

Agostino De Marco
Domenico P. Coiro

Elementi
di
Dinamica e simulazione di volo

Quaderno 3
Quaternione dell'orientamento
di un velivolo

Marzo 2017
ver. 2017.a

Dichiarazione di Copyright

- Questo testo è fornito per uso personale degli studenti. Viene reso disponibile in forma preliminare, a supporto della preparazione dell'esame di *Dinamica e simulazione di volo*.
- Sono consentite la riproduzione e la circolazione in formato cartaceo o elettronico ad esclusivo uso scientifico, didattico o documentario, purché il documento non venga alterato in alcun modo sostanziale, ed in particolare mantenga le corrette indicazioni di data, paternità e fonte originale.
- Non è consentito l'impiego di detto materiale a scopi commerciali se non previo accordo.
- È gradita la segnalazione di errori o refusi.

Copyright 2010–2017 Agostino De Marco e Domenico P. Coiro
Università degli Studi di Napoli Federico II
Dipartimento di Ingegneria Industriale

(Legge italiana sul Copyright 22.04.1941 n. 633)

Quaternione dell'orientamento di un velivolo

Read Euler: he is our master in everything.
– Pierre-Simon Laplace

Indice

3.1	Parametrizzazione dell'orientamento	3
3.2	Asse di Eulero e formulazione asse-angolo	4
3.2.1	Asse di Eulero	4
3.2.2	Rotazioni finite e parametri di Eulero	6
3.2.3	Parametri di Eulero	9
3.3	Formulazione basata sul quaternione dell'orientamento	11
3.3.1	Parametri simmetrici di Eulero-Rodrigues	11
3.3.2	Parametrizzazione di Eulero-Rodrigues	12
3.3.3	Cenni all'algebra dei quaternioni	14
3.3.4	Rotazioni finite e quaternioni	17
3.3.5	Dal quaternione dell'orientamento agli angoli di Eulero	21
3.3.6	Esempi di storie temporali del quaternione dell'orientamento	25

3.1 Parametrizzazione dell'orientamento

Nella Fisica Matematica il modo in cui viene rappresentato l'orientamento di un solido nello spazio e la sua caratterizzazione attraverso un certo numero di parametri viene spesso chiamato "parametrizzazione dell'orientamento". La definizione di orientamento del velivolo basata sugli angoli di Eulero è una parametrizzazione a tre parametri. Essi rappresentano una sequenza di rotazioni che, a meno di uno spostamento rigido dell'origine, porta una terna di riferimento, che inizialmente coincide con una terna fissa, a sovrapporsi alla terna degli assi velivolo. La parametrizzazione con angoli di Eulero esprime ovviamente il fatto che l'orientamento di un corpo rigido nello spazio ha tre gradi di libertà. Da

tale formulazione discende la costruzione della matrice di rotazione $[T]_{EB}$ che compare nella (2.12). Dal fatto che la moltiplicazione fra matrici è un'operazione *non* commutativa si può dedurre che la definizione di angoli di Eulero basata sulla sequenza di rotazioni sopra illustrata *non* è l'unica possibile. In altre parole, con una permutazione nell'ordine delle rotazioni è possibile ottenere il medesimo orientamento finale con valori differenti nella terna di angoli.

La definizione di angoli di Eulero data nel capitolo precedente è quella comunemente accettata per la descrizione dell'orientamento di un velivolo ed è basata su una sequenza di rotazioni del tipo "3-2-1". In altre discipline si parla ugualmente di angoli di Eulero ma le definizioni adottate sono basate su sequenze di rotazioni diverse. Ad esempio in astrodinamica gli angoli di Eulero sono definiti da una sequenza del tipo "3-2-3".

Il più evidente svantaggio della parametrizzazione con angoli di Eulero è il verificarsi del *gimbal lock*, cioè di una singolarità per la quale si perde un grado di libertà quando il parametro $\theta \rightarrow \pm \frac{\pi}{2}$.

La figura 2.4 mostra un esempio di orientamento del velivolo per il quale si verifica il *gimbal lock*. In tale situazione, con la fusoliera orientata in verticale e la prua verso l'alto (praticamente con $\theta = 90$ deg), è facile verificare che il medesimo orientamento del velivolo può essere dato, oltre che dalla coppia di valori angolari (ψ, ϕ) , dall'infinità di coppie $(\psi + \Delta, \phi + \Delta)$, con Δ qualsiasi. In altri termini, con la fusoliera così disposta, i parametri ψ e ϕ vengono ad appartenere allo stesso grado di libertà.

3.2 Asse di Eulero e formulazione asse-angolo

3.2.1 Asse di Eulero

L'orientamento del riferimento non inerziale, solidale al velivolo rispetto ad una terna assunta come riferimento fisso può essere descritto attraverso un approccio fornito dallo stesso Eulero. Lo sviluppo di tale formulazione è basato su una proposizione che è nota come *Teorema di Eulero*:

L'orientamento istantaneo di un corpo rigido, il cui moto è vincolato a delle rotazioni intorno a un punto fisso, può sempre essere descritto come quello originato da una singola rotazione a partire dall'orientamento iniziale, di un dato angolo e attorno a un dato asse.

L'asse di rotazione è comunemente noto come *asse di Eulero* e l'angolo di rotazione come *angolo di Eulero*. Questa parametrizzazione dell'orientamento, detta anche *parametrizzazione asse-angolo*, è rappresentata nella figura 3.2 dove e è un vettore che rappresenta la direzione dell'asse di Eulero e μ è l'angolo di Eulero. Nella rappresentazione riportata la terna di assi velivolo \mathcal{T}_B si ottiene dalla terna di assi Terra \mathcal{T}_E attraverso una singola rotazione di quest'ultima intorno all'asse e di un angolo μ .

Quando è applicata allo studio del moto di un velivolo, la parametrizzazione asse-angolo può fornire una formulazione alternativa delle trasformazioni cinematiche ausiliarie (2.26)-(2.27). La rotazione descritta dal Teorema di Eulero dà luogo ad una parametrizzazione dell'orientamento a quattro parametri: l'angolo di rotazione μ e le tre componenti (e_x, e_y, e_z) del versore e diretto secondo la direzione dell'asse di Eulero. Si osservi che i

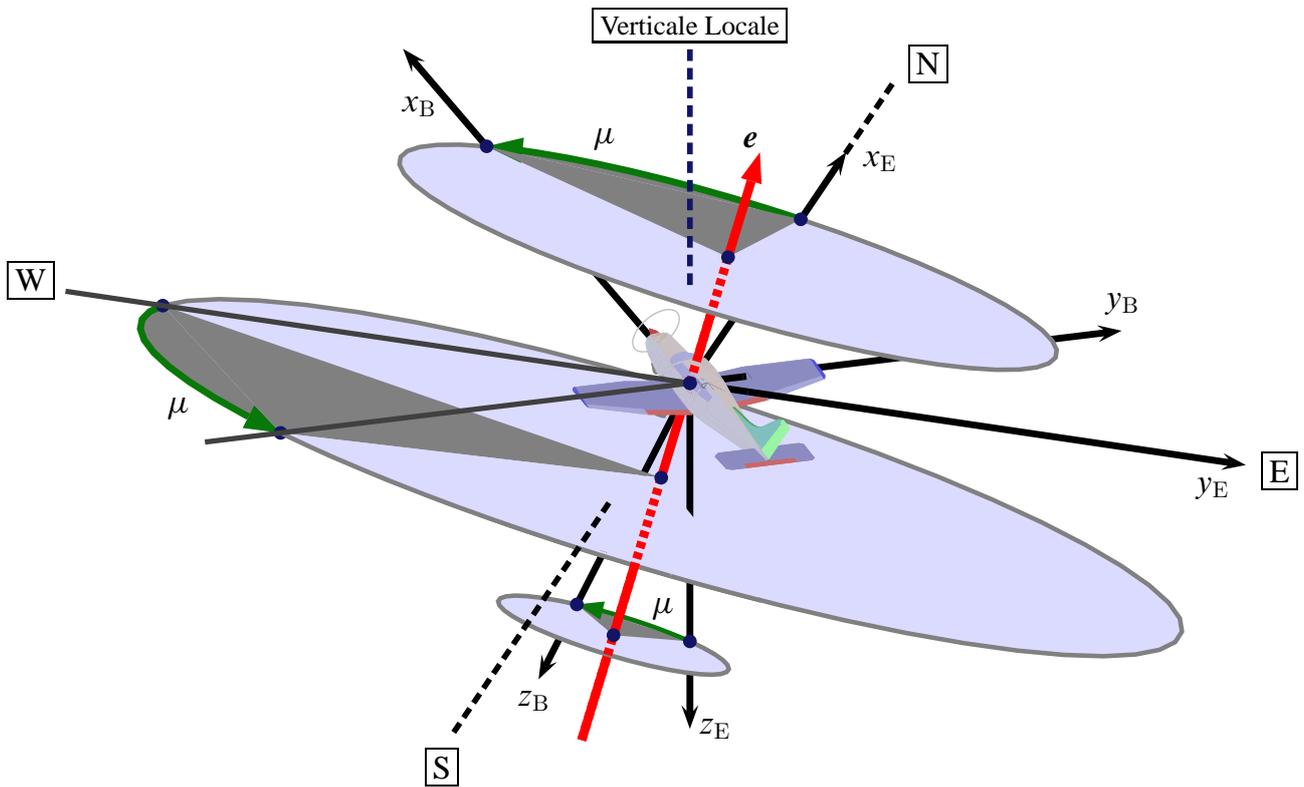


Figura 3.2 Parametrizzazione “asse-angolo” dell’orientamento del velivolo. La terna di assi velivolo \mathcal{T}_B si ottiene dalla terna di assi Terra \mathcal{T}_E attraverso una rotazione intorno all’asse di Eulero e di un angolo μ , angolo di Eulero.

(e_x, e_y, e_z, μ) . Queste dipendenze sono alternative a quelle degli elementi $T_{BE,ij}$ della matrice di trasformazione $[T_{BE}]$ dai tre angoli di Eulero (ϕ, θ, ψ) .

Dal momento che i quattro parametri suddetti rappresentano l’orientamento di un solido nello spazio, che ha tre gradi di libertà, esiste una ridondanza nella parametrizzazione asse-angolo finora descritta. È evidente, d’altra parte, che la lunghezza del vettore che descrive la direzione nello spazio dell’asse di Eulero è arbitraria. Al fine di ridurre i gradi di libertà a tre è opportuno imporre un vincolo al modulo del vettore e . Seppure tale vincolo può essere scelto in modo arbitrario, la scelta più comune e al tempo stesso più ovvia è quella di richiedere che esso abbia un modulo unitario

$$e_x^2 + e_y^2 + e_z^2 = 1 \quad (3.3)$$

Il vettore e è quindi il *versore* dell’asse di Eulero.

3.2.2 Rotazioni finite e parametri di Eulero

Per arrivare ad interpretare il significato di una parametrizzazione a quattro parametri priva di singolarità, che *non* è quella del tipo asse-angolo ma, come vedremo più avanti, è quella basata sul quaternione dell’orientamento, è opportuno a questo punto richiamare alcuni risultati di base sulle rotazioni di vettori in spazi euclidei [3].

Come illustrato nella figura 3.3, un vettore $v = P - O$ dello spazio euclideo tridimensionale, che si intende ruotare di un angolo μ attorno ad un asse dello spazio individuato

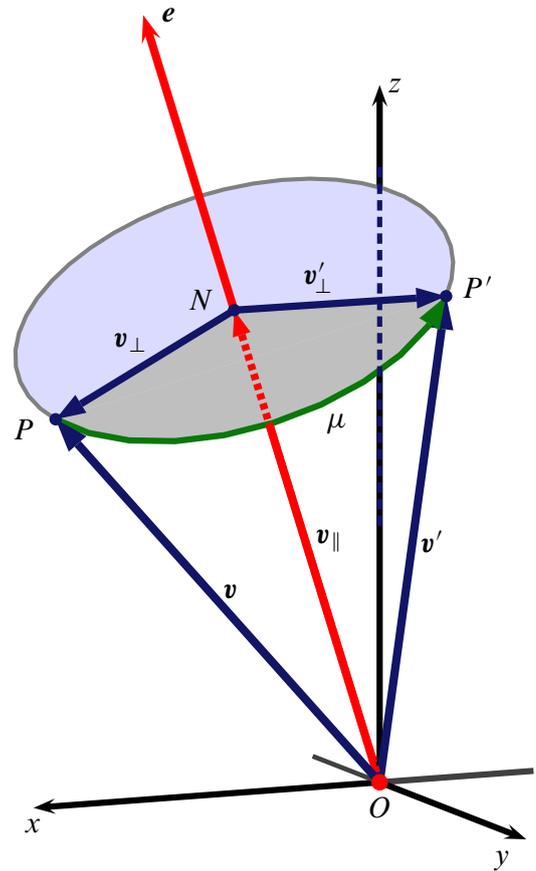


Figura 3.3 Rotazione di un angolo μ del vettore $\mathbf{v} = \mathbf{v}_\perp + \mathbf{v}_\parallel$ attorno all'asse individuato dal versore \mathbf{e} . A rotazione avvenuta il vettore \mathbf{v}' è somma dei vettori \mathbf{v}'_\perp e \mathbf{v}_\parallel .

dal versore \mathbf{e} , è esprimibile come somma $(P - N) + (N - O)$ ovvero

$$\mathbf{v} = \mathbf{v}_\parallel + \mathbf{v}_\perp = (\mathbf{v} \cdot \mathbf{e}) \mathbf{e} + [\mathbf{v} - (\mathbf{v} \cdot \mathbf{e}) \mathbf{e}] \quad (3.4)$$

dove N è la proiezione del punto P sull'asse di rotazione, essendo $(N - O) = \mathbf{v}_\parallel$ un vettore disposto lungo l'asse e $(P - N) = \mathbf{v}_\perp$ un vettore normale all'asse.

A rotazione avvenuta il vettore \mathbf{v} si trasforma in

$$\mathbf{v}' = (P' - O) \quad (3.5)$$

ma il primo addendo della somma (3.4) si conserva inalterato. Il secondo addendo \mathbf{v}_\perp viene trasformato in

$$(P' - N) = \mathbf{v}'_\perp \quad (3.6)$$

Dall'esame della figura 3.4 si può verificare che la rotazione di \mathbf{v}_\perp fornisce

$$\mathbf{v}'_\perp = \cos \mu \mathbf{v}_\perp + \sin \mu \mathbf{e} \wedge \mathbf{v}_\perp \quad (3.7)$$

Sostituendo l'espressione di \mathbf{v}_\perp come $\mathbf{v} - (\mathbf{v} \cdot \mathbf{e}) \mathbf{e}$ nella (3.7) e tenendo conto della (3.4) il vettore \mathbf{v}' si esprime infine come

$$\mathbf{v}' = \cos \mu \mathbf{v} + \sin \mu \mathbf{e} \wedge \mathbf{v} + (1 - \cos \mu) (\mathbf{v} \cdot \mathbf{e}) \mathbf{e} \quad (3.8)$$

La (3.8) si può scrivere in termini matriciali come relazione del tipo

$$\{v'\} = [T(\mathbf{e}, \mu)]\{v\} \quad (3.9)$$

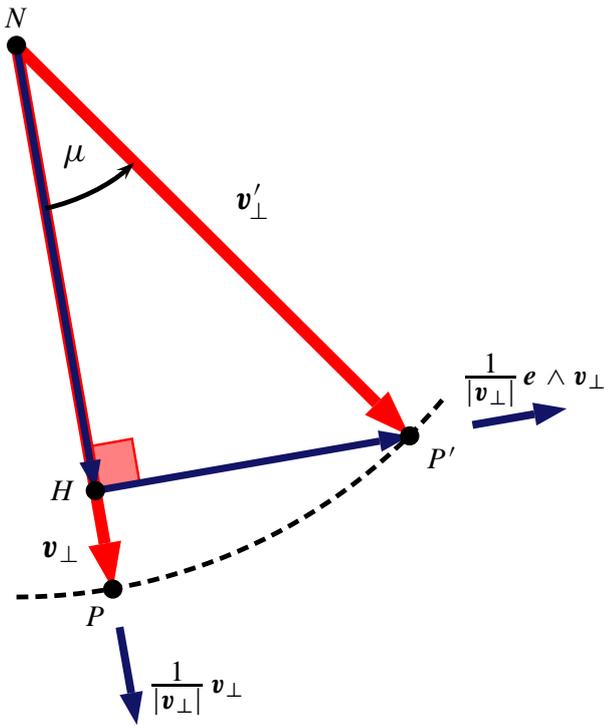


Figura 3.4 Rotazione di un angolo μ del vettore $(P - N) = \mathbf{v}_\perp$ attorno ad N , traccia dell'asse individuato dal versore \mathbf{e} normale al piano del foglio. A rotazione avvenuta il vettore $(P' - N) = \mathbf{v}'_\perp$ è somma dei vettori $(H - N) = \cos \mu \mathbf{v}_\perp$ e $(P' - H) = \sin \mu \mathbf{e} \wedge \mathbf{v}_\perp$.

dove $[T(\mathbf{e}, \mu)]$ è una matrice che rappresenta la trasformazione delle componenti di un vettore per effetto della rotazione intorno al versore \mathbf{e} dell'angolo μ . A tale scopo si introduce una matrice $[\tilde{\mathbf{e}}]$ ottenuta dalle componenti di \mathbf{e} tramite l'operatore tilde ($[\tilde{\quad}]$)

$$[\tilde{\mathbf{e}}] = \begin{bmatrix} 0 & -e_z & e_y \\ e_z & 0 & -e_x \\ -e_y & e_x & 0 \end{bmatrix} \quad (3.10)$$

La (3.10) permette di esprimere il prodotto vettoriale $\mathbf{r} = \mathbf{e} \wedge \mathbf{v}$ nel sistema di riferimento considerato come prodotto matriciale $\{r\} = [\tilde{\mathbf{e}}]\{v\}$.

Una matrice come la $[\tilde{\mathbf{e}}]$ gode di alcune proprietà notevoli (si veda ad esempio Salamin [3]). In primo luogo è possibile far vedere che, essendo \mathbf{e} un vettore di modulo unitario, si ha: $[\tilde{\mathbf{e}}]^3 = -[\tilde{\mathbf{e}}]$ (uguaglianza nota come Teorema di Cayley-Hamilton). Ne deriva che la sequenza delle potenze successive $[\tilde{\mathbf{e}}]^k$, a partire da quella di esponente nullo, $k = 0, 1, 2, 3, 4, 5, 6, 7, 8, \dots$, può essere scritta come

$$[I], [\tilde{\mathbf{e}}], [\tilde{\mathbf{e}}]^2, -[\tilde{\mathbf{e}}], -[\tilde{\mathbf{e}}]^2, [\tilde{\mathbf{e}}], [\tilde{\mathbf{e}}]^2, -[\tilde{\mathbf{e}}], -[\tilde{\mathbf{e}}]^2, \dots \quad (3.11)$$

con $[I] \equiv [\tilde{\mathbf{e}}]^0$ la matrice identica. Successivamente si può far vedere che la matrice di rotazione $[T(\mathbf{e}, \mu)]$ è la somma della serie di potenze che definisce la matrice esponenziale di $\mu [\tilde{\mathbf{e}}]$

$$[T(\mathbf{e}, \mu)] = \exp(\mu [\tilde{\mathbf{e}}]) = e^{\mu [\tilde{\mathbf{e}}]} \quad (3.12)$$

Inoltre, se si esplicitano i termini della serie (3.12) e si tiene conto della (3.11), si possono identificare nella sommatoria i termini delle serie trigonometriche che definiscono le funzioni $\sin \mu$ e $\cos \mu$. Con qualche passaggio si arriva al risultato importante che permette di esprimere la (3.12) nella forma chiusa seguente

$$[T(\mathbf{e}, \mu)] = e^{\mu [\tilde{\mathbf{e}}]} = [I] + \sin \mu [\tilde{\mathbf{e}}] + (1 - \cos \mu) [\tilde{\mathbf{e}}]^2 \quad (3.13)$$

La (3.13) è nota come *formula di Rodrigues*.

A questo punto, se si esplicita la matrice $[\tilde{e}]^2$ per semplice moltiplicazione della $[\tilde{e}]$ per se stessa,

$$[\tilde{e}]^2 = \begin{bmatrix} -e_y^2 - e_z^2 & e_x e_y & e_x e_z \\ e_y e_x & -e_x^2 - e_z^2 & e_y e_z \\ e_z e_x & e_z e_y & -e_x^2 - e_y^2 \end{bmatrix} = [e_i e_j - \delta_{ij}] \quad (i, j = x, y, z) \quad (3.14)$$

con δ_{ij} il simbolo di Kronecker (1 se $i = j$, 0 altrimenti), e si sostituisce nella (3.13) si ottiene l'espressione desiderata della matrice di rotazione

$$[T(\mathbf{e}, \mu)] = \begin{bmatrix} e_{xx} + C_\mu & e_{xy} - e_z S_\mu & e_{xz} + e_y S_\mu \\ e_{xy} + e_z S_\mu & e_{yy} + C_\mu & e_{yz} - e_x S_\mu \\ e_{xz} - e_y S_\mu & e_{yz} + e_x S_\mu & e_{zz} + C_\mu \end{bmatrix} \quad (3.15)$$

con

$$C_\mu = \cos \mu, \quad S_\mu = \sin \mu, \quad e_{ij} = e_i e_j (1 - C_\mu) \quad (3.16)$$

È facile verificare, come c'è da aspettarsi da una matrice di rotazione, che la $[T(\mathbf{e}, \mu)]$ è una matrice ortogonale: $[T(\mathbf{e}, \mu)]^{-1} = [T(\mathbf{e}, \mu)]^T$. L'inversione corrisponde in tal caso ad una rotazione $(-\mu)$ intorno allo stesso asse.

Per finire, con l'aiuto della figura 3.3, si osservi che, scambiando l'operazione di rotazione del vettore con quella di rotazione del riferimento, gli elementi della matrice colonna $\{v'\}$ sono anche le componenti, nel riferimento fin qui considerato, di quel vettore v che nel riferimento ruotato dell'angolo μ intorno all'asse $N - O$ è dato dalla colonna $\{v\}$. Secondo la notazione qui usata si ha dunque che $[T(\mathbf{e}, \mu)] = [R(\mathbf{e}, \mu)]^{-1}$ dove la matrice $[R(\mathbf{e}, \mu)]$ è

$$[R(\mathbf{e}, \mu)] = \begin{bmatrix} e_{xx} + C_\mu & e_{xy} + e_z S_\mu & e_{xz} - e_y S_\mu \\ e_{xy} - e_z S_\mu & e_{yy} + C_\mu & e_{yz} + e_x S_\mu \\ e_{xz} + e_y S_\mu & e_{yz} - e_x S_\mu & e_{zz} + C_\mu \end{bmatrix} \quad (3.17)$$

La (3.17) ha validità generale. In particolare, può essere utilizzata per riformulare le equazioni della cinematica del velivolo rigido.

3.2.3 Parametri di Eulero

Se si applica quest'ultimo risultato all'orientamento istantaneo di un aeromobile, la relazione tra le componenti di un vettore nella terna di assi velivolo e quelle corrispondenti nel riferimento Terra è data dalla

$$\begin{Bmatrix} v_{xB} \\ v_{yB} \\ v_{zB} \end{Bmatrix} = \begin{bmatrix} e_{xx} + C_\mu & e_{xy} + e_z S_\mu & e_{xz} - e_y S_\mu \\ e_{xy} - e_z S_\mu & e_{yy} + C_\mu & e_{yz} + e_x S_\mu \\ e_{xz} + e_y S_\mu & e_{yz} - e_x S_\mu & e_{zz} + C_\mu \end{bmatrix} \begin{Bmatrix} v_{xE} \\ v_{yE} \\ v_{zE} \end{Bmatrix} \quad (3.18)$$

nota come *Formula di Eulero*. La (3.18) e la sua inversa

$$\begin{pmatrix} v_{x_E} \\ v_{y_E} \\ v_{z_E} \end{pmatrix} = \begin{bmatrix} e_{xx} + C_\mu & e_{xy} - e_z S_\mu & e_{xz} + e_y S_\mu \\ e_{xy} + e_z S_\mu & e_{yy} + C_\mu & e_{yz} - e_x S_\mu \\ e_{xz} - e_y S_\mu & e_{yz} + e_x S_\mu & e_{zz} + C_\mu \end{bmatrix} \begin{pmatrix} v_{x_B} \\ v_{y_B} \\ v_{z_B} \end{pmatrix} \quad (3.19)$$

forniscono la parametrizzazione asse-angolo ed i quattro parametri (μ, e_x, e_y, e_z) prendono il nome di *parametri di Eulero*.

È possibile dimostrare (si rimanda il lettore ai riferimenti bibliografici [9] e [58] per un approfondimento) che vale la formula seguente:

$$\begin{pmatrix} \dot{\mu} \\ \dot{e}_x \\ \dot{e}_y \\ \dot{e}_z \end{pmatrix} = \begin{bmatrix} 2e_x & 2e_y & 2e_z \\ e'_{xx} + \frac{C_{\mu/2}}{S_{\mu/2}} & e'_{xy} - e_z & e'_{xz} + e_y \\ e'_{xy} + e_z & e'_{yy} + \frac{C_{\mu/2}}{S_{\mu/2}} & e'_{yz} - e_x \\ e'_{xz} - e_y & e'_{yz} + e_x & e'_{zz} + \frac{C_{\mu/2}}{S_{\mu/2}} \end{bmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (3.20)$$

con

$$e'_{ij} = -e_i e_j \frac{C_{\mu/2}}{S_{\mu/2}} \quad (3.21)$$

La legge (3.20) esprime i tassi di variazione temporale dei parametri di Eulero in termini delle componenti della velocità angolare istantanea $\boldsymbol{\Omega}$ dell'aeromobile nella terna di assi velivolo.

Come mostra la (3.20), anche la formulazione basata sui parametri di Eulero presenta una singolarità: quando $\mu = 0$ oppure $\mu = 2\pi$, cioè $S_{\mu/2} = 0$, l'integrazione nel tempo di tali equazioni è indeterminata. Questa singolarità è poi particolarmente problematica per lo studio del moto di un velivolo perché si presenterebbe quando l'asse longitudinale x_B , il cui orientamento esprime in pratica l'assetto della fusoliera, è orizzontale ed orientato verso nord (concorde ad x_E) oppure verso sud (in opposizione ad x_E). Il verificarsi di tali circostanze è molto più verosimile in condizioni di volo normale rispetto a quelle di volo acrobatico per le quali avviene il *gimbal lock* della parametrizzazione con angoli di Eulero.

Per i motivi su esposti la parametrizzazione asse-angolo non è mai direttamente utilizzata nella risoluzione delle equazioni del moto. Piuttosto, ed è il motivo per cui è stata introdotta, essa viene sfruttata attraverso un cambio di variabili per sviluppare una formulazione alternativa, sempre a quattro parametri, che risulta essere priva di singolarità. Le trasformazioni cinematiche che ne conseguono saranno basate su quattro quantità che prenderanno il nome di *componenti del quaternione dell'orientamento* del velivolo o anche di *parametri di Eulero-Rodrigues*.

3.3 Formulazione basata sul quaternione dell'orientamento

3.3.1 Parametri simmetrici di Eulero-Rodrigues

La parametrizzazione di Eulero-Rodrigues è legata a quella asse-angolo attraverso un semplice cambio di variabili. I quattro parametri utilizzati nella formulazione basata sull'asse e sull'angolo di Eulero vengono combinati tra di loro per ottenere quattro nuove grandezze che risultano molto più convenienti nella descrizione dell'orientamento. Tali grandezze si definiscono come

$$\begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} = \begin{pmatrix} \cos \frac{\mu}{2} \\ e_x \sin \frac{\mu}{2} \\ e_y \sin \frac{\mu}{2} \\ e_z \sin \frac{\mu}{2} \end{pmatrix} \quad (3.22)$$

e sono note come i quattro *parametri simmetrici di Eulero-Rodrigues* o come componenti del *quaternione di rotazione finita*. I parametri ora introdotti, così come in seguito le componenti dei quaternioni, saranno indicati come nella (3.22), facendo intendere che gli ultimi tre possono essere pensati come componenti di un vettore dello spazio euclideo tridimensionale, o alternativamente come (q_0, q_1, q_2, q_3) per comodità di notazione. Essi sono alla base di una parametrizzazione dell'orientamento e delle rotazioni di un corpo rigido che è diventata di gran lunga la più utilizzata nella simulazione del volo moderna.

L'introduzione alla comunità scientifica, alla fine dell'800, delle entità matematiche denominate *quaternioni* è dovuta al celebre matematico inglese William R. Hamilton [1]. Oggi le proprietà dell'algebra dei quaternioni vengono elegantemente utilizzate nella Fisica Matematica e nella Matematica Applicata per formulare problemi che vanno dalla rappresentazione delle rotazioni in spazi tridimensionali alla cinematica dei corpi rigidi, dall'elettromagnetismo alla relatività, dalla *computer vision* alla visualizzazione dei campi di moto in Fluidodinamica.

La parametrizzazione di Eulero-Rodrigues fu formulata ad uso dello studio della cinematica di un corpo rigido ben prima che venisse esposta la teoria dei quaternioni e della loro algebra non commutativa. Prima di Hamilton già Eulero e Gauss avevano formulato degli oggetti caratterizzati da quattro parametri (*Q-objects*). Inoltre, precedentemente nello stesso secolo, Olinde Rodrigues [2] aveva fornito delle regole di moltiplicazione tra entità a quattro parametri simili a quelle che sarebbero state inglobate in seguito nella teoria moderna dei quaternioni.

La fortuna dei quaternioni fu ostacolata dalla rivalità tra gli allievi di Hamilton ed altri due matematici dell'epoca, Gibbs ed Heaviside, ai quali si deve la nascita della moderna algebra e dell'analisi vettoriale [4]. Ma le qualità peculiari dell'algebra dei quaternioni, uniche nel loro genere perché proprie di enti a quattro parametri, oggi anche noti come *numeri ipercomplessi* (*hypercomplex numbers*), sono state riscoperte in tempi relativamente recenti (nel decennio 1960-70, [3]) per via dei vantaggi che, in diversi campi

applicativi, una formulazione basata sui quaternioni può comportare, sia in termini di esposizione formale che, come vedremo più avanti, in termini di efficienza computazionale. Inoltre, nei problemi di animazione tipici della *computer graphics* l'interpolazione degli orientamenti nello spazio di un oggetto in moto comporta un grado di regolarità non ottenibile con altri tipi di parametrizzazione, quale quella basata sugli angoli di Eulero.

In questa sede verranno presentati i concetti pertinenti alla rappresentazione dell'orientamento ed alla determinazione del moto di un velivolo.

3.3.2 Parametrizzazione di Eulero-Rodrigues

Le quattro grandezze definite con la (3.22) descrivono orientamenti a tre gradi di libertà. Dunque, sono anch'esse vincolate da una condizione che elimina la ridondanza di un parametro. La condizione si deduce semplicemente se si sommano i quadrati delle q_k

$$q_0^2 + q_x^2 + q_y^2 + q_z^2 = \cos^2 \frac{\mu}{2} + \left(e_x^2 + e_y^2 + e_z^2 \right) \sin^2 \frac{\mu}{2} \quad (3.23)$$

Se si tiene conto del vincolo (3.3) che impone al vettore di Eulero di avere modulo unitario, $\sum_{k=1}^3 e_k^2 = 1$, si ottiene banalmente che $\cos^2 \mu/2 + \sin^2 \mu/2 = 1$. Pertanto, i parametri di Eulero-Rodrigues, per come sono definiti, obbediscono implicitamente alla condizione vincolante

$$q_0^2 + q_x^2 + q_y^2 + q_z^2 = 1 \quad (3.24)$$

La trasformazione (3.18) corrispondente alla formula di Eulero può essere ora scritta in termini di seno e coseno della metà dell'angolo di Eulero applicando le formule trigonometriche:

$$\sin \mu = 2 \sin \frac{\mu}{2} \cos \frac{\mu}{2} \quad \cos \mu = \cos^2 \frac{\mu}{2} - \sin^2 \frac{\mu}{2} \quad 1 - \cos \mu = 2 \sin^2 \frac{\mu}{2}$$

ed in termini dei nuovi parametri q_k . Dopo qualche passaggio in cui si applica il cambio di variabili (3.22) e si riconosce che $e_{ij} = 2q_i q_j$ ed $S_{\mu/2}^2 = q_x^2 + q_y^2 + q_z^2$, si ottiene

$$\begin{Bmatrix} v_{x_B} \\ v_{y_B} \\ v_{z_B} \end{Bmatrix} = \begin{bmatrix} q_0^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y + q_0 q_z) & 2(q_z q_x - q_0 q_y) \\ 2(q_x q_y - q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z + q_0 q_x) \\ 2(q_z q_x + q_0 q_y) & 2(q_y q_z - q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \begin{Bmatrix} v_{x_E} \\ v_{y_E} \\ v_{z_E} \end{Bmatrix} \quad (3.25)$$

Si noti che nella trasformazione (3.25) compare la matrice di rotazione $[R(q_k)]$ ricavata a partire dalla $[R(e, \mu)]$ della parametrizzazione asse-angolo, che ha la forma

$$[R(q_k)] = \begin{bmatrix} \mathfrak{S}_q(+ - -) & \mathcal{D}_q^+(123) & \mathcal{D}_q^-(312) \\ \mathcal{D}_q^-(123) & \mathfrak{S}_q(- + -) & \mathcal{D}_q^+(231) \\ \mathcal{D}_q^+(312) & \mathcal{D}_q^-(231) & \mathfrak{S}_q(- - +) \end{bmatrix} \quad (3.26)$$

dove $\mathfrak{S}_q(\pm \pm \pm) = q_0^2 \pm q_1^2 \pm q_2^2 \pm q_3^2$, $\mathcal{D}_q^\pm(i j k) = 2q_i q_j \pm 2q_0 q_k$. La matrice inversa $[R(q_k)]^{-1}$, che è anche ovviamente la trasposta, si ottiene dalla (3.26) cambiando semplicemente il segno nelle funzioni di differenza $\mathcal{D}_q^\pm(i j k)$ e lasciando inalterate quelle

di somma \mathcal{S}_q

$$[R(q_k)]^{-1} = \begin{bmatrix} \mathcal{S}_q(+ - -) & \mathcal{D}_q^-(123) & \mathcal{D}_q^+(312) \\ \mathcal{D}_q^+(123) & \mathcal{S}_q(- + -) & \mathcal{D}_q^-(231) \\ \mathcal{D}_q^-(312) & \mathcal{D}_q^+(231) & \mathcal{S}_q(- - +) \end{bmatrix} \quad (3.27)$$

per ottenere la trasformazione inversa di quella data dalla (3.25)

$$\begin{Bmatrix} v_{x_E} \\ v_{y_E} \\ v_{z_E} \end{Bmatrix} = \begin{bmatrix} q_0^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_0 q_z) & 2(q_z q_x + q_0 q_y) \\ 2(q_x q_y + q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_0 q_x) \\ 2(q_z q_x - q_0 q_y) & 2(q_y q_z + q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \begin{Bmatrix} v_{x_B} \\ v_{y_B} \\ v_{z_B} \end{Bmatrix} \quad (3.28)$$

Le derivate nel tempo dei parametri di Eulero-Rodrigues sono legate a quelle dei parametri della formulazione asse-angolo. Semplicemente differenziando le (3.22) rispetto al tempo si ottiene

$$\begin{Bmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{Bmatrix} = \begin{Bmatrix} -\sin \frac{\mu}{2} \\ e_x \cos \frac{\mu}{2} \\ e_y \cos \frac{\mu}{2} \\ e_z \cos \frac{\mu}{2} \end{Bmatrix} \frac{\dot{\mu}}{2} + \begin{Bmatrix} 0 \\ \dot{e}_x \sin \frac{\mu}{2} \\ \dot{e}_y \sin \frac{\mu}{2} \\ \dot{e}_z \sin \frac{\mu}{2} \end{Bmatrix} \quad (3.29)$$

e tenendo conto delle (3.20) si arriva alla

$$\begin{Bmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} -e_x S_{\mu/2} & -e_y S_{\mu/2} & -e_z S_{\mu/2} \\ C_{\mu/2} & -e_z S_{\mu/2} & e_y S_{\mu/2} \\ e_z S_{\mu/2} & C_{\mu/2} & -e_x S_{\mu/2} \\ -e_y S_{\mu/2} & e_x S_{\mu/2} & C_{\mu/2} \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \quad (3.30)$$

nella quale si riconoscono come elementi della matrice, a meno del segno, proprio le componenti q_k del quaternione di rotazione. Dunque la (3.30) diventa

$$\begin{Bmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} -q_x & -q_y & -q_z \\ q_0 & -q_z & q_y \\ q_z & q_0 & -q_x \\ -q_y & q_x & q_0 \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \quad (3.31)$$

La (3.31) è una trasformazione equivalente alla (3.20) della parametrizzazione asse-angolo ed alla (2.24) della parametrizzazione classica con angoli di Eulero. Inoltre, per la linearità negli q_k degli elementi del prodotto a secondo membro della (3.31) si arriva a scrivere la

stessa trasformazione anche nella forma

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{pmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} \quad (3.32)$$

L'equazione (3.32) rappresenta la trasformazione cinematica ausiliaria in termini dei parametri di Eulero-Rodrigues. Come è evidente essa è una *parametrizzazione priva di singolarità*.

Dal punto di vista dell'esposizione delle equazioni del moto e della loro interpretazione fisica è evidente che gli angoli di Eulero, malgrado la singolarità che ne deriva, sono preferiti ai quaternioni, essendo facile immaginare gli angoli di rotta, di elevazione e di inclinazione laterale di un velivolo più che il coseno del semiangolo di Eulero ($q_0 = \cos \mu/2$) oppure il prodotto del seno di tale angolo per uno dei coseni direttori dell'asse di Eulero ($q_k = e_k \sin \mu/2$).

Dal punto di vista applicativo i vantaggi dell'utilizzo di una formulazione basata sul quaternione dell'orientamento sono chiari e si conoscono da almeno mezzo secolo. Inizialmente la parametrizzazione di Eulero-Rodrigues era giudicata più efficiente di quella basata sugli angoli di Eulero poiché la simulazione del volo ed altre discipline in cui era necessario trattare le rotazioni e l'integrazione delle equazioni del moto faceva largo uso di computer analogici. Tuttavia quelle stesse motivazioni sono ancora valide nel "mondo digitale" di oggi. È possibile dimostrare che 4 è il numero minimo di parametri da usare in una descrizione dell'orientamento e delle rotazioni di un corpo rigido affinché si abbiano delle relazioni cinematiche di trasformazione *lineari* come le (3.31) e (3.32) e prive di singolarità. Ciò fa intuire che gli errori associati all'integrazione numerica delle equazioni sono meglio controllabili con la parametrizzazione di Eulero-Rodrigues.

Il vantaggio più importante presentato dalla parametrizzazione di Eulero-Rodrigues risiede nella maggiore velocità di calcolo offerta da questa formulazione in confronto a tutte le altre oggi conosciute. Infatti la non linearità di equazioni cinematiche ausiliarie come le (3.20) e (2.24) con la presenza in esse di funzioni trigonometriche rende le altre parametrizzazioni molto più pesanti, dal punto di vista computazionale, di equazioni lineari come la (3.32). Inoltre il vantaggio di usare un quaternione per la parametrizzazione dell'orientamento viene esteso dal fatto che esso obbedisce all'*algebra dei quaternioni* introdotta da Hamilton. Esistono oggi degli algoritmi di calcolo particolarmente efficienti perché basati sulle regole di moltiplicazione fra quaternioni peculiari della loro algebra.

Per illustrare i vantaggi di tale approccio verranno presentati brevemente qui di seguito i quaternioni ed il loro formalismo a completamento dell'esposizione dei parametri simmetrici di Eulero-Rodrigues.

3.3.3 Cenni all'algebra dei quaternioni

Dal punto di vista matematico i quaternioni sono elementi di una certa algebra che gode di alcune proprietà particolari, cioè di un spazio vettoriale dotato di una particolare operazione di moltiplicazione.

È conveniente pensare ad un quaternione \mathfrak{q} come ad una collezione di quattro parametri reali dei quali il primo, q_0 , è da considerarsi la *parte scalare*, $\text{Scal}(\mathfrak{q})$, e l'insieme dei rimanenti tre, $q_1 \equiv q_x, q_2 \equiv q_y, q_3 \equiv q_z$, è da intendersi come la *parte vettoriale*, $\text{Vec}(\mathfrak{q})$. Gli scalari possono dunque essere pensati come quaternioni a parte vettoriale nulla mentre i vettori come dei quaternioni a parte scalare nulla. Ciò giustifica l'impiego delle seguenti notazioni, possibili per la rappresentazione formale di un generico quaternione,

$$\mathfrak{q} = (q_0, q_1, q_2, q_3) \quad (\text{quaterna di parametri}) \quad (3.33a)$$

$$= (q_0, \mathbf{q}) \quad (\text{scalare, vettore}) \quad (3.33b)$$

$$= q_0 + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} \quad (\text{scalare 'più' vettore}) \quad (3.33c)$$

La (3.33c) si presta ad un'interpretazione dei quaternioni come una generalizzazione dei numeri complessi, detti *numeri ipercomplessi*. Le componenti della parte vettoriale si possono pensare come i coefficienti di tre parti immaginarie differenti. La parte vettoriale, variabile in uno spazio a tre componenti reali, può essere dunque vista come parte immaginaria generalizzata del quaternione. Ciò fa intendere che i quaternioni avranno proprietà in parte simili a quelle dei vettori ed in parte simili a quelle dei numeri complessi.

I particolari quaternioni

$$\mathbf{i} = (0, \mathbf{i}) \quad \mathbf{j} = (0, \mathbf{j}) \quad \mathbf{k} = (0, \mathbf{k}) \quad (3.34)$$

corrispondono alle unità immaginarie generalizzate, dove \mathbf{i} , \mathbf{j} e \mathbf{k} sono i classici versori dello spazio euclideo a tre dimensioni. Il quaternione $\mathbb{1} = (1, \mathbf{0}) \equiv 1$ rappresenta inoltre l'unità scalare generalizzata ed infine il quaternione $\mathbb{0} = (0, \mathbf{0}) \equiv 0$ rappresenta il quaternione nullo. Se ne deduce un'altra possibile rappresentazione di un quaternione che è quella di *numero ipercomplesso*

$$\mathfrak{q} = q_0 \mathbb{1} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} \quad (3.35)$$

La (3.35) viene comunemente utilizzata nelle trattazioni di interesse squisitamente matematico e per i nostri scopi può essere banalmente sostituita con la (3.33c).

La moltiplicazione tra quaternioni, che pure vengono indicati con la notazione speciale “ \mathfrak{q} ”, non corrisponde a nessuno dei prodotti che si possono definire tra gli elementi di uno spazio euclideo tridimensionale. Anch'essa ha bisogno di una notazione apposita e nel seguito sarà indicata con il simbolo “ \boxtimes ”. Le seguenti regole di moltiplicazione definiscono i prodotti tra le unità immaginarie generalizzate:

$$\begin{array}{lll} \mathbf{i} \boxtimes \mathbf{i} = -1 & \mathbf{i} \boxtimes \mathbf{j} = \mathbf{k} & \mathbf{i} \boxtimes \mathbf{k} = -\mathbf{j} \\ \mathbf{j} \boxtimes \mathbf{i} = -\mathbf{k} & \mathbf{j} \boxtimes \mathbf{j} = -1 & \mathbf{j} \boxtimes \mathbf{k} = \mathbf{i} \\ \mathbf{k} \boxtimes \mathbf{i} = \mathbf{j} & \mathbf{k} \boxtimes \mathbf{j} = -\mathbf{i} & \mathbf{k} \boxtimes \mathbf{k} = -1 \end{array} \quad (3.36)$$

Nei prodotti incrociati delle (3.36) si riconosce una somiglianza con i prodotti vettoriali tra versori (ad esempio: $\mathbf{i} \boxtimes \mathbf{j} = \mathbf{k}$) mentre nei termini diagonali si ha una somiglianza con il prodotto dell'unità immaginaria per se stessa (ad esempio: $\mathbf{i} \boxtimes \mathbf{i} = -1$).

Il prodotto tra quaternioni gode della proprietà distributiva. Ne consegue che il

prodotto tra due quaternioni generici \mathbb{p} e \mathbb{q} , tenendo conto delle (3.36), è:

$$\begin{aligned} \mathbb{p} \boxtimes \mathbb{q} &= (p_0 + p_x \mathbf{i} + p_y \mathbf{j} + p_z \mathbf{k}) \boxtimes (q_0 + q_x \mathbf{i} + q_y \mathbf{j} + q_z \mathbf{k}) \\ &= (p_0 q_0 - p_x q_x - p_y q_y - p_z q_z) \\ &\quad + (p_0 q_x + p_x q_0 + p_y q_z - p_z q_y) \mathbf{i} \\ &\quad + (p_0 q_y - p_x q_z + p_y q_0 + p_z q_x) \mathbf{j} \\ &\quad + (p_0 q_z + p_x q_y - p_y q_x + p_z q_0) \mathbf{k} \end{aligned} \quad (3.37)$$

Come è facile verificare, la regola (3.37) si può esprimere elegantemente con la notazione (*scalare*), (*vettore*)), cioè

$$\mathbb{p} \boxtimes \mathbb{q} = \left(p_0 q_0 - \mathbf{p} \cdot \mathbf{q}, q_0 \mathbf{p} + p_0 \mathbf{q} + \mathbf{p} \wedge \mathbf{q} \right) \quad (3.38)$$

Un'altra efficace espressione della regola del prodotto fra quaternioni è data dalla formula matriciale:

$$\mathbb{p} \boxtimes \mathbb{q} = \left[\begin{array}{c|c} q_0 & -\{q\}^T \\ \{q\} & q_0 [I]_3 - [\tilde{q}] \end{array} \right] \left\{ \begin{array}{c} p_0 \\ \{p\} \end{array} \right\} \equiv [\hat{\mathbb{q}}] \{\mathbb{p}\} \quad (3.39)$$

dove $\{p\}$ e $\{q\}$ sono le matrici colonna aventi per elementi le componenti delle parti vettoriali di \mathbb{p} e \mathbb{q} , $[I]_3$ è la matrice identica 3×3 , $[\tilde{q}]$ è la matrice ottenuta applicando l'operatore tilde all'array $\{q\}$ ed infine $\{\mathbb{p}\}$ è la matrice colonna a 4 elementi rappresentativa del corrispondente quaternione.

Come si può dedurre facilmente dalla (3.38), essendo il prodotto vettoriale $\mathbf{p} \wedge \mathbf{q}$ non commutativo, *la moltiplicazione tra quaternioni non è commutativa*. Si ha infatti, dalla (3.39) che il prodotto a termini commutati $\mathbb{q} \boxtimes \mathbb{p}$ è:

$$\mathbb{q} \boxtimes \mathbb{p} = \left[\begin{array}{c|c} q_0 & -\{q\}^T \\ \{q\} & q_0 [I]_3 + [\tilde{q}] \end{array} \right] \left\{ \begin{array}{c} p_0 \\ \{p\} \end{array} \right\} \equiv [\mathbb{q}] \{\mathbb{p}\} \quad (3.40)$$

Si noti che la matrice $[\mathbb{q}]$ definita dalla (3.40) differisce dalla $[\hat{\mathbb{q}}]$ definita dalla (3.39) per il segno che compare davanti agli elementi della $[\tilde{q}]$. Le espressioni esplicite di matrici come la $[\mathbb{q}]$ e la $[\hat{\mathbb{q}}]$ sono le seguenti:

$$[\mathbb{q}] = \left[\begin{array}{c|ccc} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{array} \right], \quad [\hat{\mathbb{q}}] = \left[\begin{array}{c|ccc} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{array} \right] \quad (3.41)$$

La regola di prodotto espressa a mezzo delle relazioni matriciali (3.39)-(3.40) permette anche di asserire che

$$\mathbb{p} \boxtimes \mathbb{q} = [\mathbb{p}] \{\mathbb{q}\}, \quad \mathbb{q} \boxtimes \mathbb{p} = [\hat{\mathbb{p}}] \{\mathbb{q}\} \quad (3.42)$$

In definitiva, il prodotto fra due quaternioni viene rappresentato anche come prodotto di una matrice 4×4 , i cui elementi sono funzioni lineari delle componenti di uno solo dei due quaternioni, e di una matrice colonna contenente le componenti dell'altro quaternione.

Oltre alla regola di moltiplicazione è notevole la definizione di *aggiunto* o *coniugato* di un quaternione \mathbb{q} , indicato con \mathbb{q}^* e definito come segue:

$$\text{Adj}(\mathbb{q}) \equiv \mathbb{q}^* = (q_0, -\mathbf{q}) \quad (3.43)$$

Si noti la somiglianza con la definizione di complesso coniugato di un numero $z = a + ib$. Una definizione conseguente è quella di *lunghezza* o *norma* di un quaternione

$$|\mathbb{q}| = \sqrt{\mathbb{q}^* \boxtimes \mathbb{q}} \quad (3.44)$$

Ha senso utilizzare la funzione radice quadrata nella definizione (3.44) poiché, dalle proprietà del prodotto \boxtimes discende che la parte vettoriale del prodotto di un quaternione per il suo coniugato è nulla, cioè $\text{Vec}(\mathbb{q}^* \boxtimes \mathbb{q}) = \mathbf{0}$. Inoltre, si deduce che la norma può essere espressa come

$$|\mathbb{q}| = \sqrt{q_0^2 + \mathbf{q} \cdot \mathbf{q}} = \sqrt{\sum_{k=0}^3 q_k^2} \quad (3.45)$$

Una proprietà della norma è che la norma di un prodotto è pari al prodotto delle norme: $|\mathbb{p} \boxtimes \mathbb{q}| = |\mathbb{p}| |\mathbb{q}|$.

Dalle definizioni (3.40) e (3.42) di prodotto di quaternioni in termini matriciali si ricavano le proprietà delle matrici $[\mathbb{q}]$ e $[\mathbb{p}]$

$$[\mathbb{q}][\mathbb{q}]^T = |\mathbb{q}|^2 [I]_4, \quad [\mathbb{q}][\widehat{\mathbb{p}}] = [\widehat{\mathbb{p}}][\mathbb{q}] \quad (3.46)$$

La seconda delle (3.46) corrisponde ad una proprietà di commutabilità del prodotto di queste particolari matrici ottenute applicando gli operatori $[\]$ e $[\widehat{\ }]$ a due quaternioni. In particolare, tali operatori quando applicati all'aggiunto di un quaternione corrispondono alla trasposizione:

$$[\mathbb{q}^*] = [\mathbb{q}]^T \quad \text{e} \quad [\widehat{\mathbb{q}^*}] = [\widehat{\mathbb{q}}]^T$$

Una proprietà dei quaternioni che discende dalle regole fin qui esposte e che rappresenta una peculiarità della loro algebra è quella di poter definire l'*inverso* (o *reciproco*) di un quaternion non nullo come

$$\mathbb{q}^{-1} = \frac{1}{|\mathbb{q}|^2} \mathbb{q}^* \quad (3.47)$$

Si noti che una tale operazione, che restituisce un elemento dell'algebra tale che il prodotto $\mathbb{q} \boxtimes \mathbb{q}^{-1} = \mathbb{1} = 1$, non può essere definita per i classici vettori di uno spazio tridimensionale.

Dalla (3.47) segue la definizione di *quaternione unitario* \mathbb{u} come un quaternione che ha norma unitaria, per il quale quindi l'operazione di inversione coincide con l'operazione di coniugio: $\mathbb{u}^{-1} = \mathbb{u}^*$.

3.3.4 Rotazioni finite e quaternioni

Le definizioni esposte finora, in particolare quella di prodotto e di quaternione unitario, permettono una utile interpretazione delle rotazioni di un corpo rigido. Detto $\mathbb{v} = (0, \mathbf{v})$ un quaternione a parte scalare nulla con il quale si vuole identificare il generico vettore \mathbf{v} , una moltiplicazione a sinistra di \mathbb{v} per un quaternione unitario \mathbb{u} ed a destra per il coniugato

\mathfrak{u}^* fornisce il quaternione

$$\mathfrak{v}' = \mathfrak{u} \boxtimes (\mathfrak{v} \boxtimes \mathfrak{u}^*) \quad (3.48)$$

avente anch'esso parte scalare nulla, $\text{Scal}(\mathfrak{v}') = 0$. Orbene, è possibile dimostrare che la parte vettoriale del risultato dell'operazione (3.48) è un vettore \mathbf{v}' da interpretarsi come il risultato di una rotazione del vettore \mathbf{v} nello spazio euclideo tridimensionale (si osservi che $|\mathfrak{v}'| = |\mathfrak{u}| |\mathfrak{v}| |\mathfrak{u}| = |\mathfrak{v}| = \sqrt{\mathbf{v} \cdot \mathbf{v}}$). La rotazione che trasforma \mathbf{v} in \mathbf{v}' dipende dall'entità del quaternione unitario \mathfrak{u} . Pertanto, *un quaternione unitario può essere interpretato come un ente attraverso il quale si possono operare delle trasformazioni di rotazione di un vettore*, se quest'ultimo viene espresso come quaternione a parte scalare nulla.

Come mostrato in [3], l'applicazione della regola (3.38) al generico triplo prodotto di quaternioni $\mathfrak{p} \boxtimes (\mathfrak{q} \boxtimes \mathfrak{r})$, specializzata al caso particolare dato dalla (3.48), porta alla seguente espressione notevole:

$$\mathbf{v}' = (u_0^2 - \mathbf{u} \cdot \mathbf{u}) \mathbf{v} + (2u_0) \mathbf{u} \wedge \mathbf{v} + 2(\mathbf{v} \cdot \mathbf{u}) \mathbf{u} \quad (3.49)$$

Si noti la similarità con l'espressione (3.8) a suo tempo ricavata per le rotazioni finite intorno ad un versore \mathbf{e} che qui riportiamo per chiarezza

$$\mathbf{v}' = \cos \mu \mathbf{v} + \sin \mu \mathbf{e} \wedge \mathbf{v} + (1 - \cos \mu) (\mathbf{v} \cdot \mathbf{e}) \mathbf{e} \quad (3.8)$$

dove μ è l'angolo di rotazione intorno all'asse di Eulero di versore \mathbf{e} .

Dal momento che le (3.48)-(3.49) rappresentano una rotazione, e tenendo presente il teorema di Eulero sulle rotazioni finite, a questo punto si può prendere in considerazione il moto di un corpo rigido incernierato in un punto fisso. Tale moto sarà puramente rotazionale intorno al punto di cerniera. In particolare, il moto del corpo si identificherà con il moto di un sistema di riferimento \mathcal{T}_m ad esso solidale, con origine nel punto di cerniera, rispetto ad un riferimento fisso \mathcal{T}_f avente la stessa origine. Immaginando di congelare il moto in un generico istante t , si ha che un punto solidale al corpo rigido, identificato da \mathbf{v} nello spazio di coordinate \mathcal{T}_m , è il trasformato del punto associato al vettore \mathbf{v}' dello spazio fisso \mathcal{T}_f . Il vettore \mathbf{v} si può quindi esprimere come il vettore colonna $\{x\}_m$ delle coordinate del generico punto appartenente al corpo rigido espresse nel riferimento mobile. Per la (3.49) il vettore \mathbf{v}' si può interpretare come il vettore $\{x\}_f$ delle coordinate dello stesso punto materiale ma espresse stavolta nel riferimento fisso. In altre parole, la (3.49) è una formulazione in termini quaternionali di una trasformazione di rotazione del tipo

$$\{x\}_f = [T]_{f_m} \{x\}_m$$

Data questa interpretazione, vale la pena ribattezzare i quaternioni corrispondenti a \mathfrak{v} e \mathfrak{v}' , rispettivamente, con $\mathfrak{x} = (0, \mathbf{x})$ e $\mathfrak{x}' = (0, \mathbf{x}')$. È inoltre pensabile differenziare la (3.48) rispetto al tempo ottenendo

$$\frac{d\mathfrak{x}'}{dt} = \frac{d\mathfrak{u}}{dt} \boxtimes (\mathfrak{x} \boxtimes \mathfrak{u}^*) + \mathfrak{u} \boxtimes \left(\mathfrak{x} \boxtimes \frac{d\mathfrak{u}^*}{dt} \right) \quad (3.50)$$

La (3.50) si ottiene applicando la regola di derivazione del prodotto, valida anche per i quaternioni, tenendo conto del fatto che $\dot{\mathfrak{x}} = 0$, per definizione di corpo rigido. Il quaternione $d\mathfrak{x}'/dt$, che si dimostra avere anch'esso parte scalare nulla, rappresenta la

velocità istantanea del punto materiale considerato nel riferimento fisso (velocità assoluta)

$$\frac{d\mathbf{x}'}{dt} = \left(0, \frac{d\mathbf{x}'}{dt} \right) \quad (3.51)$$

Se a secondo membro della (3.50) si esprime \mathbf{x} in termini di \mathbf{x}' invertendo la (3.48)

$$\mathbf{x} = \mathfrak{u}^* \boxtimes (\mathbf{x}' \boxtimes \mathfrak{u}) \quad (3.52)$$

si ottiene

$$\frac{d\mathbf{x}'}{dt} = \frac{d\mathfrak{u}}{dt} \boxtimes (\mathfrak{u}^* \boxtimes \mathbf{x}') + \mathbf{x}' \boxtimes \left(\mathfrak{u} \boxtimes \frac{d\mathfrak{u}^*}{dt} \right) \quad (3.53)$$

Si può osservare a questo punto che nella (3.53), come illustrato in [5], essendo

$$\mathfrak{u}^* \boxtimes \mathfrak{u} = \mathbb{1} = (1, \mathbf{0})$$

si deduce che i quaternioni $(d\mathfrak{u}/dt) \boxtimes \mathfrak{u}^*$ ed $\mathfrak{u} \boxtimes (d\mathfrak{u}^*/dt)$ hanno entrambi parte scalare nulla e parte vettoriale opposta. Cioè, detto

$$\mathbf{w}' = \text{Vec}((d\mathfrak{u}/dt) \boxtimes \mathfrak{u}^*)$$

si ha:

$$\left(\frac{d}{dt} \mathfrak{u} \right) \boxtimes \mathfrak{u}^* = (0, \mathbf{w}')$$

ed

$$\mathfrak{u} \boxtimes \left(\frac{d}{dt} \mathfrak{u}^* \right) = (0, -\mathbf{w}')$$

Pertanto, dalla (3.53) si conclude che il quaternionone $d\mathbf{x}'/dt$ ha, come ci si aspetta, parte scalare nulla e parte vettoriale pari a $2\mathbf{w}' \wedge \mathbf{x}'$ (si rimanda al riferimento [5] per approfondimenti). In definitiva si può scrivere:

$$\frac{d\mathbf{x}'}{dt} = \left(0, \frac{d\mathbf{x}'}{dt} \right) = (0, 2\mathbf{w}' \wedge \mathbf{x}') \quad (3.54)$$

Inoltre, se si pone $2\mathbf{w}' = \boldsymbol{\Omega}'_m$ (velocità angolare istantanea del corpo rigido rappresentata nello spazio fisso) e si definisce il quaternionone

$$\mathbb{W}' = (0, \boldsymbol{\Omega}'_m) \quad (3.55)$$

è possibile dimostrare il risultato seguente:

$$\mathbb{W}' = 2 \frac{d\mathfrak{u}}{dt} \boxtimes \mathfrak{u}^* \quad \text{equivalente a} \quad \left\{ \begin{array}{c} 0 \\ \{\boldsymbol{\Omega}'_m\}_f \end{array} \right\} = 2 [\widehat{\mathbb{W}'}]^T \left\{ \begin{array}{c} \dot{u}_0 \\ \{\dot{u}\} \end{array} \right\} \quad (3.56)$$

nel quale si riconosce il vettore colonna $\{\boldsymbol{\Omega}'_m\}_f$, cioè la rappresentazione nel riferimento fisso della velocità angolare istantanea del riferimento mobile.

La (3.56) ha per inversa una relazione che fornisce la derivata temporale $d\mathfrak{u}/dt$ dato il quaternionone unitario \mathfrak{u} ed il vettore $\boldsymbol{\Omega}'_m$. Si può far vedere che tale relazione è data dalla

$$\frac{d\mathfrak{u}}{dt} = \frac{1}{2} \mathbb{W}' \boxtimes \mathfrak{u} \quad (3.57)$$

La (3.57) può essere manipolata in modo da ottenere la relazione matriciale (3.32) che qui riportiamo per comodità

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{pmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} \quad (3.32)$$

in cui avevamo chiamato (q_0, q_x, q_y, q_z) le componenti del quaternione normalizzato (ovvero unitario) dell'orientamento del velivolo. Ciò lascia dunque intendere che il quaternione unitario \mathfrak{u} che compare nella relazione differenziale (3.57) ed in quelle precedenti è proprio il quaternione dell'orientamento del velivolo ($\mathfrak{u} \equiv \mathfrak{q}$). In questa sede sarebbe troppo lungo dimostrarlo, ed anche i risultati formali che riportiamo di seguito andrebbero ricavati rigorosamente; pertanto si rimanda il lettore alla letteratura specialistica per approfondimenti.

La relazione (3.32), con $\mathfrak{u} = \mathfrak{q}$, si ottiene dalla (3.57) se si esprime la velocità angolare nel riferimento mobile anziché in quello fisso, cioè come vettore $\boldsymbol{\Omega}_m$, ponendo

$$\mathbb{W} = (0, \boldsymbol{\Omega}_m) = \begin{pmatrix} 0 \\ \{\Omega_m\}_m \end{pmatrix} = \begin{pmatrix} 0 \\ p \\ q \\ r \end{pmatrix} \quad (3.58)$$

ed applicando le formule di trasformazione corrispondenti. Si dimostra che la (3.56) è equivalente alla relazione matriciale:

$$\begin{pmatrix} 0 \\ \{\Omega_m\}_m \end{pmatrix} = 2[\mathbb{W}]^T \begin{pmatrix} \dot{u}_0 \\ \dot{\mathbf{u}} \end{pmatrix} \quad (3.59)$$

(si noti la presenza di $[\mathbb{W}]^T$ anziché di $[\widehat{\mathbb{W}}']^T$) la cui inversa è

$$\frac{d\mathfrak{u}}{dt} = \frac{1}{2} \mathfrak{u} \boxtimes \mathbb{W} \quad (3.60a)$$

equivalente a

$$\begin{pmatrix} \dot{u}_0 \\ \dot{\mathbf{u}} \end{pmatrix} = \frac{1}{2} [\mathfrak{u}] \begin{pmatrix} 0 \\ \{\Omega_m\}_m \end{pmatrix} \quad (3.60b)$$

Queste ultime formule costituiscono il risultato fondamentale che vuole essere evidenziato in questo paragrafo. Gli elementi della matrice $[\mathfrak{u}]$ che compare nella (3.60b) sono funzioni *lineari* delle componenti di \mathfrak{u} :

$$[\mathfrak{u}] = \begin{bmatrix} u_0 & -u_1 & -u_2 & -u_3 \\ u_1 & u_0 & -u_3 & u_2 \\ u_2 & u_3 & u_0 & -u_1 \\ u_3 & -u_2 & u_1 & u_0 \end{bmatrix} \quad (3.41)$$

Ciò si vede facilmente applicando la formula (3.40), riscrivendo la (3.60a) come

$$\begin{pmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{pmatrix} = \frac{1}{2} \begin{bmatrix} u_0 & -u_1 & -u_2 & -u_3 \\ u_1 & u_0 & -u_3 & u_2 \\ u_2 & u_3 & u_0 & -u_1 \\ u_3 & -u_2 & u_1 & u_0 \end{bmatrix} \begin{pmatrix} 0 \\ p \\ q \\ r \end{pmatrix}$$

per poi sfruttare la linearità dei termini della matrice e riottenere dalla (3.60b), la formula di trasformazione (3.32).

In definitiva, abbiamo delineato il collegamento diretto che esiste tra le relazioni cinematiche (3.32) associate al moto di un corpo rigido, il cui orientamento istantaneo è espresso in termini di un quaternione unitario, e le proprietà del prodotto dell'algebra dei quaternioni. In particolare, in termini quaternionali tali relazioni ausiliarie si identificano con la (3.60a) che esprime il prodotto del quaternione dell'orientamento ed il quaternione a parte scalare nulla costruito con le componenti della velocità angolare istantanea.

3.3.5 Dal quaternione dell'orientamento agli angoli di Eulero

Ai fini dell'integrazione numerica delle equazioni del moto, se la formulazione scelta è quella basata sul quaternione dell'orientamento (le cui componenti sono i parametri di Eulero-Rodrigues), la relazione differenziale (3.32) o equivalentemente la (3.60a) permettono l'aggiornamento della posizione e dell'orientamento del velivolo ad ogni passo e dunque la determinazione del moto.

Per quanto concerne la rappresentazione dell'orientamento, per la necessità, peraltro inevitabile, di dover conoscere i valori degli stessi angoli di Eulero istante per istante, resta da specificare ancora una relazione di trasformazione importante. È necessario cioè esprimere gli angoli di Eulero corrispondenti ad un dato quaternione dell'orientamento e, viceversa, noti gli angoli di Eulero, per un dato orientamento trovarne il quaternione corrispondente. La prima relazione è necessaria, ad esempio, per la rappresentazione grafica del moto. La sua inversa, invece, è richiesta nelle situazioni in cui le condizioni iniziali o alcuni vincoli particolari sono imposti in termini di angoli di *azimuth*, di *elevation* e di inclinazione laterale e se ne devono ricavare le corrispondenti relazioni in termini delle componenti del quaternione.

Come abbiamo visto nei paragrafi precedenti l'introduzione della formulazione basata sul quaternione dell'orientamento nasce dalla particolare forma delle relazioni cinematiche differenziali a cui obbediscono le variazioni dell'orientamento del velivolo. Le riportiamo qui per comodità:

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \quad (2.22)$$

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{bmatrix} 1 & \frac{\sin \phi \sin \theta}{\cos \theta} & \frac{\cos \phi \sin \theta}{\cos \theta} \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (2.24)$$

L'equazione (2.24) è l'inversa della (2.22) e fornisce i ratei di variazione degli angoli di Eulero del velivolo una volta note le componenti della sua velocità angolare istantanea. La matrice a secondo membro della (2.24) diviene singolare quando l'angolo di elevazione θ tende ad assumere il valore 90 deg. Nelle evoluzioni in cui il moto del velivolo porta al verificarsi di questa circostanza la relazione cinematica (2.24) risulta inapplicabile e la determinazione dell'orientamento dell'aeromobile negli istanti successivi necessita di un trattamento speciale.

Le componenti (q_0, q_x, q_y, q_z) del quaternione dell'orientamento sono collegate agli angoli di Eulero del velivolo tramite le relazioni seguenti:

$$\begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} = \begin{pmatrix} \cos \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2} \\ \sin \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} - \cos \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2} \\ \cos \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2} \\ \cos \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2} - \sin \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2} \end{pmatrix} \quad (3.61)$$

Come è facile verificare, queste relazioni confermano che i parametri di Eulero-Rodrigues sono le componenti di un quaternione unitario:

$$q_0^2 + q_x^2 + q_y^2 + q_z^2 = 1 \quad (3.24)$$

Inoltre, è possibile dimostrare che le (3.61) discendono direttamente dalle definizioni (3.22) se si esprimono le componenti (e_x, e_y, e_z) dell'asse di Eulero ed i termini in cui compare l'angolo di Eulero μ in funzione degli angoli di Eulero ϕ , θ e ψ (si veda, ad esempio, il testo di Phillips [9]). Le relazioni inverse delle (3.61) risultano essere le seguenti:

$$\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} = \begin{pmatrix} \operatorname{atan2}\left[2(q_0 q_x + q_y q_z), (q_0^2 - q_x^2 - q_y^2 + q_z^2)\right] \\ \operatorname{asin}\left[2(q_0 q_y - q_x q_z)\right] \\ \operatorname{atan2}\left[2(q_0 q_z + q_x q_y), (q_0^2 + q_x^2 - q_y^2 - q_z^2)\right] \end{pmatrix} \quad (3.62)$$

Le relazioni (3.62) permettono di ricostruire la terna di valori degli angoli di Eulero a partire dalla quaterna di valori delle componenti del quaternione unitario dell'orientamento. La funzione $\operatorname{atan2}(\cdot, \cdot)$ che compare nelle (3.62) è una funzione arcotangente a due argomenti come quella descritta dagli standard dei linguaggi di programmazione moderni. I due argomenti sono il numeratore ed il denominatore di un rapporto di cui si vuole l'arcotangente. Essendo gli angoli ϕ e ψ definiti in intervalli di ampiezza 2π

la funzione atan2 è necessaria nella prima ed ultima delle relazioni (3.62) poiché essa ritorna un risultato angolare nel quadrante opportuno a seconda dei segni di ciascuno dei suoi argomenti. Si osservi che per la determinazione dell'angolo θ non è richiesta una funzione a due argomenti dato che esso è definito nell'intervallo che va da $-\frac{\pi}{2}$ a $\frac{\pi}{2}$.

Si osservi che la formula (3.62) non ha validità generale. Ad esempio, essa non è applicabile nei casi in cui l'orientamento del velivolo dà luogo al *gimbal lock*. Ciò accade nelle situazioni di volo per le quali $\theta \rightarrow \frac{\pi}{2}$ (si veda la figura 2.4 a pagina 13) oppure per $\theta \rightarrow -\frac{\pi}{2}$.

D'altra parte, se si considerano le relazioni (3.61), dal punto di vista squisitamente matematico, è lecito poter verificare quale sia il risultato a primo membro se nel secondo membro si sostituisce a θ il valore $\frac{\pi}{2}$ e sia a ϕ che a ψ il valore 0. In tal caso si ottiene la quaterna di valori: (0,71, 0, 0,71, 0). Resta ben inteso che *ciò non significa che la terna degli angoli di Eulero è (0, $\frac{\pi}{2}$, 0)*. Allo stesso tempo, questa terna di valori dà luogo attraverso le (3.61) ad un quaternione perfettamente lecito. Quando questa quaterna di parametri viene sostituita nelle relazioni inverse (3.62) si ottiene

$$\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} = \begin{pmatrix} \text{atan2}(0,00, 0,00) \\ \text{asin}(1,00) \\ \text{atan2}(0,00, 0,00) \end{pmatrix} \quad (3.63)$$

Quest'ultimo passaggio permette di affermare che l'elevazione θ corrispondente è pari ad un angolo di 90 deg. Al contrario, in questa particolare situazione i valori degli angoli ϕ e ψ restano indeterminati. In altre parole, questo esempio conferma che quando l'angolo $\theta \rightarrow \pm\frac{\pi}{2}$ i rimanenti due angoli di Eulero non sono univocamente determinati.

Per avere una conferma di quest'ultima affermazione basta sostituire nelle relazioni (3.61) una diversa terna di valori, ad esempio, $\frac{\pi}{2}$ a θ e, diversamente dal caso precedente, il valore $\frac{\pi}{4}$ a ϕ ed il valore $\frac{\pi}{4}$ a ψ . Ciò darà luogo ancora una volta al quaternion (0,71, 0, 0,71, 0).

Quando l'asse velivolo x_B forma un angolo di ± 90 deg con il piano orizzontale, di fatto, l'angolo di *heading* ψ può assumere un valore del tutto arbitrario; e ciò non importa, dal momento che l'orientamento della fusoliera rimane completamente specificato dalla sola circostanza che essa è disposta in verticale. Dunque, *specificare un angolo di prua ψ quando l'angolo di elevazione è pari a ± 90 deg è come specificare la longitudine geografica di un punto situato al polo Nord o al polo Sud della superficie terrestre* [9]. Infatti, quando la latitudine è ± 90 deg non è necessario specificare anche la longitudine perché ciascuno di questi due valori descrive, da solo, un punto sulla superficie terrestre.

A causa del *gimbal lock* le trasformazioni (3.62) non possono più essere utilizzate in generale. Qualora l'angolo di elevazione θ attraversa i valori $\pm\frac{\pi}{2}$ sia l'angolo di *heading* che l'angolo di *bank* dovranno subire necessariamente un salto di π . Inoltre, per un angolo θ di $\pm\frac{\pi}{2}$ l'angolo ϕ può solo essere espresso come funzione del valore arbitrario di ψ .

if $(q_0 q_y - q_x q_z) = 0,50$ **then**

$$\begin{cases} \phi \\ \theta \\ \psi \end{cases} := \begin{cases} 2 \operatorname{asin} \left(\frac{q_x}{\cos(\pi/4)} \right) + \psi \\ \frac{\pi}{2} \\ \text{arbitrary} \end{cases}$$

else if $(q_0 q_y - q_x q_z) = -0,50$ **then**

$$\begin{cases} \phi \\ \theta \\ \psi \end{cases} := \begin{cases} 2 \operatorname{asin} \left(\frac{q_x}{\cos(\pi/4)} \right) - \psi \\ -\frac{\pi}{2} \\ \text{arbitrary} \end{cases}$$

else

$$\begin{cases} \phi \\ \theta \\ \psi \end{cases} := \begin{cases} \operatorname{atan2} \left[2 (q_0 q_x + q_y q_z), (q_0^2 - q_x^2 - q_y^2 + q_z^2) \right] \\ \operatorname{asin} \left[2 (q_0 q_y - q_x q_z) \right] \\ \operatorname{atan2} \left[2 (q_0 q_z + q_x q_y), (q_0^2 + q_x^2 - q_y^2 - q_z^2) \right] \end{cases}$$

end if

Figura 3.5 Algoritmo per la determinazione degli angoli di Eulero, note le componenti del quaternione dell'orientamento.

Dalle (3.61), i parametri di Euler-Rodrigues nelle condizioni di *gimbal lock* diventano

$$\begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} = \cos \frac{\pi}{4} \begin{pmatrix} \cos \frac{\phi}{2} \cos \frac{\psi}{2} \pm \sin \frac{\phi}{2} \sin \frac{\psi}{2} \\ \sin \frac{\phi}{2} \cos \frac{\psi}{2} \mp \cos \frac{\phi}{2} \sin \frac{\psi}{2} \\ \pm \cos \frac{\phi}{2} \cos \frac{\psi}{2} + \sin \frac{\phi}{2} \sin \frac{\psi}{2} \\ \cos \frac{\phi}{2} \sin \frac{\psi}{2} \mp \sin \frac{\phi}{2} \cos \frac{\psi}{2} \end{pmatrix} = \cos \frac{\pi}{4} \begin{pmatrix} \cos \left(\frac{\phi}{2} \mp \frac{\psi}{2} \right) \\ \sin \left(\frac{\phi}{2} \mp \frac{\psi}{2} \right) \\ \pm \cos \left(\frac{\phi}{2} \mp \frac{\psi}{2} \right) \\ \mp \sin \left(\frac{\phi}{2} \mp \frac{\psi}{2} \right) \end{pmatrix} \quad (3.64)$$

per $\theta = \pm \frac{\pi}{2}$. Queste ultime espressioni, valide in situazioni di *gimbal lock*, permettono di scrivere:

$$\begin{cases} \phi \\ \theta \\ \psi \end{cases}_{\text{gimbal lock}} = \begin{cases} 2 \operatorname{asin} \left(\frac{q_x}{\cos(\pi/4)} \right) \pm \psi \\ \pm \frac{\pi}{2} \\ \psi \text{ (valore arbitrario)} \end{cases} \quad (3.65)$$

Combinando le equazioni (3.62) e (3.65), è possibile formulare un algoritmo generale di assegnazione degli angoli di Eulero, note le componenti del quaternione dell'orientamento. Questo algoritmo è riportato nella figura 3.5.

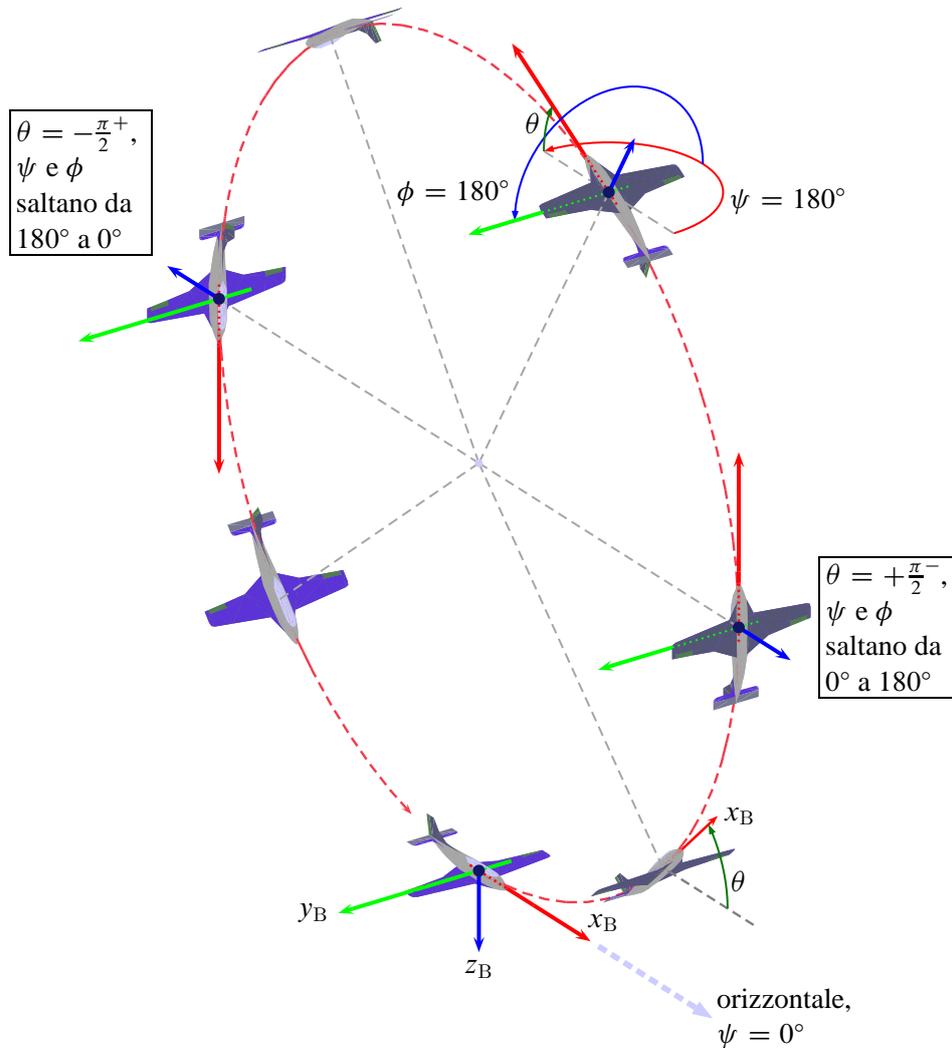


Figura 3.6 Variazione degli angoli di Eulero durante una manovra di *looping*.

Essendo arbitrario il valore dell'angolo di prua in situazioni di *gimbal lock*, negli algoritmi di integrazione numerica delle equazioni del moto è possibile scegliere un valore conveniente, ad esempio, il valore di ψ ottenuto al passo di integrazione precedente.

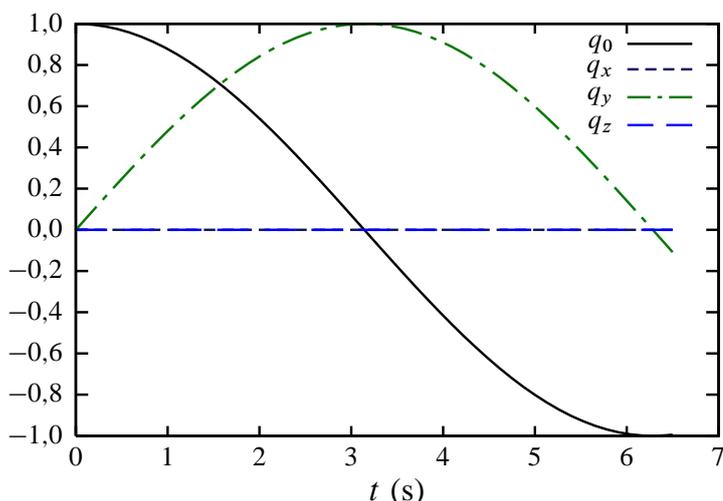
3.3.6 Esempi di storie temporali del quaternione dell'orientamento

Riportiamo due esempi di storie temporali delle componenti del quaternione dell'orientamento e delle terne corrispondenti di angoli di Eulero.

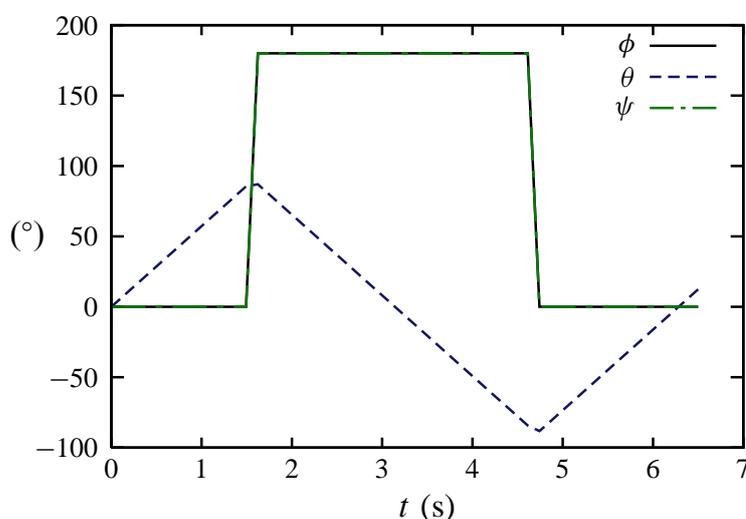
Il primo esempio è molto semplice. Esso si riferisce a un'evoluzione del velivolo in cui la velocità angolare mantiene componenti p ed r identicamente nulle mentre la componente di beccheggio q è una costante non nulla:

$$p(t) = r(t) = 0, \quad q(t) = 1,00 \text{ rad/s} \quad (3.66)$$

Queste condizioni possono essere effettivamente mantenute in situazioni reali. Ad esempio quando il pilota riesce ad effettuare una richiamata sostenuta (*looping*) come quella rappresentata nella figura 3.6, per una buona parte della manovra l'unica componente non nulla e costante della velocità angolare è proprio la q . Note le storie temporali delle



(a) Componenti del quaternione dell'orientamento.



(b) Angoli di Eulero.

Figura 3.7 Storie temporali delle componenti del quaternione dell'orientamento e degli angoli di Eulero corrispondenti per un orientamento iniziale assegnato dalle condizioni $\psi = \theta = \phi = 0$, e per delle assegnate leggi di variazione delle componenti della velocità angolare, $q(t) = 1,00 \text{ rad/s}$, $p(t) = r(t) = 0,00 \text{ rad/s}$.

componenti del vettore $\boldsymbol{\Omega}$ in assi velivolo, è possibile integrare le equazioni cinematiche:

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_x \\ \dot{q}_y \\ \dot{q}_z \end{pmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} \quad (3.67)$$

in un dato intervallo di tempo, ad esempio $[0, t_f]$, per degli assegnati valori iniziali ($t = 0$) delle componenti del quaternione. Questi ultimi possono essere fissati ipotizzando, per esempio, un volo inizialmente ad ali livellate, fusoliera orizzontale e prua nella direzione Nord. In queste condizioni iniziali l'orientamento del velivolo nello spazio è caratterizzato da angoli di Eulero tutti nulli; dalla (3.61) si ottiene:

$$q_0(0) = 1, \quad q_x(0) = q_y(0) = q_z(0) = 0 \quad (3.68)$$

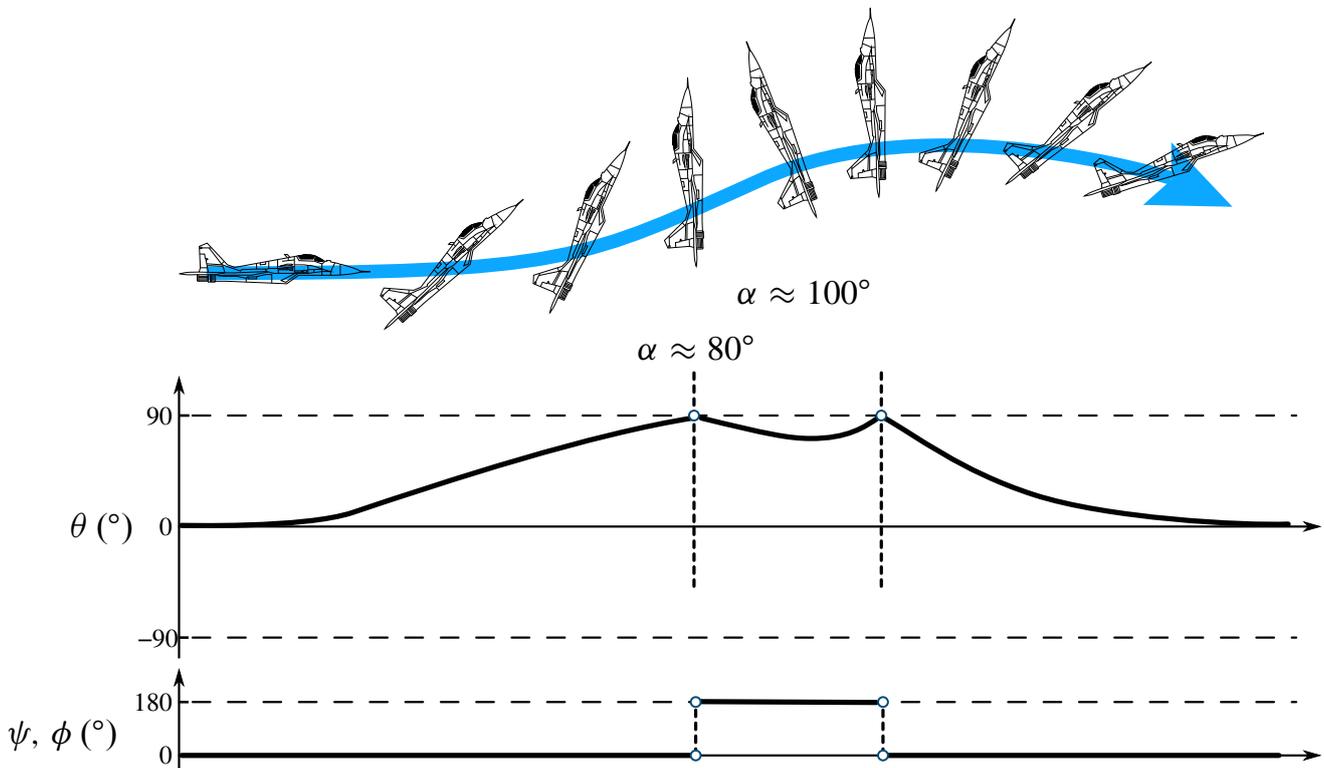


Figura 3.8 Variazione degli angoli di Eulero in una manovra acrobatica detta *manovra del cobra*.

La soluzione numerica del problema di valori iniziali (3.67)-(3.68) fornisce le storie temporali delle componenti del quaternione dell'orientamento e dei corrispondenti angoli di Eulero rappresentate nella figura 3.7. Dall'esame dei grafici e con l'ausilio della figura 3.6 si deduce che:

- come ci si aspetta, le componenti del quaternione variano con regolarità e sono comprese fra i valori -1 ed 1 ;
- l'elevazione θ varia linearmente da 0 ad un valore di poco inferiore a 90 deg; allo stesso tempo l'angolo di heading ψ e l'angolo di rollio ϕ sono costanti e nulli; non appena θ si avvicina al valore di 90 deg sia ψ che ϕ si incrementano ("saltano") di 180 deg; a questo punto θ (anziché continuare a crescere) comincia a decrescere linearmente, fino a 0 ed oltre, arrivando ad assumere valori vicini a -90 deg; in questo intervallo di tempo gli angoli ψ e ϕ si mantengono pari a 180 deg;
- per θ che si avvicina a -90 deg si ha un salto repentino dei valori di ψ e ϕ , da 180 deg a 0 deg; negli istanti successivi gli angoli di heading e di rollio si mantengono nulli mentre θ ricomincia a crescere linearmente, da valori di poco superiori a -90 deg fino a 0 deg, concludendo la traiettoria di *looping* ritornando in una condizione di orientamento identica a quella iniziale.

Il comportamento "a scatti" delle grandezze ψ e ϕ che si apprezza nella figura 3.7b è perfettamente normale e rispecchia le definizioni degli intervalli di variazione degli angoli di Eulero del velivolo.

La figura 3.8, in cui è raffigurata una manovra acrobatica detta *manovra del cobra*, rappresenta un ulteriore esempio in cui vi è variazione discontinua degli angoli ψ e ϕ per un angolo θ che varia in prossimità degli estremi del suo intervallo di definizione.

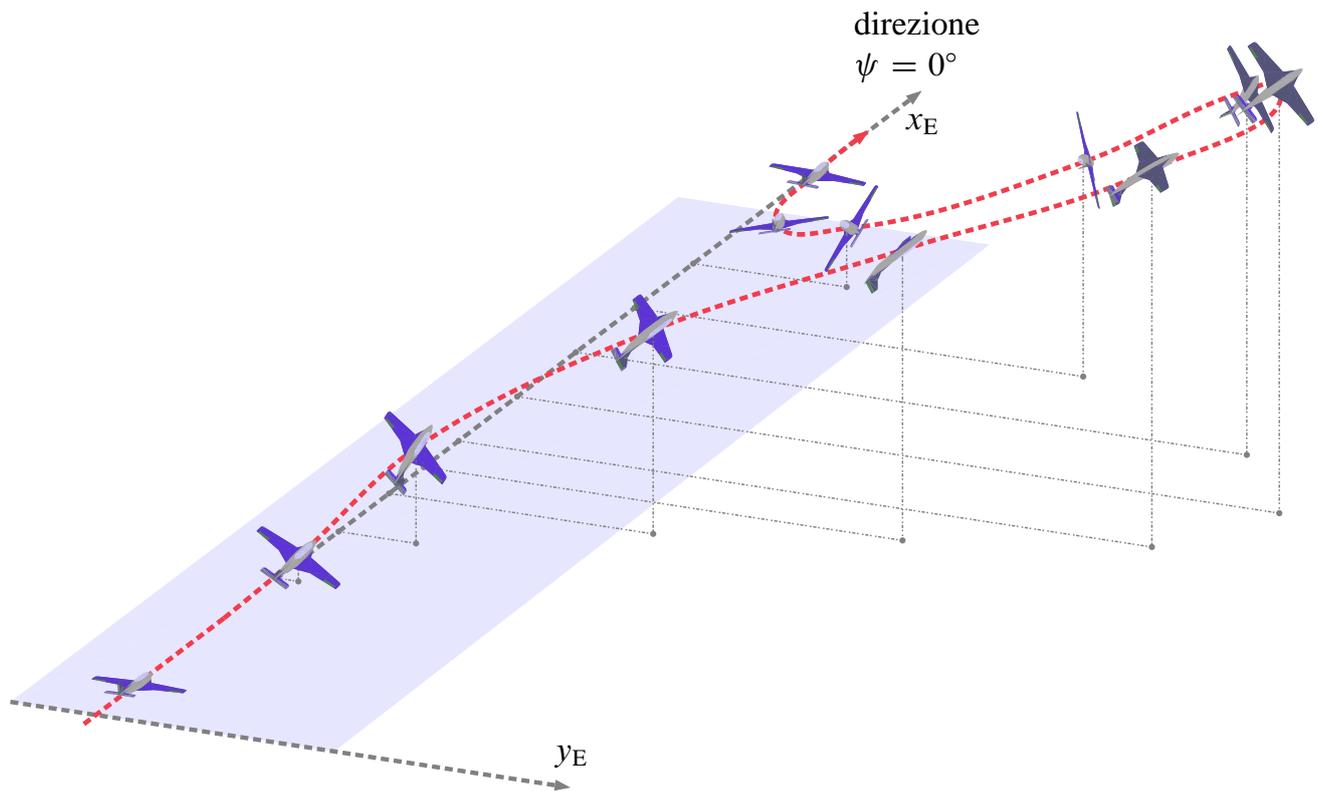


Figura 3.9 Esempio di evoluzione con moto di rotazione intorno all'asse di rollio. La manovra è caratterizzata da storie temporali dei parametri dell'orientamento simili a quelle della figura 3.10.

Esercizio 3.1: Cinematica dell'evoluzione di looping in Matlab



Utilizzando le funzioni predefinite di Matlab risolvere numericamente il problema di valori iniziali (3.67)-(3.68) nel caso di una manovra di *looping*. Confrontare i risultati con quelli rappresentati dai grafici della figura 3.7 a pagina 26.

Assegnare opportunamente gli angoli di Eulero iniziali e delle plausibili leggi temporali delle componenti della velocità angolare in assi velivolo. Ad esempio, supporre la velocità angolare di beccheggio $q(t)$ inizialmente nulla e crescente con t , fino al raggiungimento di un valore costante q_{\max} , mantenuto per quasi tutta la manovra; nella fase terminale dell'evoluzione ipotizzare un decadimento della $q(t)$ da q_{\max} a 0. Per semplicità assumere profili identicamente nulli per le velocità angolari p ed r .

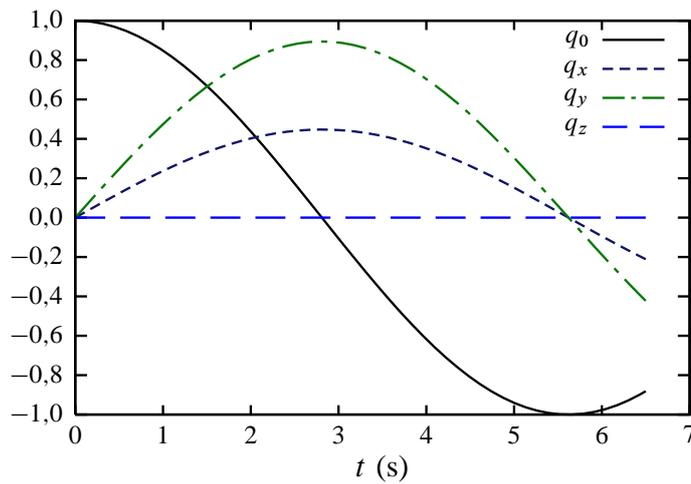
Si consulti la documentazione della funzione `ode45` e delle funzioni `angle2quat` e `quat2angle` dell'Aerospace Toolbox. Si scriva un programma di calcolo che risolve il problema di valori iniziali assegnato e disegna i grafici necessari. Si trovi una combinazione opportuna del tempo di simulazione t_f e della legge $q(t)$ tale da far compiere al velivolo un giro completo.



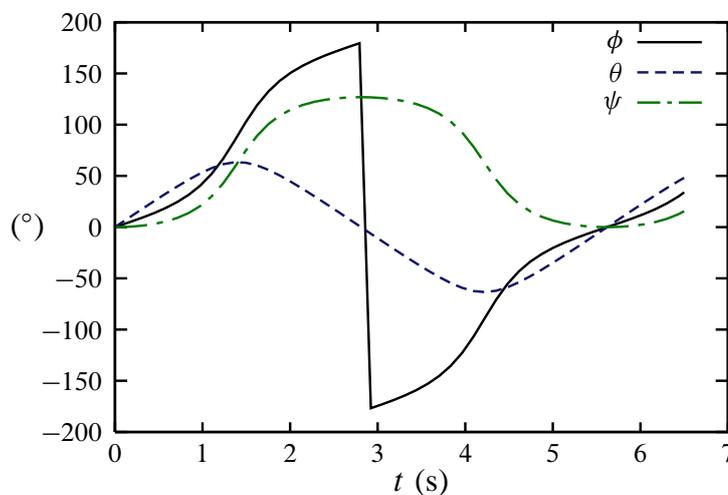
Esercizio 3.2: Traiettoria dell'evoluzione di looping in Matlab



Con riferimento all'esercizio 3.1 scrivere un codice di calcolo in Matlab che integra le equazioni della navigazione (3.28) e ricostruisce la traiettoria del baricentro del velivolo in assi Terra, note le storie temporali del quaternione e degli angoli di Eulero.



(a) Componenti del quaternione dell'orientamento.



(b) Angoli di Eulero.

Figura 3.10 Storie temporali delle componenti del quaternione dell'orientamento e degli angoli di Eulero corrispondenti per un orientamento iniziale assegnato dalle condizioni $\psi = \theta = \phi = 0$, e per delle assegnate leggi di variazione delle componenti della velocità angolare, $q(t) = 1,00 \text{ rad/s}$, $p(t) = 0,50 \text{ rad/s}$, $r(t) = 0,00 \text{ rad/s}$.

Assegnare opportunamente la posizione iniziale e delle plausibili leggi temporali delle componenti della velocità in assi velivolo. Ad esempio, supporre la componente u inizialmente pari a 250 km/h e decrescente gradualmente con t , fino al raggiungimento di un valore costante, mantenuto per tutta la manovra. Per semplicità assumere profili identicamente nulli per le componenti v e w .

Si consulti la documentazione della funzione `quat2dcm` dell'Aerospace Toolbox.



Un secondo esempio di traiettoria di volo con un'interessante successione degli orientamenti del velivolo nello spazio è rappresentato dalla figura 3.9. Questo tipo di evoluzione acrobatica è noto con il nome di *manovra di tonneau*.

Nella figura 3.10 sono riportate le storie temporali delle componenti del quaternione e degli angoli di Eulero. Le condizioni iniziali sono le stesse dell'esempio precedente, cioè le (3.68), ma il moto di rotazione assegnato è differente. Per semplicità si possono assumere le leggi seguenti: $q(t) = q_0 = 1,00 \text{ rad/s}$, $p(t) = p_0 = 0,50 \text{ rad/s}$, $r(t) = 0$.

In questo caso l'angolo di prua non subisce una variazione a scatti poiché la storia del moto è influenzata maggiormente dalla velocità angolare di rollio. Infatti, l'angolo ψ cresce nel tempo e raggiunge un massimo prima di raggiungere l'estremo superiore del suo intervallo di definizione. Successivamente, esso decresce e tende al valore nullo iniziale. D'altra parte, in questo moto a velocità angolare di rollio costante l'angolo ϕ raggiunge il limite superiore del suo intervallo di definizione. Non appena l'angolo di *bank* tende a superare i 180 deg esso subisce una variazione a scatto di -180 deg che lo riporta in prossimità dell'estremo inferiore del suo intervallo di definizione.

Esercizio 3.3: Cinematica dell'evoluzione di tonneau in Matlab



Si ripetano gli esercizi 3.1 e 3.2 nel caso di una manovra di *tonneau*. Confrontare i risultati con quelli rappresentati dai grafici della figura 3.10 nella pagina precedente.

Assegnare opportunamente gli angoli di Eulero iniziali e delle plausibili leggi temporali delle componenti della velocità angolare in assi velivolo. Ad esempio si supponga una velocità angolare di rollio $p(t)$ inizialmente nulla e crescente con t , fino a raggiungere un valore costante p_{\max} , mantenuto per gran parte della manovra; nella fase terminale dell'evoluzione ipotizzare un decadimento della $p(t)$ da p_{\max} a 0. Ipotizzare un legge temporale analoga per la componente di velocità angolare di beccheggio $q(t)$, con q_{\max} pari a circa la metà di p_{\max} . Si supponga una legge $r(t)$ della velocità angolare di imbardata identicamente nulla.

Assegnare opportunamente la posizione iniziale e delle plausibili leggi temporali delle componenti della velocità in assi velivolo. Ad esempio si supponga la componente $u(t)$ pari inizialmente a $u_0 = 380$ km/h, decrescente con t fino a un valore minimo ($\approx 0,70 u_0$) e successivamente crescente con t per raggiungere u_0 a fine manovra. Per semplicità assumere profili identicamente nulli per le componenti v e w .

Esempio di svolgimento

Una possibile via per risolvere il problema assegnato è data dal codice seguente.

```

clc; clear; close all;

% Problem solved:
% Given time histories of angular velocity components in body axes, integrate
% the quaternion components evolution equations and determine the related Euler
% angles time histories. Then, for assigned time histories of CG velocity
% components in body axes reconstruct the flight path.

t_fin = 105.0; % Simulation final time

% Initial conditions
phi0 = 0; psi0 = 0; theta0 = 0; % Euler angles
X0 = 0; Y0 = 0; Z0 = 0; % Position

%% Anonymous functions
% Tonneau, some reference values used to build
% three appropriate time histories of p, q, r
p_max = convangvel(4.0, 'deg/s', 'rad/s'); % rad/s
q_max = convangvel(2.0, 'deg/s', 'rad/s'); % rad/s
r_max = 0; % rad/s

p = @(t) ... → see Fig. 3.11a
    interp1( ...
        [0, t_fin/30, t_fin/10, t_fin/5, 0.7*t_fin, 0.9*t_fin, t_fin], ...
        [0, p_max/40, p_max*3/4, p_max, p_max, 0, 0], ...
        t, 'pchip' ...
    );

```

```

q = @(t) ...           → see Fig. 3.11a
    interp1( ...
        [0, t_fin/30, t_fin/10, t_fin/5, 0.8*t_fin, 0.9*t_fin, t_fin], ...
        [0, q_max/40, q_max*3/4, q_max, q_max, 0, 0], ...
        t, 'pchip' ...
    );
r = @(t) 0;

% Tonneau, some reference values used to build
% three appropriate time histories of u, v, w
u0 = convvel(380.0, 'km/h', 'm/s'); % m/s
v0 = convvel( 0.0, 'km/h', 'm/s'); % m/s
w0 = convvel( 0.0, 'km/h', 'm/s'); % m/s

u = @(t) ...           → see Fig. 3.11b
    interp1( ...
        [0, t_fin/30, t_fin/10, t_fin/5, 0.8*t_fin, t_fin], ...
        [u0, 0.8*u0, 0.7*u0, u0, 1.1*u0, u0], ...
        t, 'pchip' ...
    );
v = @(t) 0;
w = @(t) 0;

```

(continua)

Per contenere tutto il codice in un unico script di Matlab sono state definite delle *funzioni anonime* della variabile temporale — si vedano le assegnazioni con $@(t)$. Esse utilizzano la funzione predefinita `interp1` ed esprimono le leggi $p(t)$, $q(t)$ e $u(t)$ attraverso delle funzioni interpolanti opportunamente costruite. Le leggi assegnate diverse dalla funzione identicamente nulla sono riportate nella figura 3.11. In particolare, le leggi di p , q ed r vanno utilizzate per definire le quattro funzioni scalari che costituiscono i secondi membri della (3.67). Essi sono delle funzioni del tempo t e delle quattro componenti q_0 , q_x , q_y e q_z del quaternione dell'orientamento.

Dunque il codice precedente continua con la definizione di una funzione anonima `dQuatdt` da passare alla funzione predefinita `ode45`. Per l'integrazione del sistema di equazioni differenziali ordinarie (3.67) accoppiato alle condizioni iniziali (3.68) la funzione `ode45` deve ricevere in input una quaterna Q_0 di valori iniziali delle componenti del quaternione e un intervallo temporale $[0 \ t_fin]$. Il codice di calcolo continua come segue.

(continua dal listato precedente)

```

% Kinematic equations → see (3.67)
dQuatdt = @(t, Q) ...
    0.5*[ 0, -p(t), -q(t), -r(t);
          p(t), 0, r(t), -q(t);
          q(t), -r(t), 0, p(t);
          r(t), q(t), -p(t), 0] * Q;

%% Solution of quaternion component evolution equations
options = odeset( ...
    'RelTol', 1e-9, ...
    'AbsTol', 1e-9*ones(1,4) ...
);

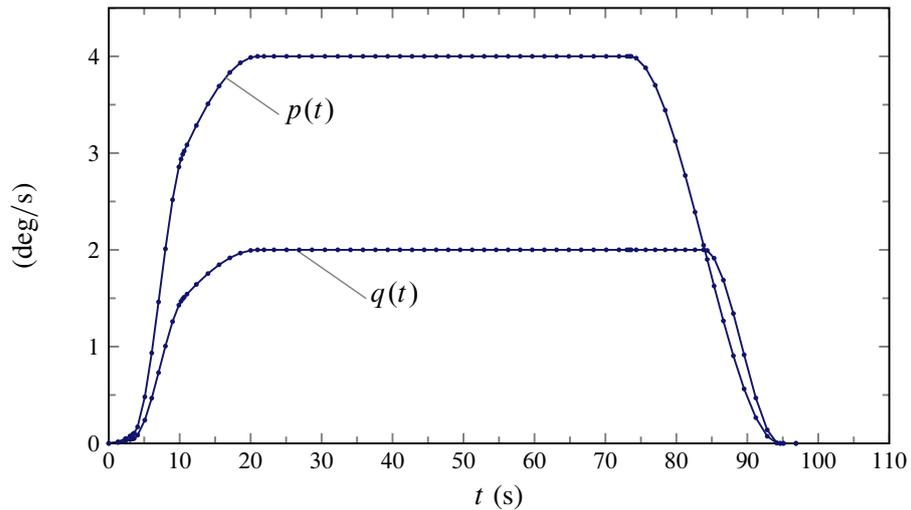
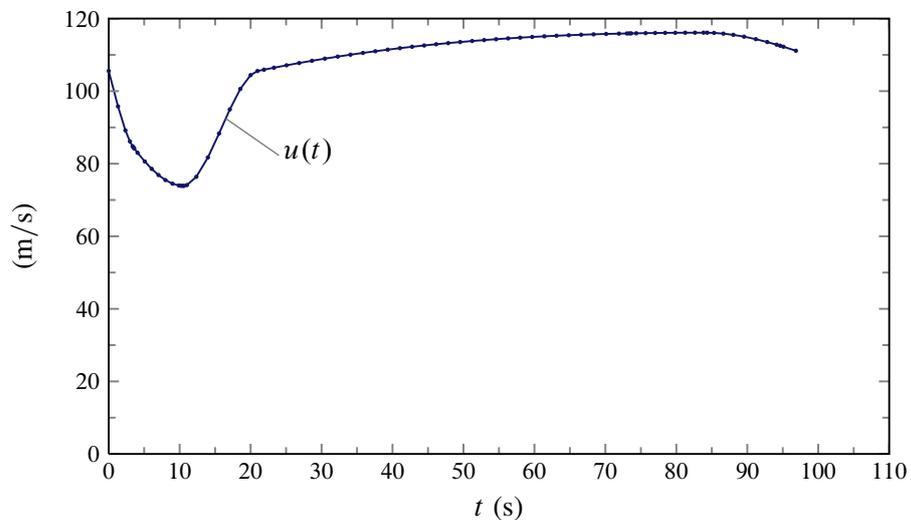
% Initial quaternion
Q0 = angle2quat(psi0, theta0, phi0); → see (3.61)

% Runge-Kutta integration
[vTime, vQuat] = ode45(dQuatdt, [0 t_fin], Q0, options);

```

(continua)

Si ottiene così il vettore $vQuat$ con la soluzione numerica contenente la sequenza degli orientamenti del velivolo. Tale sequenza è associata all'insieme discreto di istanti di tempo successivi immagazzinati nel vettore $vTime$.

(a) Componenti p e q della velocità angolare.(b) Componente u della velocità del baricentro.**Figura 3.11** Storie temporali delle componenti di velocità angolare e traslazionale assegnate nell'esercizio 3.3.

Il frammento di codice seguente prepara i grafici delle grandezze fin qui ricavate.

```

figure(1)
% Quaternion components time histories → see Fig. 3.13a
subplot 121
plot( ...
    vTime, vQuat(:,1), '-.', ...
    vTime, vQuat(:,2), '-.-', ...
    vTime, vQuat(:,3), '-.-', ...
    vTime, vQuat(:,4), ':' ...
)
legend('q_0', 'q_x', 'q_y', 'q_z')
xlabel('t (s)'); ylabel('')
title('Quaternion components')

% Euler angles time histories → see Fig. 3.13b
subplot 122
[vpsi, vtheta, vphi] = quat2angle(vQuat);
plot( ...
    vTime, convang( vpsi, 'rad', 'deg'), '-.', ...
    vTime, convang(vtheta, 'rad', 'deg'), ':', ...
    vTime, convang( vphi, 'rad', 'deg'), '-.-' )

```

(continua dal listato precedente)

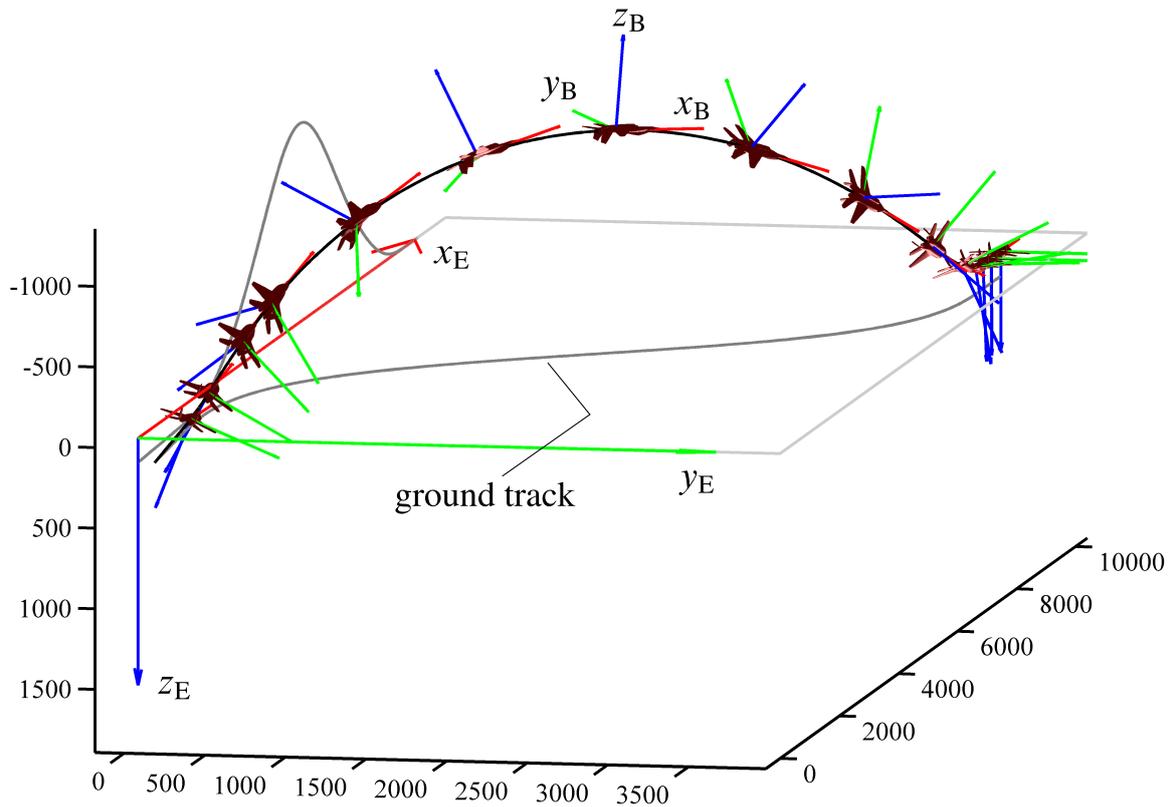


Figura 3.12 Un tonneau a destra.

```

legend('\psi', '\theta', '\phi')
xlabel('t (s)'); ylabel('(deg)')
title('Euler angles')

figure(2)
% Angular velocity components in body axes → see Fig. 3.11a
subplot 211
plot( ...
    vTime, convangvel(p(vTime), 'rad/s', 'deg/s'), ...
    vTime, convangvel(q(vTime), 'rad/s', 'deg/s'), ...
    vTime, convangvel(r(vTime), 'rad/s', 'deg/s'))
legend('p(t)', 'q(t)', 'r(t)')
xlabel('t (s)'); ylabel('(deg/s)')
title('Angular velocity components in body axes')

% Velocity components in body axes → see Fig. 3.11b
subplot 212
plot(vTime, u(vTime))
legend('u(t)')
xlabel('t (s)'); ylabel('(m/s)')
title('X-Velocity component in body axes')

```

(continua)

Noti gli orientamenti nel tempo bisogna ora risolvere il problema di valori iniziali

$$\left\{ \begin{array}{l} \left\{ \begin{array}{l} \dot{x}_{E,G} \\ \dot{y}_{E,G} \\ \dot{z}_{E,G} \end{array} \right\} = [T_{EB}(q_0, q_x, q_y, q_z)] \left\{ \begin{array}{l} u \\ v \\ w \end{array} \right\} \\ x_{E,G}(0) = 0, \quad y_{E,G}(0) = 0, \quad z_{E,G}(0) = z_{E,G,0} \end{array} \right. \quad (3.69)$$

Sia le componenti di velocità che le componenti del quaternione nelle equazioni del pro-

blema (3.69) sono trattabili a questo punto come delle funzioni note del tempo. Infatti, le funzioni $u(t)$, $v(t)$ e $w(t)$ sono un dato del problema mentre le componenti del quaternione sono state ricavate come funzioni tabulari. Esse sono espresse dalla matrice $[vTime, vQuat]$ di dimensioni $N \times 5$, con N pari a $\text{length}(vTime)$. Le $q_0(t)$, $q_x(t)$, $q_y(t)$ e $q_z(t)$ vengono quindi ricostruite attraverso funzioni anonime che forniscono i valori all'istante generico t per interpolazione a partire dai valori discreti contenuti in $vTime$ e $vQuat$.

Si definisce così una funzione anonima $dPosEdt$ con i secondi membri delle equazioni del problema (3.69). Tale funzione viene passata a `ode45` insieme al vettore $vTime$ precedentemente ricavato e insieme al vettore $PosE0$ della posizione iniziale.

Il codice che ricava la traiettoria del baricentro è il seguente.

```

(continua dal listato precedente)
% Time interpolation function for known quaternion history
Quat = @(t) ...
    [interp1(vTime,vQuat(:,1),t), ...
      interp1(vTime,vQuat(:,2),t), ...
      interp1(vTime,vQuat(:,3),t), ...
      interp1(vTime,vQuat(:,4),t)];

% Transformation matrix from Earth- to body-axes → [TBE] = [TEB]T
T_BE = @(Q) ...
    quat2dcm(Q);

% RHS of navigation equations → see (3.25), [TEB] · [u v w]T
dPosEdt = @(t,PosE) ...
    transpose(quat2dcm(Quat(t)))*[u(t);v(t);w(t)];

% Solution of navigation equations
options = odeset( ...
    'RelTol', 1e-3, ...
    'AbsTol', 1e-3*ones(3,1) ...
    );

% Initial position
PosE0 = [0;0;0];

% Runge-Kutta integration
[vTime2, vPosE] = ode45(dPosEdt, vTime, PosE0, options);

vXe = vPosE(:,1);
vYe = vPosE(:,2);
vZe = Z0 + vPosE(:,3);
(continua)
```

Il frammento di codice seguente rappresenta le storie temporali delle coordinate del baricentro.

```

(continua dal listato precedente)
% CG coordinates time history → see Fig. 3.13c
figure(3)
plot( ...
    vTime, vXe, '--', ...
    vTime, vYe, '-.-', ...
    vTime, vZe, '-.-', ...
    )
% hold on, view(3)
legend('x_{G,E}(t)', 'y_{G,E}(t)', 'z_{G,E}(t)')
xlabel('t (s)'); ylabel('m');
title('CG coordinates in Earth axes');
(continua)
```

Infine, viene riportata nello spazio tridimensionale la traiettoria e viene rappresentata la sequenza di modelli tridimensionali di velivolo orientati secondo la successione dei valori del quaternione dell'orientamento (figura 3.12).

(continua dal listato precedente)

```

%% Plot body and trajectory → see Fig. 3.12
h_fig4 = figure(4);
title('Flight path with body');
% theView = [70 20];
theView = [1,1,0.5];
plotTrajectoryAndBody(h_fig4, vXe, vYe, vZe, vQuat, 0.003, 25, theView)

```

→ see <http://wpage.unina.it/agodemar/DSV-DQV/plotTrajectoryAndBody.zip>

I grafici della figura 3.13 nella pagina successiva riassumono i risultati dell'esercizio proposto riportando le storie temporali dei parametri dell'orientamento del velivolo e della posizione del baricentro.



Esercizio 3.4: *Manovre acrobatiche*



Con riferimento alle figure 3.14 e 3.15 si riproduca la cinematica di una o più manovre acrobatiche. Si rappresentino le storie temporali degli angoli di Eulero e delle componenti del quaternione dell'orientamento del velivolo. Si rappresentino infine la traiettoria del baricentro e gli orientamenti successivi dell'aeromobile (come nella figura 3.12).



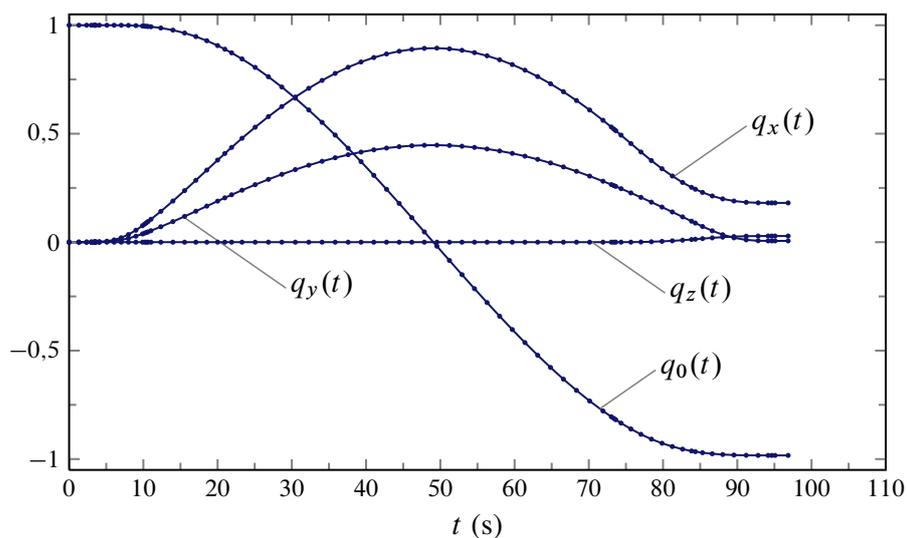
Esempio 3.1: *Manovra di rollio in Simulink*



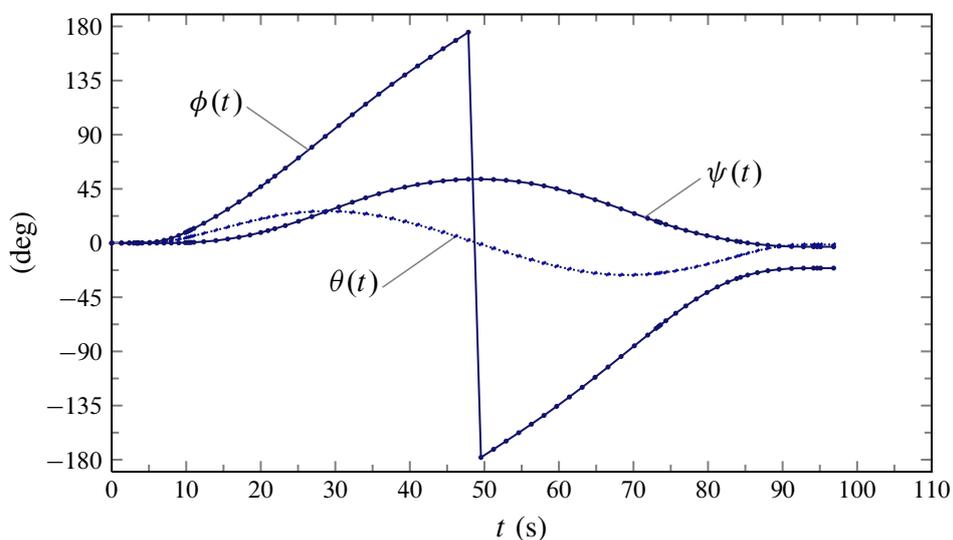
È interessante e istruttivo rivisitare i problemi affrontati negli esercizi precedenti — ad esempio l'esercizio 3.3 — realizzandone un modello Simulink. Ancora una volta si produrranno i segnali di input $[p(t), q(t), r(t)]^T$ e $[u(t), v(t), w(t)]^T$ per ottenere le storie temporali del quaternione dell'orientamento e dei tre angoli di Eulero del velivolo nonché la traiettoria del baricentro.

Un possibile modello Simulink per lo studio di un'evoluzione di rollio simile a un tonneau è dato dalla figura 3.16 a pagina 40. Si consideri il blocco di tipo Signal Builder raffigurato in dettaglio nella figura 3.17 (in Simulink Library Browser si veda il percorso `simulink/Sources`). Esso produce i segnali 'Signal 1', 'Signal 2' e 'Signal 3', i quali sono opportunamente scalati con blocchi di tipo Gain. In particolare il primo segnale viene ritardato con un blocco di tipo Transport Delay (in Simulink Library Browser si veda il percorso `simulink/Continuous`) — i guadagni e il ritardo di uno dei segnali vengono utilizzati per aggiustamenti successivi in modo da ottenere il risultato voluto. Ne risulta la terna di segnali $p(t)$, $q(t)$ ed $r(t)$ in rad/s che costituisce uno degli input della simulazione. Il blocco Scope della figura 3.17 serve a rappresentare le componenti di velocità angolare in deg/s (si veda la figura 3.19). La figura 3.18 rappresenta la finestra d'impostazione del blocco Signal Builder. Nell'esempio riportato i tre segnali hanno tutti un massimo unitario e sono stati costruiti con gli strumenti visuali forniti dalla finestra di dialogo. È anche possibile creare segnali leggendo array già presenti nel workspace di Matlab.

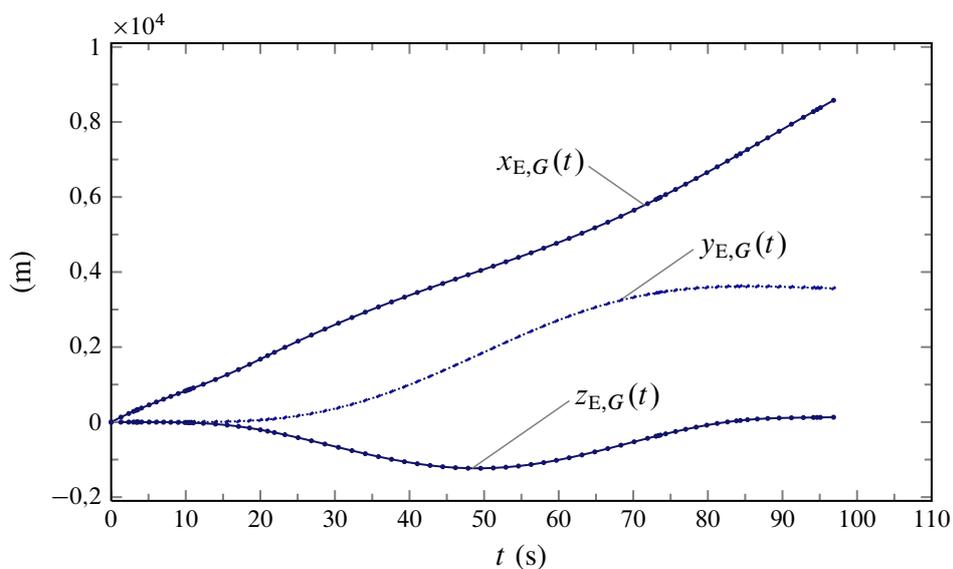
Lo schema della figura 3.20 a pagina 43 è un'altro dettaglio del modello Simulink. Esso implementa l'integrazione del sistema di equazioni differenziali (3.67). Il blocco di



(a) Componenti del quaternione dell'orientamento.



(b) Angoli di Eulero.



(c) Coordinate del baricentro.

Figura 3.13 Storie temporali dei parametri dell'orientamento e della posizione ricavati nell'esercizio 3.3.

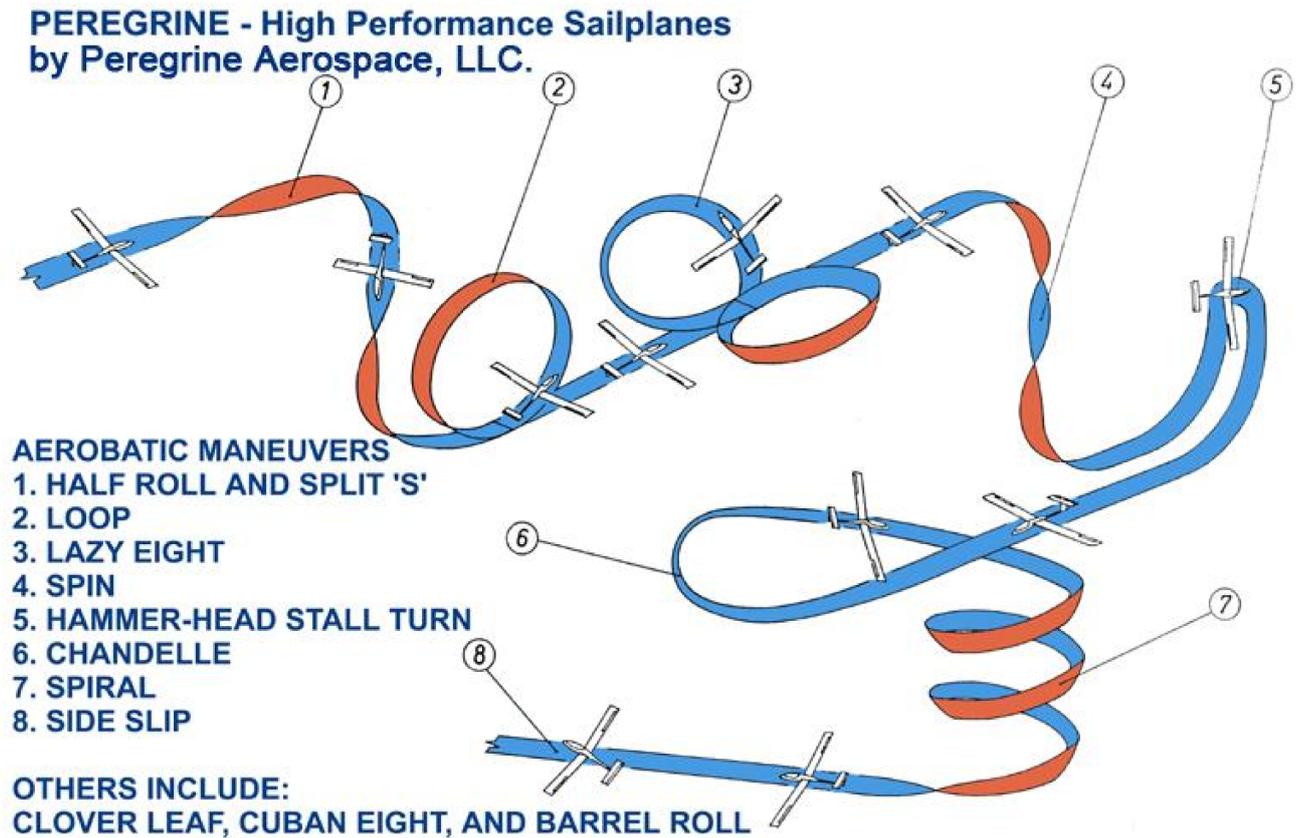


Figura 3.14 Esempi di manovre acrobatiche.

tipo Integrator (in Simulink Library Browser si veda il percorso `simulink/Continuous`) ha per uscita lo stato x del sistema, cioè il quaternione $[q_0(t), q_1(t), q_2(t), q_3(t)]^T$, e riceve in ingresso: (a) un segnale a 4 componenti che rappresenta il secondo membro della (3.67) e (b) un secondo ingresso, il segnale costante x_0 , corrispondente al valore iniziale del quaternione dell'orientamento (ottenibile, ad esempio, accedendo ad una variabile nel workspace di Matlab). L'uscita del blocco Integrator è rimandata come ingresso al blocco 'Matrix Multiply'. Quest'ultimo ha un segnale d'uscita corrispondente al secondo membro della (3.67).

Lo schema interno del sottosistema 'Antisymmetric 4×4 Matrix' della figura 3.20 è mostrato nella figura 3.21. Questo blocco di calcolo riceve in input le tre componenti di velocità angolare e serve a costruire la matrice a secondo membro dell'equazione (3.67). Al generico istante della simulazione viene assemblata una matrice quadrata, 4×4 , antisimmetrica, ottenendo in primo luogo dai tre ingressi p , q ed r i 16 segnali (di cui 4 nulli) corrispondenti ai singoli elementi. Attraverso il blocco di tipo Vector Concatenate (in Simulink Library Browser si veda il percorso `simulink/Commonly Used Blocks`) i 16 scalari vengono disposti in un array monodimensionale a gruppi di 4 rappresentanti le colonne della matrice desiderata. Infine si applica all'array un'operazione di *reshape* (corrispondente all'analogia funzione di ridimensionamento predefinita di Matlab) tramite il blocco Reshape (in Simulink Library Browser si veda il percorso `simulink/Math Operations`) e infine una moltiplicazione per $\frac{1}{2}$ (blocco Gain a destra).

Nella figura 3.22 a pagina 44 è rappresentato in dettaglio lo schema d'integrazione delle equazioni della traiettoria del baricentro. Il segnale principale è sempre costituito dal quaternione dell'orientamento, che proviene dallo schema d'integrazione discusso sopra.

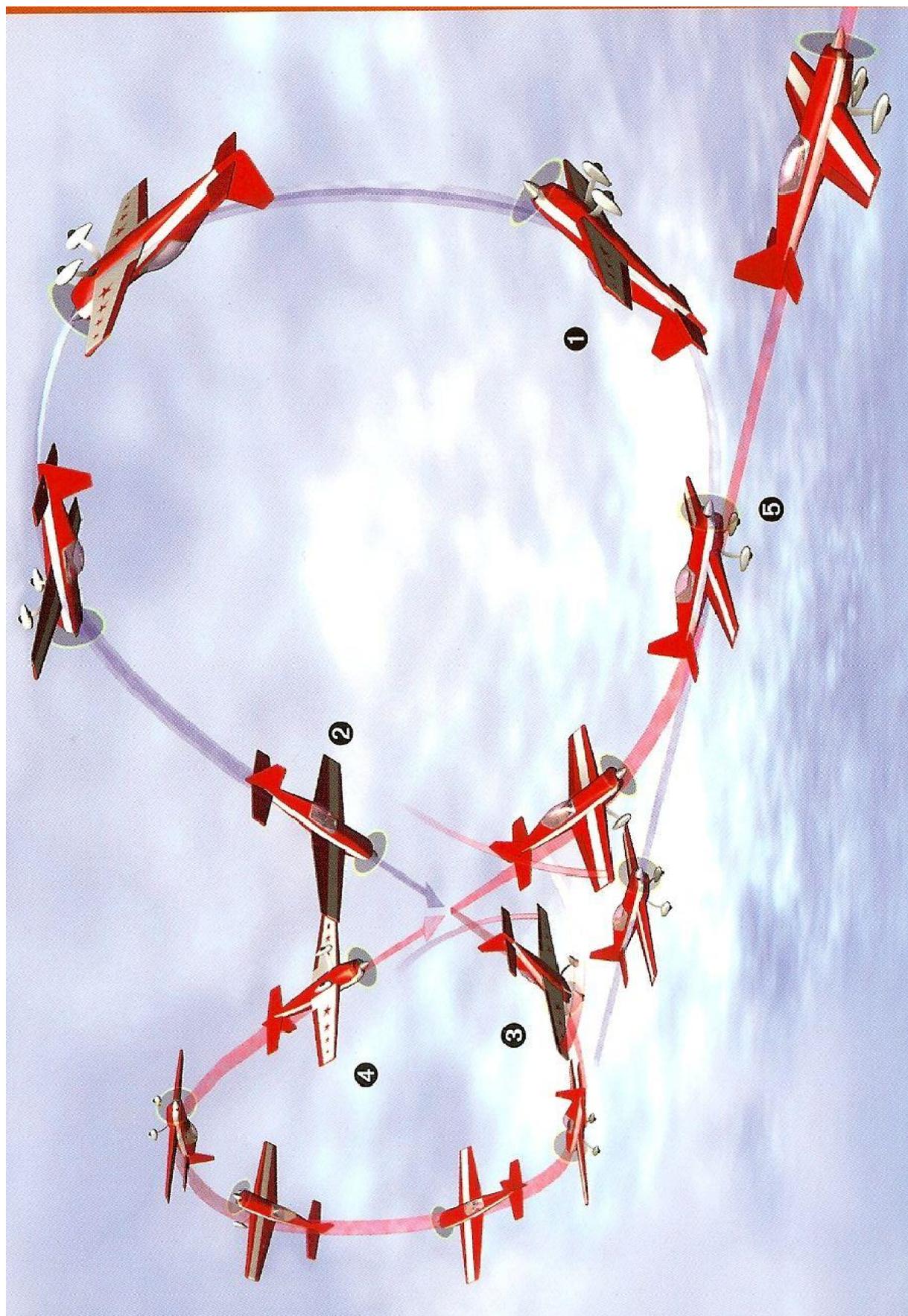


Figura 3.15 Una manovra acrobatica detta *Cuban eight*.

Il quaternione è un ingresso del blocco ‘Quaternions to Rotation Angles (Rotation Order ZYX)’ messo a disposizione dall’Aerospace Blockset (in Simulink Library Browser si veda il percorso `simulink/Aerospace Blockset/Utilities/Axes Transformations`), il quale restituisce la terna di angoli di Eulero $R_1 = \psi(t)$, $R_2 = \theta(t)$, $R_3 = \phi(t)$. A questo punto, note le storie degli angoli d’assetto, se si assegnano le componenti di velocità e una posizione iniziale del baricentro è possibile trovare la traiettoria.

Per semplicità si assume: (a) una velocità costante in modulo, pari a V_0 , e (b) un asse velivolo x_B costantemente tangente alla traiettoria ($\alpha_B(t) = \beta(t) = 0$). Si dovrà risolvere un problema di valori iniziali simile al (3.69), che nelle ipotesi appena fatte è esprimibile come segue:

$$\left\{ \begin{array}{l} \begin{array}{l} \dot{x}_{E,G} \\ \dot{y}_{E,G} \\ \dot{z}_{E,G} \end{array} = \begin{array}{l} V_N \\ V_E \\ V_D \end{array} = \begin{array}{l} V_0 \cos \theta \cos \psi \\ V_0 \cos \theta \sin \psi \\ -V_0 \sin \theta \end{array} \\ x_{E,G}(0) = 0, \quad y_{E,G}(0) = 0, \quad z_{E,G}(0) = z_{E,G,0} \end{array} \right. \quad (3.70)$$

Le componenti di velocità a secondo membro delle (3.70) sono definite attraverso i blocchi ‘Fcn: VN’, ‘Fcn: VE’ e ‘Fcn: VD’ della figura 3.22. Questi blocchi Simulink sono di tipo Fcn (in Simulink Library Browser si veda il percorso `simulink/User-Defined Functions`) e in particolare sono stati configurati come mostrato nella figura 3.23 a pagina 45. Il valore V_0 è assegnato in Matlab in una variabile `speed_ms` che è presente nel workspace quando il modello Simulink viene eseguito. I segnali corrispondenti alle componenti di velocità del baricentro in assi terra V_N , V_E , V_D sono, rispettivamente, gli ingressi dei blocchi ‘Integrator1’, ‘Integrator2’, e ‘Integrator3’ della figura 3.22. I primi due ricevono dalla porta d’ingresso della condizione iniziale un valore nullo; il terzo, corrispondente all’integrazione della terza delle (3.70), riceve un valore $z_{E,G,0}$ che viene acquisito accedendo ad una variabile nel workspace di Matlab.

Il codice Matlab da eseguire prima di lanciare il modello Simulink è il seguente.

```
%% Initial altitude
altitude0_ft = 1500.0;
altitude0_mt = convlength(altitude0_ft, 'ft', 'm');

%% Initial quaternion components
psi0 = 0; % rad
theta0 = 0; % rad
phi0 = 0; % rad
quat0 = angle2quat(psi0, theta0, phi0);

%% Speed
speed_kts = 200.0;
speed_ms = convvel(speed_kts, 'kts', 'm/s');
```

Pertanto, le costanti assegnate in ingresso come condizioni iniziali degli integratori dell’equazione dell’orientamento e della terza delle equazioni della traiettoria (3.70) sono i valori `quat0` (una matrice 4×1) e `-altitude0_mt` (uno scalare).

L’output del modello Simulink è rappresentato nelle figure 3.24, 3.25, 3.26. L’evoluzione del velivolo e la traiettoria del baricentro sono rappresentate nella figura 3.27.



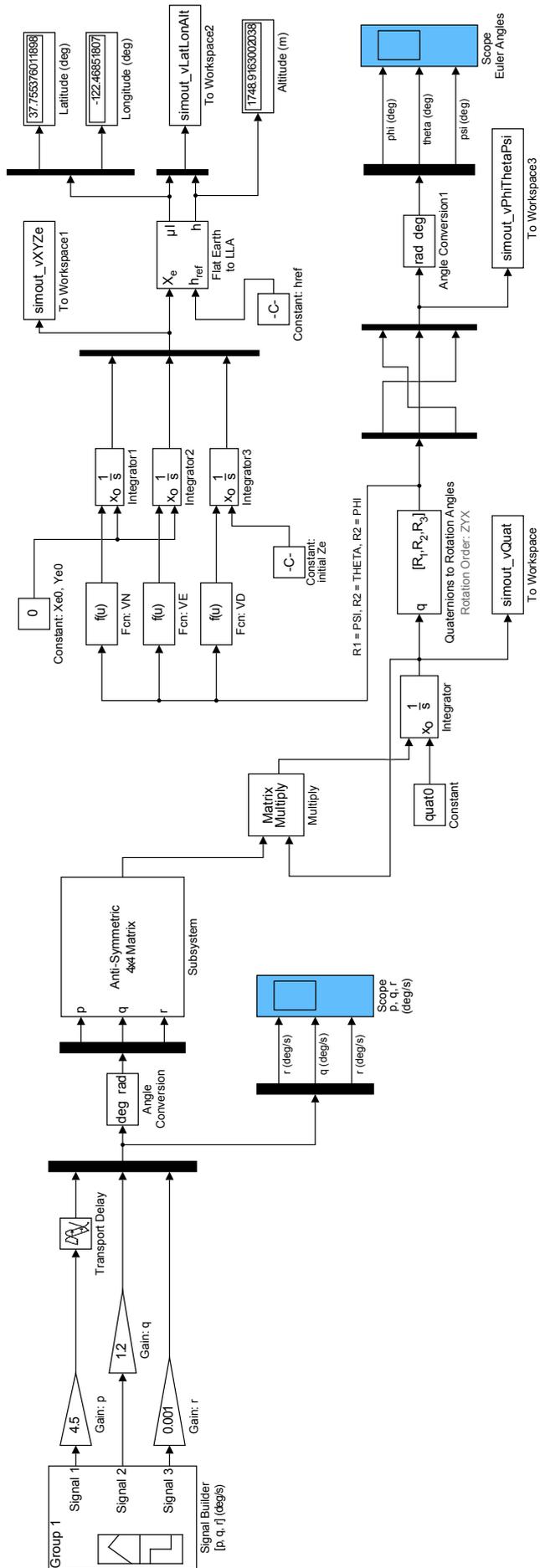


Figura 3.16 Il modello Simulink proposto nell'esercizio 3.1.

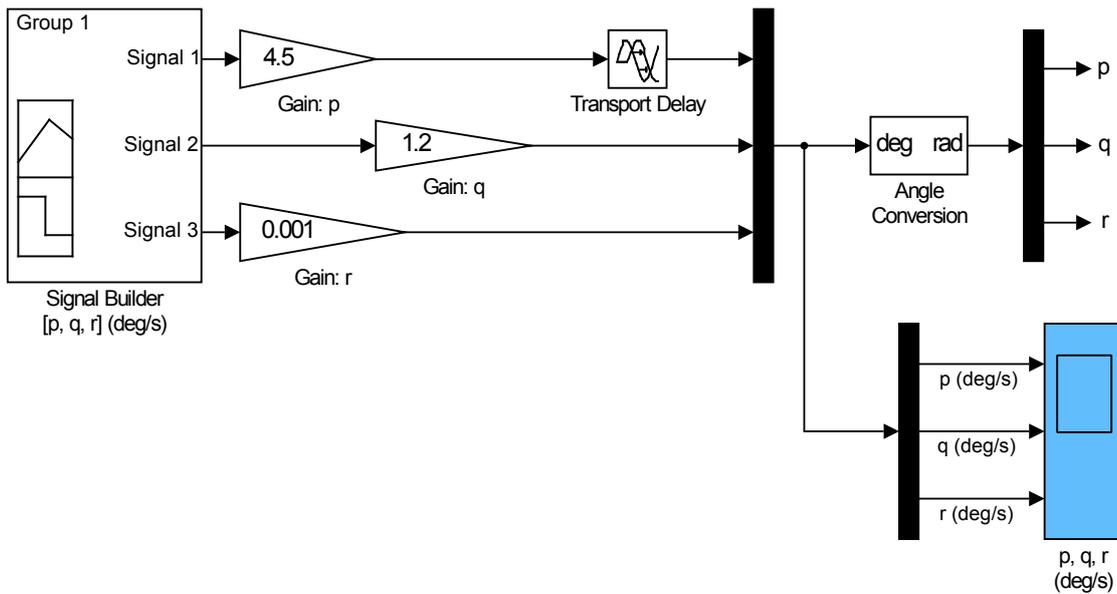


Figura 3.17 Dettaglio del modello Simulink con il blocco a sinistra di tipo Signal Builder che produce i segnali ‘Signal 1’, ‘Signal 2’ e ‘Signal 3’. I segnali sono opportunamente scalati con blocchi di tipo Gain. In particolare il primo segnale viene ritardato con un blocco di tipo Transport Delay. In uscita a destra si hanno i segnali $p(t)$, $q(t)$ ed $r(t)$ in rad/s che costituiscono l’input della simulazione. Il blocco Scope serve a rappresentare le componenti di velocità angolare in deg/s (figura 3.19).

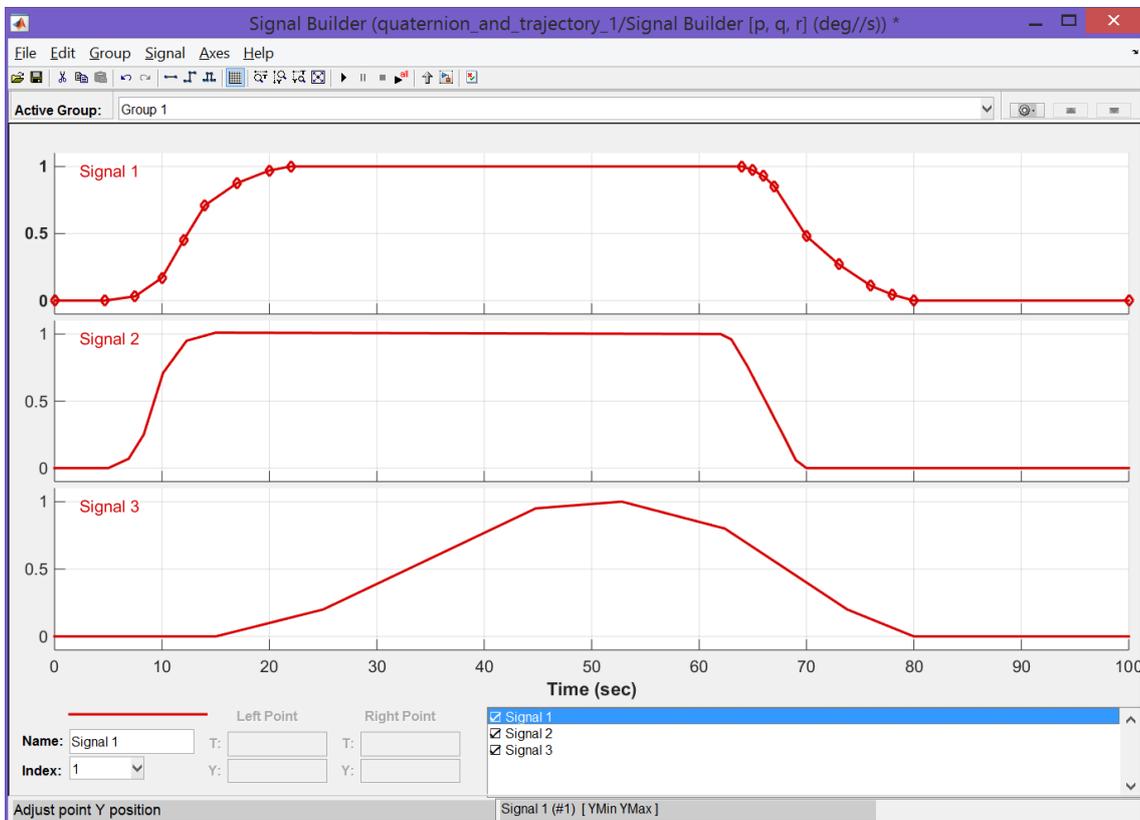


Figura 3.18 La finestra di impostazione del blocco Signal Builder. Tutti i segnali hanno un massimo unitario.

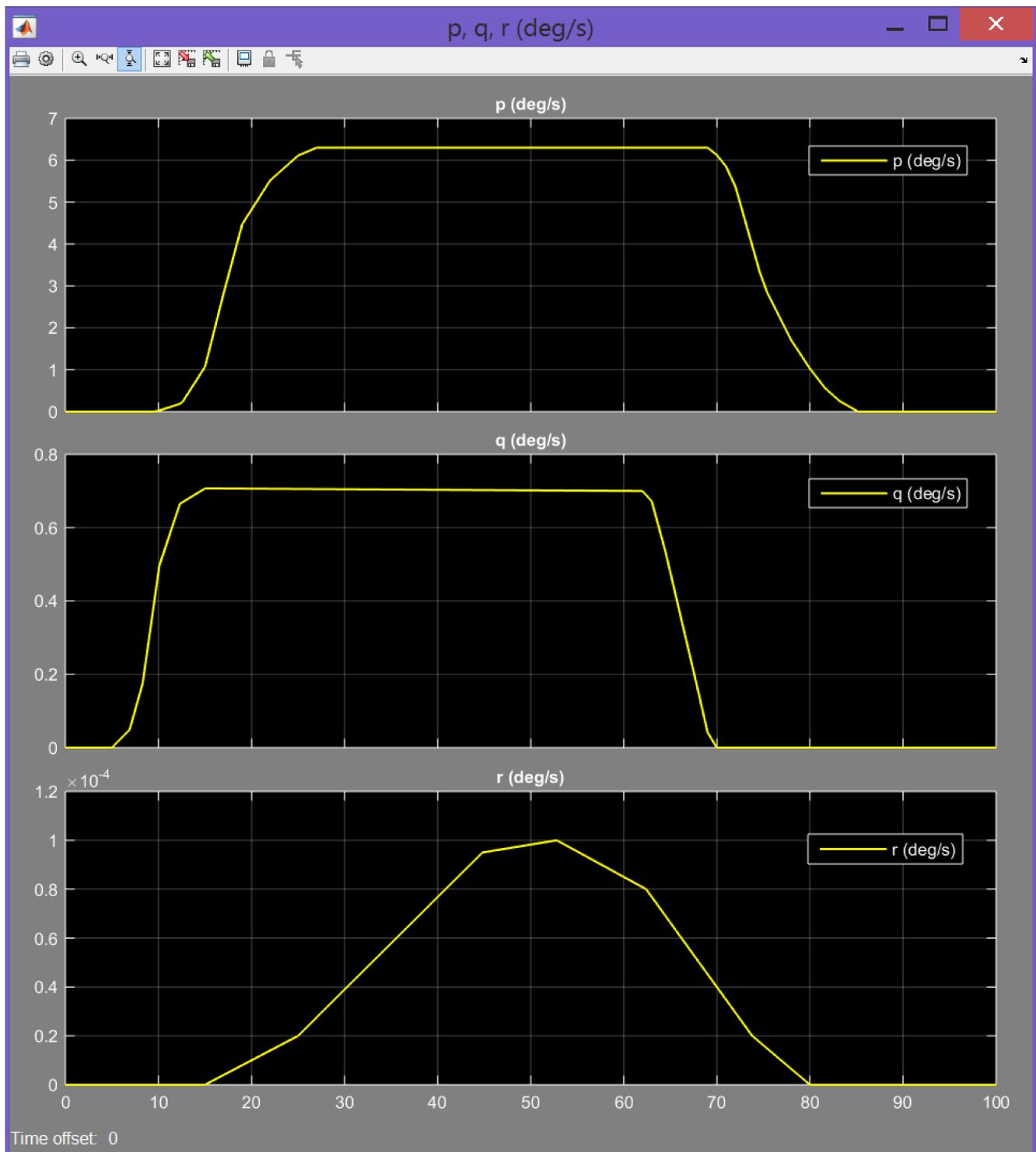


Figura 3.19 Storie temporali effettive delle componenti di velocità angolare in assi velivolo.

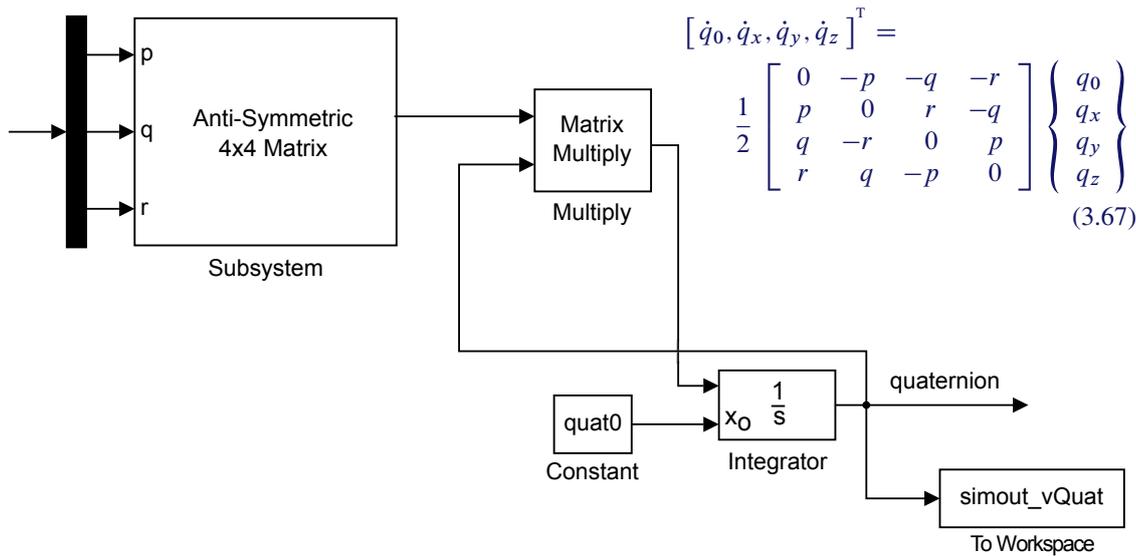


Figura 3.20 Dettaglio del modello Simulink con i blocchi per l'integrazione del sistema (3.67). Il blocco Integrator ha per uscita lo stato, cioè il quaternione $[q_0(t), q_1(t), q_2(t), q_3(t)]^T$. Questo segnale è anche l'ingresso del blocco 'Matrix Multiply', dal quale esce un segnale corrispondente al secondo membro della (3.67). Il primo blocco a sinistra, che riceve i segnali p , q ed r , è il sottosistema 'Antisymmetric 4×4 Matrix' riportato nella figura 3.21 e serve a costruire la matrice a secondo membro dell'equazione (3.67).

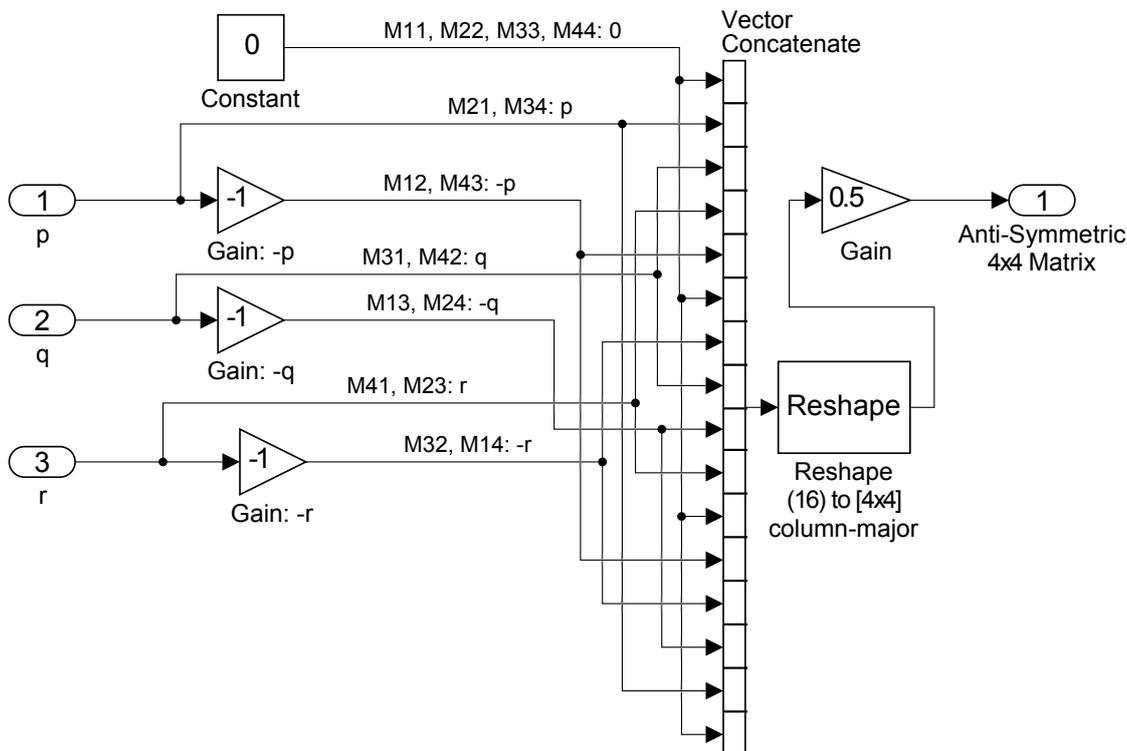


Figura 3.21 Dettaglio del modello Simulink con il sottosistema 'Antisymmetric 4×4 Matrix' della figura 3.20. Al generico istante della simulazione esso assembla la matrice quadrata, 4×4 , antisimmetrica a secondo membro dell'equazione (3.67). L'operazione viene effettuata ottenendo 16 segnali (di cui 4 nulli) dai tre ingressi p , q ed r . I 16 scalari vengono disposti in un array monodimensionale a gruppi di 4 rappresentanti le colonne della matrice desiderata. Infine si applica all'array un'operazione di *reshape* (corrispondente all'analogica funzione di ridimensionamento predefinita di Matlab).

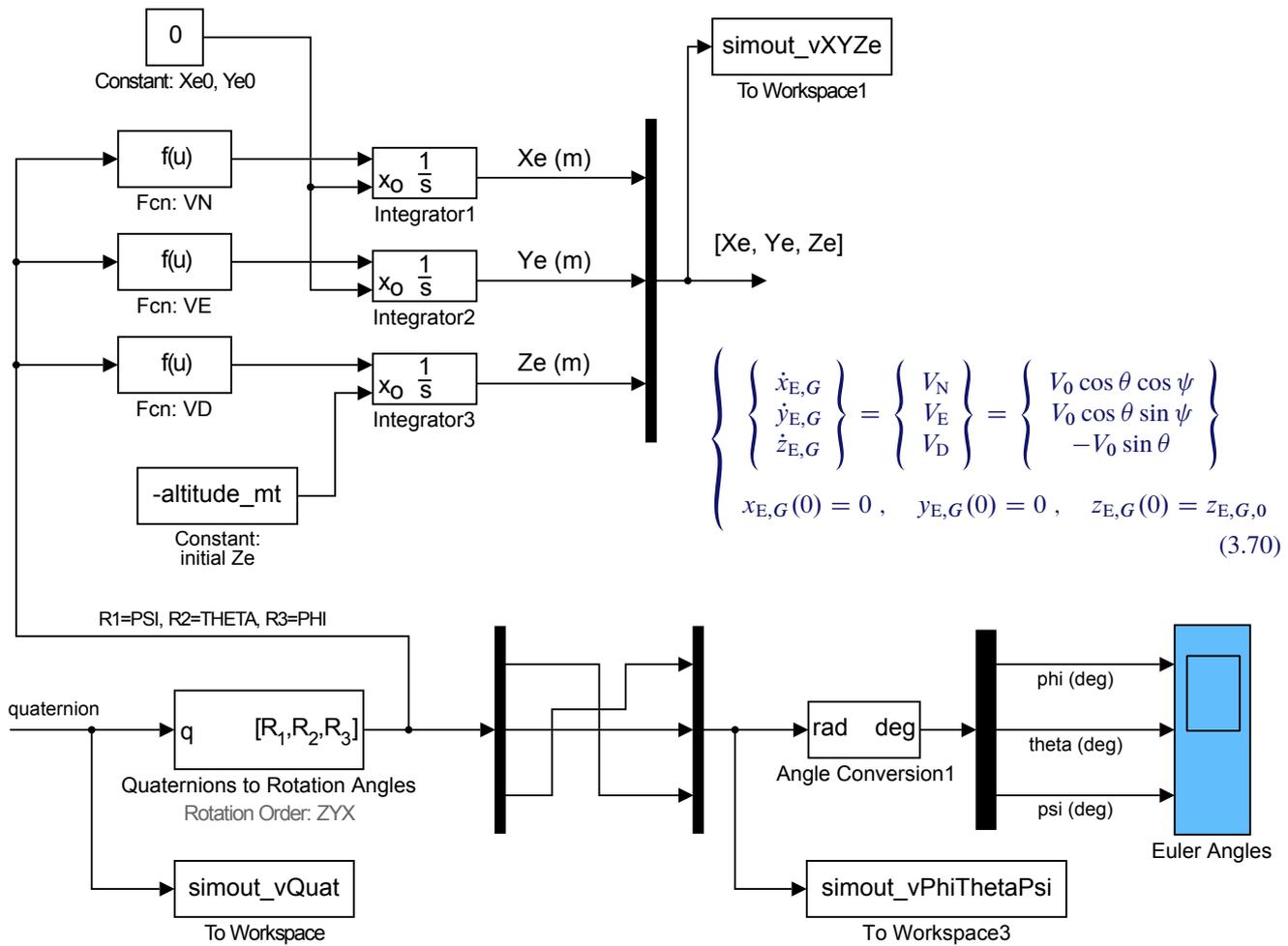


Figura 3.22 Dettaglio del modello Simulink con i blocchi di integrazione dell'equazione della traiettoria.

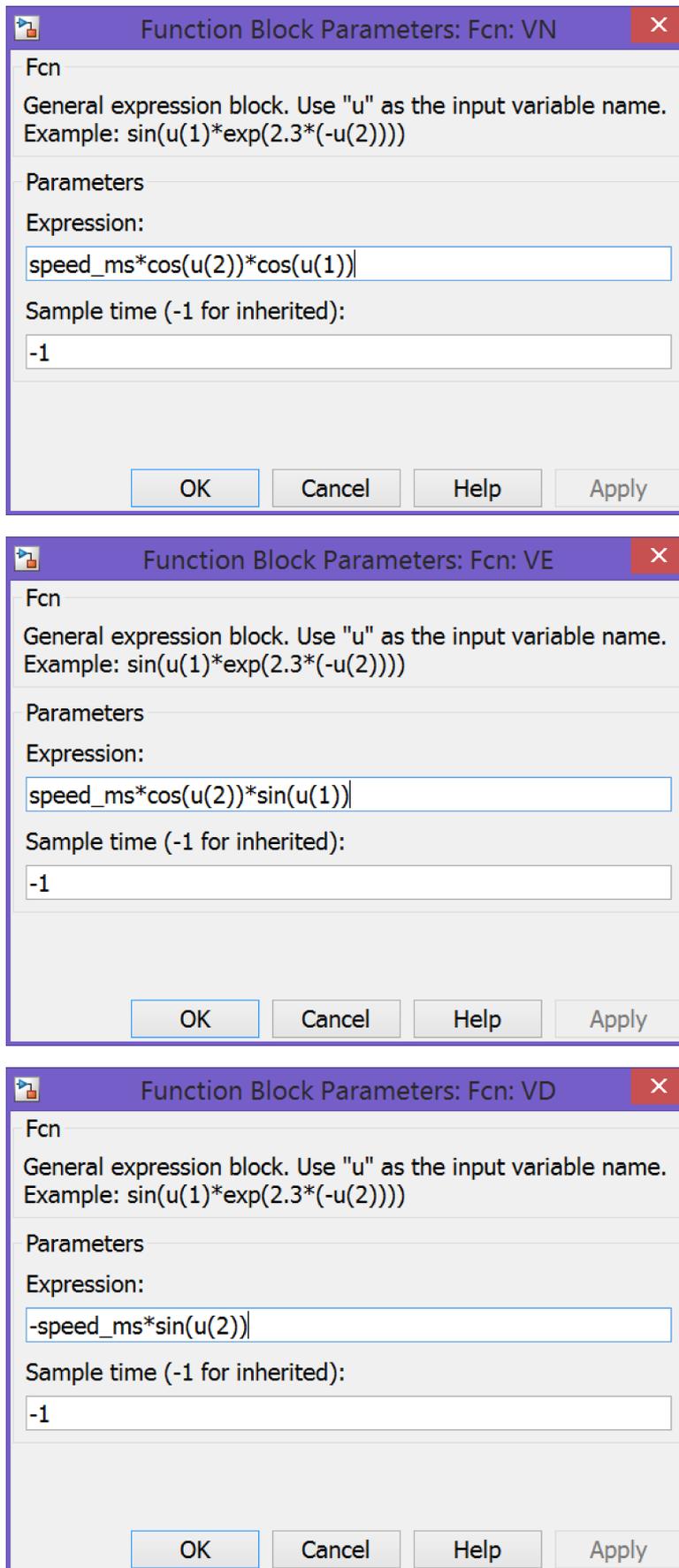


Figura 3.23 Configurazione dei blocchi di tipo User Defined Function per il calcolo delle componenti di velocità V_N , V_E e V_D (figura 3.22). La variabile `speed_ms` deve essere presente nel workspace di Matlab.

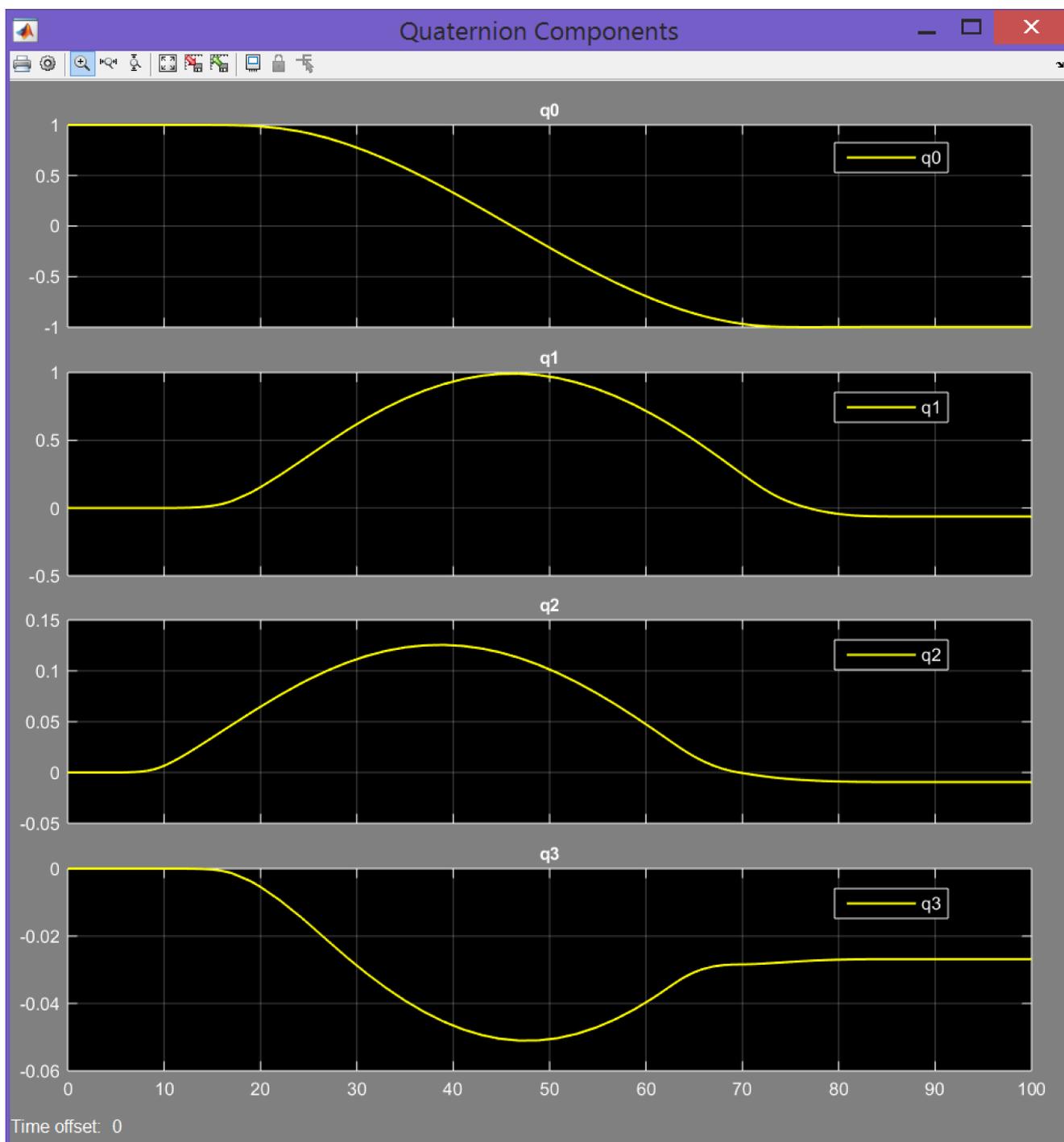


Figura 3.24 Storie temporali delle componenti del quaternione dell'orientamento. Blocco Scope non mostrato nella figura 3.20; si lascia per esercizio al lettore il compito di aggiungerne uno ottenendo i grafici qui rappresentati.

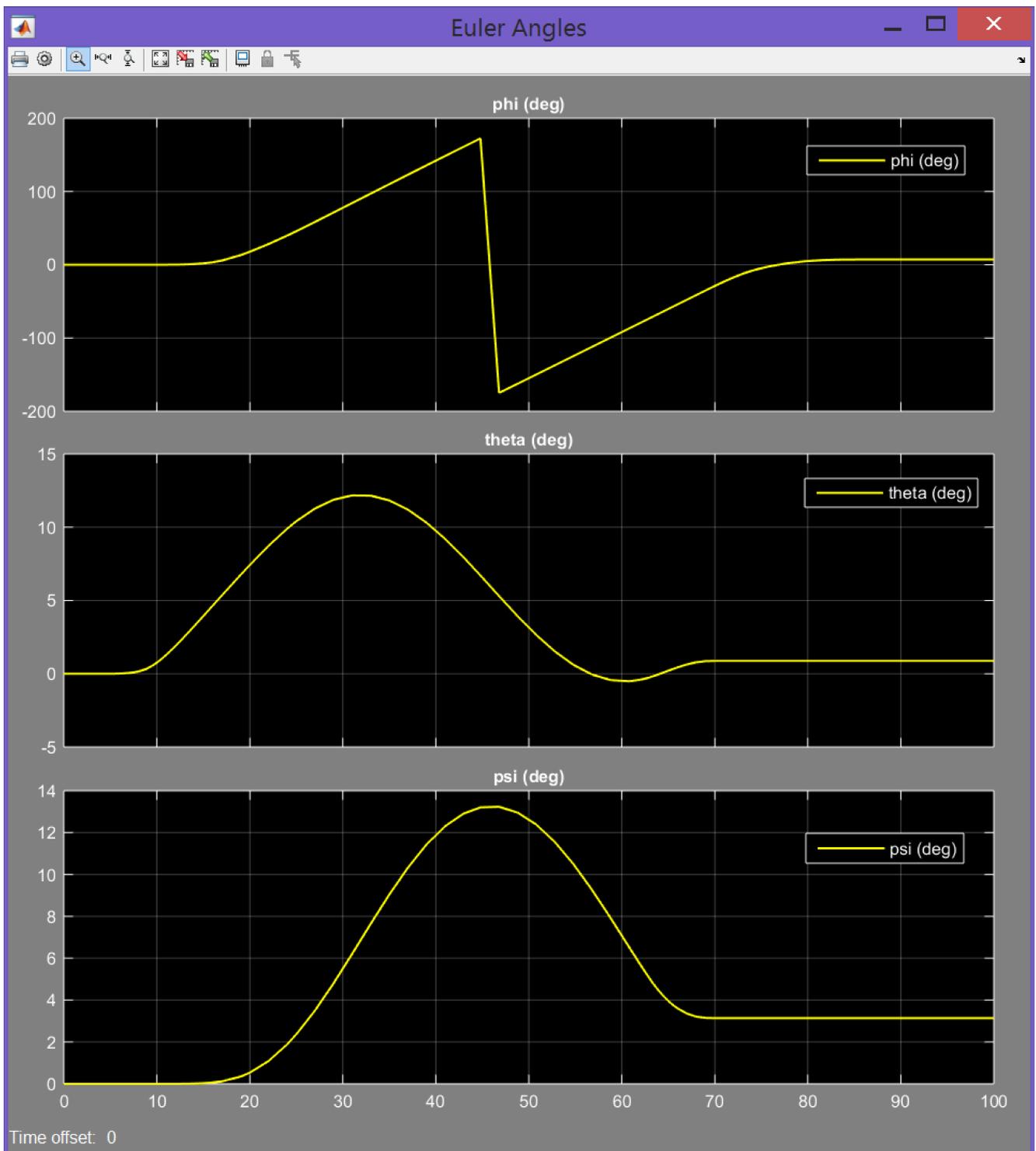


Figura 3.25 Storie temporali degli angoli di Eulero.

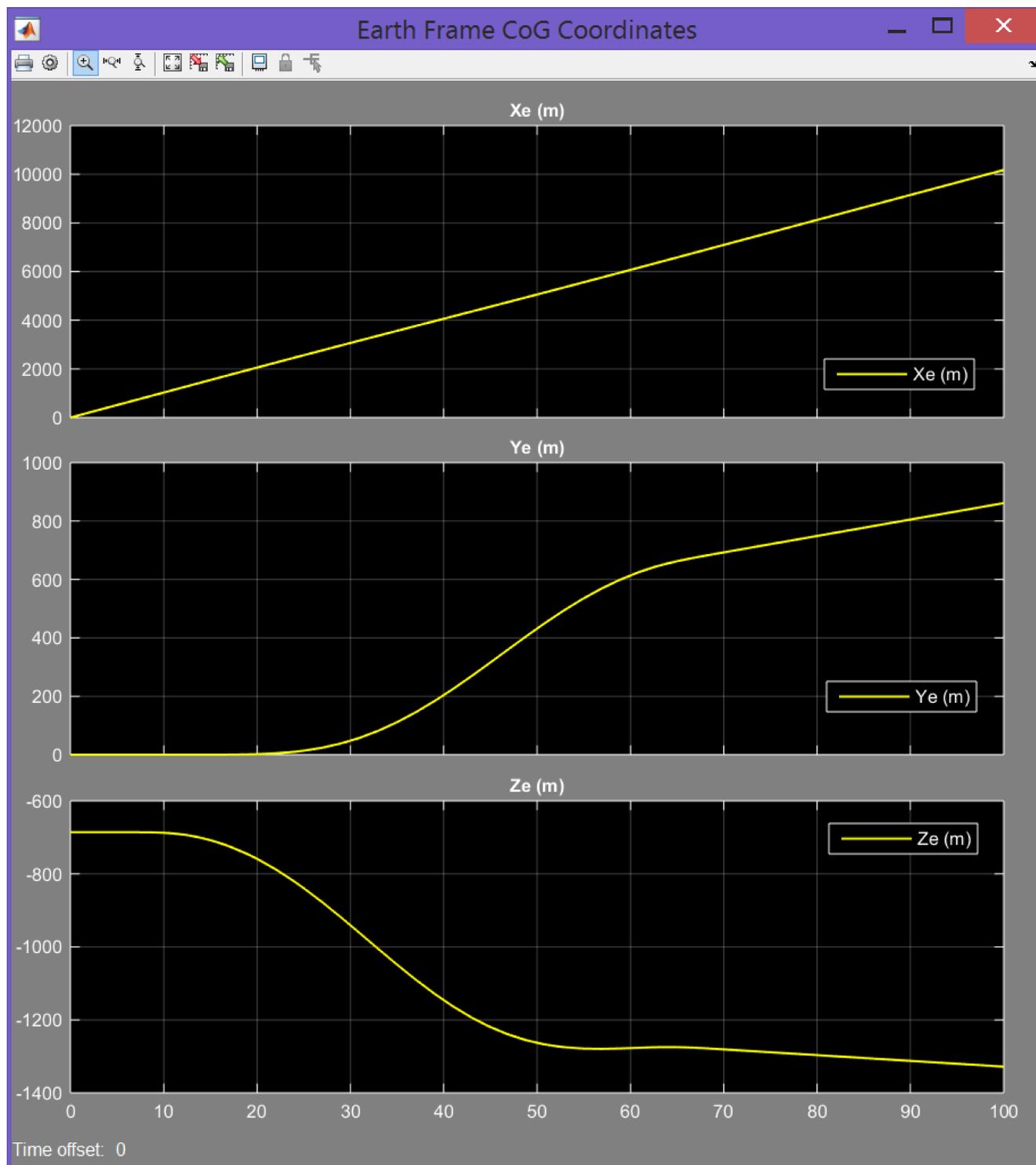


Figura 3.26 Storie temporali delle coordinate del baricentro. Blocco Scope non mostrato nella figura 3.22; si lascia per esercizio al lettore il compito di aggiungerne uno ottenendo i grafici qui rappresentati.

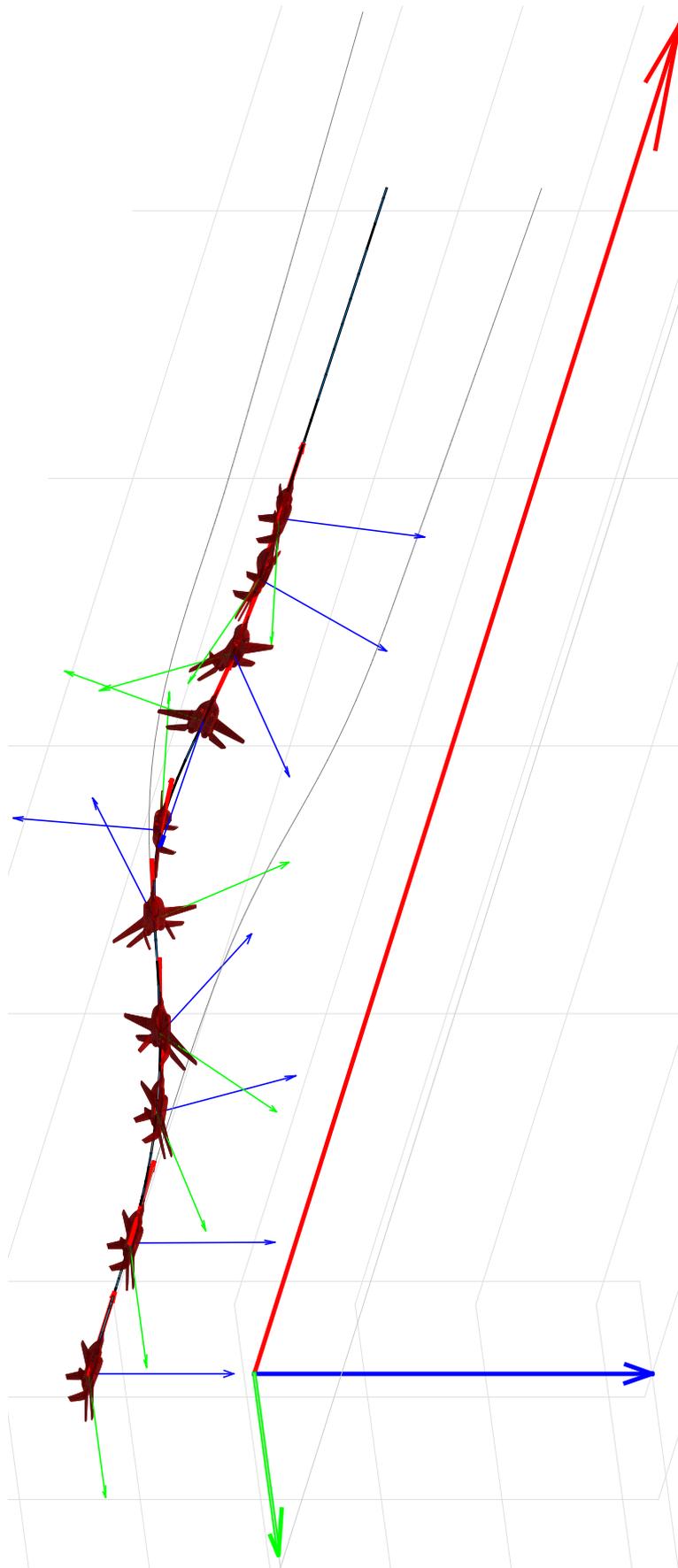


Figura 3.27 Evoluzione di rollio ottenuta con il modello Simulink dell'esercizio 3.1. La condizione $\alpha_B(t) = \beta(t) = 0$ imposta per semplificare il calcolo della traiettoria comporta una condizione di tangenza, punto per punto, dell'asse velivolo x_B alla traiettoria.

Esercizio 3.5: *Manovre acrobatiche in Simulink*

Si risolvano con Simulink i problemi proposti negli esercizi 3.1, 3.2, 3.4.



Bibliografia

- [1] W. R. Hamilton, *Lectures on Quaternions*, Hodges & Smith, 1853.
- [2] O. Rodrigues, “Des lois géométriques qui régissent les déplacements d’un système solide dans l’espace, et de la variation des coordonnées provenant de ses déplacements considérées indépendamment des causes qui peuvent les produire”, *Journal des Mathématiques Pures et Appliquées*, vol. 5, 1840.
- [3] E. Salamin, “Application of Quaternions to Computation with Rotations”, Working paper, Stanford AI Lab, 1979.
- [4] A. P. Yefremov, “Quaternions: Algebra, Geometry and Physical Theories”, *Hypercomplex Numbers in Geometry and Physics*, vol. 1, 2004.
- [5] Schwab A. L., “Quaternions, Finite Rotations and Euler Parameters”, Course notes on Applied Multibody Dynamics, Delft University of Technology, Laboratory for Engineering Mechanics, 2003.
<http://tam.cornell.edu/~als93/quaternion.pdf>.
- [6] AIAA/ANSI, *Recommended Practice for Atmospheric and Space Flight Vehicle Coordinate Systems*. R-004-1992, 1992.
- [7] G. H. Bryan, *Stability in Aviation: An Introduction to Dynamical Stability as Applied to the Motions of Aeroplanes*. Macmillan and Co., Limited, London, 1911.
- [8] D. J. Diston, *Computational Modelling of the Aircraft and the Environment. Volume 1, Platform Kinematics and Synthetic Environment*. John Wiley & Sons, Inc., 2009.
- [9] W. F. Phillips, *Mechanics of Flight*. John Wiley & Sons, Inc., 2004.
- [10] W. F. Phillips, “Phugoid Approximation for Conventional Airplanes”, *Journal of Aircraft*, Vol. 37, No. 1, January-February 2000.
- [11] W. F. Phillips, “Improved Closed-Form Approximation for Dutch-Roll”, *Journal of Aircraft*, Vol. 37, No. 1, May-June 2000.
- [12] R. Stengel, *Flight Dynamics*. Princeton University Press, Princeton, 2004.
- [13] M. R. Napolitano, *Aircraft Dynamics: From Modeling to Simulation*. John Wiley, 2012.

- [14] D. K. Schmidt, *Modern Flight Dynamics*. McGraw-Hill, 2010.
- [15] B. Stevens, F. Lewis, *Aircraft Control and Simulation*. John Wiley & Sons, Inc., 1992.
- [16] D. Stinton, *The Anatomy of the Airplane* (2nd edition). American Institute of Aeronautics and Astronautics, 1998.
- [17] B. Etkin, *Dynamics of Flight, Stability and Control*. John Wiley & Sons, New York, 1982.
- [18] M. Calcara, *Elementi di dinamica del velivolo*. Edizioni CUEN, Napoli, 1988.
- [19] L. V. Schmidt, *Introduction to Aircraft Flight Dynamics*. AIAA Education Series, 1998.
- [20] W. J. Duncan, *Control and Stability of Aircraft*. Cambridge University Press, Cambridge, 1952.
- [21] R. Jategaonkar, *Flight Vehicle System Identification: A Time Domain Methodology*. Progress in Astronautics and Aeronautics Series, 2006.
- [22] C. D. Perkins, R. E. Hage, *Aircraft Performance, Stability and Control*. John Wiley & Sons, New York, 1949.
- [23] J. R. Wright, J. E. Cooper, *Introduction to Aircraft Aeroelasticity and Loads*. John Wiley & Sons, Inc., 2007.
- [24] V. Losito, *Fondamenti di Aeronautica Generale*. Accademia Aeronautica, Napoli, 1994.
- [25] E. Torenbeek, H. Wittenberg, *Flight Physics*. Springer, Heidelberg, 2009.
- [26] P. H. Zipfel, *Modeling and Simulation of Aerospace Vehicle Dynamics*. Second Edition. AIAA Education Series, American Institute of Aeronautics and Astronautics, Reston, VA. 2007.
- [27] J. D. Mattingly, *Elements of Propulsion: Gas Turbines and Rockets*. AIAA Education Series, American Institute of Aeronautics and Astronautics, Reston, VA. 2006.
- [28] K. Hünecke, *Jet Engines. Fundamentals of Theory, Design and Operation*. Motorbooks International, 1997.
- [29] A. Linke-Diesinger, *Systems of Commercial Turbofan Engines*. Springer-Verlag, Berlin Heidelberg, 2008.
- [30] F. R. Garza, E. A. Morelli, "A Collection of Nonlinear Aircraft Simulations with MATLAB". NASA-TM-2003-212145, January 2003.
- [31] Voce WGS84 su *Wikipedia*:
http://en.wikipedia.org/wiki/World_Geodetic_System

- [32] Anonimo, *Department of Defense World Geodetic System 1984. Its Definition and Relationship with Local Geodetic Systems*. NIMA TR8350.2, Third Edition, Amendment 2. National Imagery and Mapping Agency, US Department of Defense, 2004.
- [33] J. Roskam, *Airplane Flight Dynamics and Automatic Flight Controls*. DARcorporation, 2001.
- [34] H. T. Schlichting, E. A. Truckenbrodt, *Aerodynamics of the Aeroplane*. McGraw Hill Higher Education, 2nd edition, 1979.
- [35] M. M. Munk, “The aerodynamic forces on airship hulls”. NACA-TR-184, 1924.
- [36] A. Silverstein, S. Katzoff, “Aerodynamic characteristics of horizontal tail surfaces”. NACA-TR-688, 1940.
- [37] R. I. Sears, “Wind-tunnel data on the aerodynamic characteristics of airplane control surfaces”. NACA-WR-L-663, 1943.
- [38] E. Garner, “Wind-tunnel investigation of control-surface characteristics XX: plain and balanced flaps on an NACA 0009 rectangular semispan tail surface”. NACA-WR-L-186, 1944.
- [39] J. D. Brewer, M. J. Queijo, “Wind-tunnel investigation of the effect of tab balance on tab and control-surface characteristics”. NACA-TN-1403, 1947.
- [40] S. M. Crandall, H. E. Murray, “Analysis of available data on the effects of tabs on control-surface hinge moments”. NACA-TN-1049, 1946.
- [41] B. W. McCormick, *Aerodynamics, Aeronautics, and Flight Mechanics*. John Wiley & Sons, 1979.
- [42] B. N. Pamadi, *Performance, Stability, Dynamics and Control of Airplanes*. AIAA Education Series, 1998.
- [43] A. Tewari, *Atmospheric and Space Flight Dynamics. Modelling and Simulation with Matlab and Simulink*. Birkhäuser, Berlin, 2007.
- [44] D. Howe, *Aircraft Loading and Structural Layout*. AIAA Education Series, 2004.
- [45] P. Morelli, *Static Stability and Control of Sailplanes*. Levrotto & Bella, Torino, 1976.
- [46] L. Prandtl, O. G. Tietjens, *Fundamentals of Hydro and Aeromechanics*. Dover, 1957.
- [47] R. K. Heffley, W. F. Jewell, “Aircraft Handling Qualities Data”. NASA-CR-2144, December 1972.
- [48] H. P. Stough III, J. M. Patton Jr, S. M. SliWa, “Flight Investigation of the Effect of Tail Configuration on Stall, Spin, and Recovery Characteristics of a Low-Wing General Aviation Research Airplane”. NASA-TP-1987-2644, February 1987.

- [49] J. D. Anderson, *Fundamentals of Aerodynamics*. McGraw-Hill, 3rd edition, New York, 2001.
- [50] J. J. Bertin, *Aerodynamics for Engineers*. Prentice-Hall, 4th edition, Upper Saddle River, NJ, 2002.
- [51] J. Katz, A. Plotkin, *Low-Speed Aerodynamics*. Cambridge University Press, 2nd edition, Cambridge, England, U.K., 2001.
- [52] D. E. Hoak, *et al.*, “The USAF Stability and Control Datcom”. Air Force Wright Aeronautical Laboratories, TR-83-3048, 1960 (Revised 1978).
- [53] R. T. Jones, “A Note on the Stability and Control of Tailless Airplanes”. NACA Report 837, 1941.
- [54] D. P. Coiro, F. Nicolosi, A. De Marco, N. Genito, S. Figliolia, “Design of a Low Cost Easy-to-Fly STOL Ultralight Aircraft in Composite Material”. *Acta Polytechnica*, Vol. 45 no. 4, 2005, pp. 73-80; ISSN 1210-2709.
- [55] F. Nicolosi, A. De Marco, P. Della Vecchia, “Flight Tests, Performances and Flight Certification of a Twin-Engine Light Aircraft”. *Journal of Aircraft*, Vol 48, No. 1, January-February 2011.
- [56] F. Nicolosi, A. De Marco, P. Della Vecchia, “Parameter Estimation and Flying Qualities of a Twin-Engine CS23/FAR23 Certified Light Aircraft”. AIAA-2010-7947, AIAA Atmospheric Flight Mechanics Conference, Toronto, 2010.
- [57] B. Etkin, *Dynamics of Atmospheric Flight*, Dover Publications, 2005.
- [58] L. Mangiacasale, *Flight Mechanics of a μ -Airplane*, Edizioni Libreria CLUP, Milano, 1998.
- [59] G. Mengali, *Elementi di Dinamica del Volo con Matlab*, Edizioni ETS, Pisa, 2001.
- [60] R. Nelson, *Flight Stability and Automatic Control*, McGraw-Hill, 1989.
- [61] Y. Li, M. Nahon, “Modeling and simulations of airship dynamics”, *Journal of Guidance, Controls and Dynamics*, Vol 30, No. 6, November-December 2007.
- [62] Y. Fan, F. H. Lutze, E. M. Cliff, “Time-Optimal Lateral Maneuvers of an Aircraft”, *Journal of Guidance, Controls and Dynamics*, Vol 18, No. 5, September-October 1995.
- [63] J. N. Nielsen, *Missile Aerodynamics*, AIAA, Cambridge, MA, 1988.
- [64] T. I. Fossen, *Guidance and Control of Ocean’s Vehicles*, Wiley, New York, 1998.
- [65] J. N. Newman, *Marine Hydrodynamics*, MIT Press, Cambridge, MA, 1977.
- [66] E. L. Duke, R. F. Antoniewicz, K. D. Krambeer, “Derivation and Definition of a Linear Aircraft Model”. Technical Report NASA Reference Publication RP-1207, Research Engineering, NASA Ames Research Center and NASA Dryden Flight Research Facility, 1988.

- [67] G. A. Stagg, *An Unsteady Aerodynamic Model for Use in the High Angle of Attack Regime*. MS thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1998.
- [68] Y. Fan, *Identification of an Unsteady Aerodynamic Model up to High Angle of Attack Regime*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1997.
- [69] *MATLAB Users' Guide*. The Mathworks, 2003 ed edizioni successive.
<http://www.mathworks.com/>
<http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.html>
- [70] V. Comincioli, *Analisi numerica: metodi, modelli, applicazioni*. McGraw-Hill, 1990, seconda edizione 1995.
- [71] E. Kreyszig, *Advanced Engineering Mathematics*. John Wiley & Sons, seventh edition, 1993.
- [72] C. de Boor, *A Practical Guide to Splines*. Springer-Verlag, 1978.
- [73] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical Recipes in Fortran: The Art of Scientific Computing*. Cambridge University Press, 1992.
- [74] G. Dahlquist, A. Bjorck, *Numerical Methods. Volume I: Fundamentals of Numerical Discretization*. John Wiley & Sons, 1988.
- [75] R. D. Richtmyer, K. W. Morton, *Difference Methods for Initial Value Problems*. Wiley-Interscience, 1967.
- [76] C. Hirsch, *Numerical Computation of Internal and External Flows*. John Wiley & Sons, 1994.
- [77] R. D. Finck, "USAF Stability and Control Datcom". AFWAL-TR-83-3048, October 1960, Revised 1978.
- [78] S. R. Vukelich, J. E. Williams, "The USAF Stability and Control Digital Datcom". AFFDL-TR-79-3032, Volume I, April 1979, Updated by Public Domain Aeronautical Software 1999.
- [79] W. B. Blake, "Prediction of Fighter Aircraft Dynamic Derivatives Using Digital Datcom". AIAA-85-4070, AIAA Applied Aerodynamics Conference, Colorado Springs, Colorado, 1985.
- [80] Autori Vari, Distribuzione ufficiale di Digital Datcom, sito internet:
<http://wpage.unina.it/agodemar/DSV-DQV/Digital-Datcom-Package.zip>
- [81] B. Galbraith, "Digital Datcom+", Holy Cows, Inc., sito internet: <http://www.holycows.net/datcom/>