equazioni di navigazione standard, dobbiamo trasformare le accelerazioni sentite dagli accelerometri prima di inserirle all'interno dell'equazione.

Dunque, occorre la conoscenza, istante per istante, della matrice di rotazione che trasforma le coordinate assi-corpo in coordinate NED.

Conoscere istante per istante la matrice \mathbf{C}_b^n , significa conoscere l'equazione differenziale che descrive l'evoluzione della matrice di rotazione tra body e NED in funzione delle rotazioni che subisce il nostro velivolo ($\boldsymbol{\omega}_{b/i}^b$) e in funzione delle rotazioni del sistema di riferimento NED rispetto al sistema inerziale ($\boldsymbol{\omega}_{n/i}^n$).

Andiamo allora a studiare come ricavare il modello matematico sopra citato.

Ricordando la (2.29), ed eseguendo dei semplici calcoli, si ha che:

$$\dot{\mathbf{C}}_b^n = (\boldsymbol{\omega}_{b/n}^n \times) \mathbf{C}_b^n = \mathbf{C}_b^n (\boldsymbol{\omega}_{b/n}^b \times) \quad (3.14)$$

Ma sappiamo anche che:

$$\boldsymbol{\omega}_{b/n}^b = \boldsymbol{\omega}_{b/i}^b - \boldsymbol{\omega}_{n/i}^b \quad (3.15)$$

Allora la (3.14) diventa:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n (\boldsymbol{\omega}_{b/i}^b \times) - \mathbf{C}_b^n (\boldsymbol{\omega}_{n/i}^b \times) \quad (3.16)$$

Dove, ricordando la (2.27), si può scrivere:

$$(\boldsymbol{\omega}_{n/i}^b \times) = \mathbf{C}_b^n (\boldsymbol{\omega}_{n/i}^n \times) \mathbf{C}_b^n$$

Ed infine:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n (\boldsymbol{\omega}_{b/i}^b \times) - (\boldsymbol{\omega}_{n/i}^n \times) \mathbf{C}_b^n \quad (3.17)$$

Dove, $\boldsymbol{\omega}_{b/i}^b$ sono esattamente le uscite dei giroscopi, mentre $\boldsymbol{\omega}_{n/i}^n$ vale:

$$\boldsymbol{\omega}_{n/i}^n = \boldsymbol{\omega}_{n/e}^n + \boldsymbol{\omega}_{e/i}^n = \boldsymbol{\omega}_{n/e}^n + \boldsymbol{\Omega}^n \quad (3.18)$$

La quale, con le equazioni (3.9) sappiamo calcolare [3].

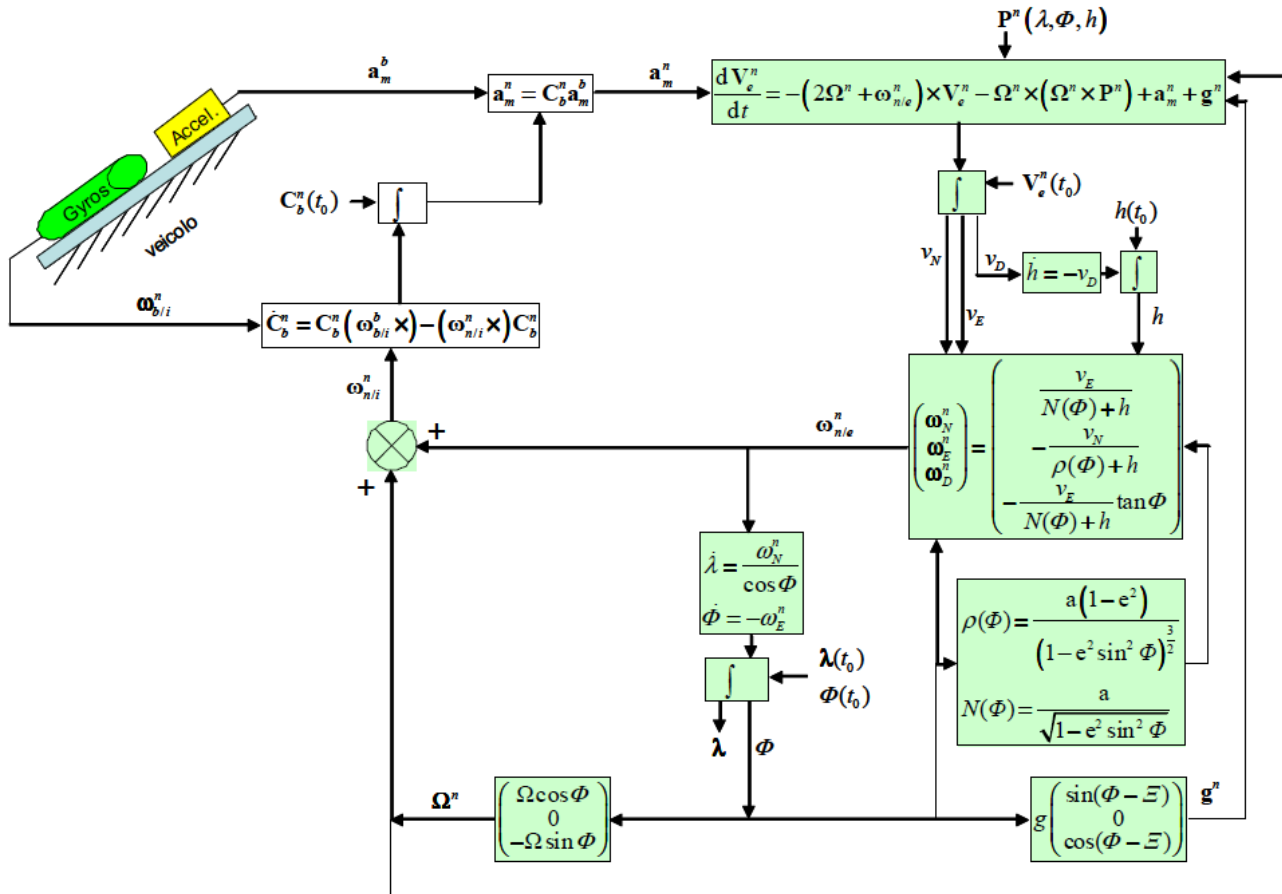


Figura 14: Modello generale di un sistema Strapdown



4. Sensori inerziali

4.1. Definizioni

Il sensore è un trasduttore (“un dispositivo di qualsiasi genere destinato a trasmettere dell’energia da un punto a un altro alterandone alcune delle caratteristiche che la identificano; per trasduttore si intende anche un dispositivo in grado di trasformare una forma di energia in un’altra forma tramite uno o più sistemi di trasformazione” [6]) che si trova in diretta interazione con il sistema misurato ed è, in ambito strettamente metrologico, riferito solamente al componente che fisicamente effettua la trasformazione della grandezza d’ingresso in un segnale di altra natura. I dispositivi in commercio spesso integrano al loro interno anche alimentatori stabilizzati, amplificatori di segnale, dispositivi di comunicazione remota, ecc. In quest’ultimo caso si preferisce definirli trasduttori. [7]

Dunque, il trasduttore è progettato per rilevare un certa grandezza fisica, tuttavia risulta molto difficile costruire un dispositivo selettivo che non risulti influenzato da altre grandezze fisiche; infatti, come poi vedremo più in dettaglio, la temperatura sarà uno dei fattori che influenza le caratteristiche fisiche di quasi tutti i materiali, ed in particolare quelli a semiconduttore; [8] questo diverrà motivo di errore nella misurazione delle nostre quantità e dovremo dunque modellizzare matematicamente questi errori, che poi come vedremo risulteranno per la maggior parte dovuti a processi di natura stocastica, per avvicinare il nostro modello simulato di sensore a quello reale e poter così valutare la bontà dei sensori in commercio che prenderemo in considerazione.

In particolare, come detto in precedenza, ci limiteremo a modellizzare gli accelerometri ed i giroscopi; essi vengono anche definiti sensori inerziali, in quanto forniscono, relativamente, accelerazione e velocità angolare rispetto ad un sistema di riferimento inerziale, ovvero rispetto ad un sistema in cui un corpo accelera se e soltanto se la risultante delle forze che agiscono sul corpo è diversa da zero. [9]

4.2. Accelerometri

Principio di funzionamento

Il principio di funzionamento di un accelerometro è molto semplice: generalmente, esso è costituito da una massa¹¹ m che è libera di spostarsi avanti ed indietro lungo una direzione prefissata (lungo il suo asse sensitivo) all'interno di un *case*. La massa è collegata ad una parete interna del *case* mediante una molla, avente costante elastica k , ed uno smorzatore, avente coefficiente di attrito viscoso λ come è possibile evincere dalla Figura 15.

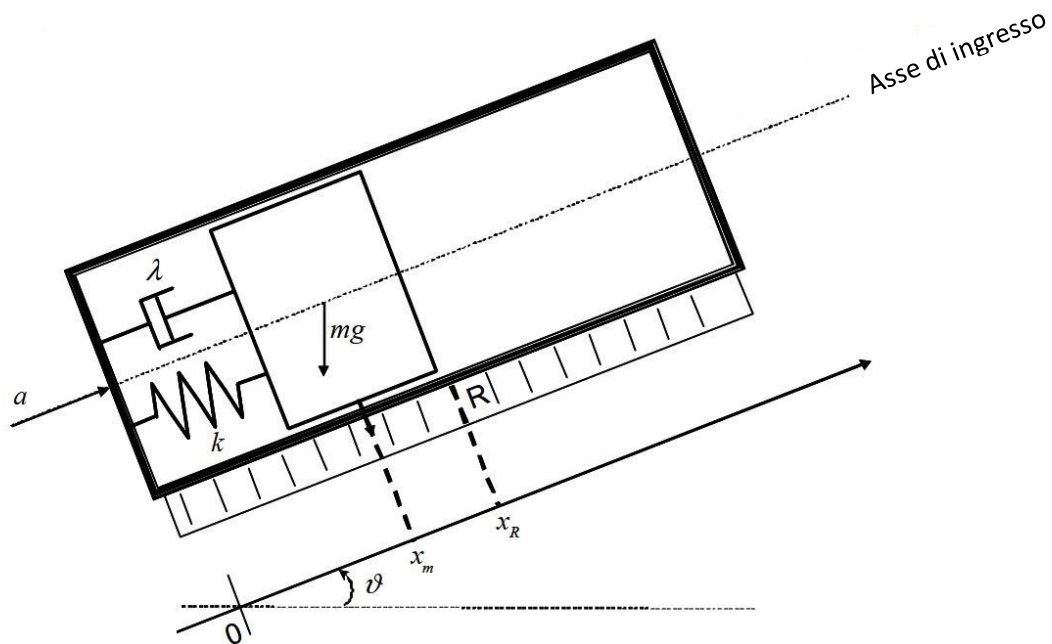


Figura 15: Schema generale di un accelerometro con sistema massa-molla-smorzatore

In condizioni di quiete, ovvero quando risulta che l'accelerazione $a = 0$, il cursore sarà posizionato su R (rispetto al sistema di riferimento inerziale esterno, la posizione corrisponderà a x_R), che rappresenta il valore di riposo.

Modellizzando matematicamente il nostro accelerometro, mediante un'analisi dinamica delle forze in gioco, avremo che agiscono contemporaneamente tre forze sulla nostra massa di prova:

— forza di richiamo elastico della molla (proporzionale a k);

¹¹ La massa è detta di prova ed è dotata di un cursore che scorre su una scala graduata, che fornisce l'entità dello spostamento.

- forza di attrito viscoso dello smorzatore (proporzionale a λ);
- forza di gravità¹² (proporzionale a m);

Dunque, sottoponendo il *case* ad un'accelerazione costante che diremo a , diretta lungo l'asse d'ingresso, la massa di prova si sposterà compiendo piccole oscillazioni e raggiungendo, dopo un certo transitorio τ , la nuova posizione di equilibrio. Come prima, considereremo questa nuova posizione rispetto al sistema di riferimento inerziale esterno, ovvero diremo che la nuova posizione di equilibrio corrisponderà a χ_m .

Dunque, applicando l'equilibrio delle forze in gioco, avremo:

$$m\ddot{\chi}_m = -k(\chi_m - \chi_R) - \lambda(\dot{\chi}_m - \dot{\chi}_R) - mg \sin \theta \quad (4.1)$$

Ponendo:

$$\begin{cases} d = \chi_m - \chi_R \\ a = \ddot{\chi}_R \end{cases}$$

Avremo che la (4.1) diventa:

$$m(\ddot{d} + a) = -kd - \lambda\dot{d} - mg \sin \theta \quad (4.2)$$

La quale diviene:

$$\ddot{d} = -\frac{k}{m}d - \frac{\lambda}{m}\dot{d} - g \sin \theta - a \quad (4.3)$$

Dall'equazione (4.3), trasformando secondo Laplace¹³, passando dunque dal dominio del tempo, al dominio della variabile complessa s , avremo che:

¹² Considereremo la sua proiezione sull'asse longitudinale dell'accelerometro.

¹³ Ciò che ci interessa in questo momento è la funzione di trasferimento, quindi considereremo che la distanza d , abbia condizioni iniziali nulle in quanto guarderemo solo la risposta forzata.

$$s^2 D(s) = -\frac{k}{m} D(s) - \frac{\lambda}{m} s D(s) - g \sin \vartheta - U(s) \quad (4.4)$$

L'equazione (4.4) pone:

$$D(s) = \mathcal{L}\{d(t)\} \rightarrow \text{uscita}$$

$$U(s) = \mathcal{L}\{-g \sin \theta - a(t)\} \rightarrow \text{ingresso}$$

Dove:

$$\mathcal{L}\{f(t)\} = F(s) = \int_{-\infty}^{+\infty} e^{-st} f(t) dt = \int_0^{+\infty} e^{-st} f(t) dt \quad (4.5)$$

È la generica trasformata di Laplace di una funzione del tempo¹⁴, a condizione che l'integrale della (4.5) converga.

Dunque, avremo che:

$$D(s) = G(s)U(s) \quad (4.6)$$

Dove, la funzione di trasferimento sarà:

$$G(s) = \frac{1}{s^2 + \frac{\lambda}{m}s + \frac{k}{m}} \quad (4.7)$$

Sostituendo poi la variabile s con $j\omega$, dove ω è la frequenza e j l'unità complessa, avremo:

$$G(s) = \frac{1}{s^2 + 2\delta\omega_n s + \omega_n^2} \quad (4.8)$$

¹⁴ In particolare, $f(t)$ deve essere regolare, continua a tratti, deve risultare che $|f(t)| < M \cdot \exp(\alpha t)$, con $M, \alpha > 0$ e deve essere un segnale ($f(t) = 0$ per $t < 0$).

Dove, con δ abbiamo indicato il coefficiente di smorzamento e con ω_n la pulsazione naturale.

In particolare, essi valgono:

$$\omega_n = \sqrt{\frac{k}{m}}$$

$$\delta = \frac{\lambda}{2m\omega_n} = \frac{\lambda}{2\sqrt{km}}$$

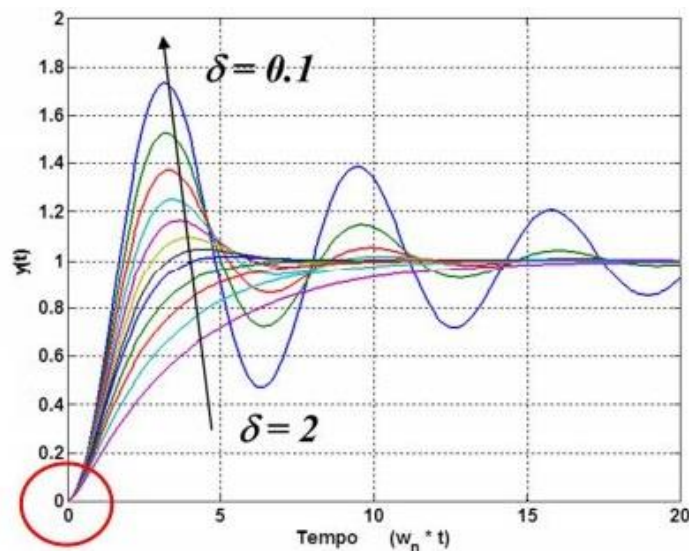


Figura 16: Variazione uscita al variare del coefficiente di smorzamento con ingresso a gradino.

Dunque, a seconda della massa presente all'interno dell'accelerometro, della costante elastica k e del coefficiente di attrito viscoso λ , possiamo dunque agire sul transitorio impiegato dal nostro sensore (in termini di oscillazione della misura e di velocità di risposta) in base alle nostre esigenze ed ai materiali.

Ricordando poi la (3.12), avevamo che l'accelerazione misurata dagli accelerometri, come anche l'accelerazione gravitazionale, deve essere scritta in assi NED:

$$\mathbf{a}_i^n = \mathbf{a}_m^n + \mathbf{g}^n = \mathbf{C}_b^n \mathbf{a}_m^b + \mathbf{g}^n \quad (4.9)$$

Dove:

$$\mathbf{g}^n = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} \quad \& \quad \mathbf{a}_m^b = \begin{pmatrix} a_{m1} \\ a_{m2} \\ a_{m3} \end{pmatrix}$$

Andiamo ora a studiare i diversi tipi di accelerometri che possono essere implementati all'interno dei nostri velivoli, rimandando al capitolo riguardante il simulatore un'analisi più attenta riguardante il confronto di qualità fra i vari sensori.

Classificazione e tipologie di accelerometri

- *MEMS (Micro Electro-Mechanical System)*

I microsistemi elettromeccanici sono un insieme di dispositivi di varia natura (meccanici, elettrici ed elettronici) integrati in forma altamente miniaturizzata su uno spesso substrato di materiale semiconduttore, ad esempio silicio, che coniugano le proprietà elettriche degli integrati a semiconduttore con proprietà opto-meccaniche. Si tratta dunque di sistemi *intelligenti* che abbinano funzioni elettroniche, di gestione dei fluidi, ottiche, biologiche, chimiche e meccaniche in uno spazio ridottissimo, integrando la tecnologia dei sensori e degli attuatori e le più diverse funzioni di gestione dei processi. [10]

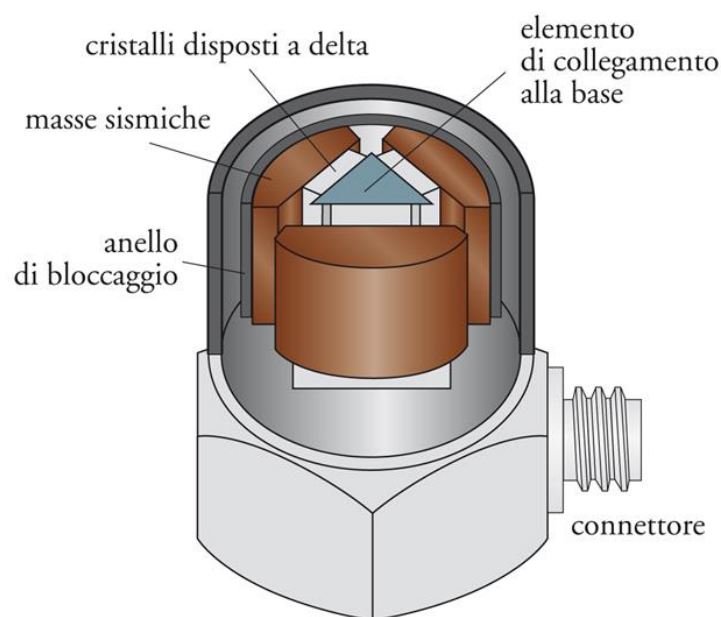


Figura 17: Accelerometro MEMS

L'accelerometro riportato in Figura 17, rappresenta un tipico esempio di accelerometro piezoelettrico. L'effetto piezoelettrico subentra quando, deformandosi, i materiali generano al loro interno delle cariche elettriche¹⁵.

In questo caso, i materiali piezoelettrici risultano essere i cristalli adiacenti alle masse sismiche.

I cristalli possono essere sia naturali, come ad esempio il quarzo, che sintetici, come il solfato di litio.

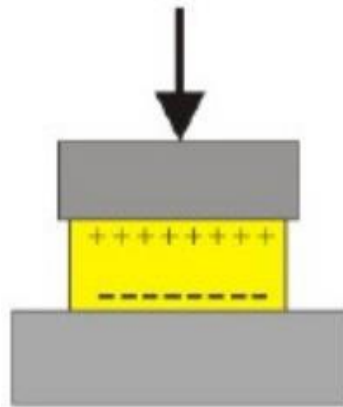


Figura 18: Materiali piezoelettrici

In genere, il materiale piezoelettrico può essere sottoposto ad un precarico opportuno per poter misurare sia accelerazioni positive che negative senza porre in trazione il materiale piezoelettrico e senza rinunciare al segno nel segnale d'uscita. Infatti, il precarico induce una tensione iniziale che progressivamente scompare. [11]

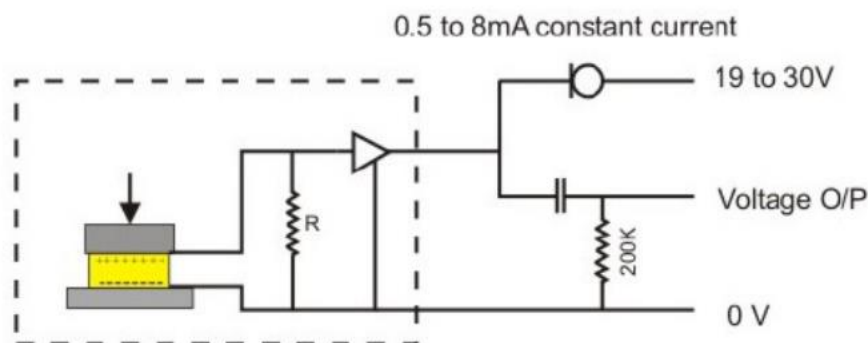


Figura 19: Misurazione deformazione materiale mediante misura della tensione

¹⁵ Il fenomeno piezoelettrico è reversibile.

- Accelerometro a pendolo

Questo tipo di accelerometri consistono di una massa sospesa e vincolata da una molla a balestra (leaf-spring) in modo che il sistema sia simile ad un pendolo.

La massa è analoga ad un'armatura di un trasformatore ad E. Inoltre, essa è chiusa all'interno di un case riempito con un fluido che opera da smorzatore, il quale ha la funzione di smorzare le oscillazioni del pendolo aumentando così la velocità di risposta del sensore.

L'intensità dell'accelerazione sarà direttamente proporzionale allo spostamento del pendolo, mentre il suo verso sarà l'opposto del verso di oscillazione del pendolo.

In Figura 21 c'è un classico esempio di accelerometro a pendolo con struttura di un trasformatore ad E, che presenta dei limiti fisici imposti dagli stop alla misurazione dell'intensità dell'accelerazione [12]. Questi limiti possono essere anche eliminati inserendo un elettrocalamita che permette di mantenere il pendolo in posizione centrale, utilizzando come ingresso la stessa tensione misurata direttamente dagli elettrodi tra cui oscilla la massa ferrosa (il pendolo).

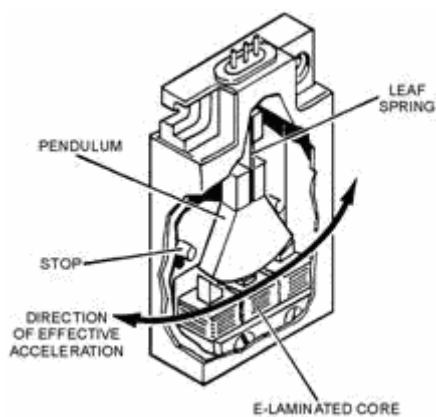


Figura 21: Accelerometro a pendolo

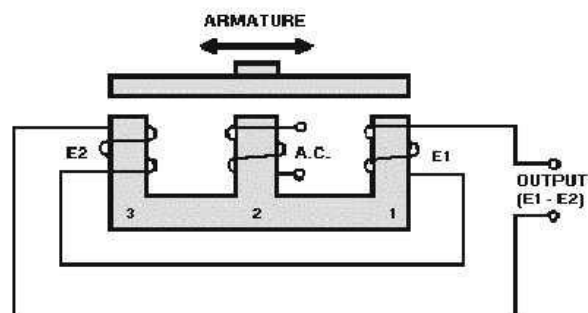


Figura 20: Trasformatore ad E

- Accelerometri laser

Ultimamente si sta puntando all'eliminazione di eventuali errori che occorrono negli accelerometri convenzionali, per via della larghezza di banda sul quale operano, ricorrendo allo studio degli effetti fotoelastici, i quali interessano la frequenza del laser, che permettono una maggior precisione nella determinazione delle accelerazioni.

La generica struttura di un accelerometro laser consiste in una massa di prova che è interposta tra due o più sensori microlaser fotoelastici, agganciati ad un case.

In genere, il *case* esterno viene precaricato con una forza iniziale F_0 che agisce in egual misura sui sensori 1 e 2.

La costruzione dell'accelerometro è una configurazione differenziale, nel senso che la forza agente su un sensore laser, dovuta ad un'accelerazione esterna subita dal corpo, fa sì che decrementi la forza F_0 presente sul secondo sensore:

$$\begin{cases} F_1(t) = F_0 + F_{a1}(t) \\ F_2(t) = F_0 - F_{a2}(t) \end{cases}$$

Con:

$F_{a1}(t)$ & $F_{a2}(t)$ → Forze di inerzia agenti sui cristalli laser

Sono inoltre presenti delle guide parallele per forzare la massa di prova a non andare in direzione diagonale senza falsare il suo spostamento lungo la direzione *sensitiva*.

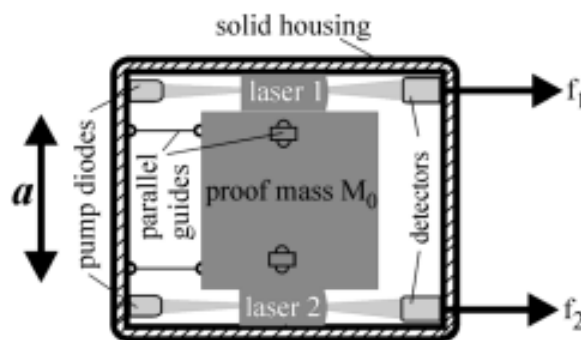


Figura 22: Accelerometro laser

Il modello appena descritto è assimilabile al modello dinamico in Figura 23.

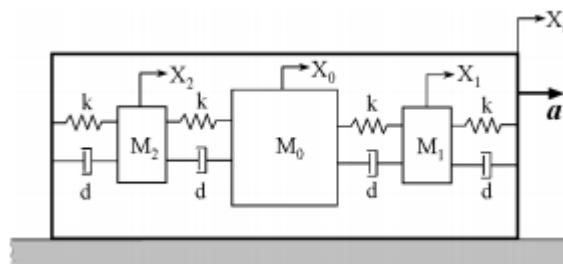


Figura 23: Semplificazione dinamica

Dove M_0 è la nostra massa di prova, M_1 ed M_2 sono le masse dei cristalli laser, mentre k e d sono rispettivamente la costante elastica e il coefficiente di attrito viscoso da determinare empiricamente [13].

In particolare, il modello matematico sarà:

$$\begin{aligned} M_1(\ddot{x}_1 + \ddot{x}_e) + d\dot{x}_1 + kx_1 + d(\dot{x}_1 - \dot{x}_0) + k(x_1 - x_0) &= 0 \\ M_0(\ddot{x}_0 + \ddot{x}_e) + d(\dot{x}_0 - \dot{x}_2) + k(x_0 - x_2) + d(\dot{x}_0 - \dot{x}_1) + k(x_0 - x_1) &= 0 \\ M_2(\ddot{x}_2 + \ddot{x}_e) + d\dot{x}_2 + kx_2 + d(\dot{x}_2 - \dot{x}_0) + k(x_2 - x_0) &= 0 \end{aligned}$$

Chiudiamo, con quest'ultima parentesi, la trattazione degli accelerometri ed andiamo a trattare la seconda categoria di sensori di nostro interesse: i giroscopi.

4.3. Giroscopi

Principio di funzionamento

Come introdotto in precedenza, per poter mantenere allineati gli accelerometri lungo le loro direzioni preferenziali, nel nostro caso lungo gli assi della terna destrorsa NED, abbiamo bisogno di una tipologia di sensori che ci permetta di calcolare le velocità angolari, ovvero le variazioni che subiscono gli angoli compresi tra la direzione attuale degli accelerometri e quella preferenziale.

Come per gli accelerometri, il principio di funzionamento varia al variare del meccanismo su cui si basa il nostro giroscopio. Per brevità di trattazione, vedremo in maniera più dettagliata il principio di funzionamento dei giroscopi MEMS e, successivamente, faremo dei brevi accenni a quello delle altre tipologie esistenti.

Cominciamo dicendo che i giroscopi MEMS si basano sugli effetti della forza di Coriolis.

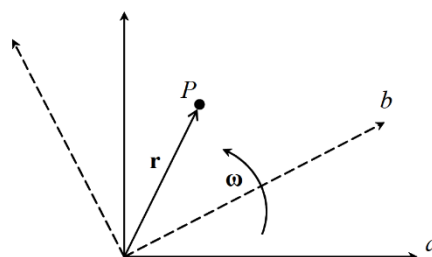


Figura 24: Rotazione tra due sistemi di riferimento

Assumendo che:

- \mathbf{r}^a *Vettore posizione di P con componenti in a*
- \mathbf{r}^b *Vettore posizione di P con componenti in b*
- \mathbf{C}_a^b *Matrice di rotazione da a in b*
- $\boldsymbol{\omega}^a$ *Velocità di rotazione di b rispetto ad a con componenti in a*
- $\boldsymbol{\omega}^b$ *Velocità di rotazione di b rispetto ad a con componenti in b*

Possiamo scrivere:

$$\mathbf{r}^a = \mathbf{C}_b^a \mathbf{r}^b$$

Da cui, derivando rispetto al tempo, si ha:

$$\dot{\mathbf{r}}^a = \dot{\mathbf{C}}_b^a \mathbf{r}^b + \mathbf{C}_b^a \dot{\mathbf{r}}^b = \mathbf{C}_b^a (\boldsymbol{\omega}^b \times) \mathbf{r}^b + \mathbf{C}_b^a \dot{\mathbf{r}}^b \quad (4.10)$$

Derivando poi una seconda volta rispetto al tempo:

$$\begin{aligned} \ddot{\mathbf{r}}^a &= \dot{\mathbf{C}}_b^a (\boldsymbol{\omega}^b \times) \mathbf{r}^b + \mathbf{C}_b^a (\dot{\boldsymbol{\omega}}^b \times) \mathbf{r}^b + \mathbf{C}_b^a (\boldsymbol{\omega}^b \times) \dot{\mathbf{r}}^b + \dot{\mathbf{C}}_b^a \dot{\mathbf{r}}^b + \mathbf{C}_b^a \ddot{\mathbf{r}}^b = \\ &= \mathbf{C}_b^a (\boldsymbol{\omega}^b \times) (\boldsymbol{\omega}^b \times) \mathbf{r}^b + \mathbf{C}_b^a (\dot{\boldsymbol{\omega}}^b \times) \mathbf{r}^b + \mathbf{C}_b^a (\boldsymbol{\omega}^b \times) \dot{\mathbf{r}}^b + \mathbf{C}_b^a (\boldsymbol{\omega}^b \times) \dot{\mathbf{r}}^b + \mathbf{C}_b^a \ddot{\mathbf{r}}^b = \\ &= \boldsymbol{\omega}^a \times (\boldsymbol{\omega}^a \times \mathbf{r}^a) + \dot{\boldsymbol{\omega}}^a \times \mathbf{r}^a + 2\boldsymbol{\omega}^a \times \mathbf{v}_b^a + \mathbf{a}_b^a \end{aligned}$$

Per concludere, esprimendo tutto in funzione dell'accelerazione di P, definita in b ed espressa in assi a, avremo:

$$\mathbf{a}_b^a = \ddot{\mathbf{r}}^a - \boldsymbol{\omega}^a \times (\boldsymbol{\omega}^a \times \mathbf{r}^a) - \dot{\boldsymbol{\omega}}^a \times \mathbf{r}^a - \underbrace{2\boldsymbol{\omega}^a \times \mathbf{v}_b^a}_{\text{Accelerazione di Coriolis}} \quad (4.11)$$

Dunque, dato che a noi serve la forza di Coriolis, basterà moltiplicare l'accelerazione per una massa.

Classificazione e tipologie di giroscopi

- Giroscopi MEMS

Quando una massa m si muove in una direzione v ed è presente anche una velocità angolare Ω , come mostrato in Figura 25, nascerà la succitata forza di Coriolis.

Lo spostamento fisico risultante, che oscillerà nel tempo, sarà successivamente letto da una *struttura sensitiva capacitiva*.

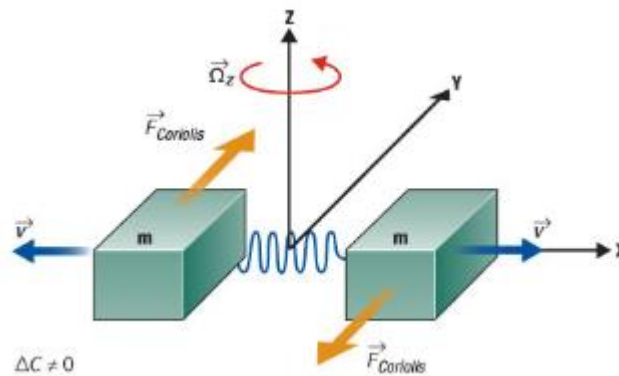


Figura 25: Giroscopio MEMS a forchetta

La maggior parte dei giroscopi MEMS usa una configurazione detta a forchetta: due masse oscillano e si muovono costantemente in direzioni opposte, come mostrato in Figura 25; quando si trova a coesistere anche una velocità angolare, nascerà una forza di Coriolis su entrambe le masse, con direzione mutuamente opposta, causando una variazione della capacità elettrica.

Questo valore differenziale della capacità sarà proporzionale alla velocità angolare e sarà successivamente convertita in una tensione d'uscita o LSB (Least Significant Bit) per giroscopi digitali. N.B.: Se viene applicata un'accelerazione lineare alle due masse, portandole a muoversi nella stessa direzione, non sarà rilevata nessuna variazione della capacità elettrica (i giroscopi MEMS non sono sensitivi alle accelerazioni lineari come ad esempio tilt, impatti o vibrazioni).

- Giroscopi meccanici

La componente primaria di questo tipo di giroscopio è un rotore avente un momento di inerzia J e una velocità angolare ω . Il momento d'inerzia del rotore è determinato dalla distribuzione della

propria massa rispetto al proprio asse di rotazione; aumentando la massa, o il diametro del rotore, aumenterà quindi il momento d'inerzia.

Il momento angolare L del rotore è dato dal prodotto del suo momento d'inerzia per la velocità angolare ω ; esso indica la tendenza dell'asse di rotazione a rimanere parallelo a se stesso e ad opporsi ad ogni tentativo di cambiare il suo orientamento; maggiore è il momento angolare e maggiore sarà la capacità del rotore di reagire alle coppie esterne e, nel nostro caso, maggiore sarà la capacità dello stabilizzatore di annullare il rollio.

Un giroscopio meccanico ha generalmente tre assi: asse di spin (asse attorno a cui ruota il rotore), asse di input (asse a cui sono applicati gli input) ed asse di output (asse trasversale attorno cui il giroscopio ruota o processa in reazione all'input). [14]

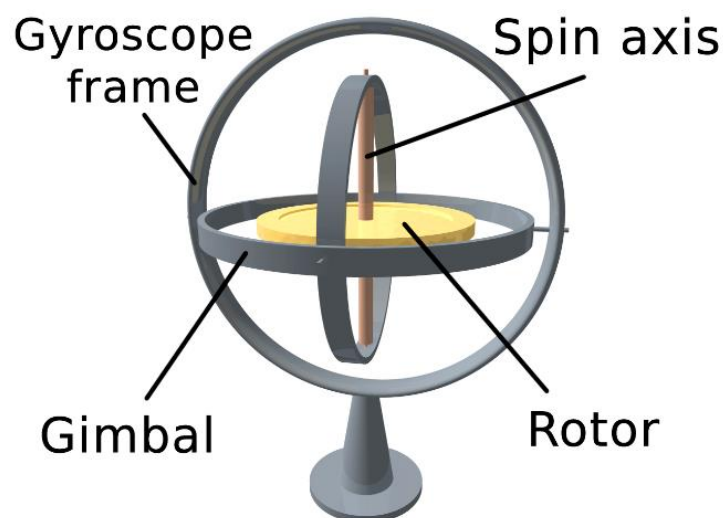


Figura 26: Giroscopio meccanico

- *Giroscopi ottici (FOG-Fiber Optical Gyro)*

I giroscopi ottici, detti anche FOG, basano il loro principio di funzionamento sul cosiddetto effetto Sagnac; questo consiste in una variazione di fase indotta tra due onde propagate, in versi opposti, all'interno di un cammino chiuso (in genere ha una forma a spirale con tanti anelli), quando il piano di rotazione subisce una rotazione angolare.

Di seguito, in Figura 27, è mostrato il principio di funzionamento.

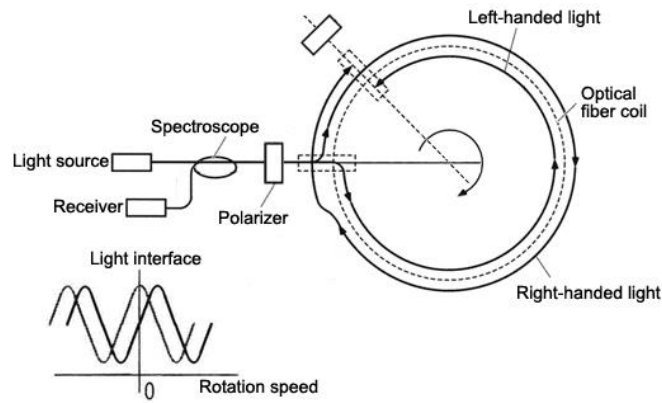


Figura 27: FOG - Fiber Optical Gyro

In generale, lo schema basilico di un FOG è simile ad un interferometro passivo, dove la l'ac-coppiatore di fibra ottica funziona in modo tale da dividere la radiazione proveniente dalla sorgente luminosa in due onde che si propagano in versi opposti all'interno di una spirale: una in senso orario e l'altra in senso antiorario.

Dopo la loro propagazione, all'interno di un photodetector, vengono ricombinate. La differenza di fase è perciò accumulata lungo tutta la spirale per ottenere una maggior qualità nella risposta, avendo uno strumento compatto. [15]

In particolare, supponiamo di avere 2 raggi di luce, L_1 (orario) e L_2 (antiorario), e supponiamo che la nostra piattaforma, come si vede in Figura 27, stia ruotando in senso orario.

Dunque, avremo che, utilizzando rispettivamente i pedici 1 e 2 per le rispettive quantità riferite ai due raggi di luce:

$$\begin{aligned}
 t_1 &= \frac{2\pi R + \omega R t_1}{c} \\
 t_2 &= \frac{2\pi R - \omega R t_2}{c}
 \end{aligned}
 \tag{4.12}$$

Dove:

- c *Velocità di propagazione del raggio luminoso*
- R *Raggio spirale¹⁶*

¹⁶ Supponendo che ogni anello della spirale abbia lo stesso raggio R.

- ω Velocità angolare piattaforma

Dunque, risolvendo le equazioni (4.12) rispetto al tempo e facendone la differenza:

$$t_1 = \frac{2\pi R}{c - \omega R} \quad \& \quad t_2 = \frac{2\pi R}{c + \omega R}$$

$$\Delta t = t_1 - t_2 = \frac{2\pi R}{c - \omega R} - \frac{2\pi R}{c + \omega R} = \frac{4\pi R^2 \omega}{c^2 - \omega^2 R^2} \quad (4.13)$$

Come detto in precedenza, in genere non è un solo anello, quindi dovremmo moltiplicare il tutto per il numero di anelli della spirale e successivamente trasformare quella differenza di tempi in differenza di frequenza ed ottenere così lo *shift* di fase che subiscono le radiazioni per via della rotazione della piattaforma.

- Giroscopi laser

Quest'ultimo tipo di giroscopi combina le funzioni sulla generazione della frequenza ottica dei giroscopi ottici e la rotazione sentita dentro un oscillatore laser all'interno di un anello a forma di cavità.

Tipicamente, essi consistono in un blocco solido di materiali vetroceramici, a forma quadrata o triangolare, in cui è introdotto un laser. Gli elettrodi, i quali in generale sono una miscela di elio o neon, forniscono un guadagno per il laser medio; per via del loro indice di rifrazione pari a 1, essi generano due raggi indipendenti in direzioni opposte attorno alla cavità.

Per supportare il laser, il cammino ottico è strutturato in modo che ci sono essere un numero intero di lunghezze d'onda; inoltre, attorno al cammino stesso, deve avvenire un'oscillazione alla stessa frequenza di quest'ultime.

La differenza tra le frequenze delle due lunghezze d'onda sarà [16]:

$$\Delta f = \frac{4A\Omega}{\lambda_s P}$$

Dove:

- A Area
- P Perimetro della cavità
- λ_s Lunghezza d'onda della luce nel laser medio
- Ω Velocità angolare della rotazione

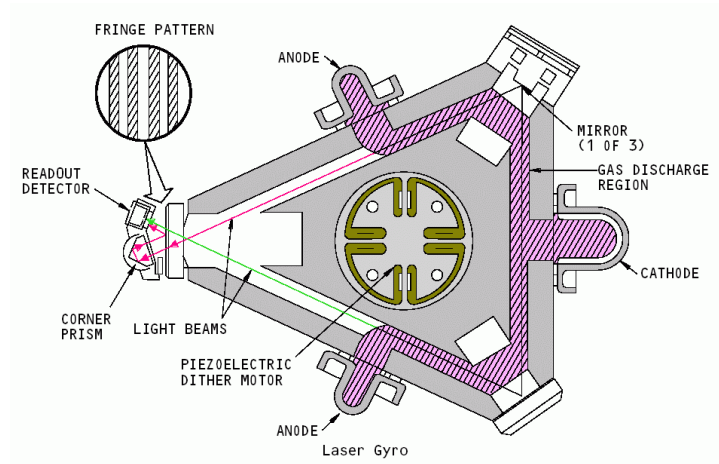


Figura 28: Giroscopio Laser

4.4. Errori sensori

Trovare dei modelli matematici che riescano a descrivere gli errori di cui sono affetti i sensori inerziali può essere uno studio molto intenso, dovuto all'elevato numero di test/prove sperimentali ed analisi statistiche richieste.

Gli output dei sensori inerziali sono influenzati da un gran numero di sorgenti di errori; essi possono essere classificati come errori deterministici e stocastici.

Gli errori deterministici sono dei disturbi non casuali, come ad esempio la dipendenza dalla temperatura della tecnologia MEMS. Al contrario, gli errori stocastici sono disturbi introdotti nel segnale da processi *random* che rappresentano la controparte dei processi deterministici.

L'approccio mediante la funzione di correlazione e la sua corrispondente trasformata nel dominio delle frequenze (trasformando dunque secondo Fourier), la densità spettrale di potenza (PSD – Power Spectral Density), rappresenta la caratterizzazione basica e la modellizzazione degli errori stocastici. [17]

In particolare, vedremo qualitativamente quattro sorgenti di errori che influenza le nostre misure e successivamente, nel capitolo riguardante il simulatore, attraverso i grafici sperimentali, faremo un'analisi quantitativa di questi errori.

Inoltre assumeremo che gli errori che agiscono sugli accelerometri e sui giroscopi abbiano analogia natura, così da poterli studiare una singola volta e applicare la seguente trattazione ad entrambi.

In altre parole, per sinteticità, tratteremo le equazioni dei giroscopi e successivamente implementeremo il tutto anche agli accelerometri.

Dunque, l'equazione di partenza che assumeremo sarà:

$$\boldsymbol{\omega}_m(t) = (\mathbf{I}_{3 \times 3} + \mathbf{M}(SF, MA))\boldsymbol{\omega}(t) + \mathbf{b}(t) + \boldsymbol{\eta}(t) \quad (4.14)$$

Dove:

- $\boldsymbol{\omega}_m(t)$ *Vettore velocità angolare misurata dal giroscopio*
- $\boldsymbol{\omega}(t)$ *Vettore velocità angolare vera*
- $\mathbf{M}(SF, MA)$ *Matrice errore di misura dovuta al fattore di scala e all'errato allineamento del sensore¹⁷*
- $\mathbf{b}(t)$ *Vettore deriva (denominata bias per gli accelerometri e drift per i giroscopi)*
- $\boldsymbol{\eta}(t)$ *Vettore rumore bianco*

N.B.: L'equazione (4.14) è un'equazione vettoriale poiché esprime la misura della velocità angolare lungo i tre assi.

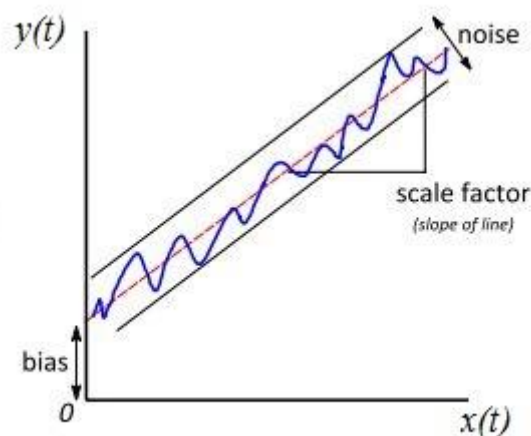


Figura 29: Errori dei sensori

¹⁷ Esso racchiude gli errori deterministici

Fattore di scala (SF - Scale Factor)

L'errore dovuto al fattore di scala è la relazione che sussiste tra l'input e l'output. Se l'ingresso è al 100%, l'uscita aspettata sarà al 100%.

L'uscita attuale è il risultato di un effetto lineare; essa è proporzionale all'ingresso ma scalata.

In altre parole, potremmo considerare il fattore di scala come l'inclinazione del segnale del sensore, come mostrato in Figura 29.

Considerare questo errore come un fattore lineare, è un'approssimazione che non tiene conto della possibile non linearità del rapporto fra ingresso ed uscita.

Spesso, i costruttori, combinano la parte lineare (SF) e la parte non lineare (SFL –Scale Factor Linearity) del fattore di scala in un unico valore.

In genere, diciamo che questo tipo di errore, appare perlopiù durante brusche accelerazioni e rotazioni. [18]

Infine, per descrivere il fattore di scala, può essere utilizzata anche la sua stabilità (Scale Factor Stability), che consiste nella capacità del sensore inerziale di sentire in maniera accurata la velocità angolare (o l'accelerazione) a differenti valori di variazione degli angoli (o per accelerazioni di intensità differenti).

Questo valore si presume essere la variazione del fattore di scala con la temperatura e la sua ripetibilità, che è espressa come parte per milione (ppm). [19]

Errore di allineamento (Misalignment)

L'errore di allineamento dei sensori è spesso una considerazione chiave per ottenere elevate prestazioni nei sistemi di controllo automatico che utilizzano IMU fornite di sensori MEMS.

Per i giroscopi nelle IMU, questo fattore descrive la differenza angolare che sussiste tra ogni asse di rotazione giroscopico ed il sistema pseudo-inerziale [20];

In altre parole, esso può essere causato da un'imperfezione nel montare i sensori e spesso risulta esserci una non-ortogonalità degli assi del sistema di riferimento che abbiamo definito body.

Come naturale conseguenza, abbiamo che ogni asse è affetto dalla misura degli altri due.

Deriva (Bias)

- Constant Bias

Il fattore di deriva nella misurazione della velocità angolare (o dell'accelerazione) è definito come il valor medio dell'uscita del sensore quando non c'è nessuna rotazione angolare (ovvero il valore di offset dell'output dal valor vero), espressa in $^{\circ}/h_r$ (per le accelerazioni si esprime in μg).

Anche un piccolo errore di constant bias, integrato nel tempo, comporta errori che crescono nel tempo e possono divenire significativi.

Infatti, avremo che per i giroscopi, supponendo in prima approssimazione di avere solo l'errore dovuto al bias, l'angolo misurato sarà:

$$\begin{aligned}\theta_m(t) &= \theta_0(t) + \int_{t_0}^t \omega_m(\tau) d\tau = \theta_0(t) + \int_{t_0}^t [\omega(\tau) + b(\tau)] d\tau \\ \theta_m(t) &= \theta(t) + \int_{t_0}^t b(\tau) d\tau = \theta(t) + t * b_c\end{aligned}\quad (4.15)$$

Mentre, per l'accelerometro, dovendo effettuare una doppia integrazione nel tempo, l'errore di posizione cresce con il quadrato del tempo:

$$\begin{aligned}v_m(t) &= v_0(t) + \int_{t_0}^t a_m(\tau) dt = v_0(t) + \int_{t_0}^t [a(\tau) + b(\tau)] dt = v(t) + t * b_c \\ p_m(t) &= p_0(t) + \int_{t_0}^t [v(\tau) + \tau * b_c] d\tau = p(t) + \frac{t^2}{2} b_c\end{aligned}\quad (4.16)$$

In genere, per la stima del valore costante del bias, si effettuano delle lunghe prove, durante le quali il sensore non è sottoposto a rotazioni, e si assume il valor medio delle varie uscite come il valore costante del bias. Una volta a conoscenza di questo valore, lo si sottrae semplicemente dall'uscita. [21]

- Bias Instability

Generalmente, la deriva di un sensore non assume sempre lo stesso valore, ma, per via del così detto *flicker noise*, che sussiste all'interno dell'apparecchiatura elettronica, esso tende a differire dal suo valor medio.

La stabilità del bias descrive, dunque, quanto il bias di un sensore può variare durante uno specifico intervallo di tempo, tipicamente di 100 secondi, in condizioni prefissate (in genere si assume che la temperatura sia costante). Essa viene inoltre specificata come il valore 1σ con unità di $^{\circ}/h_r$ (per le accelerazioni si esprime in μg).

Il *flicker noise* è un rumore con uno spettro $1/f$, e si può osservare nei componenti elettronici per frequenze relativamente basse; infatti per alte frequenze esso tende ad essere messo in ombra dal rumore bianco [21].

Esistono diverse modellizzazioni matematiche per rappresentare questa variazione del bias, tra cui le due più utilizzate sono:

a. Rate Random Walk Model

$$\frac{d}{dt}b(t) = \dot{b}(t) = \eta_b \quad (4.17)$$

b. First Order Gauss-Markov Model

$$\frac{d}{dt}b(t) = \dot{b}(t) = -\frac{1}{\tau}b(t) + \eta_b \quad (4.18)$$

Dove η_b è un rumore bianco con autocovarianza pari a:

$$E[\eta_b(t)\eta_b^T(\tau)] = \sum_b \delta(t-\tau) = \sigma_b^2 \mathbf{I} \delta(t-\tau)$$

Dove, σ_b^2 è la rate random walk variance ed il generico $\delta(t)$ è il delta di Dirac [22].

Rumore bianco (White Noise)

L'output di un sensore, generalmente, è perturbato da rumori di natura termo-meccanica che variano con una velocità molto più elevata dei campioni in uscita dal sensore; in altre parole, hanno una frequenza molto maggiore rispetto a quella del sensore.

Dunque, ciò che vedremo saranno dei campioni ottenuti dal sensore e perturbati da una sequenza di rumore bianco, la quale è modellizzata come una sequenza di variabili aleatorie (random) incorrelate fra loro e aventi valor medio nullo.

Con questa modellizzazione, si considera che ogni variabile aleatoria ha una distribuzione identica, che noi assumeremo essere la distribuzione normale (distribuzione gaussiana), ed una varianza finita σ^2 .

Nota: Distribuzione gaussiana e teorema del limite centrale

In particolare, la funzione di densità di probabilità di una variabile aleatoria Gaussiana è:

$$f_X(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4.19)$$

Dove σ è la deviazione standard e μ è il valor medio.

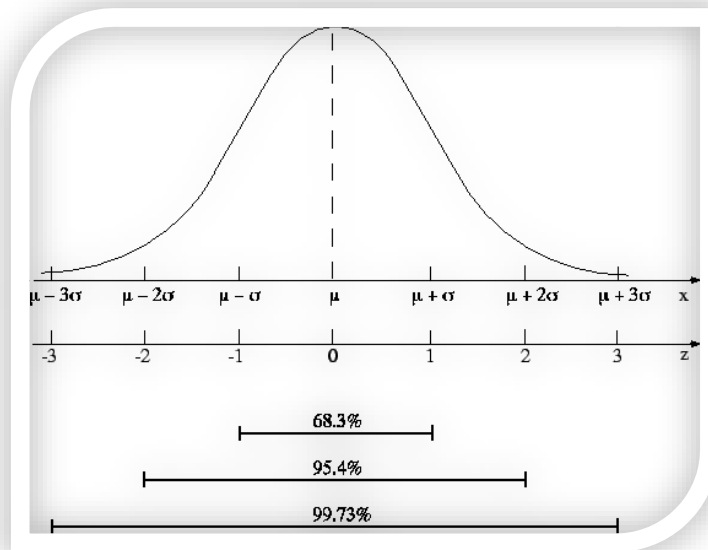


Figura 30: Distribuzione Gaussiana

Il motivo per cui generalmente viene utilizzata questo tipo di distribuzione in questi casi è dettato dall' esistenza del *teorema del limite centrale*, il quale afferma che:

Date n variabili aleatorie X_1, X_2, \dots, X_n indipendenti, con valori medi $\mu_1, \mu_2, \dots, \mu_n$ e varianze $\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2$, la distribuzione della variabile aleatoria

$$Y = \frac{1}{\sqrt{n}} \sum_{i=1}^n \frac{X_i - \mu_i}{\sigma_i}$$

sotto condizioni molto generali tende al crescere di n a quella di una gaussiana con valor medio 0 e varianza 1. [23]

Fine nota

Per vedere quali effetti questo rumore ha sul nostro segnale integrato, effettuiamo una semplice analisi in cui si assume che, per effettuare l'integrazione, si utilizza la regola dei rettangoli.

Denotiamo con N_i l' i -esima variabile aleatoria nella sequenza del rumore bianco. Assumiamo inoltre che le variabili N_i siano identicamente distribuite con valor medio, $E[N_i] = 0$, nullo e varianza $Var(N_i) = \sigma^2$.

Per ciò che abbiamo detto precedentemente $Cov(N_i, N_j) = 0 \forall i \neq j$.

Dunque, integrando mediante la regola dei rettangoli un rumore bianco $\eta(t)$, su un intervallo di tempo $t = n \cdot \delta t$:

$$\int_0^t \eta(\tau) d\tau = \delta t \sum_{i=1}^n N_i \quad (4.20)$$

Dove, n è il numero dei campioni ricevuti dal sensore durante il periodo t e δt è il tempo che intercorre tra due campioni consecutivi.

Utilizzando le semplici relazioni:

$$\begin{aligned} E[aX + bY] &= aE[X] + bE[Y] \\ Var(aX + bY) &= a^2Var(X) + b^2Var(Y) \end{aligned}$$

Dove, a e b sono costanti e X e Y sono variabili aleatorie

Dunque, la (4.20) si può riscrivere come:

$$E\left[\int_0^t \eta(\tau) d\tau\right] = \delta t \cdot n \cdot E\left[\sum_{i=1}^n N_i\right] = 0$$

$$\text{Var}\left(\int_0^t \eta(\tau) d\tau\right) = \delta t^2 \cdot n \cdot \text{Var}(N) = \delta t^2 \cdot n \cdot \sigma^2 = \delta t \cdot t \cdot \sigma^2$$

Il rumore introduce un errore del tipo *random walk*¹⁸ a media nulla all'interno del segnale integrato, la cui deviazione standard è:

$$\sigma_\theta(t) = \sigma \cdot \sqrt{\delta t \cdot t}$$

la quale cresce con la radice quadrata del tempo.

Comunemente, i costruttori specificano delle grandezze per indicare quantitativamente il rumore bianco di cui soffre il sensore, tra cui la misura dell'*angle random walk* (ARW), il quale è pari a:

$$ARW = \sigma_\theta(1)$$

Esso si calcola in $^\circ/\sqrt{h_r}$ (per le accelerazioni si esprime in $\mu g/\sqrt{Hz}$) ed è legato alla PSD del rumore bianco mediante la relazione [21]:

$$ARW \left[\frac{^\circ}{\sqrt{h_r}} \right] = \frac{1}{60} \sqrt{PSD \left[\left(\frac{^\circ}{h_r} \right)^2 \frac{1}{Hz} \right]} \quad (4.21)$$

E dunque, la PSD del nostro rumore bianco sarà allora pari a:

$$PSD \left[\left(\frac{^\circ}{s} \right)^2 \frac{1}{Hz} \right] = \frac{60^2}{3600^2} ARW \left[\frac{^\circ}{\sqrt{h_r}} \right] \quad (4.22)$$

Abbiamo così modellizzato anche l'ultimo errore che perturba le uscite dei nostri sensori.

Come anticipato nell'introduzione, ciò che stiamo andando a studiare sono le prestazioni di INS di livello *navigation* e *tactical grade*; di conseguenza, supporremo che tutti gli errori sistematici, quali lo *factor scale*, il *misalignment* e il *constant bias*, siano stati già compensati in maniera corretta

¹⁸ Un *random walk* è definito come un processo che consiste di una serie di step, in cui la direzione e l'ampiezza degli steps è determinata in maniera casuale.

mediante tarature e prove sperimentali, mentre l'Angular Random Walk e la Bias Instability perturberanno le uscite dei nostri sensori in maniera non trascurabile.

In altre parole, senza ledere di generalità, all'interno del nostro simulatore implementeremo solo gli errori di natura stocastica e non deterministica.



5. MATLAB

5.1. Introduzione al potenziale di MATLAB

Come anticipato nell'introduzione, per scrivere e compilare il simulatore, abbiamo utilizzato un linguaggio di programmazione che è risultato molto utile ed appropriato: MATLAB.

“Esso fornisce strumenti per acquisire, analizzare e visualizzare i dati, consentendo una comprensione approfondita dei dati in una frazione del tempo che servirebbe usando fogli di calcolo o linguaggi di programmazione tradizionali. È inoltre possibile documentare e condividere i risultati attraverso grafici e report, oppure sotto forma di codice MATLAB pubblicato.

Il linguaggio MATLAB offre un supporto nativo per le operazioni vettoriali e matriciali fondamentali per risolvere problemi ingegneristici e scientifici velocizzando lo sviluppo e l'esecuzione.

Grazie al linguaggio MATLAB, la scrittura di programmi e lo sviluppo di algoritmi sono più veloci rispetto a quanto accade con i linguaggi tradizionali, poiché non serve eseguire attività amministrative di basso livello quali dichiarare le variabili, specificare i tipi di dato e allocare la memoria. In molti casi, il supporto per le operazioni vettoriali e matriciali elimina la necessità di cicli *for*. Di conseguenza, una riga di codice MATLAB può spesso sostituire varie righe di codice C o C++.

Esso mette a disposizione le funzioni dei tradizionali linguaggi di programmazione, inclusi il controllo di flusso, la gestione degli errori e la programmazione a oggetti (OOP). Si possono usare tipi di dato di base o strutture di dati avanzate, oppure definire tipi di dato personalizzati.

Si possono produrre risultati immediati eseguendo interattivamente i comandi uno alla volta. Questo approccio consente di esplorare rapidamente opzioni multiple ed eseguire iterazioni fino a raggiungere una soluzione ottimale. È possibile registrare le fasi interattive sotto forma di script e funzioni, che possono essere riutilizzate per automatizzare il lavoro”. [24]

Purtroppo, per brevità di descrizione, non descriveremo appieno il programma e daremo per scontato che il lettore abbia già delle basi discrete al fine della comprensione degli script formulati, i quali verranno di seguito mostrati.

In particolare, ai fini della scrittura del simulatore, ci servirà un tool di MATLAB denominato Simulink che racconteremo brevemente nel prossimo paragrafo.

5.2. Simulink

“Simulink è un ambiente grafico per la simulazione multidominio e il Model-Based Design. Supporta la progettazione a livello di sistema, la simulazione, la generazione automatica del codice, il testing e la verifica di sistemi embedded.

Simulink offre un editor grafico, librerie di blocchi personalizzabili e solutori per la modellazione e la simulazione di sistemi dinamici. È integrato con MATLAB, consente di incorporare gli algoritmi MATLAB nei modelli e di esportare i risultati delle simulazioni in MATLAB per ulteriori analisi”. [25]

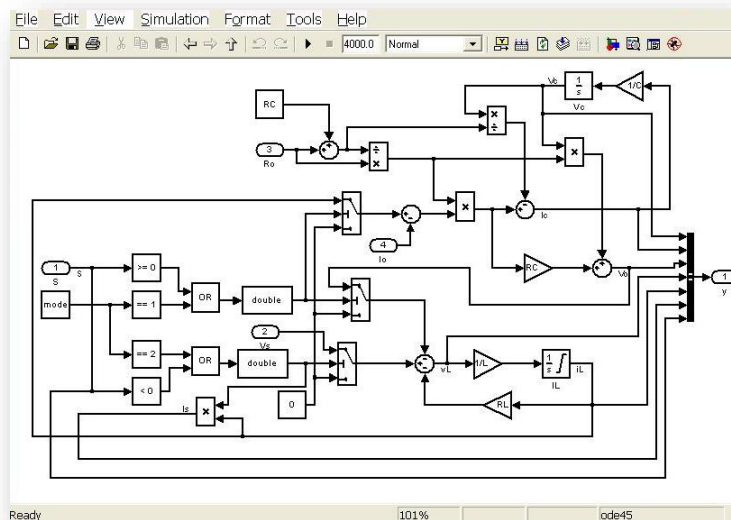


Figura 31: Ambiente di programmazione Simulink

È sembrato più che giusto fare una breve introduzione a questo linguaggio di programmazione in quanto si è rivelato molto utile ai fini della nostra trattazione, ma per brevità di trattazione, supporremo che il lettore conosca già questo ambiente di programmazione e, di conseguenza, passeremo subito alla descrizione del nostro simulatore, senza dilungarci troppo sulla trattazione delle librerie di Simulink.

Portando giusto un esempio per introdurre questo linguaggio di programmazione, ovvero, volendo implementare la (4.3) all'interno di Simulink, allora avremo un algoritmo simile a quello in Figura 32.

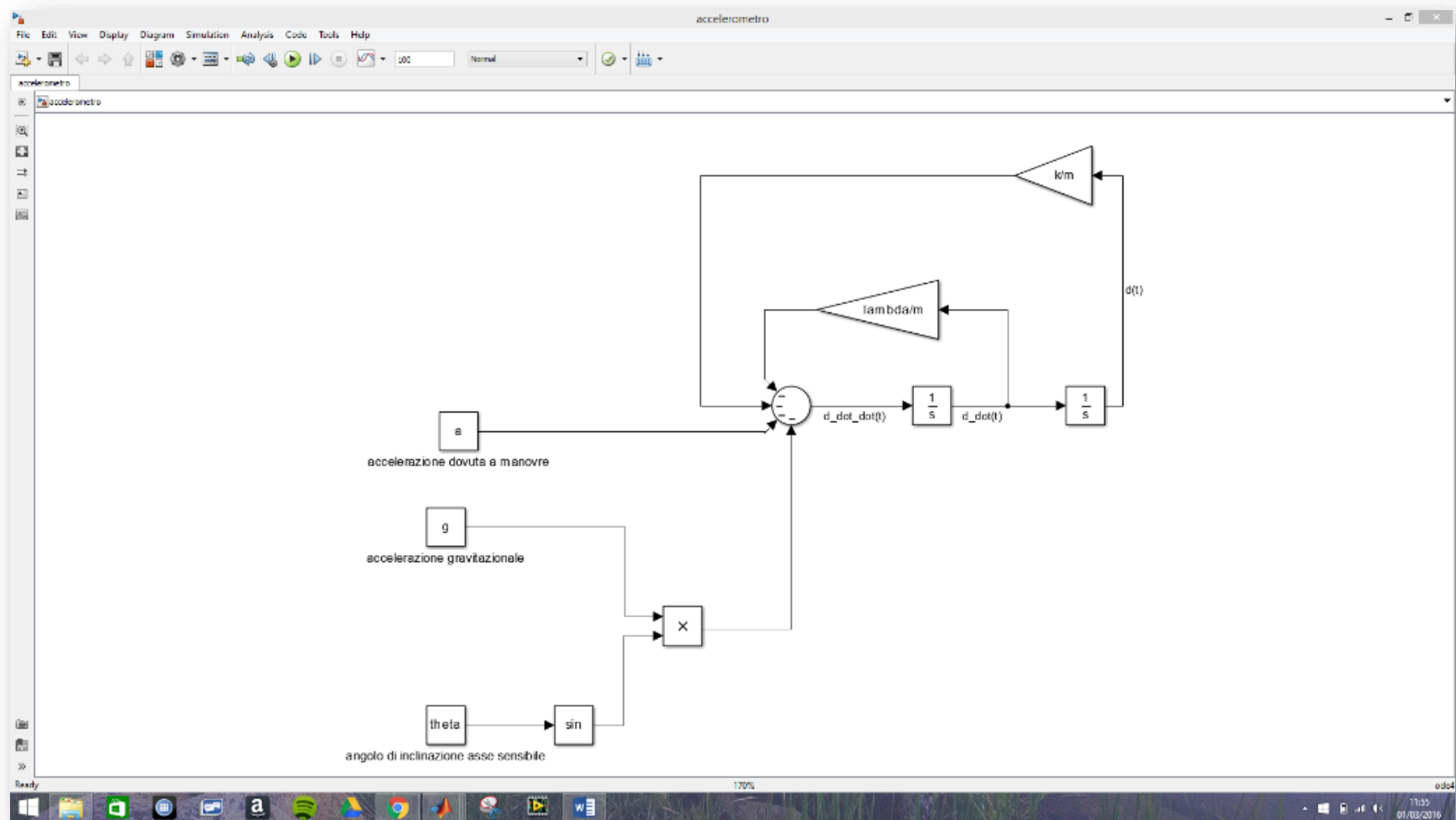


Figura 32: Generica modellizzazione su Simulink di un accelerometro MEMS

6. Simulatore

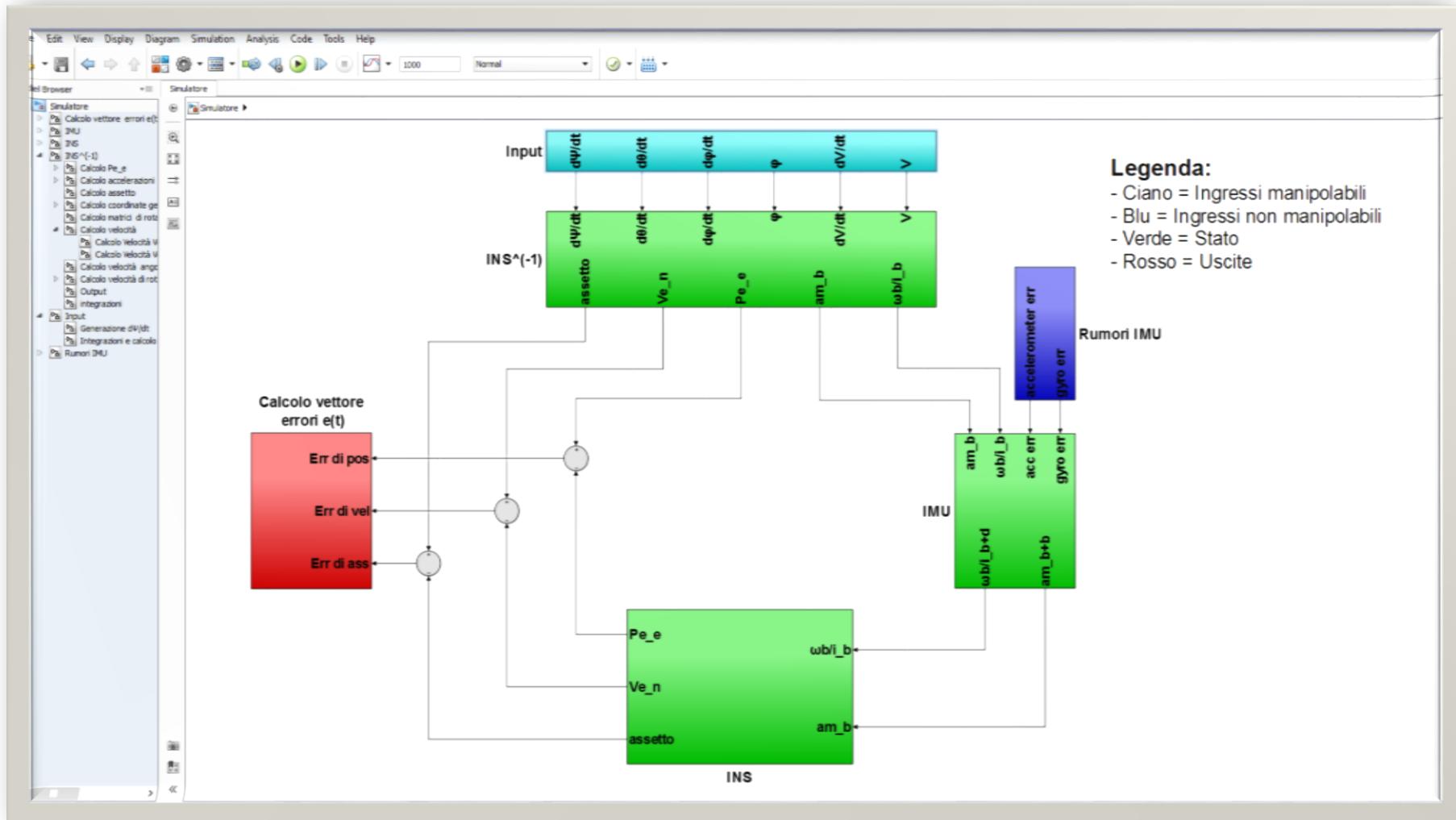


Figura 33: Layout Simulatore

6.1. Input

Definizione della traiettoria

Per poter ottenere dei dati qualitativamente e quantitativamente realistici, riguardo gli errori presenti all'interno dei sensori, è stata implementata all'interno del simulatore una traiettoria realistica effettuata al di sopra dell'aeroporto di Crotone.

In particolare, la traiettoria che è stata seguita è una procedura d'attesa denominata *holding pattern* e comunemente chiamata anche con il nome di traiettoria a biscotto (per via della sua forma): essa è costituita da due tratti rettilinei raccordati da due virate di 180° ciascuna. [26]

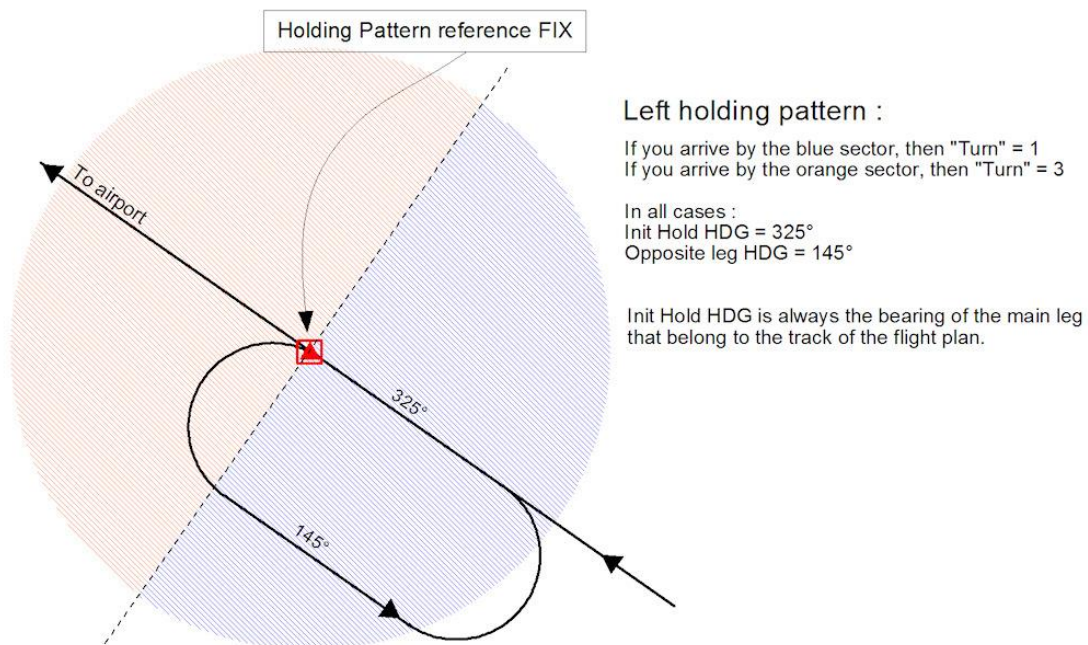


Figura 34: Holding Pattern

I dati della nostra traiettoria sono:

- λ_0 → Longitudine iniziale = 0°
- Φ_0 → Latitudine iniziale = 0°

- h_0 → Quota iniziale¹⁹ = 1000 m
- L_1 → Lunghezza tratto rettilineo = 6000 m
- R → Raggio di curvatura tratto in virata = 2063 m
- t_1 → Tempo trascorso prima dell'immissione all'interno della traiettoria pari a 100 s
- V_0 → Modulo velocità iniziale²⁰ 108 m/s
- ψ_0 → Angolo di prua iniziale = 180°
- θ_0 → Angolo di elevazione iniziale = 0°
- φ_0 → Angolo di rollio iniziale = 0°

Inoltre, il simulatore è stato progettato per far eseguire al nostro velivolo virtuale un numero n di giri (di biscotti), come si potrà evincere dall'algoritmo che verrà mostrato di seguito, ma da ora supporremo che il nostro aeromobile compia sempre tre giri, in modo tale che, durante il confronto tra i vari sensori, non ci siano elementi che possano differire e creare delle distorsioni che porterebbero inevitabilmente a giudizi falsati.

Combinando i dati che abbiamo a disposizione, si possono ottenere le seguenti:

- $t_2 = L_1/V_0$ Tempo trascorso per compiere il tratto rettilineo [s]
- $L_2 = R * \pi$ Lunghezza tratto curvilineo \approx 6480 m
- $t_3 = L_2/V_0$ Tempo trascorso per compiere il tratto curvilineo [s]
- $T = 2 * t_2 + 2 * t_3$ Tempo trascorso per compiere un intero biscotto [s]
- $\Omega = \pi/t_3$ Rateo di virata [rad/s]

All'interno del nostro modello supporremo di effettuare delle virate coordinate, le quali unite alle informazioni di quota costante e velocità costante, eguagliano la variazione dell'angolo di prua ($\dot{\psi}$) al rateo di virata (Ω), durante le virate.

Grazie alle informazioni sui tempi, diviene più semplice implementare la variazione dell'angolo di prua, la quale si può implementare sul nostro simulatore mediante un semplice algoritmo su Matlab:

```
%% Calcolo psi_dot
% rateo = rateo di virata
% t_1 = tempo di inizio manovra (del primo biscotto)
```

¹⁹ Quota costante durante tutta la manovra.

²⁰ Senza ledere di generalità, si può assumere che esso resti costante durante tutto il volo.

```

% t_2 = tempo che il velivolo impiega per completare un tratto rettilineo
% t_3 = tempo che il velivolo impiega per compiere un tratto curvilineo
% bisc = numero di "biscotti" da far eseguire al velivolo
% T = periodo di un biscotto 2*t_2+2*t_3
% t = tempo di simulazione corrente (in Simulink)

t_star=t-t_1;

a=fix(t_star/T); % numero di volte in cui è stato eseguito
                % il biscotto completo (in tempo reale)

if a>=bisc || bisc==0
    psi_dot=0;
else
    if t_star<0
        psi_dot=0;
    elseif t_star>=T*a && t_star<t_2+T*a
        psi_dot=0;
    elseif t_star>=t_2+T*a && t_star<t_2+t_3+T*a
        psi_dot=rateo;
    elseif t_star>=t_2+t_3+T*a && t_star<2*t_2+t_3+T*a
        psi_dot=0;
    elseif t_star>=2*t_2+t_3+T*a && t_star<2*t_2+2*t_3+T*a
        psi_dot=rateo;
    else
        psi_dot=0;
    end
end
end

```

Il quale entrerà all'interno della funzione implementata nel blocco Simulink, come mostrato in Figura 35:

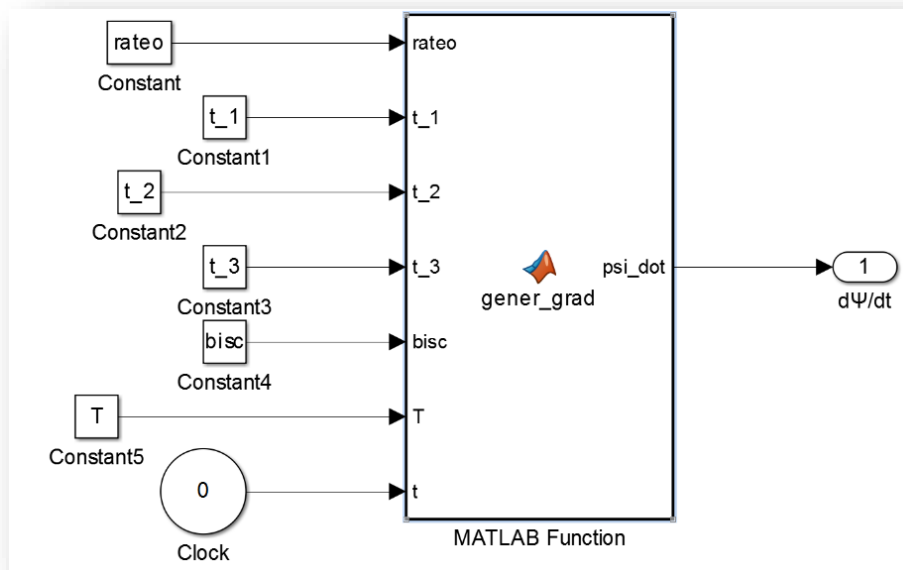


Figura 35: Calcolo variazione angolo di prua

Andiamo ora a vedere come possiamo ricavare le informazioni riguardo le componenti di velocità rispetto agli assi NED che dovremo implementare all'interno del nostro simulatore.

Modellazione matematica del velivolo come punto materiale 3D

Senza ledere di generalità, andiamo a trovare il legame che sussiste tra la velocità del velivolo, esprimendola in assi NED per implementarla con maggiore facilità all'interno dell'equazione di navigazione standard, e l'assetto del velivolo stesso assumendo che il velivolo sia assimilabile ad un punto materiale.

Inoltre, elimineremo eventuali fattori che tengono in considerazione la forma terrestre, ovvero assumeremo la terra piatta, e la sua rotazione, Terra ferma, e fattori quali l'angolo d'attacco del velivolo.

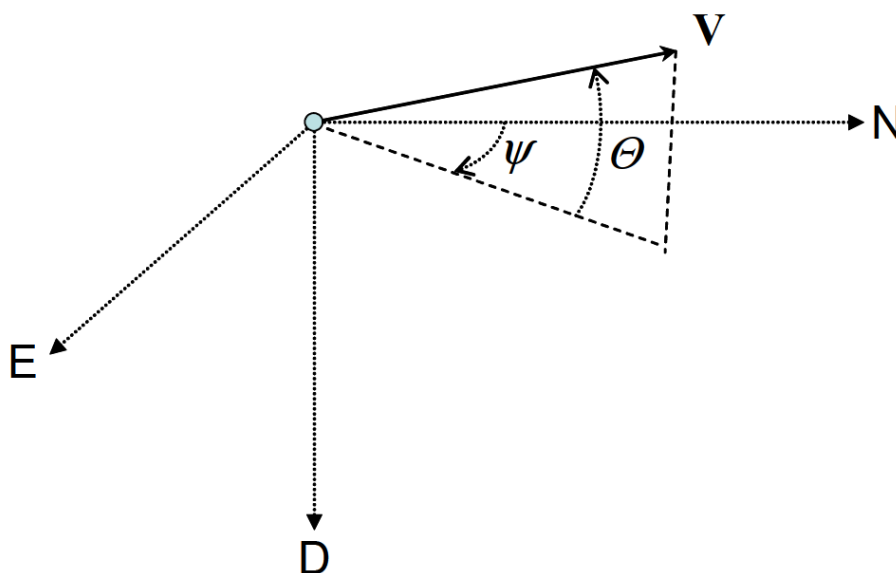


Figura 36: Rappresentazione velocità in assi NED

In particolare, come si evince dalla Figura 36, la velocità del velivolo sarà messa in funzione degli angoli di prua (ψ) e di elevazione (θ), per via del fatto che il nostro corpo è un punto materiale 3D, e successivamente, dovremo, in funzione della traiettoria che genereremo, valutare anche l'angolo di inclinazione (φ).

Allora, esprimere la velocità, che supporremo definita in assi ECEF, in coordinate NED, significa trovare il seguente vettore:

$$\mathbf{V}_e^n = (V_N, V_E, V_D)$$

Le quali componenti, come si può evincere geometricamente dalla Figura 15, saranno uguali a:

$$\begin{cases} V_N = V \cos \theta \cos \psi \\ V_D = V \cos \theta \sin \psi \\ V_E = -V \sin \theta \end{cases} \quad (6.1)$$

E, le rispettive derivate ordinarie, dunque rispetto al tempo, saranno:

$$\begin{cases} \dot{V}_N = \dot{V} \cos \theta \cos \psi - \dot{\theta} V \sin \theta \cos \psi - \dot{\psi} V \cos \theta \sin \psi \\ \dot{V}_E = \dot{V} \cos \theta \sin \psi - \dot{\theta} V \sin \theta \sin \psi + \dot{\psi} V \cos \theta \cos \psi \\ \dot{V}_D = -\dot{V} \sin \theta - \dot{\theta} V \cos \theta \end{cases} \quad (6.2)$$

Queste semplici relazioni rappresentano *vincoli* a cui sarà sottoposto il nostro simulatore, il quale, in funzione della traiettoria che farà seguire al velivolo, dovrà generare i dati mancanti. [27]

Implementazione angolo di banco

Dunque, abbiamo già visto come simulare l'angolo di prua istante per istante, abbiamo supposto che la traiettoria sarà a quota costante e dunque che l'angolo di elevazione debba restare sempre nullo, andiamo ora a vedere come possiamo valutare istante per istante l'angolo di rollio così da simulare l'assetto del velivolo ad ogni istante.

All'interno del simulatore, l'angolo φ è stato generato mediante un modello dinamico del secondo ordine del tipo:

$$\ddot{\varphi} + 2\delta\omega_n\dot{\varphi} + \omega_n^2\varphi = \omega_n^2 \tan^{-1}\left(\dot{\psi} \frac{V}{g}\right) \quad (6.3)$$

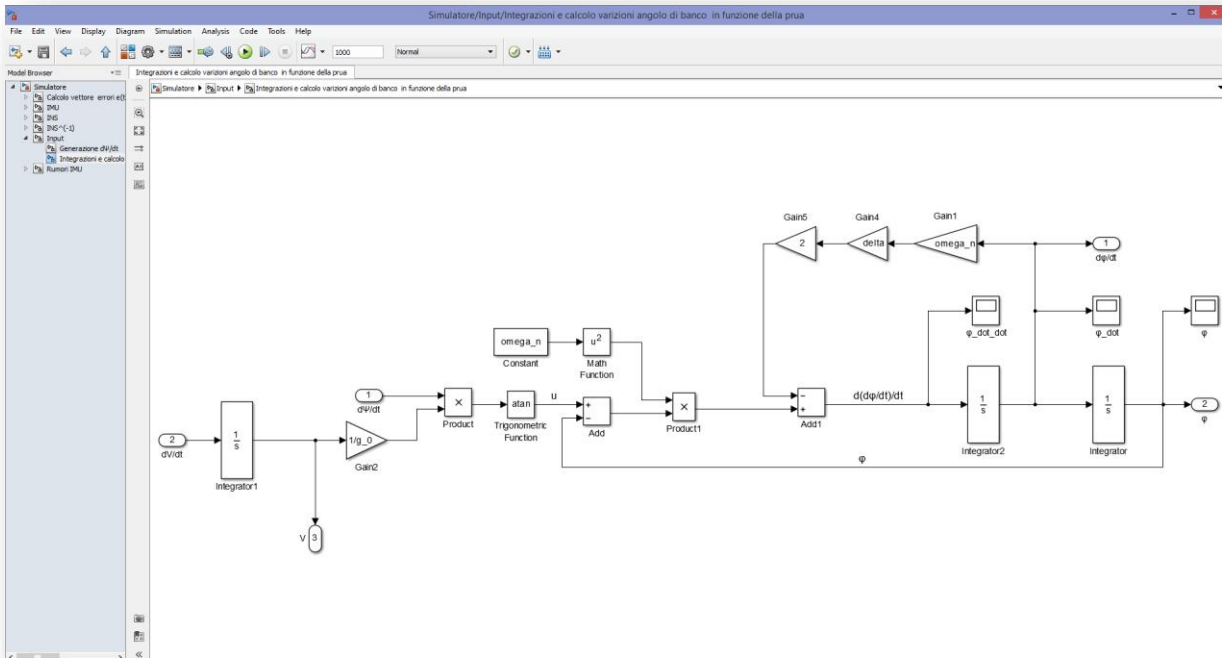


Figura 37: Implementazione angolo di banco su Simulink

Con questa modellizzazione, durante le virate si registrano i seguenti andamenti:

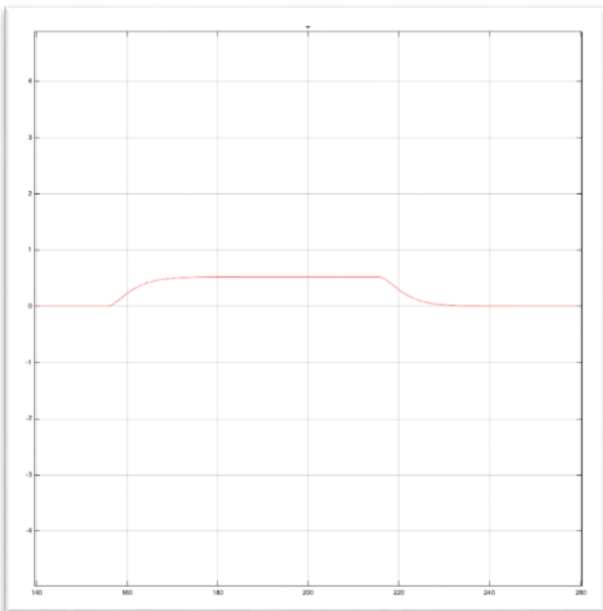


Figura 39: Andamento della φ

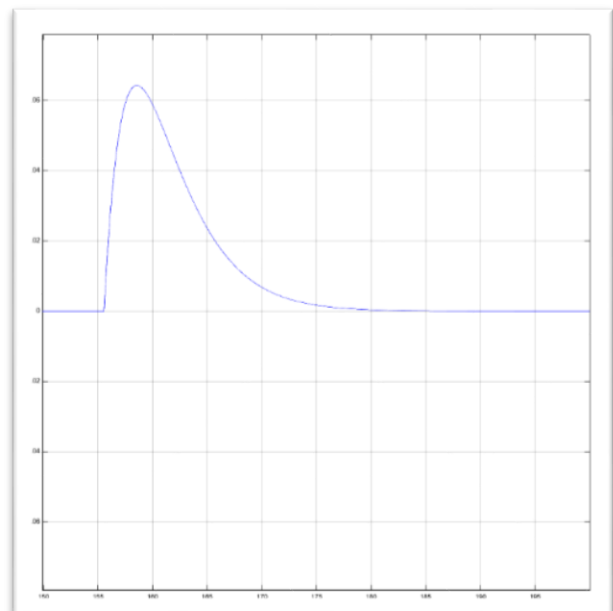


Figura 38: Andamento della $\dot{\varphi}$

Con questa descrizione chiudiamo la prima parte del nostro simulatore e andiamo ora a studiare la seconda, relativa all'inversione delle equazioni di navigazione standard per ottenere le uscite, prive di errori, dei sensori in funzione della traiettoria generata.

6.2. INS^{-1}

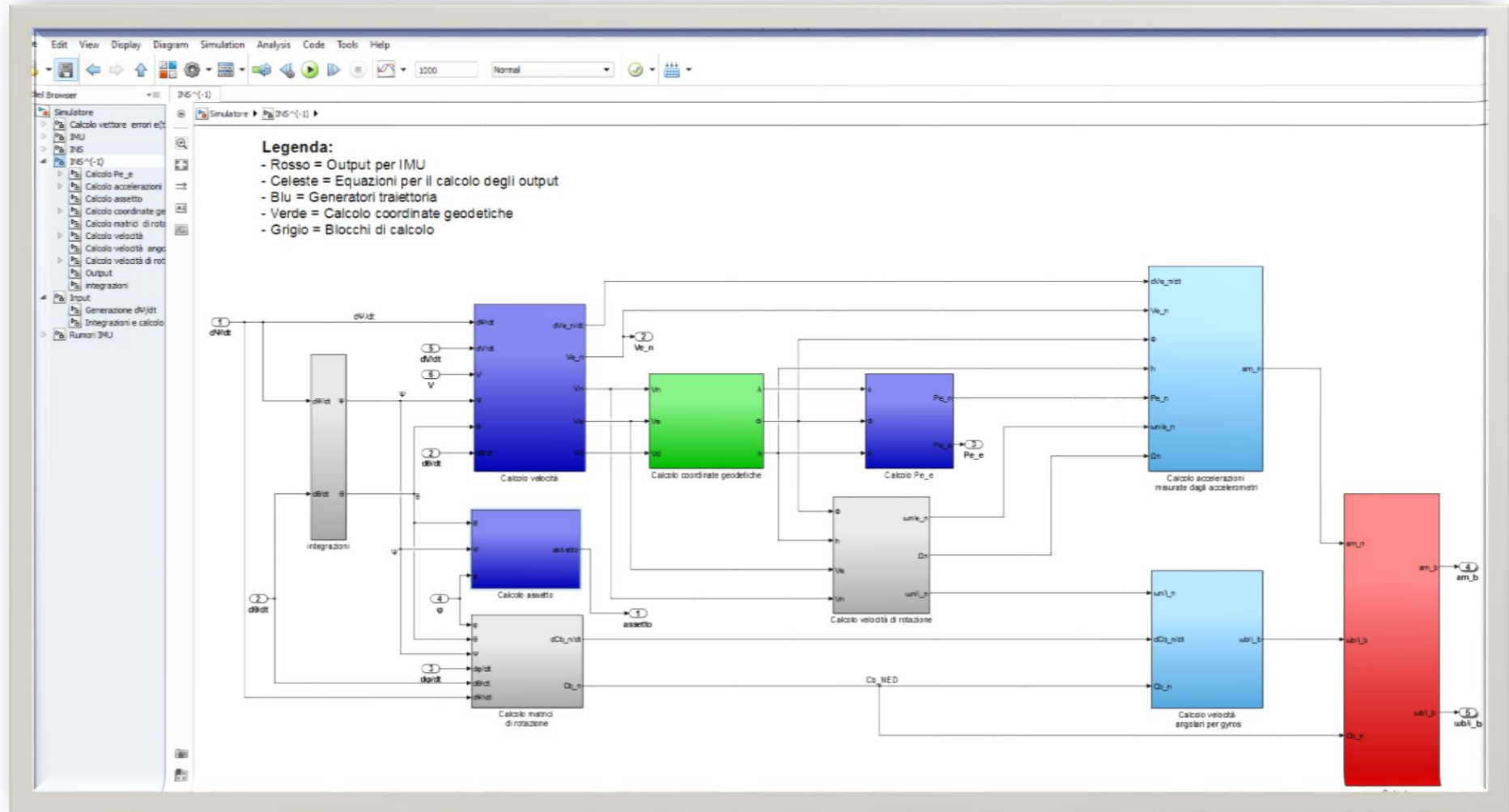


Figura 40: Blocco INS^{-1}

Calcolo output accelerometri

Come abbiamo già visto in precedenza, nell'equazione (3.12), abbiamo che, a secondo membro, figurano le accelerazione fornite dagli accelerometri in assi NED.

In particolare, mettendo tutto in funzione dell'uscita degli accelerometri, abbiamo che la (3.12) diviene:

$$\mathbf{a}_m^n = \dot{\mathbf{V}}_e^n - \mathbf{g}^n + \boldsymbol{\Omega}^n \times (\boldsymbol{\Omega}^n \times \mathbf{P}^n) + (2\boldsymbol{\Omega}^n + \boldsymbol{\omega}_{n/e}^n) \times \mathbf{V}_e^n \quad (6.4)$$

Dove, dalla conoscenza degli angoli di assetto e del modulo della velocità istante per istante, per le (6.1) ricaviamo il vettore velocità in assi NED e mediante le (6.2) ricaviamo anche la derivata ordinaria delle stesse.

Inoltre, i vettori $\boldsymbol{\omega}_{n/e}^n$ e $\boldsymbol{\Omega}^n$ li conosciamo dalle (3.9), il vettore \mathbf{P}^e lo abbiamo già ricavato nell'equazione (2.4), dunque basterà pre-moltiplicarlo per la matrice di rotazione da ECEF a NED riportata nella (2.13) per ottenere \mathbf{P}^n in funzione delle coordinate geodetiche, le quali sono funzioni delle velocità traslazionali e possiamo ricavarle dalle equazioni (3.11). Considerando infine che:

$$\mathbf{g}^n = \begin{pmatrix} g \sin(\Phi - \Xi) \\ 0 \\ g \cos(\Phi - \Xi) \end{pmatrix}$$

dove Φ e Ξ sono rispettivamente la latitudine geodetica e quella geocentrica; allora possiamo dire che conosciamo le accelerazioni \mathbf{a}_m^n , ovvero le accelerazioni che subisce il nostro corpo rispetto al sistema inerziale, definite in assi body ma con coordinate in assi NED.

Ciò che dobbiamo fare ora sarà trasformare le coordinate del vettore delle accelerazioni appena ricavato in coordinate body, così da avere esattamente gli output dei sensori.

Calcolo matrici di rotazione

Dunque, sappiamo che:

$$\mathbf{a}_m^b = \mathbf{C}_n^b \mathbf{a}_m^n$$

Abbiamo già visto in precedenza che, dalla conoscenza dell'assetto, si può ricavare la matrice di rotazione da NED a body mediante la (2.11), ma per calcolare anche le uscite dei giroscopi, abbiamo bisogno anche della conoscenza della derivata temporale di questa matrice, infatti, esprimendo la (3.16) in funzione di $\boldsymbol{\omega}_{b/i}^b$:

$$\boldsymbol{\omega}_{b/i}^b = \mathbf{C}_n^b \dot{\mathbf{C}}_b^n + \boldsymbol{\omega}_{n/i}^b \quad (6.5)$$

Dove $\boldsymbol{\omega}_{n/i}^b$ lo si calcola con la (3.18), di cui conosciamo tutti i termini; mentre la derivata della matrice di rotazione, che, come abbiamo già detto in precedenza, non può essere ricavata utilizzando le librerie di Simulink, se si vogliono evitare errori di calcolo numerico dati dalle derivate, deve essere ricavata esternamente al simulatore e successivamente implementata.

Allora, essa sarà uguale a²¹:

$$\frac{d}{dt} \mathbf{C}_b^n(\psi, \theta, \varphi) = \dot{\mathbf{C}}_b^n = \begin{pmatrix} -\dot{\psi}c\theta s\psi - \dot{\theta}s\theta c\psi & -\dot{\psi}(s\theta s\varphi s\psi + c\varphi c\psi) + \dot{\theta}c\theta s\varphi c\psi + \dot{\varphi}(s\theta c\varphi c\psi + s\varphi s\psi) & \dot{\psi}(s\varphi c\psi - s\theta c\varphi s\psi) + \dot{\theta}c\theta c\varphi c\psi + \dot{\varphi}(c\varphi s\psi - s\theta s\varphi c\psi) \\ \dot{\psi}c\theta c\psi - \dot{\theta}s\theta s\psi & \dot{\psi}(s\theta s\varphi c\psi - c\varphi s\psi) + \dot{\theta}c\theta s\varphi s\psi + \dot{\varphi}(s\theta c\varphi s\psi - s\varphi c\psi) & \dot{\psi}(s\theta c\varphi c\psi + s\varphi s\psi) + \dot{\theta}c\theta c\varphi s\psi + \dot{\varphi}(s\theta s\varphi s\psi - c\varphi c\psi) \\ -\dot{\theta}c\theta & \dot{\varphi}c\varphi c\theta - \dot{\theta}s\varphi s\theta & -\dot{\varphi}s\varphi c\theta - \dot{\theta}c\varphi s\theta \end{pmatrix} \quad (6.6)$$

Dunque, gli output del nostro blocco saranno:

- \mathbf{a}_m^b Uscite accelerometri
- $\boldsymbol{\omega}_{b/i}^b$ Uscite giroscopi

²¹ Per brevità di notazione, si sono sostituiti gli operatori *seno* e *coseno*, rispettivamente, con *s* e *c*.

- $$\begin{cases} \mathbf{V}_e^n \\ \mathbf{P}_e^e \\ (\psi, \theta, \varphi) \end{cases}$$
 Parametri traiettoria

6.3. IMU e rumori dei sensori

Nel paragrafo precedente abbiamo calcolato le uscite che idealmente i nostri sensori dovrebbero fornire se fossero privi di errori.

In questo nuovo paragrafo dobbiamo dunque implementare ciò che abbiamo spiegato nella sezione dedicata agli errori dei sensori così che il nostro simulatore divenga più realistico.

Come detto in precedenza, non implementeremo tutte le sorgenti di errori, assumendo dunque che gli errori deterministici siano nulli, ma implementeremo solo ed esclusivamente gli errori stocastici che per noi saranno il *bias instability* e l'*angular random walk*, in maniera additiva.

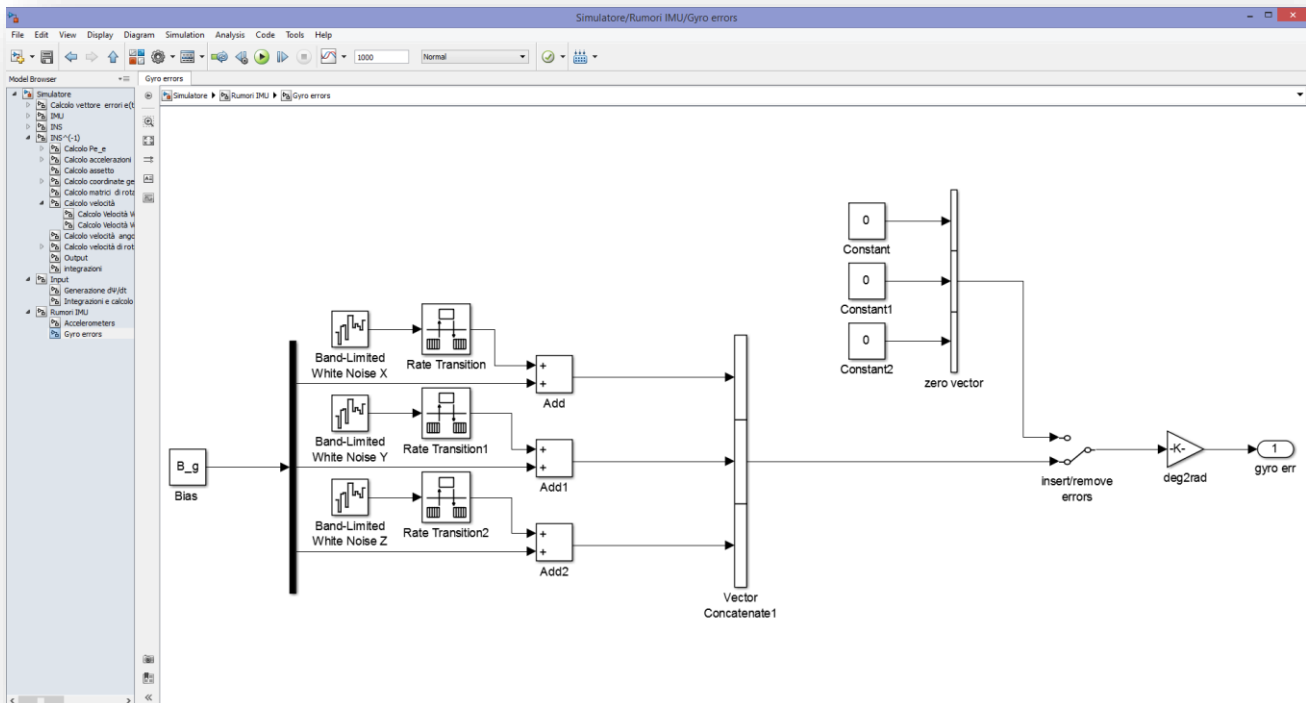


Figura 41: Implementazione errori sui giroscopi

Abbiamo dunque imposto che, ad esempio, per i giroscopi deve valere che:

$$\omega_m(t) = \omega(t) + \mathbf{b} + \eta(t)$$

Dove:

- $\omega_m(t)$ *Velocità angolare misurata*
- $\omega(t)$ *Velocità angolare vera*
- \mathbf{b} *Bias*
- η *Rumore bianco*

In particolare, abbiamo considerato il bias come una variabile aleatoria, che resta costante durante tutta la simulazione²², ma varia per ogni simulazione; abbiamo inoltre assunto che esso ha deviazione standard pari al valore della bias instability, valor medio nullo e distribuzione normale.

Per quanto riguarda il rumore bianco, anch'esso è pensato come una variabile aleatoria con distribuzione gaussiana; il suo valor medio è nullo e ha una varianza che è pari al prodotto tra la PSD, ricavata dalla conoscenza dell'ARW, come spiegato nell'equazione (4.22), e la frequenza del rumore bianco stesso, che in mancanza di specificazioni del costruttore, si assume pari a quella del sensore.

Lo stesso ragionamento è stato applicato alla terna degli accelerometri.

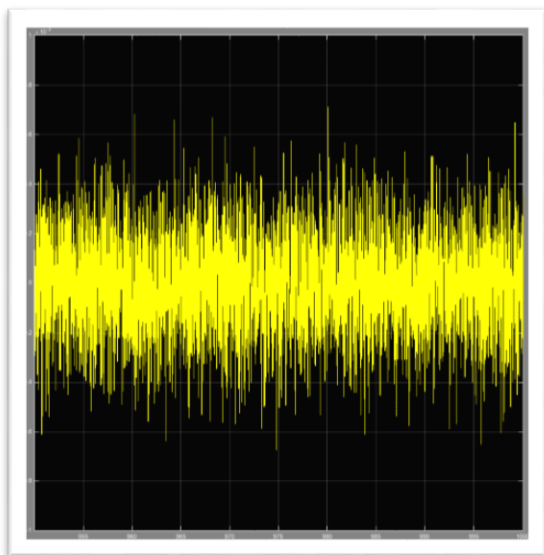


Figura 42: *Andamento tipico di un White Noise*

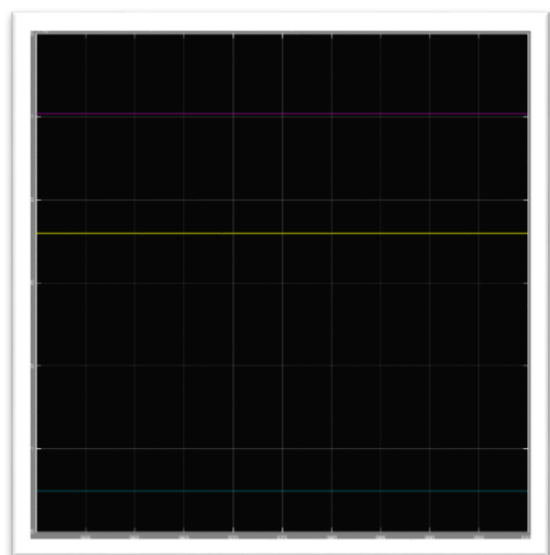


Figura 43: *Andamento Bias relativo a tre giroscopi*

²² Ciò è lecito se si considera che effettivamente la simulazione non dura abbastanza per far sì che esso vari.

6.4. INS

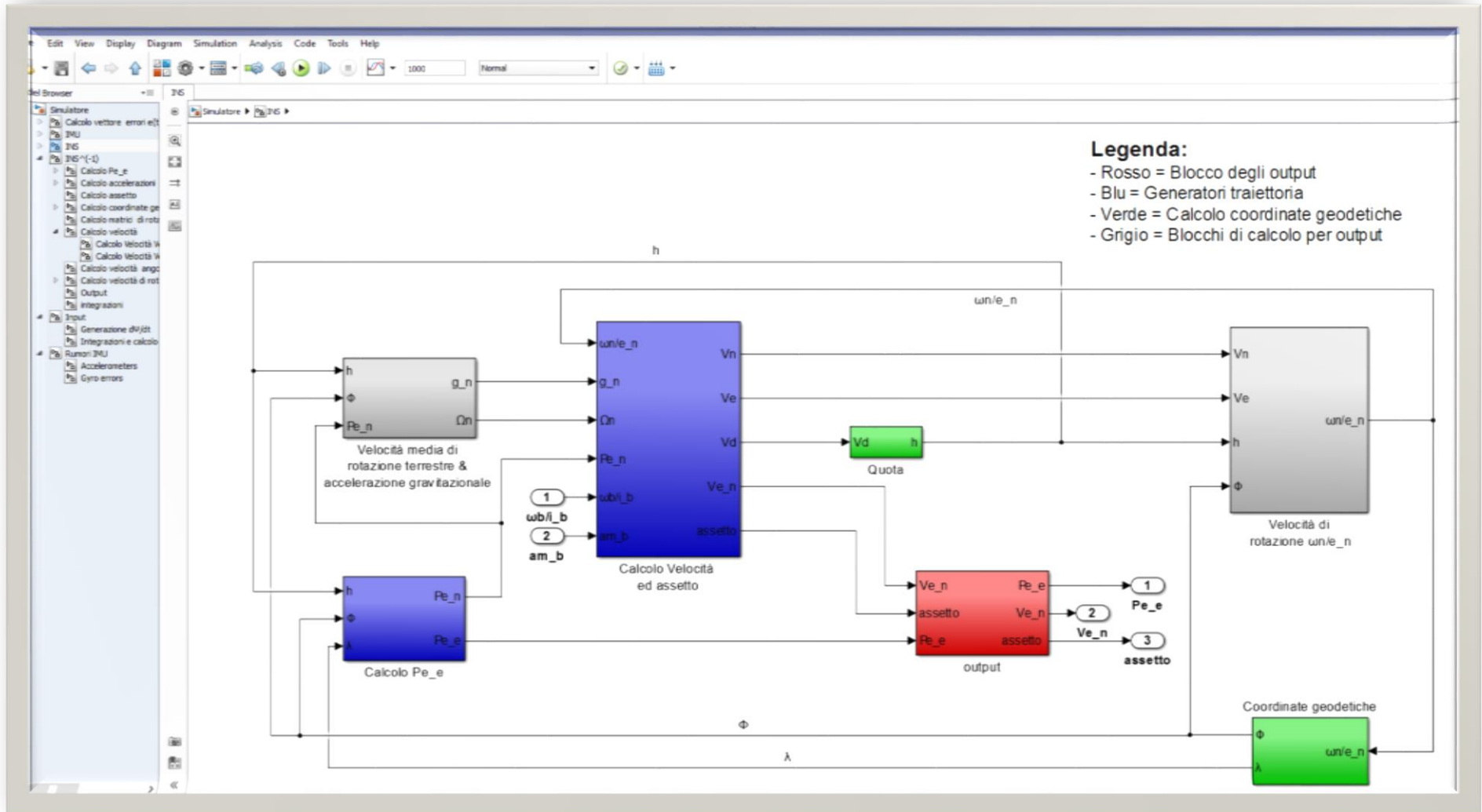


Figura 44: Blocco INS

Dunque, dobbiamo ora ricostruire, a partire dalla conoscenza delle informazioni provenienti dai sensori e dalla conoscenza delle condizioni iniziali, la traiettoria.

Come abbiamo ripetuto più volte, la legge cinematica che descrive la nostra traiettoria, ricavando la variazione della velocità, è l'equazione vettoriale di navigazione standard.

Per comodità la riporteremo qui sotto:

$$\dot{\mathbf{V}}_e^n = \mathbf{a}_m^n + \mathbf{g}^n - \boldsymbol{\Omega}^n \times (\boldsymbol{\Omega}^n \times \mathbf{P}^n) - (2\boldsymbol{\Omega}^n + \boldsymbol{\omega}_{n/e}^n) \times \mathbf{V}_e^n$$

Andiamo ora studiare come ricavare i termini mancanti.

Accelerazioni ed assetto

Abbiamo che le accelerazioni fornite dagli accelerometri sono in assi corpo e dunque dovremo trasformarle in assi NED prima di implementarle all'interno dell'equazione di navigazione; ciò è possibile pre-moltiplicando il vettore \mathbf{a}_m^n per la matrice di rotazione \mathbf{C}_b^n .

La conoscenza di quest'ultima impone la conoscenza dell'assetto (di cui conosciamo le condizioni iniziali).

L'algoritmo utilizzato su Simulink per risalire a queste quantità è mostrato in Figura 45.

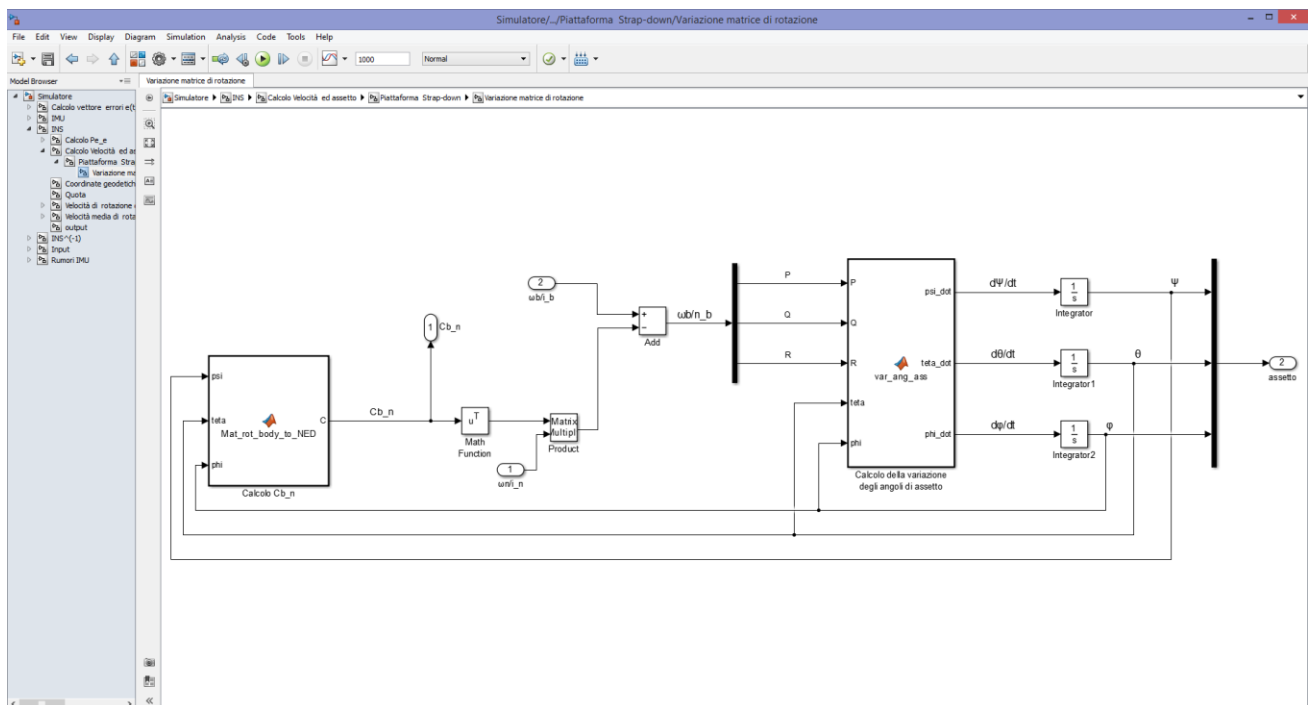


Figura 45: Calcolo assetto INS

Le equazioni applicate all'interno dell'algoritmo sono:

$$\boldsymbol{\omega}_{b/n}^b = \boldsymbol{\omega}_{b/i}^b - (\mathbf{C}_b^n)^T \boldsymbol{\omega}_{n/i}^n = \begin{pmatrix} P \\ Q \\ R \end{pmatrix}$$

Applicando poi le equazioni di propagazione degli angoli d'assetto, ovvero le (2.19), mettendo in retroazione, come mostrato nella Figura 45, la misura degli angoli di elevazione e rollio, dopo che sono stati integrati rispetto al tempo, si ottengono le stesse misure degli angoli di assetto.

Mediante questi angoli, si può successivamente costruire la matrice di rotazione da body a NED e di conseguenza avere una conoscenza di quest'ultima in tutti gli istanti di tempo.

È da notare che per eseguire queste operazioni è possibile utilizzare diverse strade; abbiamo utilizzato quest'ultima perché è quella che ci ha consentito di ottenere le nostre incognite mediante solo tre integrazioni e dunque ci ha portato ad un'accuratezza di calcolo maggiore.

Infine, una volta a conoscenza dell'assetto e dunque della matrice di rotazione, si possono esprimere le accelerazioni in assi NED.

Vettore posizione e coordinate geodetiche

Ciò che ci interessa ricavare ora è il vettore \mathbf{P}_e^n in funzione delle coordinate geodetiche.

Ricordiamo che esso è uguale a:

$$\mathbf{P}_e^n = \mathbf{C}_e^n \mathbf{P}_e^e$$

Il vettore posizione definito in ECEF e calcolato in coordinate ECEF, come abbiamo visto più volte, è ricavato tramite l'equazione (2.4), la quale si basa sulla conoscenza delle coordinate geodetiche.

Come abbiamo precedentemente mostrato, tramite le equazioni (3.11), la propagazione delle coordinate geodetiche, ovvero la variazione nel tempo della longitudine e della latitudine, escludendo per ora la quota, è esprimibile mediante il vettore $\boldsymbol{\omega}_{n/e}^n$. In particolare, sappiamo che:

$$\begin{cases} \dot{\lambda} = \frac{\omega_N^n}{\cos \Phi} \\ \dot{\Phi} = -\omega_E^n \end{cases}$$

Dunque, se conosciamo le loro condizioni iniziali, possiamo integrarle rispetto al tempo e conoscere il loro andamento durante la simulazione.

Per quanto riguarda la quota, avremo che essa è facilmente ricavabile; infatti, essa può essere espressa in funzione della velocità come:

$$\dot{h} = -V_D$$

L'ultimo termine che ci resta da conoscere per chiudere il nostro quadro è $\omega_{n/e}^n$; ma, ricordando la (3.9), essa può essere ricavata in funzione delle stesse coordinate geodetiche e delle componenti della velocità espresse in assi NED.

Dunque, considerando che Ω^n e \mathbf{g}^n li conosciamo dato che li abbiamo ricavati rispettivamente nella (3.9) e nella (3.13), allora abbiamo ora tutti i termini che ci servono per risolvere istante per istante l'equazione vettoriale ordinaria, non lineare, di navigazione.

6.5. Output simulatore

La parte finale del nostro simulatore compara la traiettoria vera, ovvero quella priva degli errori stocastici dei sensori, con la traiettoria perturbata da quest'ultimi.

Per facilitare la lettura dei grafici, si è pensato di plottare i valori assoluti degli errori di interesse ed in particolare, sono presenti errori del tipo:

- $\mathbf{e}_p(t) = |\mathbf{P}_{e\ vera}^e(t) - \mathbf{P}_{e\ calcolata}^e(t)|$ *Errore di posizione*
- $\mathbf{e}_V(t) = |\mathbf{V}_{e\ vera}^n(t) - \mathbf{V}_{e\ calcolata}^n(t)|$ *Errore di velocità*
- $\mathbf{e}_\psi(t) = |\psi_{vera}(t) - \psi_{calcolata}(t)|$ *Errore angolo di prua*
- $\mathbf{e}_\theta(t) = |\theta_{vera}(t) - \theta_{calcolata}(t)|$ *Errore angolo di elevazione*
- $\mathbf{e}_\varphi(t) = |\varphi_{vera}(t) - \varphi_{calcolata}(t)|$ *Errore angolo di banco*

Dove, gli errori di posizione e di velocità a loro volta hanno tre componenti, le quali esprimono i singoli errori relativi ai tre assi.

7. Risultati dei test

7.1. Grafici e valori significativi

Grazie all’ausilio del simulatore, di cui abbiamo ampiamente spiegato la struttura nel capitolo che precede, siamo riusciti a valutare la bontà di alcuni sensori, portando alla luce dei dati che mostrano la netta superiorità, nell’accuratezza, dei giroscopi laser e FOG sui MEMS.

Per via del campo trattato, il settore aeronautico, ci siamo limitati a valutare i seguenti sensori²³:

Giroscopio/ Accelerometro	Casa Produttrice	Modello	Tipo	ARW, Noise density [deg/ $\sqrt{h_r}$], [$\mu\text{g}/\sqrt{Hz}$]	Bias Instability [deg/ h_r], [$\mu\text{g}/Hz$]
Giroscopio	SBG	Ellipse series	MEMS	0.18	8
Giroscopio	Moog Crossbow	DG600A-100	FOG	0.02	0.2
Giroscopio	iNAV-RQH	1001	Laser	0.0012	0.02
Accelerometro	iNAV-RQH	0018	MEMS	8	60

Detto ciò, andiamo ora a vedere i risultati delle prove sperimentali, ricordando che la traiettoria a biscotto è stata eseguita tre volte, con un tempo di simulazione pari a 1000 secondi, un sample time di 0.01 secondi ed utilizzando il solutore “ode4(Runge-Kutta)” dove *ode* sta per *ordinary differential equation* e 4 è il grado.

²³ Generalmente si utilizzano solo accelerometri del tipo MEMS, mentre i giroscopi variano a seconda della precisione voluta.

- *Configurazione Accelerometro MEMS – Giroscopio MEMS*

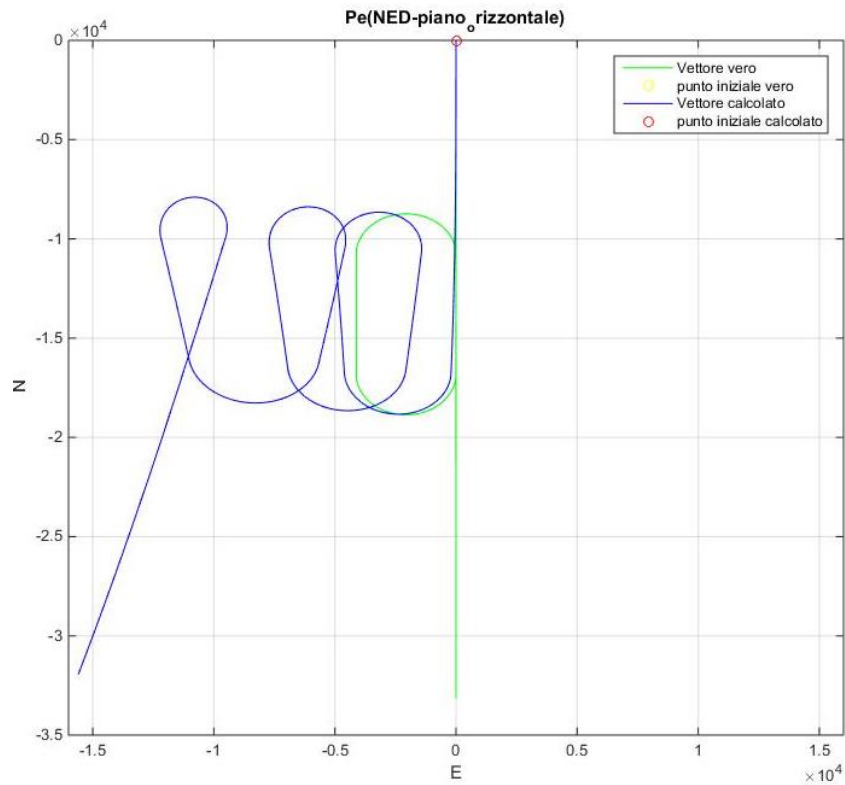


Figura 46: Accelerometro MEMS - Giroscopio MEMS – Piano orizzontale

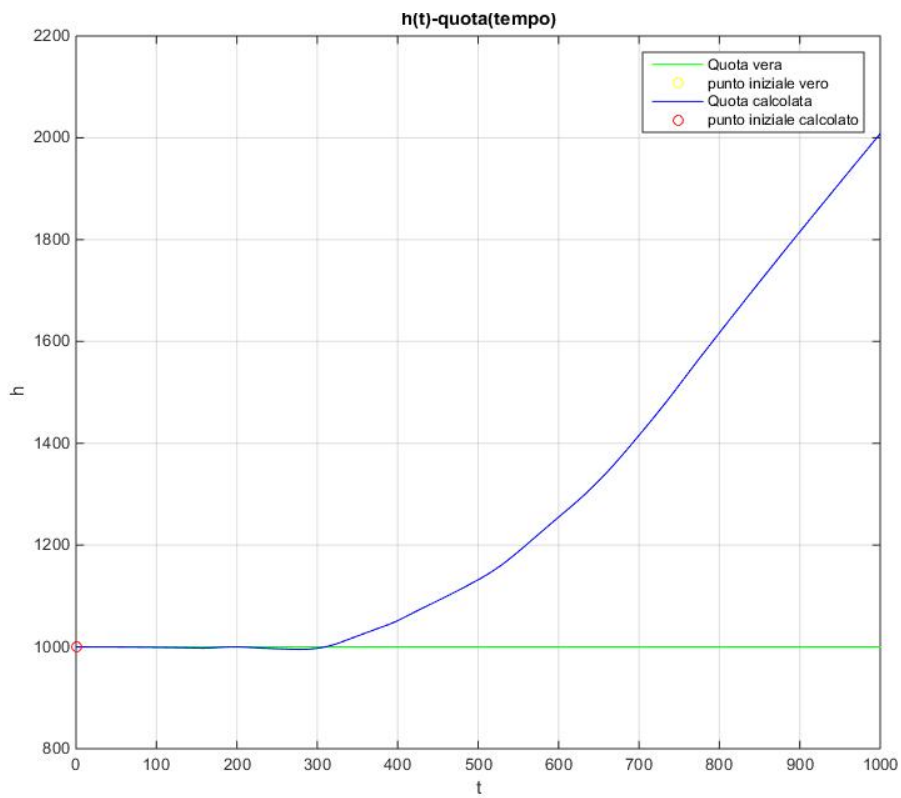


Figura 47: Accelerometro MEMS - Giroscopio MEMS - quota

Risultati dei test

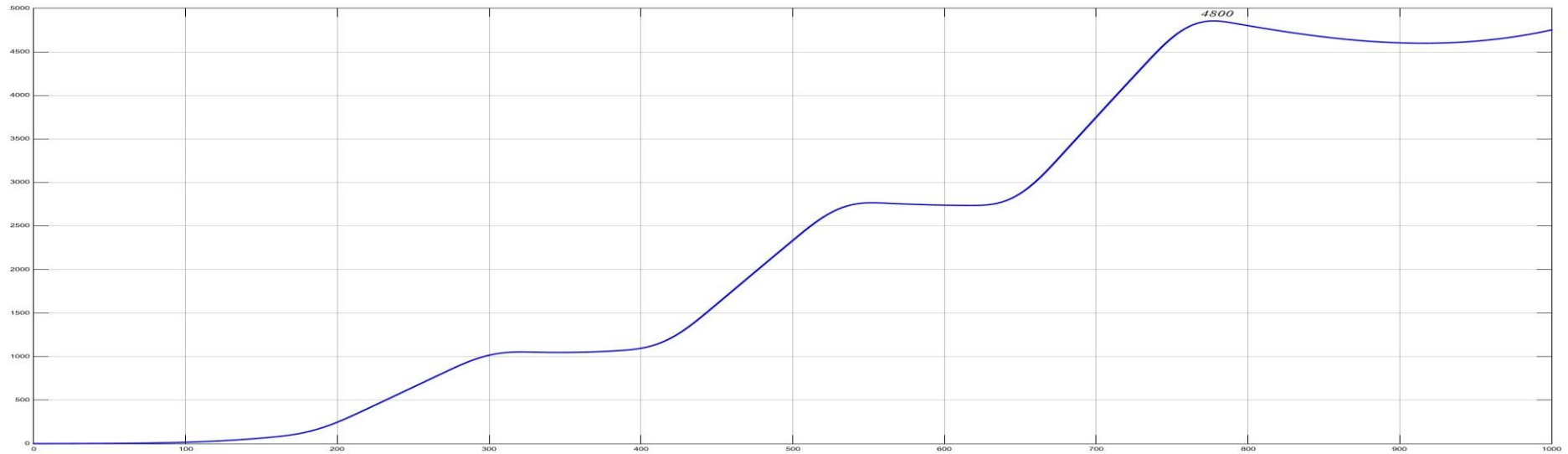


Figura 48: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell'errore di posizione sul canale orizzontale [m]

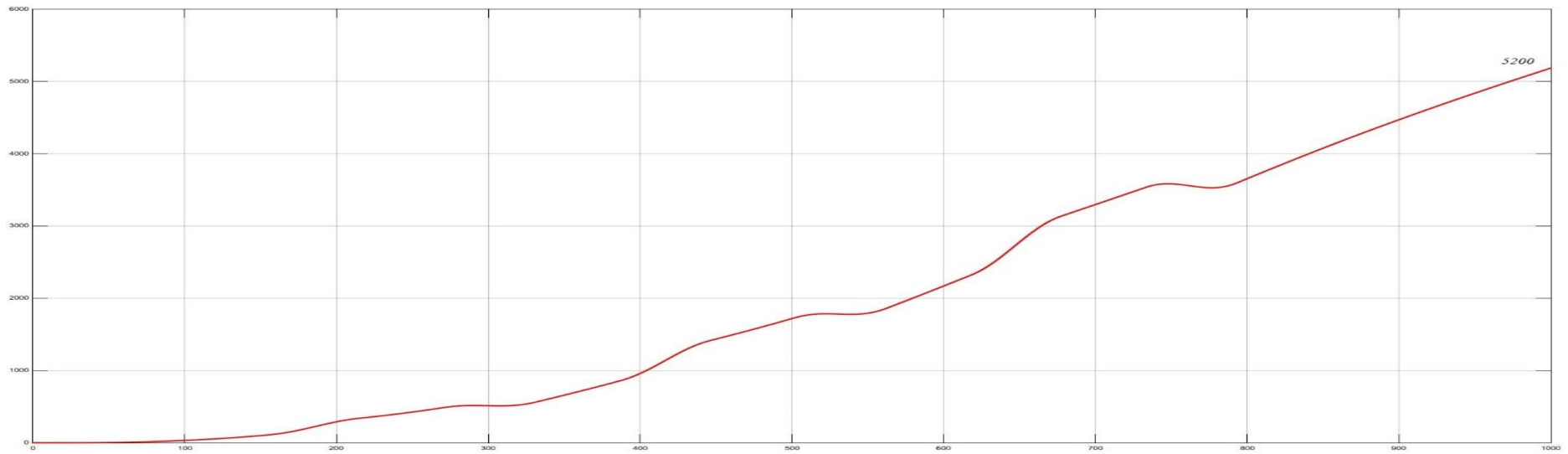


Figura 49: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell'errore di posizione sul canale verticale [m]

Risultati dei test

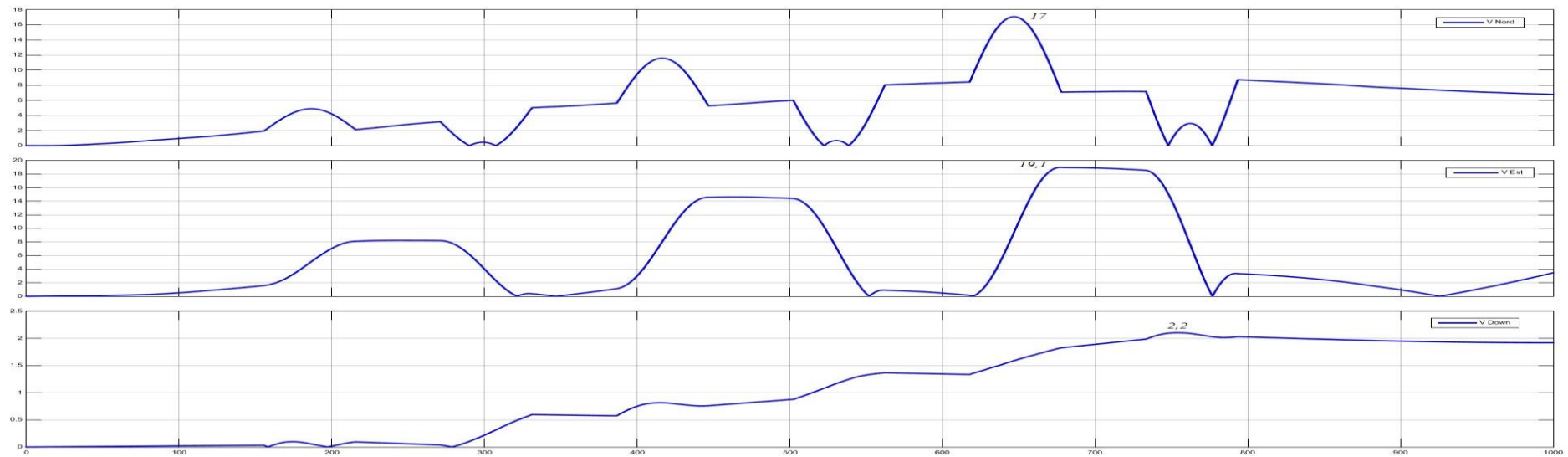


Figura 50: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell'errore sulle componenti di velocità traslazionale [m/s]

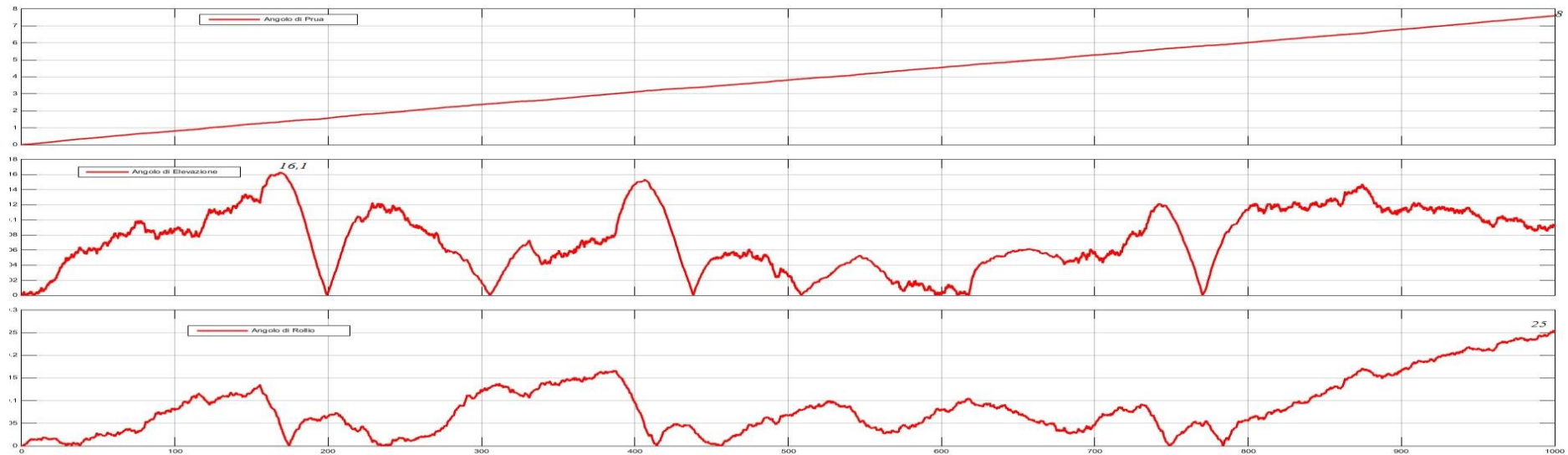


Figura 51: Accelerometro MEMS - Giroscopio MEMS – Valore assoluto dell'errore sugli angoli d'assetto [deg]

- Configurazione Accelerometro MEMS – Giroscopio FOG

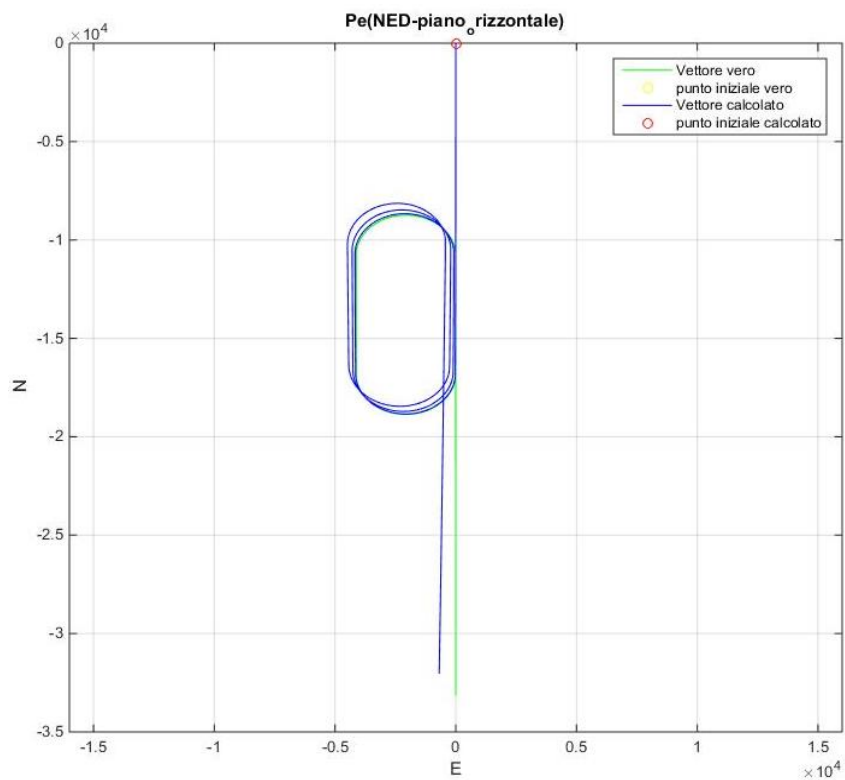


Figura 53: Accelerometro MEMS - Giroscopio FOG – Piano orizzontale

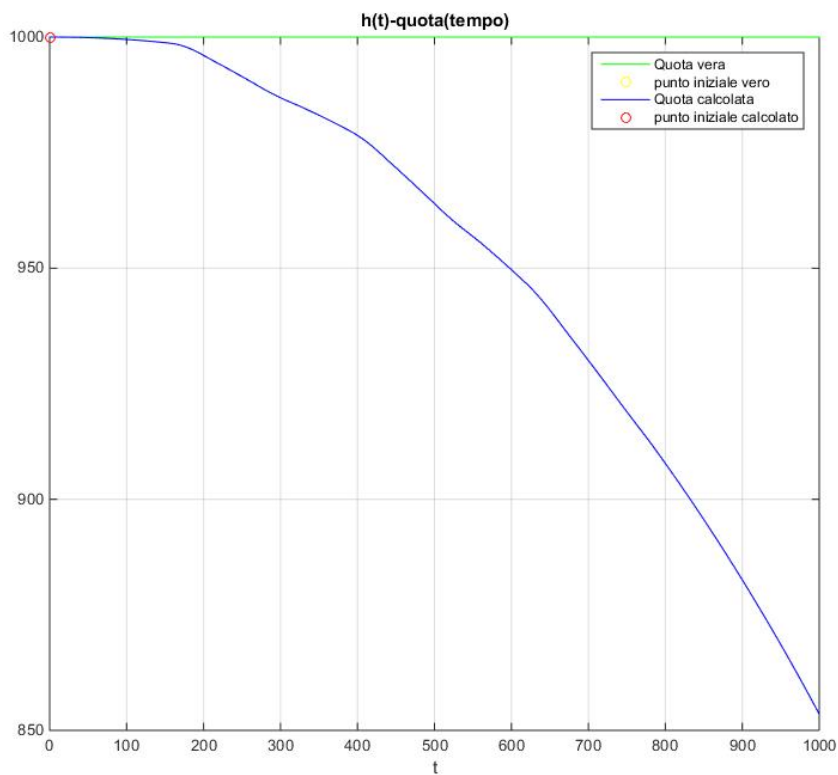


Figura 52: Accelerometro MEMS - Giroscopio FOG - quota

Risultati dei test

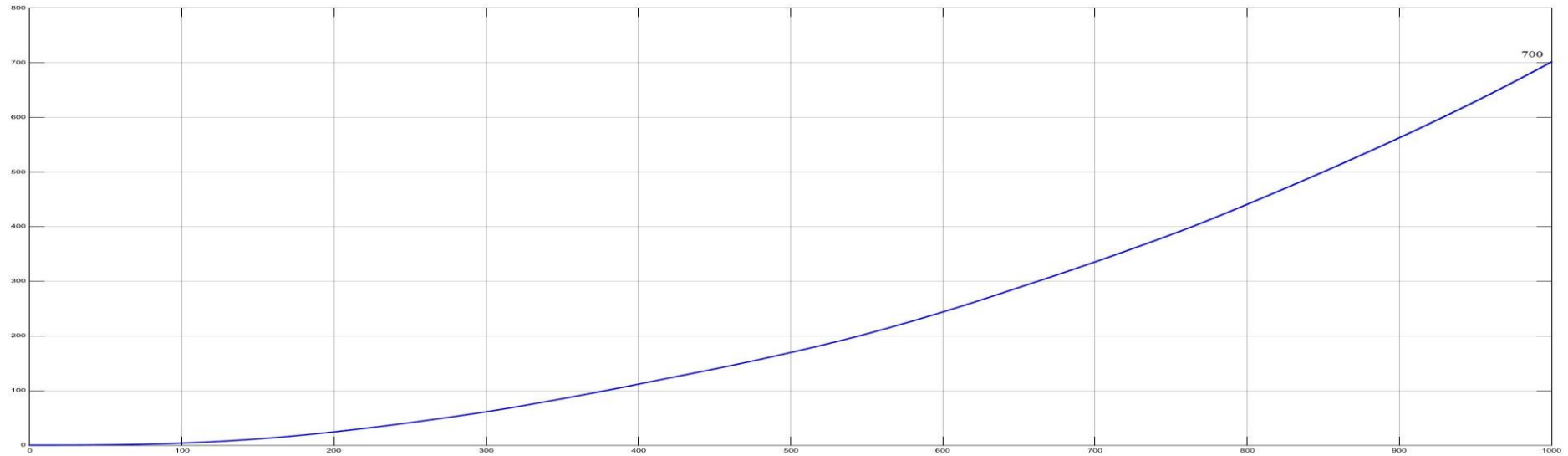


Figura 54: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell'errore di posizione sul canale orizzontale [m]

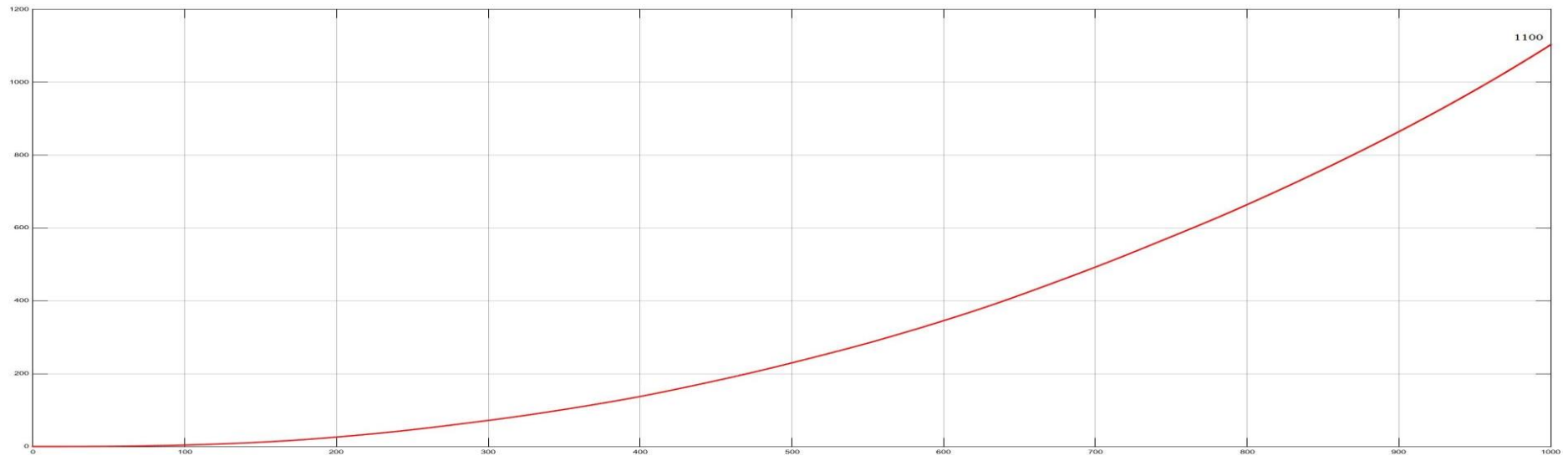


Figura 55: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell'errore di posizione sul canale verticale [m]

Risultati dei test

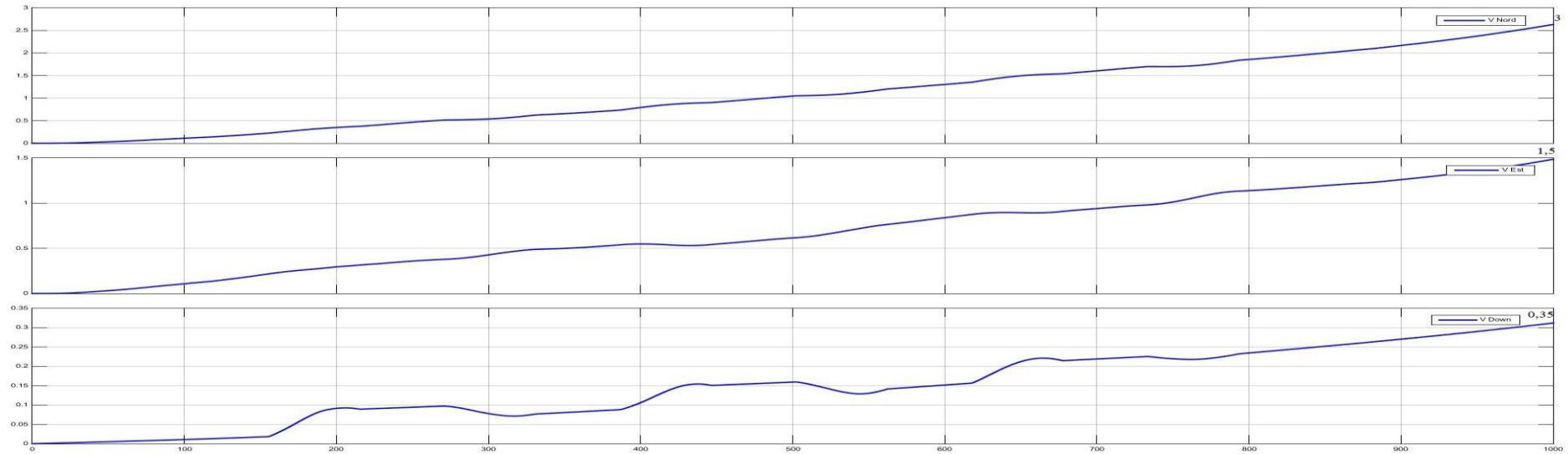


Figura 56: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell'errore sulle componenti di velocità traslazionale [m/s]

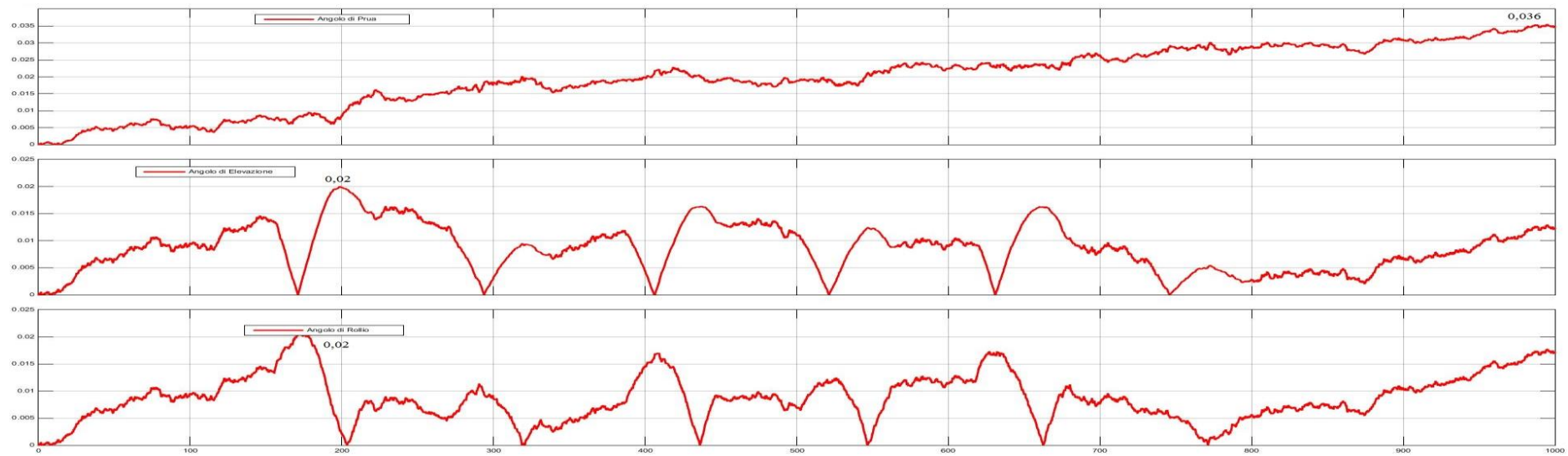


Figura 57: Accelerometro MEMS - Giroscopio FOG – Valore assoluto dell'errore sugli angoli d'assetto [deg]

- *Configurazione Accelerometro MEMS – Giroscopio Laser*

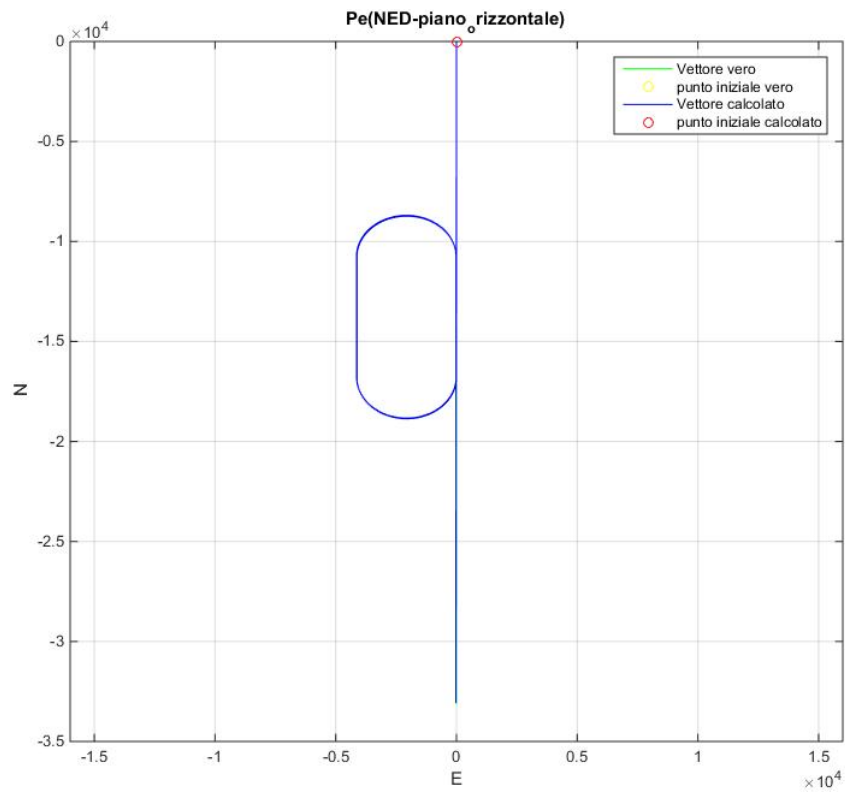


Figura 59: Accelerometro MEMS - Giroscopio Laser – Piano orizzontale

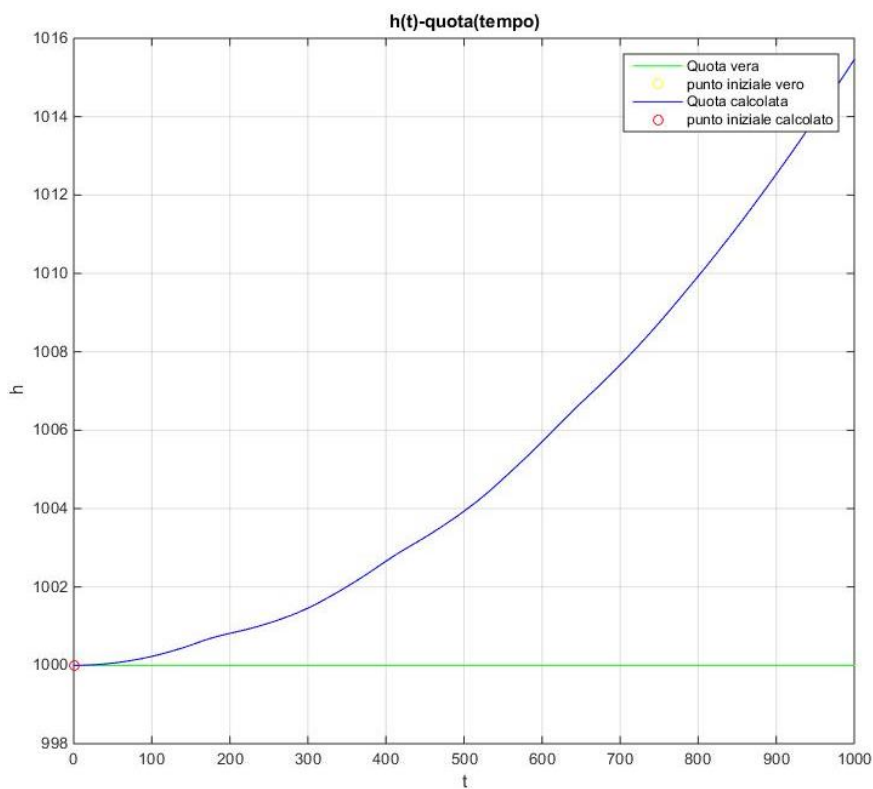


Figura 58: Accelerometro MEMS - Giroscopio Laser - quota

Risultati dei test

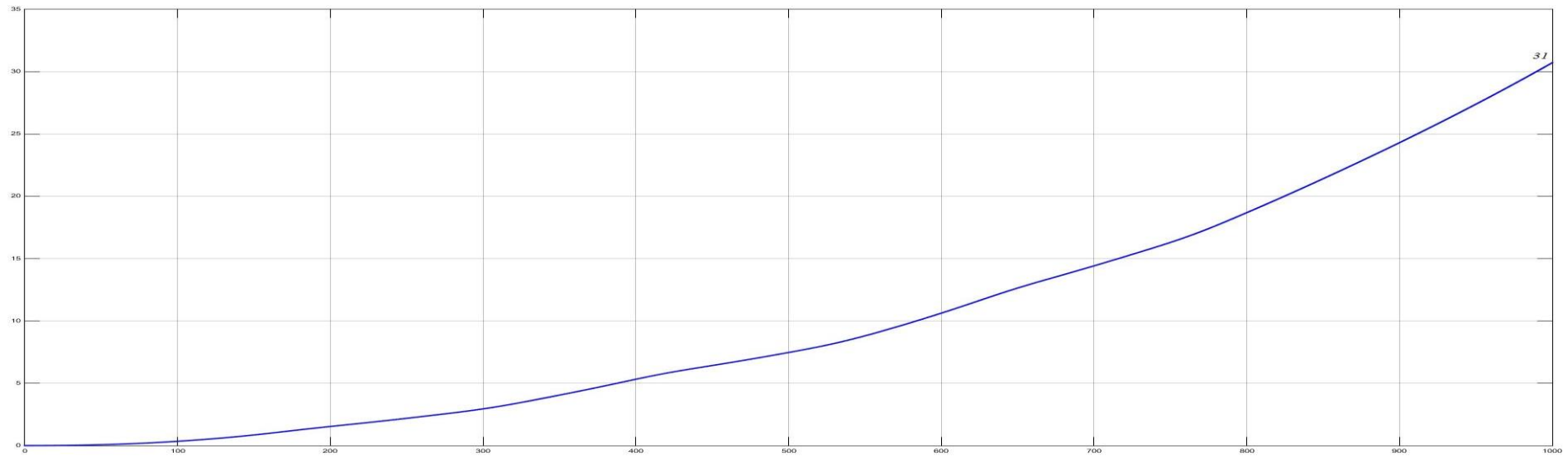


Figura 61: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell'errore di posizione sul canale orizzontale [m]

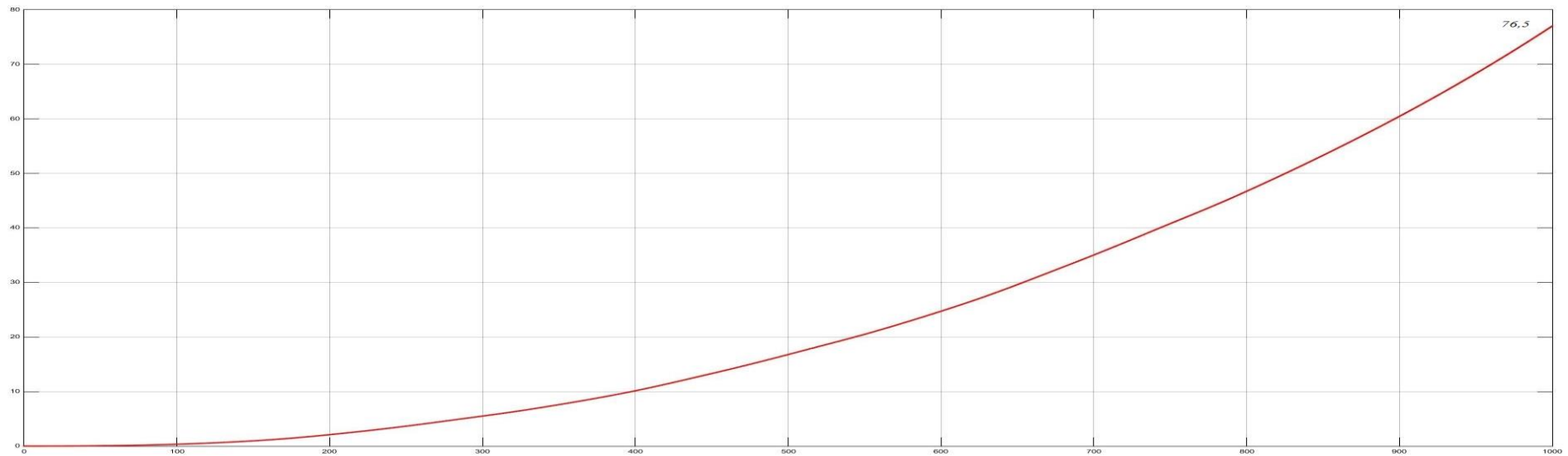


Figura 60: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell'errore di posizione sul canale verticale [m]

Risultati dei test

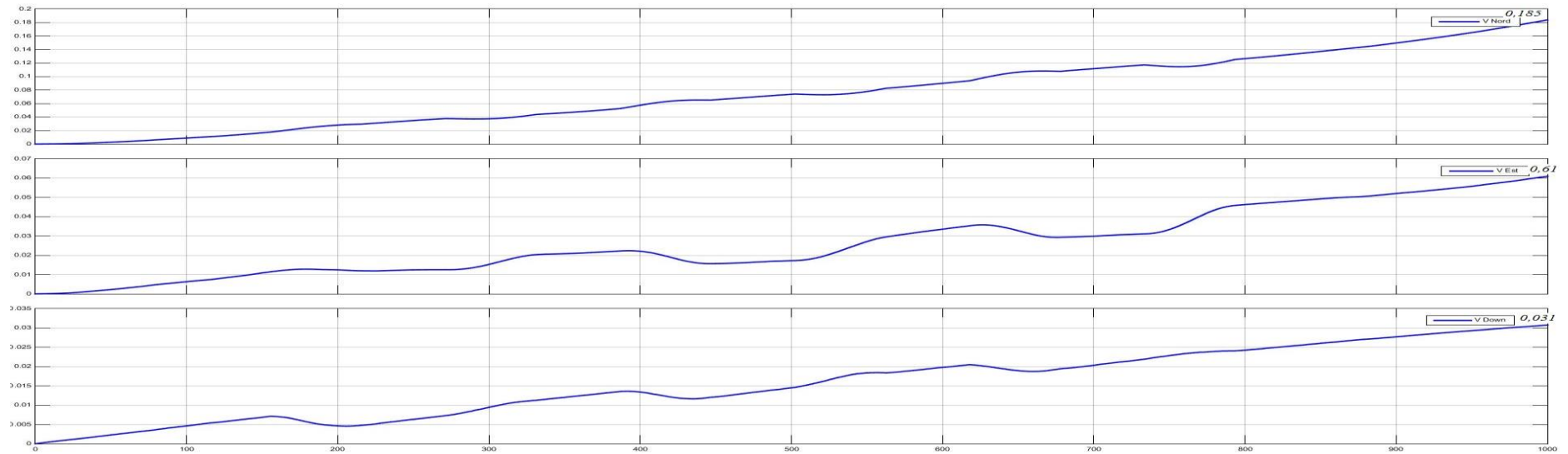


Figura 62: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell'errore sulle componenti di velocità traslazionale [m/s]

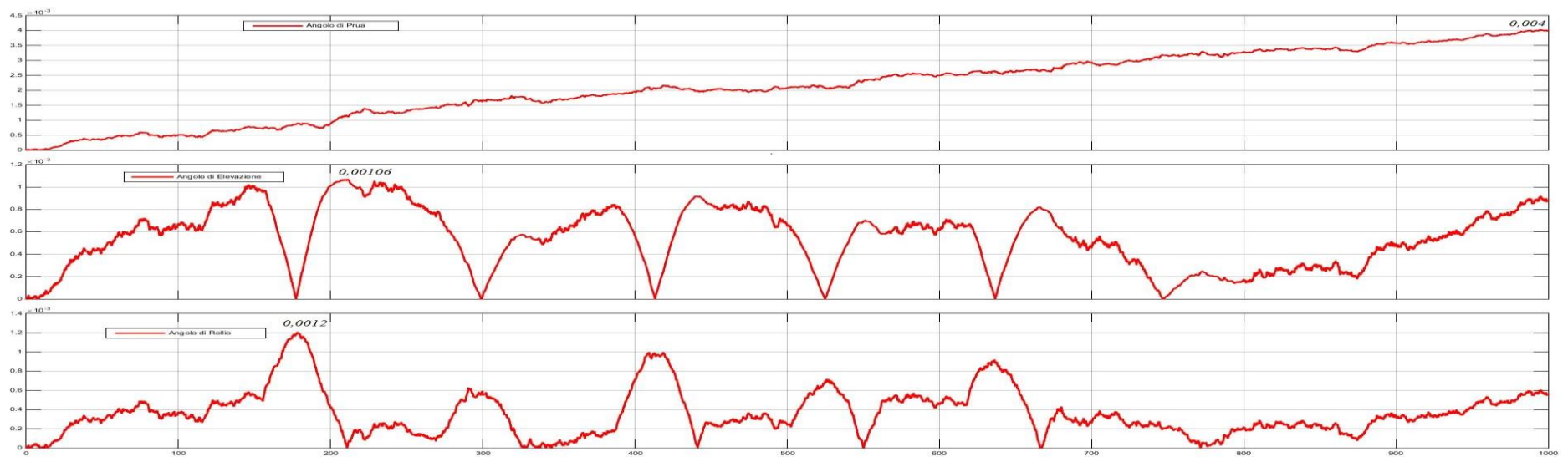


Figura 63: Accelerometro MEMS - Giroscopio Laser – Valore assoluto dell'errore sugli angoli d'assetto [deg]

Tramite l'analisi effettuata, possiamo sicuramente affermare che, a livello qualitativo, i giroscopi laser hanno una precisione maggiore dei FOG, i quali hanno, a loro volta, una precisione maggiore dei MEMS.

Infatti, alla fine della simulazione, quantitativamente, registriamo i seguenti dati²⁴:

Errori	Errore di Posizione (Nord) [m]	Errore di Posizione (Est) [m]	Errore di Posizione (Down) [m]	Errore di Velocità [m/s]	Errore di Assetto [deg]
Giroscopi					
MEMS	$1,461 * 10^3$	$9,996 * 10^3$	$1,744 * 10^4$	52.5176	0.0435
FOG	76,3275	492,6841	471,1635	1,5943	0,0015
Laser	15,86	26,3	77,02	0,196	$9,982 * 10^{-5}$

Per concludere, abbiamo riportato qui di seguito gli algoritmi utilizzati per plottare le traiettorie simulate, tralasciando ciò che non è stato utilizzato ai fini della nostra ricerca.

```

scelta1=menu('Scegli cosa visualizzare:', 'Vettore posizione', 'Vettore velocità (NED) ');

switch scelta1
case 1
    scelta2=menu('NED o ECEF?:', 'NED', 'ECEF');
    switch scelta2
    case 1
        [NED,NED_c]=from_ECEF_to_NED(Pe_e,Pe_e_c);
        scelta3=menu('Scegli cosa visualizzare:', 'Piano orizzontale', 'variazione della quota (canale verticale)', 'Entrambi');

        switch scelta3
        case 1
            plot_P_NED_orizz(NED,NED_c);
        case 2
            plot_h_func_t(h_vero,h_calc,tout);
        case 3
            subplot(1,2,1);
            plot_P_NED_orizz(NED,NED_c);
            subplot(1,2,2);
            plot_h_func_t(h_vero,h_calc,tout);
        end
    case 2
        plot_P_ECEF(Pe_e,Pe_e_c);
    end
case 2
    plot_V(Ve_n,Ve_n_c);

```

²⁴ Tutti i dati sono in valore assoluto e, per quanto riguarda la velocità e l'assetto, è stata presa come errore la radice quadrata della somma dei quadrati degli errori presenti sulle tre componenti del vettore velocità/assetto (*Root Sum Square*).

```
end
```

Dove:

```
% Plot del Piano Orizzontale
```

```
function plot_P_NED_orizz(NED,NED_c)
```

```

    plot(NED(2,:),NED(1,:), 'g');
    hold on
    plot(NED(2,1),NED(1,1), 'yo');
    plot(NED_c(2,:),NED_c(1,:), 'b');
    hold on
    plot(NED_c(2,1),NED_c(1,1), 'ro');
    grid on
    title Pe(NED-piano_orizzontale)
    xlabel E
    ylabel N
    legend('Vettore vero', 'punto iniziale vero', 'Vettore calcolato', 'punto
        iniziale calcolato');
    hold off

```

```
end
```

```
% Plot della quota in funzione del tempo
```

```
function plot_h_func_t(h,hc,t)
```

```

    plot(t,h, 'g');
    hold on
    plot(t(1),h(1), 'yo');
    plot(t,hc, 'b');
    hold on
    plot(t(1),hc(1), 'ro');
    grid on
    title h(t)-quota(tempo)
    xlabel t
    ylabel h
    legend('Quota vera', 'punto iniziale vero', 'Quota calcolata', 'punto ini-
        ziale calcolato');
    hold off

```

```
end
```



Conclusioni

Lo studio effettuato in questa tesi è stato molto utile al fine dell'identificazione qualitativa e quantitativa dei rumori che perturbano le uscite dei sensori inerziali. Grazie ad esso siamo riusciti a quantificare queste perturbazioni, valutando le grandi differenze che sussistono all'interno di giroscopi con diverse tecnologie.

In particolare, abbiamo identificato quegli errori, non prevedibili deterministicamente, responsabili di distorsioni nelle misurazioni, le quali possono essere valutate solo mediante lo studio dei processi stocastici che li governano.

Naturalmente, questo è uno studio di prima approssimazione, utile per un primo approccio a questo mondo, che porta a capire l'importanza dell'implementazione di filtri (come ad esempio il filtro di Kalman) all'interno dei nostri algoritmi per correggere questi errori, che non possono assolutamente essere trascurati, soprattutto per sensori di basso costo come i MEMS.

In qualunque settore, ogni giorno, si cerca di trovare la miglior soluzione al minor prezzo possibile e questo è possibile solo utilizzando sensori che non siano molto precisi e costosi, come ad esempio i sensori laser di cui abbiamo visto l'enorme capacità di accuratezza, ma sensori di basso costo e, magari, anche di bassa precisione, ma di cui se ne riescono a modellizzare matematicamente gli errori per poi riuscire ad eliminarli; questa è una delle tante ragioni per cui esistono ancora sensori quali i MEMS che, nonostante la loro scarsa affidabilità e precisione, sono molto utilizzati nel settore aeronautico come in quello automobilistico.

Riferimenti

- [1] National Imagery and Mapping Agency, «Department of Defense World Geodetic System 1984, its Definition and Relationships whit Local Geodetic Systems,» third edition, 3 january 2000.
- [2] S. M. S. Di Gennaro, *Appunti di Modellistica e Controllo di Strutture Spaziali*, 22/01/2007.
- [3] M. Zanzi, «Dispense di Elaborazione Dati per la Navigazione - Sistemi di Riferimento e Navigazione Inerziale,» 20 04 2015. [Online]. Available: <http://campus.unibo.it/185346/1/Sistemi%20di%20Riferimento%20e%20Navigazione%20Inerziale.pdf>.
- [4] J. L. David Titterton, *Strapdown Inertial Navigation Technology*, 2nd Edition, 2004.
- [5] C. Verplaetse, «Can A Pen Remember What It Has Written Using Inertiali Navigation? : An Evaluation Of Current Accelerometer Technology,» 1995.
- [6] «Wikipedia, L'Enciclopedia Libera,» [Online]. Available: <https://it.wikipedia.org/wiki/Trasduttore>.
- [7] «Wikipedia, L'Enciclopedia Libera,» [Online]. Available: <https://it.wikipedia.org/wiki/Sensore>.
- [8] C. Secchi, «Sensori e Trasduttori - Ingegneria e Tecnologia dei sistemi di controllo,» 2005-2006. [Online]. Available: <http://www.automazione.ingre.unimore.it/pages/corsi/materialeDidattico/ITSC0607/ITSC10-Sensori.pdf>.
- [9] M. Zanzi, «Dispense di Elaborazione Dati per la Navigazione - Accelerometri e Giroscopi,» 08 04 2015. [Online]. Available: <http://campus.unibo.it/183482/1/Accelerometri%20e%20Gyros.pdf>.
- [10] L. I. Wikipedia, «MEMS,» [Online]. Available: <https://it.wikipedia.org/wiki/MEMS>.
- [11] I. P. Cmaps, «Accelerometro,» [Online]. Available: <http://skat.ihmc.us/rid=1H20KPZL4-6BM3G6-RS7/accelerometro.pdf>.
- [12] N. E. a. E. T. S. (NEETS), «Principles of Synchros, Servos, and Gyros,» [Online]. Available: <http://www.rfcafe.com/references/electrical/NEETS-Modules/NEETS-Module-15-3-21-3-27.htm>.
- [13] N. M. M. S. Wolfgang Holzapfel, «INERTIAL GRADE LASER ACCELEROMETER PRACTICABILITY AND BASIC EXPERIMENTS,» in *XVII IMEKO World Congress - Metrology in the 3rd Millennium*, Dubrovnik, Croatia, 2003.
- [14] ITEMare, «Teoria del funzionamento MC2,» [Online]. Available: http://www.tonissi.com/rt_files/doc/Funzionamento_MC2.pdf.
- [15] M. N. S. D. Sabina Merlo, «"The Electro-Optical Gyroscope" - Fiber Optic Sensors,» A. Higuera, J. Wiley and Sons, 2002.
- [16] R. R. Jeng-Nan Juang, «EVALUATION OF RING LASER AND FIBER OPTIC GYROSCOPE TECHNOLOGY,» School of Engineering, Mercer University, Macon, GA 31207 USA.
- [17] P. P. J. K. J. D.-C. A. B. Javier Hidalgo, «Improving Planetary Rover Attitude Estimation via MEMS Sensor Characterization,» Creative Commons Attribution license, Basel, Switzerland., 2012.

- [18] Novatel, «IMU Errors and Their Effects,» 21 February 2014. [Online]. Available: <http://www.novatel.com/assets/Documents/Bulletins/APN064.pdf>.
- [19] H. Hou, «Modeling Inertial Sensors Errors Using Allan Variance,» UCGE Reports - Department of Geomatics Engineering, Calgary, 2004.
- [20] M. Looney, «Analog Devices - The Basics of MEMS IMU/Gyroscope Alignment,» June 2015. [Online]. Available: http://www.analog.com/library/analogdialogue/archives/49-06/IMU_Gyroscope.html.
- [21] O. J. Woodman, «An introduction to inertial navigation,» University of Cambridge, 2007.
- [22] M. E. Pittelkau, «Rotation Vector in Attitude Estimation,» *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS*, vol. Vol. 26, n. n. 6, pp. Vol. 26, No. 6, November–December 2003, November–December 2003.
- [23] M. C. Leonardo Calandrino, *Lezioni di Comunicazioni Elettriche*, Bologna: Pitagora Editrice, 2013.
- [24] Mathworks, «MATLAB - Il linguaggio del calcolo tecnico,» [Online]. Available: <http://it.mathworks.com/products/matlab/>.
- [25] Mathworks, «Simulink - Simulazione e progettazione model-based,» [Online]. Available: <http://it.mathworks.com/products/simulink/>.
- [26] V. Nastro, *Assistenza al volo e controllo del traffico aereo*, ULRICO HOEPLI MILANO, 2008.
- [27] M. Zanzi, «Dispense di Elaborazione Dati per la Navigazione - Guida,» 12 05 2015. [Online]. Available: <http://campus.unibo.it/188061/1/Guida.pdf>.