

**POLITECNICO DI TORINO**

**III Facoltà di Ingegneria  
Corso di Laurea specialistica in Ingegneria Elettronica**

**Tesi di laurea**

**Progetto AraMiS:  
Progetto del software indirizzato alla  
determinazione ed al controllo  
dell'assetto del satellite**



Relatore:  
prof. Claudio Passerone

Candidato:  
Manuel Bonjean

2010

*Ai miei genitori*

# Sommario

Scopo della presente tesi è l'implementazione ed il relativo collaudo del sistema software ACS (Attitude Control System), indirizzato alla determinazione ed al controllo dell'assetto del satellite modulare AraMiS nello spazio. La determinazione ed il controllo dell'assetto del satellite rappresentano dunque il problema su cui è focalizzato lo studio.

Durante l'attività sono stati opportunamente definiti i sistemi di riferimento utilizzati nella descrizione delle geometrie spaziali in esame e sono stati sviluppati degli appositi algoritmi, scritti in linguaggio di programmazione C e C++, finalizzati alla soluzione del problema in analisi. È stata posta molta attenzione nello sviluppo di algoritmi di calcolo il più possibile indipendenti dalla struttura modulare del satellite cui risultano essere rivolti. In questo modo lo stesso software risulta utilizzabile senza, o al limite con ridotte, modifiche su un gran numero di satelliti AraMiS, aventi configurazioni anche molto differenti tra loro.

La prima fase dell'attività è stata focalizzata sullo studio dell'orbita del satellite. In questa sede sono stati opportunamente sviluppati due algoritmi di calcolo della posizione del satellite rispetto al sistema di riferimento inerziale, preliminarmente definito. Il primo algoritmo consiste nell'applicazione diretta della teoria sviluppata da Keplero e risulta, pertanto, semplice e poco accurato. La scarsa accuratezza è dovuta essenzialmente alla modellazione del problema come un sistema formato soltanto da due corpi, la Terra ed il satellite, ed alla totale mancanza di contemplazione delle perturbazioni presenti nel sistema analizzato. Il secondo algoritmo menzionato è raffigurato dal software SGP4 (Simplified General Perturbation), rilasciato da NASA/NORAD. Questo software è stato analizzato ed opportunamente adattato, in vista dell'integrazione dello stesso nel sistema ACS. In particolare sono state opportunamente eliminate tutte le routine riguardanti le orbite satellitari alte, dal momento che AraMiS è indirizzato verso applicazioni spaziali in orbita bassa. Per entrambi gli algoritmi, gli elementi orbitali, opportunamente codificati nel formato della TLE, definito dal NORAD, ed il tempo in cui si desidera conoscere l'informazione circa la posizione del satellite, rappresentano i parametri d'ingresso. Il tempo è da intendere come il lasso di tempo trascorso dalla particolare data di creazione della TLE in analisi. L'algoritmo SGP4 risulta, pertanto, essere più complesso e più accurato del precedente. L'accuratezza dell'algoritmo è approssimativamente pari ad 1 Km, nel periodo di validità di una TLE (una settimana generalmente). In fase di collaudo dell'algoritmo si è fatto uso dell'ambiente STK ed i valori di posizione da esso calcolati sono stati fissati come i riferimenti del problema in analisi. Il confronto dei risultati ottenuti ha dato esito positivo.

La determinazione dell'assetto del satellite si basa sulla lettura delle informazioni fornite dai sensori di posizione del Sole e dai sensori di campo magnetico (vettori di misura), disposti diversamente sulla struttura del satellite, e sulla conoscenza dei modelli della posizione del Sole e del vettore campo magnetico terrestre rispetto allo spazio inerziale (vettori di riferimento). Dopo uno studio dei parametri disponibili nella rappresentazione dell'orientamento di un oggetto nello spazio, è stato deciso di utilizzare la rappresentazione mediante matrice dei coseni direttori nel corso dello sviluppo dell'algoritmo di determinazione dell'assetto del satellite. Per quanto riguarda il modello della posizione del Sole è stato sviluppato un apposito algoritmo di calcolo, basato sulla conoscenza dei parametri orbitali dell'orbita della Terra. Questo algoritmo calcola la posizione del Sole in funzione di una particolare data e risulta utilizzabile per ricavare la posizione di tutti i pianeti del sistema solare. La posizione del Sole, calcolata mediante quest'algoritmo, è stata opportunamente confrontata con i valori tabulati in appositi documenti astronomici. Questo confronto ha dimostrato l'attendibilità dei risultati ottenuti. Per quanto riguarda il modello del campo magnetico terrestre, invece, si è fatto uso dei modelli IGRF e WMM. Durante l'implementazione dell'algoritmo in questione sono state analizzate, in principio, le particolari situazioni dettate dal satellite avente

soltanto sensori di posizione del Sole e dal satellite avente esclusivamente sensori di campo magnetico. Questa fase iniziale ha permesso lo sviluppo di alcuni algoritmi di calcolo preliminari, utili nella messa a punto dell'algoritmo generale finale. In particolare, è stata percepita la necessità di conoscere almeno due vettori di misura non paralleli e due vettori di riferimento non paralleli. Soltanto in questa situazione risulta, infatti, possibile specificare in modo completo l'assetto del satellite. Nel caso di satellite avente unicamente sensori di posizione del Sole, data la particolare geometria del sensore considerato, è possibile specificare l'assetto in modo completo, nel caso in cui almeno due facce non parallele, aventi il sensore, risultano illuminate dalla luce solare. Nel caso di satellite avente soltanto sensori di campo magnetico, invece, data la particolare geometria del sensore considerato, mediante due facce non parallele e provviste di sensore risulta possibile specificare soltanto un vettore di misura. Questo ha consigliato di impiegare tutti i sensori di campo magnetico presenti a bordo satellite nel calcolo di un unico vettore di misura, più accurato. Così facendo, evidentemente, un satellite fornito esclusivamente di sensori di campo magnetico non ha la possibilità di conoscere completamente il suo assetto nello spazio, ma esclusivamente l'orientamento di un suo asse. Queste considerazioni hanno premesso l'implementazione di un algoritmo di determinazione dell'assetto del satellite generalizzato, basato sulla conoscenza di un numero variabile di vettori di misura e di riferimento. L'algoritmo finale si basa pertanto sulla lettura delle informazioni fornite da un numero variabile di sensori di posizione del Sole e di sensori di campo magnetico e sul riempimento delle righe di alcune matrici temporanee, utilizzate nel calcolo definitivo dell'assetto del satellite. Quest'algoritmo risulta essere indipendente dalla configurazione del satellite, adattabile al numero di sensori presenti (la possibilità di guasti sugli stessi è contemplata) ed integrabile con eventuali sensori d'assetto inseriti in futuro. I risultati ottenuti dall'algoritmo sono stati opportunamente confrontati con i valori attesi, costruendo delle opportune matrici d'assetto d'errore e calcolando gli angoli di Eulero ad esse associati, al fine di riuscire ad interpretare meglio i valori ottenuti. I risultati finali dimostrano l'attendibilità dell'algoritmo sviluppato.

Il controllo dell'assetto del satellite si basa sull'attuazione di opportune ruote di reazione disposte diversamente sulla struttura del satellite. Le ruote di reazione rappresentano un esempio di attuatore largamente utilizzato in ambito aerospaziale. Le rotazioni di questi attuatori generano del momento angolare e sono quindi in grado di governare la dinamica del satellite. Il tipo di controllo sviluppato risulta essere un controllo ad anello aperto, finalizzato allo spostamento punto-punto del satellite. Ovviamente un requisito fondamentale del controllo dell'orientamento del satellite è dato dalla conoscenza dell'assetto attuale dello stesso. Durante l'attività sono state studiate le principali definizioni della dinamica del satellite nello spazio ed è stata fissata la tecnica di controllo da adottare. Inoltre è stato fissato il formato del comando previsto in ricezione dal sistema ACS. Il comando previsto è composto di una matrice di rotazione, riferita allo spazio inerziale, ed un tempo. La prima consiste nella rappresentazione dell'assetto desiderato del satellite rispetto allo spazio inerziale, il secondo consiste nella specifica della tempistica massima in cui portare a termine la manovra del satellite. L'approccio adottato nello sviluppo dell'opportuno algoritmo di controllo dell'assetto del satellite consiste nella scomposizione della rotazione, necessaria ad allineare l'assetto del satellite secondo il particolare assetto desiderato, nella sequenza di tre rotazioni successive eseguite attorno agli assi principali d'inerzia del satellite. In questo modo si generano delle rotazioni del satellite più fluide e più semplici da analizzare (prive di moti di precessione), in virtù delle caratteristiche dettate dagli assi principali d'inerzia. Durante la scomposizione della rotazione in analisi è stato deciso di utilizzare la rappresentazione dell'assetto mediante quaternioni unitari. Questi importanti parametri d'assetto permettono, infatti, di eliminare eventuali singolarità della rappresentazione e di effettuare una fattorizzazione relativamente semplice, secondo delle direzioni prestabilite. L'algoritmo si preoccupa quindi di ricavare il quaternioni associato alla particolare matrice di rotazione che si vuole scomporre e di fattorizzarlo nel prodotto di tre quaternioni speciali, riferiti agli assi principali d'inerzia del satellite. Ottenute le tre singole rotazioni è stato possibile sviluppare un algoritmo di previsione del comando (momento angolare)

che le ruote di reazione devono generare al fine di ottenere realmente le rotazioni del satellite calcolate. Si tenga presente che, nel caso generale, non è possibile controllare direttamente una rotazione attorno ad un asse principale, dal momento che le ruote di reazione non sono, sempre nel caso generale, disposte in modo da avere l'asse di rotazione coincidente con questi assi. In questa fase sono stati sviluppati due algoritmi a hoc, uno indirizzato verso i satelliti AraMiS con sezione cubica o ottagonale ed uno indirizzato verso i satelliti AraMiS con sezione esagonale. In entrambi i casi i satelliti in esame sono opportunamente forniti di una terna d'assi ortogonale e solidale con la struttura meccanica, definita in fase di progetto. I satelliti a configurazione esagonale hanno la peculiarità di non ammettere la presenza di alcuna ruota di reazione su un asse della terna appena citata. Questo tuttavia non complica eccessivamente l'algoritmo di controllo. Sostanzialmente si è deciso di far fronte alla mancanza della ruota di reazione sul particolare asse, generando del momento angolare come sovrapposizione dei contributi apportati dalle rotazioni delle ruote di reazione aventi l'asse di rotazione disposto sullo stesso piano dell'asse privo di ruote. Gli algoritmi sviluppati si basano sulla lettura del numero di ruote di reazione presenti sul satellite (la possibilità di guasti sulle ruote è contemplata) e sul calcolo del comando in funzione di questo numero. Con lo scopo di effettuare il collaudo dell'algoritmo di controllo dell'assetto del satellite è stato opportunamente sviluppato un modello, in ambiente Simulink, della dinamica del satellite. Così facendo è stato possibile verificare il funzionamento del software sviluppato fissando un assetto del satellite iniziale ed un assetto del satellite desiderato. I risultati ottenuti sono stati visualizzati generando dei grafici raffiguranti gli angoli di Eulero dell'assetto e le velocità angolari del satellite e mostrano chiaramente l'attendibilità degli algoritmi implementati.

L'intero sistema ACS è stato opportunamente descritto in UML utilizzando l'ambiente Visual Paradigm. La descrizione del sistema è avvenuta mediante il diagramma dei casi d'uso ed il diagramma delle classi. Utilizzando il tool Instant Generator di Visual Paradigm, il sistema ACS, scritto in principio in linguaggio C, è stato tradotto in linguaggio C++. Il software C++ generato è stato nuovamente collaudato ed ha riconfermato l'attendibilità del sistema.

# Indice

1	Introduzione .....	10
2	I principali sistemi di riferimento utilizzati .....	14
2.1	La definizione dei sistemi di riferimento .....	14
2.2	Il sistema di riferimento solidale con la rotazione della Terra.....	15
2.3	Le coordinate celesti .....	17
2.3.1	Il sistema di coordinate orizzontale .....	17
2.3.2	Il sistema di coordinate orario.....	18
2.3.3	Il sistema di coordinate equatoriali .....	19
2.4	I principali sistemi di coordinate orbitali .....	20
2.4.1	Il sistema di riferimento PQW .....	20
2.4.2	Il sistema di riferimento RSW .....	21
3	I principali concetti dell'astrodinamica e le orbite kepleriane.....	23
3.1	Le grandezze principali utilizzate nella descrizione dell'orbita del satellite .....	23
3.2	Gli elementi orbitali classici .....	27
3.3	Il formato delle Two-Line Element set (TLE) fornito dal NORAD.....	28
3.4	Il problema di Keplero .....	29
4	Il calcolo della posizione del satellite rispetto al sistema geocentrico inerziale IJK.....	31
4.1	L'algoritmo di calcolo della posizione del satellite rispetto al sistema di riferimento geocentrico inerziale IJK basato sulla teoria sviluppata da Keplero .....	31
4.1.1	La soluzione dell'equazione di Keplero.....	31
4.1.2	Il calcolo della posizione del satellite rispetto al sistema IJK.....	33
4.2	Le principali perturbazioni dell'orbita satellitare.....	35
4.2.1	Le perturbazioni introdotte dalla presenza del terzo corpo.....	36
4.2.2	Le perturbazioni dovute alla non sfericità della Terra .....	37
4.2.3	Le perturbazioni causate dall'attrito aerodinamico dell'atmosfera.....	38
4.2.4	Le perturbazioni causate dalla pressione esercitata dalla radiazione solare .....	39
4.3	Il calcolo della posizione del satellite rispetto al sistema geocentrico inerziale IJK mediante l'algoritmo SGP4.....	40
4.4	Il confronto dei risultati ottenuti mediante l'ambiente STK.....	43
5	Il problema della determinazione dell'assetto nello spazio .....	45
5.1	Il problema generale della determinazione dell'assetto del satellite e la matrice dei coseni direttori.....	45
5.2	La rappresentazione dell'assetto del satellite mediante l'asse e l'angolo di Eulero .....	48
5.3	La rappresentazione dell'assetto del satellite mediante i quaternioni unitari .....	50
5.4	La rappresentazione dell'assetto del satellite mediante gli angoli di Eulero .....	52
6	I sistemi di riferimento del satellite e delle tiles ed i sensori d'assetto del satellite .....	55
6.1	I sistemi di riferimento del satellite e delle tiles e le configurazioni del satellite .....	55
6.1.1	I sistemi di riferimento del satellite e delle tiles .....	55
6.1.2	Il satellite a configurazione cubica.....	56
6.1.3	I satelliti a configurazione esagonale ed ottagonale.....	57
6.2	I sensori d'assetto.....	59
6.2.1	Il sensore di posizione del Sole.....	59
6.2.2	Il sensore di campo magnetico.....	60
6.2.3	Il giroscopio .....	62
7	I modelli .....	64
7.1	Il modello della posizione del Sole rispetto al sistema geocentrico inerziale IJK.....	64

7.1.1	L'algorithmo di calcolo della posizione del Sole.....	64
7.1.2	Verifica dell'algorithmo di calcolo implementato .....	67
7.2	Il modello del campo magnetico terrestre.....	68
7.2.1	Il campo magnetico terrestre.....	68
7.2.2	I parametri utilizzati nella descrizione del campo magnetico terrestre.....	70
7.2.3	I principi su cui si basano i modelli del campo magnetico terrestre.....	72
8	La determinazione dell'assetto del satellite .....	75
8.1	La determinazione dell'assetto mediante sensori di posizione del Sole.....	75
8.2	La determinazione dell'assetto mediante sensori di campo magnetico.....	79
8.2.1	L'algorithmo di determinazione del vettore campo magnetico terrestre rispetto al sistema UVW del satellite.....	80
8.3	La determinazione dello spin del satellite.....	83
8.3.1	L'algorithmo di determinazione dello spin del satellite.....	83
8.4	La generalizzazione della soluzione del problema della determinazione dell'assetto.....	85
8.4.1	L'algorithmo di determinazione dell'assetto del satellite.....	85
9	La dinamica del satellite nello spazio .....	89
9.1	Le grandezze fondamentali .....	89
9.2	Gli assi principali d'inerzia .....	92
9.3	Le equazioni dinamiche .....	94
10	Il controllo dell'assetto del satellite .....	97
10.1	Le ruote di reazione.....	97
10.1.1	Il principio di funzionamento.....	98
10.2	L'algorithmo di controllo dell'orientamento del satellite.....	99
10.2.1	Il calcolo della matrice di manovra.....	99
10.2.2	La scomposizione della matrice di manovra nella sequenza di tre singole rotazioni successive mediante la scomposizione dei quaternioni unitari.....	101
10.2.3	Il calcolo del comando da fornire alle ruote di reazione.....	105
10.3	La simulazione dell'algorithmo mediante Simulink.....	111
11	La descrizione del sistema ACS mediante il linguaggio UML.....	123
11.1	Il paradigma Object-Oriented .....	123
11.2	Il linguaggio UML .....	124
11.2.1	Gli Use Case Diagrams .....	125
11.2.2	I Class Diagrams .....	126
11.3	Il diagramma dei casi d'uso del sistema ACS.....	128
11.4	Il diagramma delle classi del sistema ACS.....	129
11.4.1	Test.....	133
11.4.2	ACS.....	135
11.4.3	Orbits.....	136
11.4.4	elements_s.....	136
11.4.5	Magnet .....	138
11.4.6	Position.....	142
11.4.7	elsetrec .....	144
11.4.8	Satellite.....	148
11.4.9	Tile .....	151
12	I test effettuati sul sistema ACS.....	154
12.1	Le grandezze di test.....	154
12.2	Test 1: valori esatti forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 5 magnetometri funzionanti .....	158
12.3	Test 2: rumore sui valori forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 5 magnetometri funzionanti .....	163

12.4	Test 3: rumore sui valori forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 2 magnetometri funzionanti .....	167
12.5	Test 4: rumore sui valori forniti dai sensori, 1 tile illuminata con sensore di Sole funzionante e 2 magnetometri funzionanti.....	168
12.6	Test 5: determinazione dello spin del satellite .....	168
13	Conclusioni .....	170
	Bibliografia .....	172

# Elenco delle figure

Figura 1.1 Architettura modulare del satellite AraMiS.....	13
Figura 1.2 Satellite AraMiS a configurazione prismatica.....	13
Figura 2.1 Uso di latitudine e longitudine nella misura della posizione di un punto sulla Terra. ....	15
Figura 2.2 Sistema di riferimento solidale con la rotazione della Terra (coordinate rettangolari). ...	16
Figura 2.3 Sistema di riferimento celeste.....	18
Figura 2.4 Sistema inerziale geocentrico. ....	19
Figura 2.5 Sistema di riferimento PQW.....	20
Figura 2.6 Sistema di riferimento RSW.....	21
Figura 3.1 Geometria di un'orbita ellittica con i suoi parametri principali. ....	24
Figura 3.2 Tipi di sezioni coniche.....	25
Figura 3.3 Elementi orbitali classici. ....	27
Figura 3.4 Formato delle Two-Line Element set. ....	28
Figura 3.5 Anomalia vera ed anomalia eccentrica. ....	30
Figura 4.1 Semiparametro p.....	34
Figura 4.2 Variazioni secolari e periodiche di un elemento orbitale. ....	35
Figura 4.3 Diagramma di flusso di massima dell' algoritmo SGP4. ....	41
Figura 4.4 Errori algoritmo SGP4 in double e float.....	42
Figura 4.5 Simulazione di un'orbita mediante l'ambiente STK. ....	43
Figura 4.6 Errori SGP4 fornito dal NORAD e SGP4 di STK.....	44
Figura 5.1 Problema della determinazione dell'assetto nello spazio. ....	46
Figura 5.2 Rotazione attorno ad un asse. ....	48
Figura 6.1 Satellite a configurazione cubica.....	56
Figura 6.2 Satellite a configurazione esagonale.....	57
Figura 6.3 Satellite a configurazione ottagonale.....	58
Figura 6.4 Prototipo del satellite AraMiS a configurazione esagonale.....	58
Figura 6.5 Geometria del sensore di posizione del Sole. ....	59
Figura 6.6 Magnetoresistenza a film sottile. ....	61
Figura 6.7 Lettura della velocità angolare attorno all'asse Z della tile. ....	62
Figura 6.8 Principio di funzionamento del giroscopio a vibrazione. ....	63
Figura 7.1 Orbite dei pianeti del sistema solare. ....	68
Figura 7.2 Campo magnetico terrestre.....	69
Figura 7.3 Gli elementi magnetici.....	70
Figura 8.1 Coordinate sferiche della direzione del Sole rispetto al sistema di riferimento XYZ. ....	76
Figura 8.2 Indeterminazione dell'assetto. ....	79
Figura 8.3 Geometria del sensore di campo magnetico. ....	81
Figura 8.4 Diagramma di flusso dell' algoritmo di determinazione dell'assetto. ....	88
Figura 9.1 Momento angolare di un punto materiale. ....	90
Figura 9.2 Momento della forza applicata ad un punto materiale. ....	95
Figura 10.1 Power Management tile con ruota di reazione. ....	97
Figura 10.2 Principio di funzionamento della ruota di reazione.....	98
Figura 10.3 Momento angolare in funzione del tempo.....	106
Figura 10.4 Satellite con numerose ruote di reazione.....	107
Figura 10.5 Mancanza di ruote di reazione su un asse geometrico.....	109
Figura 10.6 Diagramma di flusso dell' algoritmo di controllo dell'assetto. ....	110
Figura 10.7 Assetto del satellite (angoli di Eulero). ....	116
Figura 10.8 Zoom assetto del satellite (angoli di Eulero). ....	117

Figura 10.9 Velocità angolari del satellite relative agli assi ABC. ....	118
Figura 10.10 Zoom velocità angolare relativa alla prima rotazione (asse principale). ....	119
Figura 10.11 Velocità angolari del satellite rispetto agli assi UVW. ....	120
Figura 10.12 Zoom velocità angolare relativa alla prima rotazione (asse del satellite). ....	121
Figura 10.13 Descrizione della dinamica del satellite mediante lo schema a blocchi in Simulink. ....	122
Figura 11.1 Diagramma dei casi d'uso. ....	125
Figura 11.2 Tipi di relazione. ....	126
Figura 11.3 Rappresentazione di una classe in UML. ....	127
Figura 11.4 Diagramma dei casi d'uso del sistema ACS. ....	128
Figura 11.5 Suddivisione del sistema ACS in classi. ....	131
Figura 11.6 Diagramma delle classi del sistema ACS. ....	132
Figura 12.1 Orientamento delle tiles fissato rispetto al sistema del satellite UVW. ....	156
Figura 12.2 Assetto del satellite in IJK dopo la manovra. ....	161
Figura 12.3 Assetto del satellite in IJK dopo la manovra. ....	162
Figura 12.4 Assetto del satellite in IJK dopo la manovra. ....	165
Figura 12.5 Assetto del satellite in IJK dopo la manovra. ....	166

# Capitolo 1

## 1 Introduzione

Il lavoro di tesi si è focalizzato sul progetto del sistema software indirizzato alla determinazione ed al controllo dell'orientamento del satellite universitario AraMiS. Questo tipo di sistema è sovente indicato in letteratura con il nome ACS, acronimo di *Attitude Control System*.

AraMiS, acronimo di Architettura Modulare per Satelliti, è un progetto iniziato nel 2006 presso il Politecnico di Torino. Lo scopo del progetto risulta essere la realizzazione di satelliti di piccole dimensioni aventi una vera e propria struttura di tipo modulare. La modularità permette di ottenere un decremento importante del costo del progetto stesso e quindi fornisce all'università la possibilità d'interessarsi allo spazio. Il costo di una missione spaziale è, infatti, il principale ostacolo che si presenta dinanzi al comune interesse all'ambito spaziale, da parte di aziende ed università. L'idea alla base del progetto in questione risulta essere lo sviluppo di unità apposite intercomunicanti e distribuite, costruite con componenti COTS (Commercial Off The Shelf), al fine di incrementare la tolleranza ai guasti e consentire un grazioso degrado delle prestazioni, pur mantenendo i costi a livelli accettabili. I satelliti artificiali di piccole dimensioni sono generalmente suddivisi in funzione del loro peso. In particolare si parla di *micro-satellite*, quando la massa del satellite è compresa tra 10-100 Kg, si parla di *nano-satellite*, quando la massa è compresa tra 1-10 Kg, si parla di *pico-satellite*, quando la massa è compresa tra 0.1-1 Kg. Il modo più efficiente per ridurre il costo del progetto di un satellite di piccole dimensioni è quello di ridurre il più possibile i costi legati al progetto e non ricorrenti alla fabbricazione. Sono proprio questi costi, infatti, a rappresentare oltre il 90% del bilancio totale. La loro riduzione può essere ottenuta soltanto attraverso la condivisione della progettazione tra un gran numero di missioni spaziali. Il riutilizzo dei progetti risulta essere la logica alla base del progetto AraMiS, ovvero lo sviluppo di un'architettura modulare costituita da un piccolo numero di moduli flessibili che possono essere riutilizzati in diverse missioni. Il riutilizzo dello stesso modulo più volte permette, quindi, la suddivisione dei costi relativi al progetto, alla qualificazione ed al testing, e la riduzione delle tempistiche di attesa prima del lancio. L'architettura base del satellite AraMiS si basa su una o più mattonelle intelligenti (le *tiles*). La maggioranza di esse sono disposte sulla parte esterna del satellite e svolgono una duplice funzione, meccanica e funzionale. La parte interna del satellite è generalmente lasciata vuota. E' questa, infatti, ad ospitare il *payload* della particolare missione spaziale. Il *payload* rappresenta lo scopo per il quale avviene la particolare missione. Questa parte è progettata a hoc per ogni missione. In genere il *payload* è raffigurato da un particolare strumento scientifico presente a bordo (un telescopio, per esempio). Ogni tile è progettata, costruita e collaudata in quantità relativamente grandi.

Le tiles esterne possono essere sostanzialmente di due tipi:

- *Power management tiles*: sono composte principalmente da un pannello solare, una batteria ricaricabile per l'accumulo dell'energia, un caricabatteria, un modulo di housekeeping che si preoccupa di tener traccia delle tensioni, correnti, temperature della tile e dai sensori e dagli attuatori indirizzati alla determinazione ed al controllo dell'orientamento del satellite.
- *Telecommunication tiles*: sono composte principalmente da un ricetrasmittitore, un modem, un amplificatore di potenza (per la trasmissione) ed un amplificatore a basso rumore (per la ricezione) ed un sistema di antenne.

Componendo insieme un certo numero di queste mattonelle è possibile creare un satellite a forma cubica o prismatica con dimensioni più o meno grandi, in funzione della particolare missione spaziale.

Determinare l'orientamento di un satellite, o di un oggetto nello spazio in generale, significa misurare (mediante opportuni *sensori*), e quindi comprendere, com'è disposta una terna d'assi cartesiani ortogonali e solidali con l'oggetto in esame (opportunamente definita in fase di progetto), rispetto ad un'altra terna cartesiana presa come riferimento.

Controllarne l'orientamento, invece, significa agire sul suo orientamento attuale (mediante opportuni *attuatori*) e variarlo a proprio piacimento, secondo le particolari necessità dettate dalla specifica missione spaziale.

Il sistema sviluppato raffigura un sottosistema molto importante del satellite. Quest'ultimo si occupa della determinazione della posizione del satellite nell'orbita, della rilevazione del suo orientamento nello spazio e del controllo del suo orientamento (detto anche *puntamento* o *assetto*). Questo sottosistema, in virtù delle sue funzionalità offerte, ricopre un ruolo molto importante in qualsiasi missione spaziale.

Il primo passo verso la realizzazione di questo sistema consiste nello studio e nell'analisi delle geometrie nello spazio. In quest'ambito ricoprono una notevole importanza i parametri, utilizzati nella descrizione della particolare geometria spaziale presa in esame, e le definizioni di opportuni sistemi di riferimento. Durante la descrizione dell'orientamento del satellite sono, di fatto, disponibili diversi parametri e, nel corso del lavoro, sono stati analizzati confrontandone le caratteristiche, al fine di evidenziare i più consoni da utilizzare. Risulta di fondamentale importanza, inoltre, conoscere e comprendere quali sistemi di riferimento è conveniente adottare e come effettuare le dovute trasformazioni tra i vari sistemi di riferimento, definiti in fase di progetto. In una missione spaziale, infatti, vengono in genere definiti diversi sistemi di riferimento e risulta di sostanziale importanza riuscire a legarli insieme correttamente, mediante opportune operazioni geometriche. Una definizione non corretta di un sistema di riferimento è, infatti, una comune fonte di errore nell'analisi di una missione spaziale.

Durante la prima fase del lavoro, gli algoritmi di calcolo sviluppati sono stati implementati e testati utilizzando il linguaggio di programmazione C. Nella fase successiva, gli stessi algoritmi sono stati organizzati in *classi* di dati sfruttando il linguaggio descrittivo UML (*Unified Modeling Language*) e tradotti, infine, nel linguaggio di programmazione orientato agli oggetti C++. Tale accorgimento permette di sfruttare i benefici dettati dal paradigma *Object-Oriented*, tra cui il più importante risulta sicuramente essere dato dal riutilizzo del software sviluppato. Inoltre, durante l'attività, sono stati utilizzati ambienti software appositi. I più importanti sono STK (Satellite Tool Kit), utilizzato durante le simulazioni riguardanti l'orbita satellitare, e Matlab, sfruttato in modo particolare durante le verifiche dei risultati ottenuti, soprattutto per quanto riguarda i calcoli coinvolgenti matrici e vettori, e durante la visualizzazione grafica dei risultati stessi.

Da un punto di vista complessivo il lavoro di tesi svolto è possibile suddividerlo in quattro fasi distinte:

- Studio dell'orbita del satellite ed implementazione di opportuni algoritmi di calcolo della posizione dello stesso nell'orbita.
- Studio del problema della determinazione dell'assetto del satellite e sviluppo di opportuni algoritmi di calcolo, basati sulla lettura di opportuni sensori d'assetto presenti a bordo satellite.
- Studio del problema del controllo ad anello aperto dell'orientamento del satellite ed implementazione di appositi algoritmi di previsione del comando da fornire ad opportuni attuatori presenti a bordo satellite.
- Astrazione dei dati utilizzati negli algoritmi sviluppati e suddivisione degli stessi in apposite classi di dati.

Durante la prima fase dell'attività sono state quindi studiate ed analizzate le definizioni principali e fondamentali nella descrizione della posizione del satellite nella sua orbita attorno alla Terra. Successivamente sono stati implementati appositi algoritmi di calcolo indirizzati alla determinazione della posizione del satellite nell'orbita.

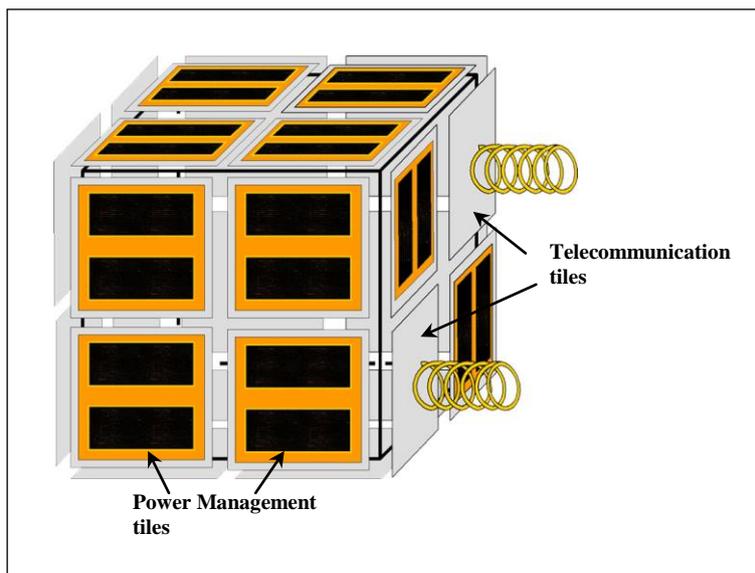
La seconda fase può, a sua volta, essere suddivisa in due fasi distinte. Queste ultime due possono essere riassunte nella determinazione di punti noti nello spazio, detti anche *vettori di riferimento* (come la posizione del Sole, o dei pianeti del sistema solare in generale, la posizione della Terra e la direzione ed il verso del vettore campo magnetico terrestre), e nello sviluppo di un opportuno algoritmo di determinazione dell'orientamento del satellite. L'algoritmo in questione sfrutta le posizioni note calcolate nella fase precedente (i modelli) e si occupa di ricavare l'orientamento del satellite, interrogando appositi sensori posizionati diversamente sulla struttura del satellite stesso (*sensori di posizione del Sole, sensori di campo magnetico, sensori giroscopici*) ed effettuando il confronto tra i valori di modello ed i valori misurati, detti anche *vettori di misura*.

Durante la terza fase sono state studiate ed analizzate le definizioni principali riguardanti la dinamica del satellite ed è stato sviluppato un algoritmo apposito indirizzato al controllo del suo puntamento. Il controllo dell'orientamento del satellite ricopre un ruolo fondamentale ed indispensabile in una missione spaziale. Si pensi, per esempio, alla necessità di comunicazione tra il satellite e la stazione di Terra. A tal scopo il satellite è provvisto di un opportuno modulo di antenne di comunicazione. Quest'ultimo, durante le comunicazioni desiderate, deve avere la possibilità di essere perennemente orientato verso la stazione comunicante. Molti satelliti, inoltre, ospitano a bordo strumenti scientifici (telescopi, oppure sensori vari in generale) i quali devono avere la possibilità di lavorare in direzioni opportune dello spazio, secondo la particolare applicazione richiesta ed in un determinato istante di tempo.

Durante la quarta ed ultima fase gli algoritmi sviluppati nelle fasi precedenti sono stati inglobati in un unico progetto, chiamato appunto sistema ACS. Quest'ultimo è stato opportunamente descritto in UML ed è stato analizzato e testato da un punto di vista complessivo, generando un programma principale interagente con tutti gli algoritmi rilevanti presenti al suo interno.

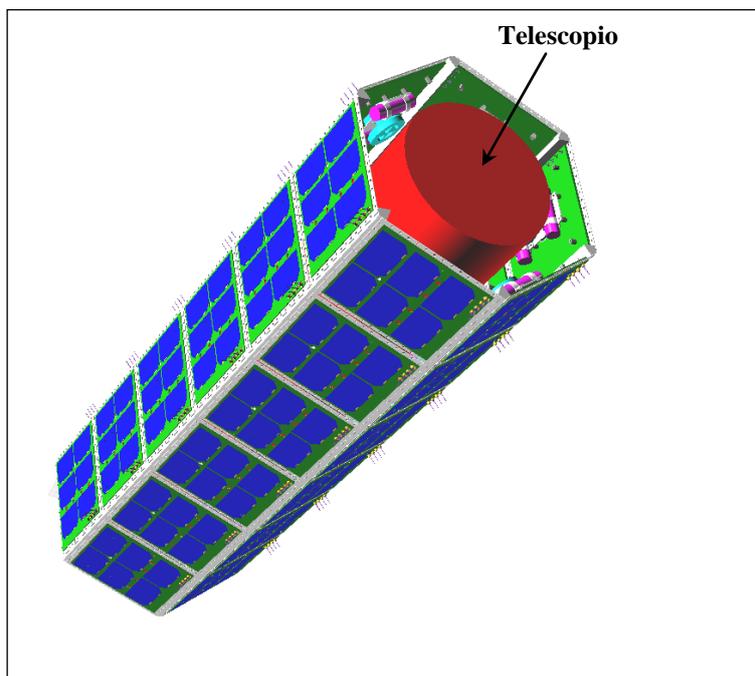
La tesi si compone di capitoli di carattere prettamente teorico, in cui sono riportate le definizioni delle grandezze principali utilizzate e le formulazioni dei problemi affrontati da un punto di vista generale, e di capitoli di carattere implementativo, in cui sono analizzati gli algoritmi di calcolo sviluppati.

Si riporta in figura 1.1 un esempio del satellite AraMiS a configurazione cubica avente quattro tiles per ogni faccia e due Telecommunication tiles (l'immagine riportata è disponibile in rete: <http://polimage.polito.it/aramis/img/aramis8.jpg>):



**Figura 1.1 Architettura modulare del satellite AraMiS.**

In figura 1.2, invece, si riporta una possibile configurazione prismatica del satellite AraMiS con sezione esagonale. Nella figura è evidenziato il telescopio presente al suo interno, il quale costituisce il particolare payload della missione spaziale cui risulta essere rivolto:



**Figura 1.2 Satellite AraMiS a configurazione prismatica.**

# Capitolo 2

## 2 I principali sistemi di riferimento utilizzati

In questo capitolo si riportano le definizioni di base utili nello studio di una geometria spaziale e sono descritti i principali sistemi di riferimento utilizzati nella determinazione della posizione di un oggetto nello spazio tridimensionale. In particolare sono descritti il sistema di riferimento solidale con la rotazione della Terra, il sistema di riferimento geocentrico inerziale (fisso nello spazio), i sistemi di riferimento celesti in generale ed i sistemi di riferimento basati sul piano orbitale.

### 2.1 La definizione dei sistemi di riferimento

Uno dei primi requisiti indispensabili nella descrizione dell'*orbita* di un satellite, ovvero la sua traiettoria nello spazio, è la definizione di un opportuno sistema di riferimento.

Un sistema di coordinate, indicato con i simboli XYZ, si definisce specificandone l'origine, il piano fondamentale e la direzione principale. Inoltre è necessario specificare il verso, o direzione positiva. Molti sistemi di riferimento hanno un verso dettato dalla regola della mano destra, la direzione positiva di ogni asse forma una terna ortogonale orientata come il pollice, l'indice, ed il medio della mano destra (indice diretto verso l'asse X, medio diretto verso l'asse Y e pollice diretto verso l'asse Z). Si utilizzano tre versori per descrivere la terna ortogonale, indicati con i simboli  $\hat{x}$ ,  $\hat{y}$ ,  $\hat{z}$ , e si esprime ogni altro vettore come combinazione lineare della base formata dai tre versori definiti. Completata la definizione del sistema di riferimento è possibile localizzare un punto qualsiasi, vale a dire l'orientamento del vettore congiungente l'origine del sistema di riferimento prefissato con il punto in esame, specificando le sue *coordinate rettangolari* oppure le sue *coordinate sferiche*. Le coordinate rettangolari rappresentano le componenti del vettore rispetto al sistema XYZ definito. Le coordinate sferiche, invece, rappresentano due spostamenti angolari da compiere sul sistema XYZ al fine di raggiungere il particolare punto interessato. In questo caso è necessario specificare i riferimenti da cui sono calcolati gli spostamenti angolari, ovvero gli assi del sistema di riferimento, e la convezione adottata per i segni degli stessi. Non esiste quindi una regola fissa, di volta in volta si specifica la convenzione utilizzata. E' possibile passare dalle coordinate rettangolari a quelle sferiche mediante semplici operazioni trigonometriche.

In linea di principio è possibile definire liberamente qualsiasi tipo di sistema di riferimento. Si cercherà tuttavia di utilizzare il sistema di riferimento più appropriato alla particolare missione spaziale in esame e che riesca a ridurre gli errori. La più comune sorgente di errore nell'analisi di una geometria spaziale è, infatti, la definizione incorretta dei diversi sistemi di riferimento coinvolti. Nei problemi riguardanti l'analisi delle orbite, oppure le geometrie sulla superficie della Terra, si fissa tipicamente il centro della Terra (sistema *geocentrico*) come il centro del sistema di coordinate. Si fissa, invece, la posizione del satellite, come centro del sistema di coordinate, nei problemi relativi alla posizione apparente ed al moto di oggetti osservati dallo spazio. I tipici modi per fissare un sistema di coordinate sono rispetto allo spazio inerziale, rispetto alla direzione della Terra o altri oggetti, visti dal satellite o da uno strumento a bordo del satellite stesso.

## 2.2 Il sistema di riferimento solidale con la rotazione della Terra

Per descrivere la posizione di un oggetto sulla superficie della Terra si sfruttano i concetti di *longitudine* e *latitudine*, definiti come parametri di tipo sferico. Un *meridiano* è l'intersezione del piano passante attraverso l'asse di rotazione terrestre e la superficie della Terra. Il *meridiano di riferimento* della Terra è quello passante attraverso il Royal Observatory di Greenwich (Inghilterra). I *poli* sono definiti dall'intersezione dell'asse di rotazione della Terra con la superficie esterna della stessa. L'*equatore* è il luogo dei punti sulla superficie della Terra disposti su un piano perpendicolare all'asse di rotazione della stessa e passante attraverso il centro di massa della stessa. Tutti i cerchi paralleli all'equatore sono invece detti *paralleli*. La longitudine di un punto è definita come lo spostamento angolare, misurato positivo verso Est, compreso tra il meridiano di riferimento ed il meridiano passante per il punto in questione. La convenzione di adottare come positiva la longitudine misurata verso Est è stata stabilita durante l'International Meridian Conference (Washington, 1884). La longitudine ha valori compresi tra  $0^\circ$  e  $360^\circ$  quando è misurata esclusivamente verso Est e valori compresi tra  $0^\circ$  e  $\pm 180^\circ$  quando è misurata sia verso Est, sia verso Ovest. Il piano equatoriale terrestre crea il piano di riferimento per misurare la latitudine. La latitudine è definita, infatti, come lo spostamento angolare compreso tra il piano equatoriale ed il punto interessato, con valori compresi tra  $0^\circ$  e  $\pm 90^\circ$ . La latitudine è definita positiva nell'emisfero boreale. L'angolo complementare alla latitudine è detto *colatitudine*, ovvero è dato dalla differenza tra  $90^\circ$  ed il valore di latitudine. Si riporta in figura 2.1 lo schema della geometria appena descritta, un punto P sulla superficie della Terra è descritto dalla sua longitudine, indicata con  $lon_P$ , e dalla sua latitudine, indicata con  $lat_P$ :

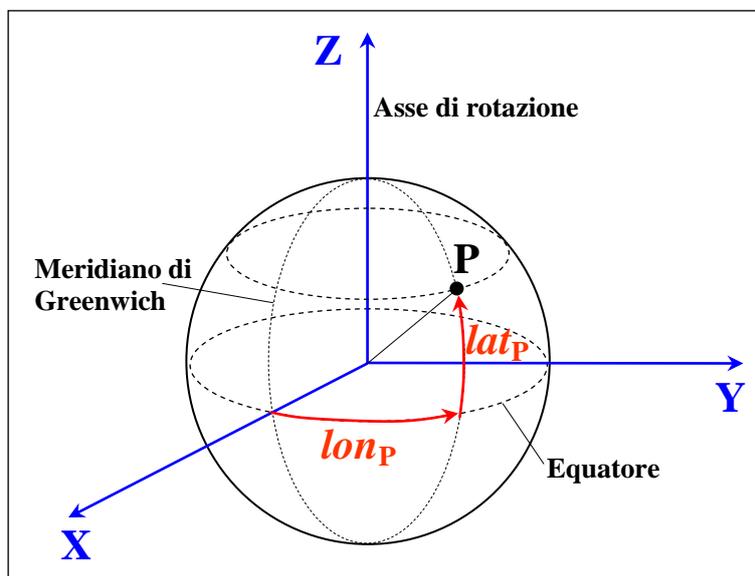


Figura 2.1 Uso di latitudine e longitudine nella misura della posizione di un punto sulla Terra.

Utilizzando le coordinate sferiche, quindi, la posizione del punto P è definita dalla seguente coppia:

$$P \equiv [lon_P, lat_P] \quad (2.1)$$

Questo sistema di riferimento è, di fatto, il più utilizzato nella descrizione della posizione di una località sulla superficie del globo terrestre. Lo stesso punto P è tuttavia possibile localizzarlo utilizzando le coordinate rettangolari. Si riporta in figura 2.2 la geometria in questo caso:

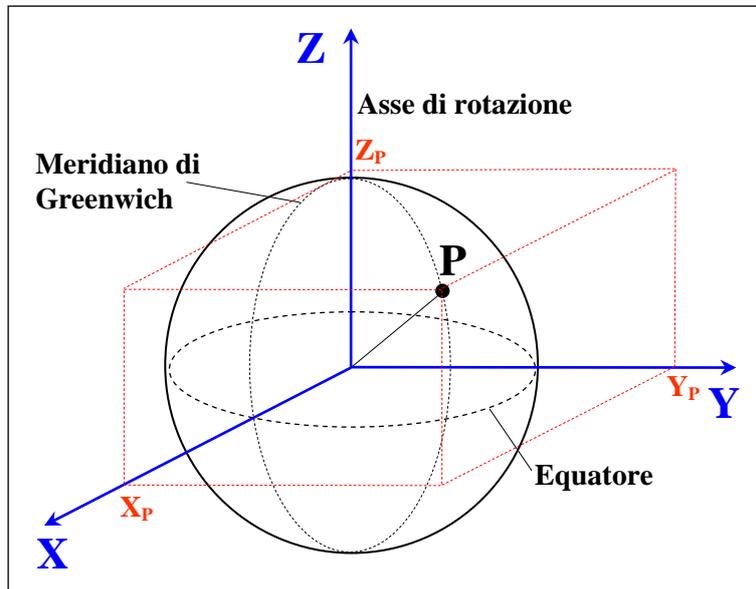


Figura 2.2 Sistema di riferimento solido con la rotazione della Terra (coordinate rettangolari).

Utilizzando le coordinate rettangolari la posizione del punto P è definita dalla terna seguente:

$$P \equiv [X_P, Y_P, Z_P] \quad (2.2)$$

Il sistema di coordinate rettangolari così definito è riportato anche con l'acronimo ECEF (Earth Centered Earth Fixed).

È possibile passare dal sistema di riferimento sferico a quello rettangolare sfruttando la trigonometria. Si riportano di seguito le relazioni necessarie:

$$X = \cos(lon_P) \cdot \cos(lat_P) \quad (2.3)$$

$$Y = \sin(lon_P) \cdot \cos(lat_P) \quad (2.4)$$

$$Z = \sin(lat_P) \quad (2.5)$$

Si noti che nelle relazioni precedenti è stata effettuata la normalizzazione rispetto al raggio terrestre, al fine di ottenere dei versori. Nel caso in cui non si desideri eseguire questa normalizzazione è necessario moltiplicare opportunamente tutte le relazioni per il valore dettato dal raggio della Terra. Per il passaggio inverso, invece, si sfruttano le relazioni seguenti:

$$\tan(lon_P) = Y / X \quad (2.6)$$

$$\sin(lat_P) = Z \quad (2.7)$$

## 2.3 Le coordinate celesti

I concetti analizzati nel paragrafo precedente sono utili nella descrizione della posizione di un punto sulla superficie della Terra. Generalizzando il discorso è possibile definire la posizione di un punto qualsiasi nello spazio, introducendo le coordinate *celesti*. Si noti, a tal scopo, che le distanze tra gli oggetti celesti naturali sono molto grandi rispetto alla dimensione della Terra. Questi oggetti risultano fissi rispetto ad una *sfera celeste*, o si muovono sulla superficie interna della stessa con l'osservatore situato al centro. La sfera celeste è una sfera immaginaria di raggio unitario molto utilizzata per rappresentare direzioni nello spazio. La rappresentazione mediante sfera celeste risulta molto utile nel focalizzare gli aspetti di tipo geometrico e fisico, molto importanti nell'analisi di una missione spaziale. Si definiscono *cerchi massimi* qualsiasi circonferenza che divida la sfera celeste in due semi-sfere identiche, tutte le circonferenze contenenti il centro della sfera sono quindi cerchi massimi. Tutti i cerchi non massimi sono quindi definiti *cerchi minimi*. I *poli celesti* (Nord e Sud) sono definiti dall'intersezione dell'asse di rotazione terrestre con la sfera celeste. In altre parole, sono definiti estendendo la definizione dell'asse di rotazione della Terra, fino a farlo intersecare con la superficie interna della sfera celeste. L'*equatore celeste* è definito dalla proiezione dell'equatore terrestre sulla sfera celeste. Insieme all'equatore si trasferiscono sulla sfera celeste anche tutti gli altri paralleli. Questi ultimi rappresentano tutti i cerchi minori della sfera celeste, ovvero tutte le *latitudini celesti* (in analogia alla latitudine terrestre) dal polo Nord celeste, indicato dalla stella Polare, sino al polo Sud. Allo stesso modo si riporta sulla sfera celeste la proiezione del meridiano passante nel punto in cui avviene l'osservazione, ovvero il cerchio massimo della volta celeste passante per i poli celesti e per i *poli dell'orizzonte* (descritti nel seguito). Questa proiezione genera quello che è detto *meridiano locale*. Insieme al meridiano locale, si proiettano sulla sfera, tutti gli altri meridiani che andranno a formare la seconda coordinata, ovvero la *longitudine celeste* (in analogia con la longitudine terrestre).

Per identificare un punto qualsiasi sulla sfera celeste è possibile sfruttare essenzialmente tre sistemi di riferimento differenti, descritti nel seguito della discussione.

### 2.3.1 Il sistema di coordinate orizzontale

Il primo sistema di riferimento descritto è conosciuto con il nome di sistema di riferimento *orizzontale*. Questo sistema di riferimento utilizza il piano dell'orizzonte (piano che divide la parte di sfera celeste visibile dall'osservatore da quella non visibile) come il piano fondamentale. L'origine del sistema di riferimento è fissata dalla posizione dell'osservatore. Un punto sulla sfera celeste è specificato mediante due coordinate sferiche, indicate con il nome di *azimut* e di *altezza*. Questo sistema è anche conosciuto con il nome di sistema *altazimutale*. L'*azimut* si misura in gradi lungo l'orizzonte, dal punto vero Sud, ovvero il punto in cui il meridiano locale interseca l'orizzonte (dal lato opposto del polo Nord celeste rispetto alla normale al piano orizzontale), in misura crescente verso Ovest. Si noti che è possibile trovare la definizione dell'azimut a partire dal vero Nord (punto diametralmente opposto al vero Sud), secondo la particolare applicazione. L'altezza, invece, si misura in gradi, da  $0^\circ$  a  $90^\circ$ , a partire dal piano dell'orizzonte verso un punto immaginario posto sulla verticale dell'osservatore, detto *Zenit*, e da  $0^\circ$  a  $-90^\circ$  verso il punto diametralmente opposto allo Zenit, detto *Nadir*. Zenit e Nadir rappresentano quindi i *poli dell'orizzonte*. Anche in questo caso è possibile trovare delle applicazioni in cui è definito come

positivo lo spostamento angolare verso il Nadir. Questo sistema di riferimento così definito è funzione della località in cui è situato l'osservatore e dell'ora particolare in cui avviene l'osservazione. Si riporta in figura 2.3 la geometria di questo sistema di riferimento, un punto P è localizzato sulla sfera celeste mediante l'azimut e l'altezza (l'origine O del sistema coincide con la posizione dell'osservatore):

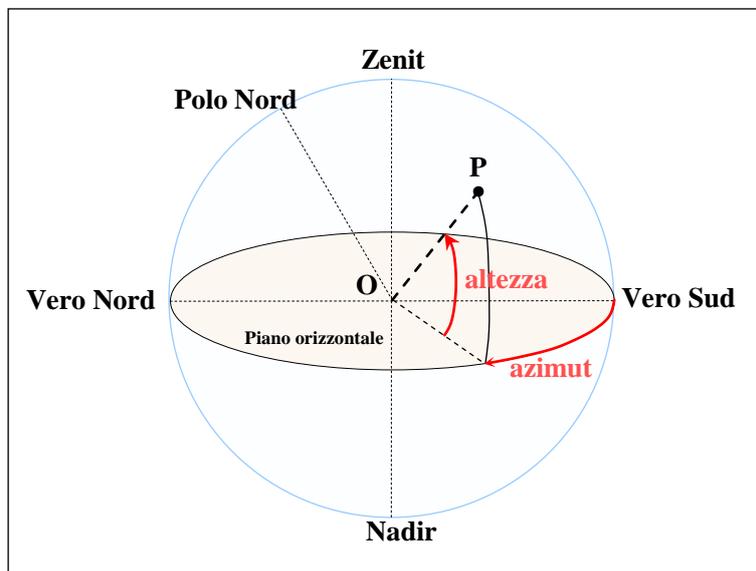


Figura 2.3 Sistema di riferimento celeste.

### 2.3.2 Il sistema di coordinate orario

Il secondo sistema di riferimento descritto è riconosciuto con il nome di sistema di coordinate *orario*. Questo sistema definisce un punto sulla sfera celeste mediante due coordinate sferiche, dette *declinazione*, che sostituisce l'altezza introdotta nel sistema precedente, ed *angolo orario*, che sostituisce l'azimut introdotto in precedenza. La declinazione si misura in gradi, con valori compresi tra  $0^\circ$  e  $+90^\circ$  partendo dall'equatore celeste verso il polo Nord celeste e tra  $0^\circ$  e  $-90^\circ$  dall'equatore celeste verso il polo Sud celeste. Il concetto di angolo orario deriva invece dall'estensione dei meridiani terrestri sulla sfera celeste, come introdotto in precedenza. L'angolo orario di un particolare oggetto nello spazio risulta quindi essere la distanza angolare, misurata sull'equatore celeste, compresa tra il meridiano celeste passante per l'oggetto in questione ed il meridiano locale. Questa grandezza è espressa normalmente in ore e minuti ed è misurata positiva verso Ovest. Si tenga presente che una rotazione di  $360^\circ$  è equivalente ad una rotazione di 24 ore. Quindi i punti cardinali Ovest ed Est sono quei punti in cui l'angolo orario vale 6 ore e 18 ore, rispettivamente. In questo sistema di riferimento la declinazione rimane fissa per qualsiasi oggetto preso in considerazione, dal momento che tale grandezza è riferita all'equatore, invariabile rispetto al cambiamento di latitudine dell'osservatore. L'angolo orario, invece, risulta essere una grandezza variabile in funzione del tempo, ovvero della longitudine del luogo di osservazione.

### 2.3.3 Il sistema di coordinate equatoriali

La Terra e la sua orbita attorno al Sole formano le basi utili nella definizione del terzo sistema di riferimento celeste, caratterizzato dall'essere fisso nello spazio, detto sistema di coordinate *equatoriali*. Si definisce *eclittica* il piano contenente l'orbita media della Terra attorno al Sole. Quando si osserva il Sole dalla Terra, esso sembra muoversi su questo piano. L'angolo compreso tra il piano dell'eclittica ed il piano equatoriale prende il nome di *obliquità dell'eclittica*. Il valore di questo angolo è approssimativamente pari a  $23.5^\circ$  ed è variabile nel tempo a causa delle perturbazioni. L'intersezione di questi due piani è d'aiuto nel fissare una direzione principale. La linea di intersezione è chiamata *linea dei nodi*. Il Sole attraversa questa intersezione due volte ogni anno. Questi due punti sono chiamati *equinozi*: uno è definito quando il Sole si trova nel nodo ascendente (*equinozio vernale*, 21 Marzo), ovvero quando attraversa il piano equatoriale passando da Sud a Nord, e l'altro quando il Sole si trova nel nodo discendente (*equinozio autunnale*, 23 Settembre), ovvero quando attraversa il piano equatoriale da Nord verso Sud. La direzione dell'equinozio vernale è sovente indicata come il *primo punto dell'Ariete* o *punto vernale* ed è la direzione fondamentale del sistema inerziale geocentrico. Il nome è dovuto al fatto che tale direzione durante i primi anni d.C. indicava la direzione della costellazione dell'Ariete. A causa della precessione degli equinozi, il punto vernale si sposta approssimativamente di  $0.014^\circ$  ogni anno. Le coordinate celesti richiedono quindi una corrispondente data di riferimento per definire in modo accurato tale direzione di riferimento. I sistemi più utilizzati sono quelli detti TOD (True Of Date), ovvero quelli che si occupano di prendere in considerazione tale variazione e di correggere opportunamente la direzione in funzione della particolare data. Il termine punto dell'Ariete non è cambiato nel tempo. Oggi giorno, in realtà, tale direzione di riferimento indica la direzione della costellazione dei Pesci.

Si riporta in figura 2.4 lo schema di principio del sistema inerziale geocentrico, indicato con le lettere IJK. L'origine è situata nel centro della Terra, il piano fondamentale è rappresentato dal piano equatoriale, l'asse I è diretto verso il punto vernale, l'asse K è diretto secondo l'asse di rotazione terrestre, verso il polo Nord, l'asse J è definito di conseguenza per formare una terna destrorsa, in modo che una rotazione antioraria di  $90^\circ$  attorno all'asse K porti l'asse I a coincidere con l'asse J:

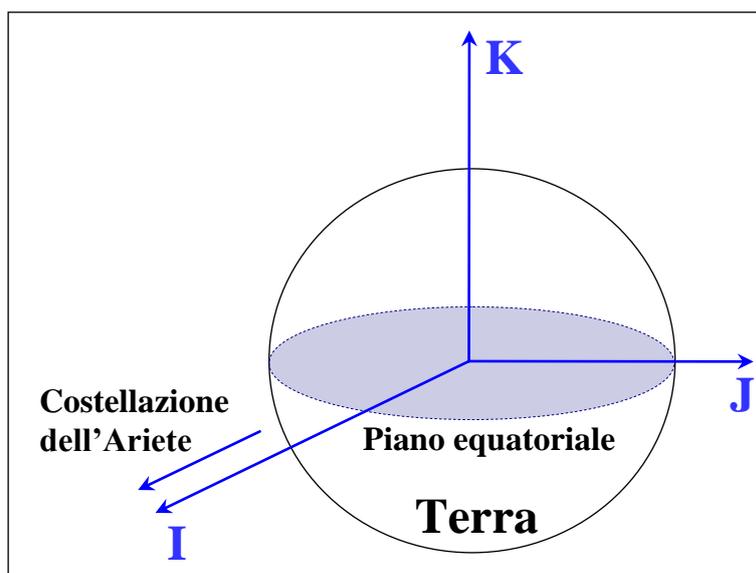


Figura 2.4 Sistema inerziale geocentrico.

Il sistema di riferimento inerziale geocentrico viene anche detto ECI (Earth Centered Inertial). Il sistema di riferimento IJK sarà considerato come sistema di riferimento assoluto e, nel seguito del testo, tutte le direzioni saranno riferite ad esso.

Anche in questo caso è possibile specificare la posizione di un punto mediante le coordinate sferiche, fornendo l'informazione delle due posizioni angolari associate ad un particolare punto interessato. I nomi in questo caso cambiano nuovamente; si parla di *ascensione retta* e di *declinazione* e sono gli equivalenti di longitudine e latitudine rispettivamente. La declinazione risulta essere la medesima coordinata introdotta nel sistema orario (si veda paragrafo precedente). L'ascensione retta, invece, è definita come lo spostamento angolare compreso tra il punto vernale ed il meridiano passante per il punto interessato. Questa grandezza è generalmente misurata in ore e minuti e risulta positiva se misurata in senso antiorario.

Il sistema inerziale geocentrico e quello solidale con la rotazione della Terra rappresentano i principali sistemi di riferimento basati sulla Terra (Earth-based Systems).

## 2.4 I principali sistemi di coordinate orbitali

In questo paragrafo si descrivono i due principali sistemi di coordinate aventi come piano fondamentale il piano orbitale (Satellite-based Systems).

### 2.4.1 Il sistema di riferimento PQW

Il primo sistema di coordinate analizzato è chiamato PQW (detto Perifocal Coordinate System). Si riporta in figura 2.5 lo schema di principio della geometria in esame:

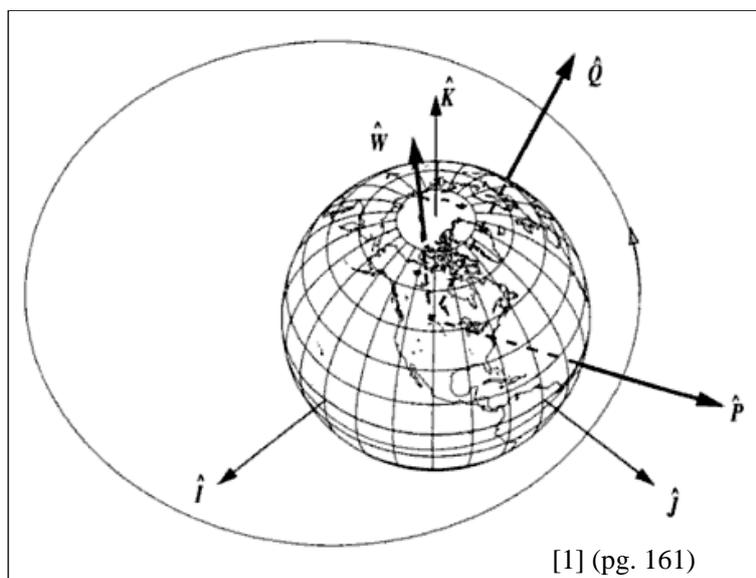


Figura 2.5 Sistema di riferimento PQW.

Nella figura precedente sono stati riportati i versori del sistema PQW e del sistema geocentrico inerziale IJK. Il piano fondamentale del sistema PQW risulta essere il piano orbitale e la sua origine è fissata in modo che risulti coincidente con il centro della Terra. Gli assi P e Q sono situati sul piano orbitale. L'asse P è fissato in modo da indicare la direzione del perigeo. L'asse W è fissato in modo che risulti normale al piano dell'orbita. L'asse Q, infine, è definito di conseguenza, in modo da formare una terna ortogonale, nella quale una rotazione antioraria di  $90^\circ$  attorno all'asse W porta l'asse P a coincidere con l'asse Q.

## 2.4.2 Il sistema di riferimento RSW

Il secondo sistema di coordinate analizzato è chiamato RSW (detto Satellite Coordinate System). Si riporta anche per questo, in figura 2.6, lo schema di principio della geometria in esame indicando gli assi RSW e gli assi del sistema geocentrico inerziale IJK:

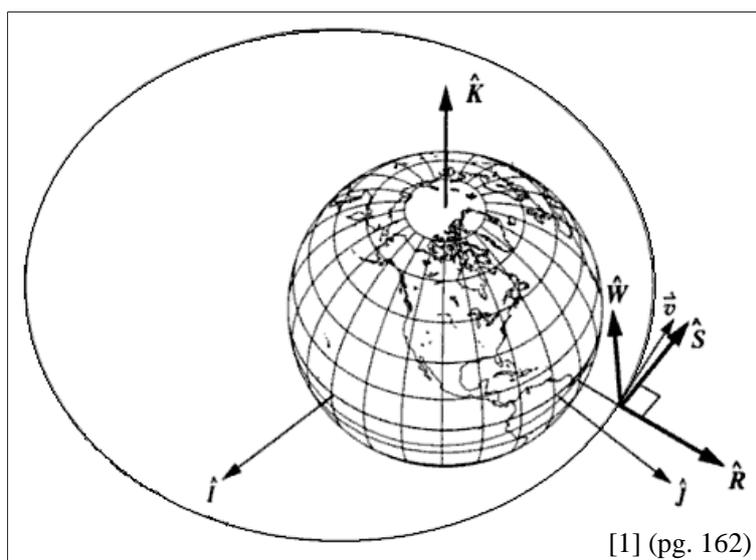


Figura 2.6 Sistema di riferimento RSW.

Questo sistema di riferimento, a differenza del sistema PQW, definito nel paragrafo precedente, non è fisso nello spazio e si muove insieme al moto del satellite. L'origine del sistema di riferimento è fissata in modo da coincidere con la posizione del satellite attorno alla Terra. L'asse R è definito in modo da risultare orientato come il vettore congiungente il centro della Terra al punto indicante la posizione del satellite, lungo il vettore radiale dello stesso. L'asse così definito segue il satellite durante il suo moto nell'orbita. L'asse W è definito in modo che risulti normale al piano orbitale. L'asse S è definito di conseguenza in modo da formare una terna ortogonale, nella quale una rotazione antioraria di  $90^\circ$  attorno a W porta l'asse R a coincidere con l'asse S. Nel caso particolare di orbita circolare, oppure di orbita ellittica in prossimità dell'apogeo e del perigeo, quest'asse è diretto verso il vettore velocità.

Sovente, al posto del sistema RSW, si utilizza il sistema RPW, chiamato sistema *roll, pitch, yaw* (molto utilizzato in ambito aeronautico). Questo sistema risulta essere molto simile al sistema appena descritto ed è possibile definirlo a partire dal sistema RSW, avendo cura di effettuare delle

semplici modifiche sul sistema stesso. Indicando con  $\hat{r}$ ,  $\hat{p}$  e  $\hat{w}$  i versori del sistema RPW, è possibile definire il loro orientamento a partire dall'orientamento definito per gli assi del sistema RSW. In particolare il versore  $\hat{r}$  è diretto in modo analogo all'asse S, il versore  $\hat{p}$  risulta essere orientato nella direzione opposta rispetto all'asse R, ovvero è orientato in modo da indicare il centro della Terra e, infine, il versore  $\hat{w}$  completa la terna cartesiana ed è opposto al vettore velocità angolare del satellite.

# Capitolo 3

## 3 I principali concetti dell'astrodinamica e le orbite kepleriane

In questo capitolo si riportano i principali concetti della meccanica celeste e sono descritti i principali parametri utili nella descrizione dell'orbita del satellite.

### 3.1 Le grandezze principali utilizzate nella descrizione dell'orbita del satellite

L'astrodinamica è una branca della meccanica celeste che si occupa di studiare il moto degli oggetti costruiti dall'uomo nello spazio, i quali risultano essere soggetti sia a forze di tipo naturale, sia a forze di tipo artificiale.

Nella descrizione dell'orbita di un satellite ci si avvale della teoria sviluppata da Keplero. Si riportano le sue tre leggi cardine riguardanti il moto planetario, le stesse saranno applicate anche nel caso di satelliti orbitanti attorno alla Terra e si parlerà appunto di orbite kepleriane:

**PRIMA LEGGE:** l'orbita di ogni pianeta ha la forma di un'ellisse, con il Sole situato in uno dei due fuochi.

**SECONDA LEGGE:** la linea congiungente il centro del pianeta con il centro del Sole percorre aree uguali in tempi uguali.

**TERZA LEGGE:** il quadrato del periodo di rivoluzione di un pianeta è proporzionale al cubo della distanza media dal Sole.

In figura 3.1 si riportano i parametri principali utilizzati nella descrizione di un'orbita ellittica. L'*eccentricità*,  $e$ , dell'ellisse (non rappresentata in figura 3.1) è data dal rapporto  $c/a$ , dove  $c$  raffigura la semidistanza tra i fuochi ed  $a$  rappresenta il semiasse maggiore dell'ellisse (in figura 3.1 è stato rappresentato l'asse maggiore, ovvero la distanza  $2a$ ). Questo parametro fornisce una misura di quanto la forma dell'ellisse si discosta da quella di una circonferenza (un'ellisse con eccentricità nulla è, di fatto, una circonferenza).

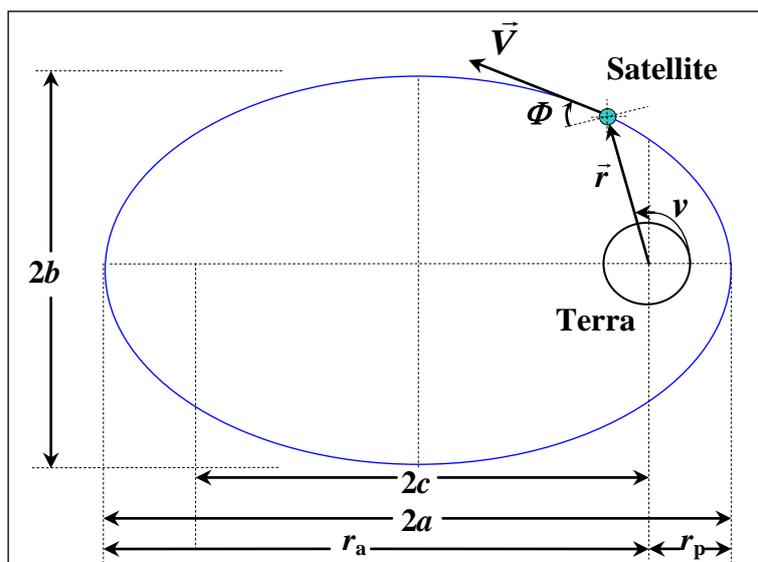


Figura 3.1 Geometria di un'orbita ellittica con i suoi parametri principali.

Nella figura precedente è possibile osservare i parametri seguenti:

$\vec{r}$  : il vettore posizione del satellite rispetto al centro della Terra.

$\vec{V}$  : il vettore velocità del satellite (centrato nella posizione del satellite e orientato verso la direzione tangente all'ellisse, con verso diretto secondo il moto del satellite).

$\Phi$  : l'angolo di volo, ovvero l'angolo compreso tra il vettore velocità ed una linea perpendicolare alla direzione specificata dal vettore di posizione,  $\vec{r}$ .

$a$  : il semiasse maggiore dell'ellisse.

$b$  : il semiasse minore dell'ellisse.

$c$  : la distanza tra il centro dell'orbita ed uno dei due fuochi.

$\nu$  : l'angolo polare dell'ellisse, chiamato *anomalìa vera*. Quest'importante angolo è misurato nella direzione del moto del satellite, dal *perigeo* verso la direzione specificata dal vettore di posizione. Il perigeo, per definizione, rappresenta il punto dell'orbita del satellite più vicino alla Terra. Il punto diametralmente opposto, ovvero quello più distante dalla Terra, è invece indicato con il termine *apogeo*.

$r_a$  : il raggio di apogeo, ovvero la distanza tra il centro della Terra e l'apogeo.

$r_p$  : il raggio di perigeo, ovvero distanza tra il centro della Terra ed il perigeo.

Un contributo decisivo alla teoria sviluppata da Keplero fu dato dal lavoro svolto da Isaac Newton. Combinando la sua seconda legge del moto con la sua Legge di Gravitazione Universale si ottengono, infatti, le basi matematiche con cui analizzare le orbite satellitari. La sua legge di gravitazione afferma che ogni particella di materia nell'universo attrae ogni altra particella di materia, con una forza che risulta proporzionale al prodotto delle masse ed inversamente proporzionale al quadrato della distanza tra le stesse. Quindi, nel caso di due masse rappresentate dalla Terra e dal satellite, il valore della forza dovuta alla gravità è dato dalla relazione seguente:

$$F = -G \cdot M \cdot m / r^2 \equiv -\mu m / r^2 \quad (3.1)$$

Dove  $F$  rappresenta il modulo della forza dovuta alla gravità,  $G$  rappresenta la costante di gravitazione universale,  $M$  rappresenta la massa della Terra,  $m$  rappresenta la massa del satellite,  $r$  rappresenta la distanza compresa tra il centro della Terra ed il satellite, e  $\mu \equiv G \cdot M$  rappresenta la costante gravitazionale della Terra ( $\mu = 398,600.5 \text{ km}^3 \text{ sec}^{-2}$ ).

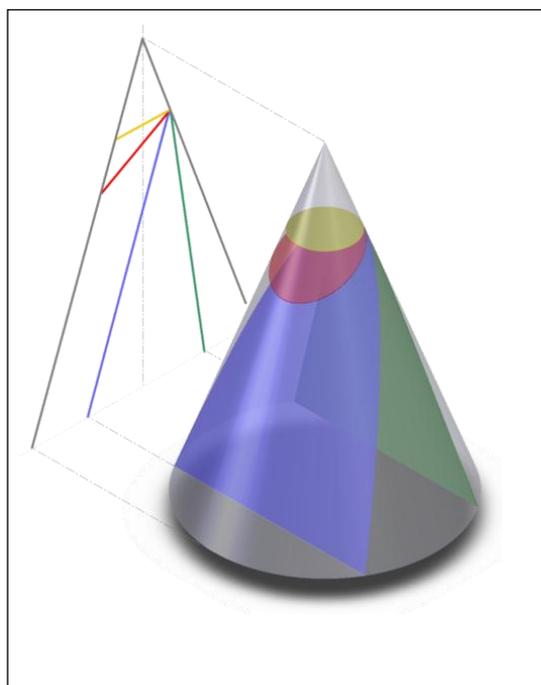
Abbinando la seconda legge di Newton (secondo principio della dinamica, ovvero quello in cui si afferma che l'accelerazione di un oggetto è proporzionale al prodotto della forza applicata sullo stesso ed inversamente proporzionale alla sua massa) con la legge di gravitazione si ottiene la seguente relazione, nota con il nome di *equazione del moto dei due corpi*:

$$\ddot{\mathbf{r}} + \mu \mathbf{r}^{-2} = 0 \quad (3.2)$$

Quest'equazione descrive l'evoluzione del vettore di posizione del satellite nella sua orbita attorno alla Terra. Questa relazione è derivata sotto le seguenti ipotesi:

- La forma della Terra è considerata una sfera perfetta.
- La massa della Terra è decisamente più grande della massa del satellite.
- L'unica forza agente sul sistema considerata risulta essere quella di gravità.
- Nel sistema sono presenti soltanto due corpi, ovvero la Terra ed il satellite (problema dei due corpi).

Una possibile soluzione dell'espressione 3.2 risulta essere data dall'*equazione polare di una sezione conica*. Una sezione conica, per definizione, è una curva piana, ovvero il luogo dei punti ottenibili intersecando la superficie di un cono circolare retto con un piano. Si riportano in figura 3.2 i diversi tipi di sezioni coniche effettivamente ottenibili (l'immagine riportata è disponibile in rete: [http://it.wikipedia.org/wiki/Sezione\\_conica](http://it.wikipedia.org/wiki/Sezione_conica)). I vari piani, intersecando il cono, possono descrivere una circonferenza (rappresentata con il colore giallo), un'ellisse (rappresentata con il colore rosso), una parabola (rappresentata con il colore blu) ed un'iperbole (rappresentata con il colore verde):



**Figura 3.2** Tipi di sezioni coniche.

Nel caso in cui l'orbita sia descritta da una parabola, oppure da un'iperbole, si dice che l'orbita in questione è *aperta* ed il satellite riesce a fuggire dall'attrazione gravitazionale esercitata dalla Terra. Questi casi particolari non sono trattati nella descrizione seguente, mentre si analizza il caso di orbite *chiuse*, ovvero le orbite caratterizzate dall'aver una forma descrivibile mediante una

circonferenza oppure un'ellisse. Si noti che risulta possibile considerare l'orbita ellittica come situazione generale e percepire l'orbita circolare come un caso particolare del primo, in cui si abbia la condizione di eccentricità nulla. La soluzione dell'equazione del moto dei due corpi, nel caso di orbita ellittica, conduce alla scrittura di una relazione che lega il modulo del vettore di posizione del satellite con i parametri dell'orbita, nel modo seguente:

$$r = a (1 - e^2) / (1 + e \cdot \cos v) \quad (3.3)$$

Utilizzando l'equazione del moto dei due corpi è possibile ricavare un parametro dell'orbita, detto *energia meccanica specifica* totale, o energia meccanica per unità di massa del sistema, definito nel modo seguente:

$$\varepsilon = V^2 / 2 - \mu / r = - \mu / (2a) \quad (3.4)$$

L'equazione precedente è sovente indicata con il nome di *equazione dell'energia* e risulta essere data dalla somma dell'energia cinetica per unità di massa e dell'energia potenziale per unità di massa, rispettivamente. Dal momento che le forze presenti nel sistema sono conservative, l'energia risulta essere una costante. Il termine riguardante l'energia potenziale, ovvero il termine  $-\mu / r$ , risulta essere sempre negativo, per qualsiasi raggio di valore finito. Questo porta ad affermare che l'energia meccanica specifica di un'orbita ellittica risulta essere sempre negativa. Inoltre, sempre osservando l'equazione 3.4 precedente, risulta semplice evincere che il satellite si muove con un moto più veloce in prossimità del perigeo e con un moto più lento in prossimità dell'apogeo.

Utilizzando l'equazione dell'energia, ed ipotizzando che la particolare orbita in esame sia descrivibile mediante una circonferenza (il semiasse maggiore, in questo caso, risulta essere uguale al raggio della circonferenza), è possibile ricavare la velocità del satellite, mediante la relazione seguente:

$$V = (\mu / r)^{1/2} \quad (3.5)$$

E' possibile osservare che la velocità del satellite è imposta dall'altitudine in cui esso si trova. In effetti, la posizione del satellite è determinata dall'equilibrio della sua energia cinetica dovuta alla sua velocità e dall'attrazione gravitazionale dovuta alla presenza della Terra.

Un'altra quantità importante ed associata all'orbita del satellite è data dal *momento angolare specifico*, indicato con il simbolo  $\vec{h}$ , il quale rappresenta il momento angolare totale del satellite diviso la sua massa. Questa grandezza è possibile valutarla attraverso l'operazione di prodotto vettoriale tra i vettori di posizione e di velocità, nel modo seguente:

$$\vec{h} = \vec{r} \times \vec{V} \quad (3.6)$$

La seconda legge di Keplero afferma che il momento angolare risulta essere costante in modulo e direzione, nel caso del problema dei due corpi. Quindi, il piano orbitale definito dai vettori di posizione e di velocità rimane fisso nello spazio inerziale.

Nel seguito della discussione si riportano i principali parametri che sono utilizzati nella descrizione dell'orbita del satellite.

## 3.2 Gli elementi orbitali classici

Durante la soluzione dell'equazione del moto dei due corpi risultano necessarie sei costanti di integrazione, vale a dire le condizioni iniziali. Conoscendo le tre componenti della posizione e della velocità del satellite in un dato istante, specificate in un prefissato sistema di riferimento, è possibile, in via teorica, calcolare e conoscere la posizione e la velocità del satellite in qualsiasi altro istante di tempo. In modo alternativo è possibile descrivere l'orbita utilizzando cinque termini indipendenti dal tempo ed un solo parametro dipendente dal tempo (nel caso precedente tutte e sei le costanti di integrazione risultano dipendenti dal tempo). Questi parametri, comunemente utilizzati in ambito spaziale, sono detti *elementi orbitali classici* e sono rappresentati in figura 3.3:

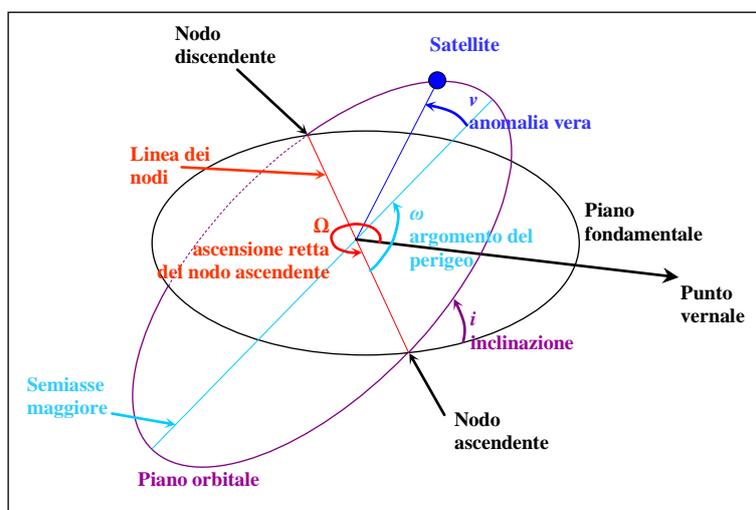


Figura 3.3 Elementi orbitali classici.

Gli elementi orbitali classici, rappresentati nella figura precedente, sono elencati di seguito:

$a$ : *semiasse maggiore*, parametro rappresentante la dimensione dell'ellisse.

$e$ : *eccentricità*, parametro descrivente la misura della forma dell'ellisse.

$i$ : *inclinazione*, parametro rappresentante lo spostamento angolare compreso tra il piano fondamentale (il piano equatoriale in genere) ed il piano orbitale.

$\Omega$ : *ascensione retta del nodo ascendente*, parametro raffigurante lo spostamento angolare compreso tra la direzione fissata dal punto vernale e la direzione dettata dal nodo ascendente.

$\omega$ : *argomento del perigeo*, parametro rappresentante lo spostamento angolare compreso tra la direzione specificata dal nodo ascendente e la direzione del perigeo.

$v$ : *anomalia vera*, parametro raffigurante lo spostamento angolare compreso tra la direzione specificata dal perigeo e la direzione dettata dal vettore di posizione del satellite, misurato nella direzione del moto del satellite.

### 3.3 Il formato delle Two-Line Element set (TLE) fornito dal NORAD

Una rappresentazione simile agli elementi orbitali classici, descritta nel paragrafo precedente, ma non identica, risulta essere data dalle *Two-Line Element set* (TLE). Le TLE rappresentano un insieme di elementi orbitali utili nella descrizione dell'orbita di un satellite. Il loro formato è specificato dal NORAD (North American Aerospace Defense Command). Il NORAD è un'organizzazione congiunta del Canada e degli Stati Uniti che si occupa di fornire un quadro d'insieme sulla situazione (natura, posizione, direzione e velocità) di ogni aviogetto, oppure oggetto spaziale, nell'ambito aerospaziale del Nord America.

Si riporta di seguito una breve descrizione del formato della TLE. Questi elementi orbitali sono stati utilizzati durante lo sviluppo dell'algoritmo indirizzato al calcolo della posizione del satellite nella sua orbita. Durante la vita operativa del satellite questi parametri rappresentano dei valori noti e sono periodicamente aggiornati, al fine di avere a disposizione una descrizione dell'orbita sempre la più attendibile possibile.

Il formato della TLE è riportato nella figura 3.4 seguente:

```

AAAAAAAAAAAAAAAAAAAAAAAAA
1 NNNNNU NNNNNAAA NNNNN.NNNNNNNNN +.NNNNNNNN +NNNNN-N +NNNNN-N N NNNNN
2 NNNNN NNN.NNNN NNN.NNNN NNNNNNNN NNN.NNNN NNN.NNNN NN.NNNNNNNNNNNNNNN

```

**Figura 3.4** Formato delle Two-Line Element set.

Nella linea 0 è indicato il nome del particolare oggetto spaziale (24 caratteri) in questione. Le linee 1 e 2, invece, contengono lo standard Two-Line Element set utilizzato dal NORAD e dalla NASA (National Aeronautics and Space Administration) per descrivere l'orbita di qualsiasi oggetto orbitante attorno alla Terra.

Nelle tabelle seguenti è riportata la descrizione del formato in questione (il parametro NSSDC ID, citato nella tabella seguente, rappresenta un sistema internazionale di numerazione dei satelliti):

LINEA 1	
COLONNA	DESCRIZIONE
01	Numero della linea
03-07	Numero del satellite
08	Classificazione (U = non classificato)
10-11	NSSDC ID (ultime due cifre dell'anno del lancio)
12-14	NSSDC ID (numero che identifica il lancio dall'inizio dell'anno)
15-17	NSSDC ID (codice di lettere che rappresenta il numero sequenziale di un pezzo nel lancio)
19-20	Epoca (ultime due cifre dell'anno)
21-32	Epoca (giorno dell'anno e parte frazionale del giorno)
34-43	Derivata prima temporale del moto medio, $\dot{n}/2$
45-52	Derivata seconda temporale del moto medio (punto decimale assunto), $\ddot{n}/6$
54-61	$B^*$ , coefficiente di resistenza aerodinamica (punto decimale assunto)
63	Tipo di effemeridi
65-68	Numero elemento
69	Checksum (Modulo 10)

LINEA 2	
COLONNA	DESCRIZIONE
01	Numero della linea
03-07	Numero del satellite
09-16	Inclinazione, $i$ [gradi]
18-25	Ascensione retta del nodo ascendente, $\Omega$ [gradi]
27-33	Eccentricità, $e$ (punto decimale assunto)
35-42	Argomento del perigeo, $\omega$ [gradi]
44-51	Anomalia media, $M$ [gradi]
53-63	Moto medio, $\bar{n}$ [Rivoluzioni al giorno]
64-68	Numero di rivoluzioni all'epoca [rivoluzioni]
69	Checksum (Modulo 10)

Le quantità indipendenti richieste per fare il calcolo della posizione del satellite nell'orbita sono sei e sono date da:  $\bar{n}$ ,  $e$ ,  $i$ ,  $\Omega$ ,  $\omega$ ,  $M$ . Le variabili  $\dot{n}/2$ ,  $\ddot{n}/6$ ,  $B^*$ , invece, sono utilizzate per descrivere l'effetto delle perturbazioni sul moto del satellite.

Questa serie di parametri rappresentano quindi le quantità note ed indispensabili utilizzate durante la soluzione del problema del calcolo della posizione del satellite (presentato nel capitolo 4).

### 3.4 Il problema di Keplero

In questo paragrafo si presenta il problema di base riguardante le orbite dei satelliti, quello riconosciuto come *problema di Keplero*.

Il problema di Keplero consiste nel ricavare la posizione raggiunta dal satellite dopo un certo lasso di tempo, nota la posizione iniziale del satellite nell'orbita. Si tratta, quindi, di propagare la posizione attuale del satellite nell'orbita in funzione del tempo. Per risolvere questo problema Keplero introdusse un parametro, detto *anomalia media*, ed indicato con il simbolo  $M$ . Questo nuovo parametro rappresenta il moto equivalente angolare, uniforme ed immaginario, del satellite su di una circonferenza ausiliaria di raggio pari al semiasse maggiore. E' facile intuire come questa grandezza corrisponde all'anomalia vera nel caso particolare di un'orbita circolare. Per definizione vale la relazione seguente:

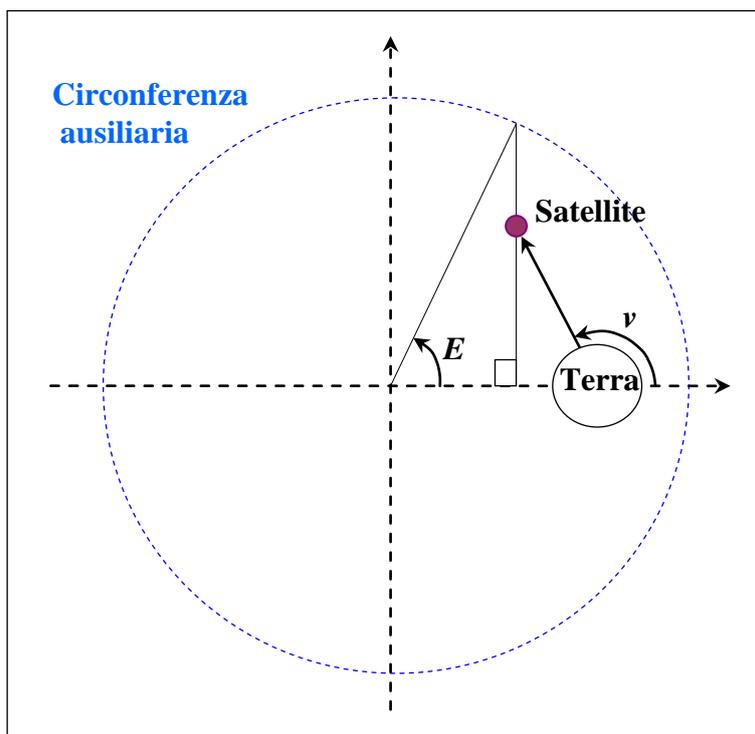
$$M - M_0 \equiv n (t - t_0) \quad (3.7)$$

Dove  $M_0$  rappresenta l'anomalia media al tempo  $t_0$  (istante iniziale) e  $n$  rappresenta il *moto medio*, o velocità angolare media, determinata a partire dal valore del semiasse maggiore dell'orbita secondo la relazione:

$$n \equiv (\mu / a^3)^{1/2} \quad (3.8)$$

Questa soluzione fornisce la posizione media e la velocità media del satellite su delle orbite di tipo circolare. Le orbite satellitari, tuttavia, sono generalmente di tipo ellittico ed il raggio, di conseguenza, varia costantemente nell'orbita in funzione del tempo. Dal momento che la velocità risulta essere dipendente dal raggio, anch'essa risulta essere variabile nel tempo. Per risolvere questo problema è possibile definire una variabile intermedia, chiamata *anomalia eccentrica*, ed

indicata con il simbolo  $E$ , utile nel caso generale di orbite ellittiche. Si riporta in figura 3.5 la relazione tra anomalia vera ed anomalia eccentrica:



**Figura 3.5 Anomalia vera ed anomalia eccentrica.**

Dalla figura 3.5 risulta semplice evincere che l'anomalia eccentrica è definita dallo spostamento angolare compreso tra l'asse maggiore e la direzione dettata dall'intersezione della linea ortogonale all'asse maggiore e passante attraverso la posizione del satellite nell'orbita e la circonferenza ausiliaria di raggio pari al semiasse maggiore.

La relazione che lega l'anomalia media con l'anomalia eccentrica è nota con il nome di *equazione di Keplero* e possiede la forma seguente:

$$M = E - e \sin(E) \quad (3.9)$$

Nel caso di orbita circolare risulta semplice intuire che le tre anomalie descritte coincidono.

Con lo scopo di risolvere il problema di Keplero, è possibile calcolare dapprima l'anomalia media, dato un certo intervallo di tempo. A questo punto si calcola l'anomalia eccentrica dall'equazione 3.9 precedente. Tale equazione non ammette forma chiusa e per risolverla si deve ricorrere a metodi di tipo numerico, come presentato nel capitolo 4.

# Capitolo 4

## 4 Il calcolo della posizione del satellite rispetto al sistema geocentrico inerziale IJK

In questo capitolo si presenta il problema della determinazione della posizione del satellite nella sua orbita attorno alla Terra, rispetto al sistema di riferimento geocentrico inerziale IJK e sono analizzati due algoritmi di soluzione, entrambi implementati utilizzando il linguaggio di programmazione C. Il primo, più semplice e più veloce dal punto di vista computazionale, si basa esclusivamente sulla teoria sviluppata da Keplero. Il secondo, invece, prende in considerazione una serie di perturbazioni che incidono sull'orbita (SGP4, algoritmo rilasciato da NASA/NORAD) e risulta pertanto più accurato e più pesante dal punto di vista computazionale rispetto al precedente.

### 4.1 L'algoritmo di calcolo della posizione del satellite rispetto al sistema di riferimento geocentrico inerziale IJK basato sulla teoria sviluppata da Keplero

Il primo algoritmo presentato, come preannunciato, è esclusivamente basato sulla teoria sviluppata da Keplero (si rimanda il lettore al capitolo 3, dove sono descritti i punti principali). Nello sviluppo dell'algoritmo è stato ipotizzato di conoscere gli elementi orbitali, codificati nel formato dettato dalla TLE. Il metodo presentato riceve quindi in ingresso una TLE e ricava il vettore di posizione del satellite, indicato con il simbolo  $\vec{r}_{IJK}$ , rispetto al sistema di riferimento geocentrico inerziale IJK, in funzione di una precisa data d'interesse.

#### 4.1.1 La soluzione dell'equazione di Keplero

Il primo passo verso la soluzione del problema della determinazione della posizione del satellite consiste nello sviluppare un procedimento, di tipo numerico, finalizzato alla soluzione dell'equazione di Keplero (si veda capitolo 3, paragrafo 4). Il modo più semplice per risolvere numericamente tale equazione si fonda sull'utilizzo del concetto di *approssimazione successiva*. Gli algoritmi che fanno uso di questo concetto implementano una serie di iterazioni. Quest'ultime sono ripetute finché una determinata condizione diviene soddisfatta. Risulta quindi di fondamentale importanza ottenere una rapida convergenza dell'algoritmo, al fine di risparmiare tempo durante i vari computi. Il criterio presentato in questo paragrafo utilizza le derivate successive della funzione

in esame nel calcolo della soluzione finale ed è detto *metodo di Newton-Raphson*. L'idea di base è la seguente:

Si vuole risolvere una funzione,  $f(y) = 0$ , nella variabile  $y$ . Si assuma  $y = x + \delta$ , dove  $x$  rappresenta una stima di  $y$  e  $\delta$  una piccola correzione. E' dunque possibile sviluppare in serie di Taylor la funzione, ottenendo l'espressione seguente:

$$0 = f(y) = f(x + \delta) \approx f(x) + \dot{f}(x) \delta + \ddot{f}(x) \delta^2 / (2!) + \dots \quad (4.1)$$

L'equazione 4.1, nell'ipotesi di poter trascurare i termini superiori al primo ordine, appare chiaramente solubile nella variabile  $\delta$  e si ottiene la relazione seguente:

$$\delta \approx -f(x) / \dot{f}(x) \quad (4.2)$$

La soluzione ricercata si raggiunge dopo una serie di iterazioni dell'algoritmo. Il valore calcolato al passo  $n$  è quindi utilizzato come valore di ingresso al passo  $n + 1$ . E' possibile indicare i valori assunti durante il processo di reiterazione utilizzando la notazione  $y_n = x_n + \delta_n$ . In questo modo è possibile rappresentare la stima di  $y$ .

Il processo illustrato è ripetuto finché  $\delta_n < \varepsilon$ , dove  $\varepsilon$  rappresenta la tolleranza desiderata. Il risultato finale del metodo in esame è quindi possibile scriverlo nella forma specificata nella relazione 4.3 seguente:

$$x_{n+1} = x_n + \delta_n = x_n - f(x) / \dot{f}(x) \quad (4.3)$$

Durante la risoluzione dell'equazione di Keplero la funzione in analisi è data da  $f(x) = f(E) = M - E + e \sin(E)$ , dove  $M$  rappresenta l'anomalia media,  $E$  l'anomalia eccentrica ed  $e$  l'eccentricità dell'orbita. La derivata prima di tale funzione risulta quindi essere  $\dot{f}(x) = -1 + e \cos(E)$ . La stima iniziale deve essere sufficientemente vicina alla soluzione finale, al fine di non violare l'assunzione lineare del metodo di Newton-Raphson. Inoltre è importante controllare che i valori iniziali dell'anomalia media siano situati nel range  $\pm 2\pi$ . La tolleranza richiesta è, in genere, approssimativamente pari a  $\varepsilon = 10^{-8}$ . L'algoritmo di soluzione dell'equazione di Keplero, quindi, riceve in ingresso i valori dell'anomalia media e dell'eccentricità e ricava il valore dell'anomalia eccentrica che soddisfa il vincolo sulla tolleranza richiesta. Questo algoritmo è articolato in due passi distinti:

1. Determinazione di un valore iniziale corretto:

Se il valore di  $M$  è compreso tra  $(-\pi, 0)$  oppure  $M > \pi$ , si impone  $E = M - e$ ; se, invece, le condizioni precedenti non sono soddisfatte si impone  $E = M + e$ .

2. Esecuzione delle iterazioni:

Si calcola il valore:

$$E_{n+1} = E_n + (M - E_n + e \sin(E_n)) / (1 - e \cos(E_n)) \quad (4.4)$$

Il calcolo 4.4 precedente è ripetuto finché  $|E_{n+1} - E_n| < \varepsilon$ .

### 4.1.2 Il calcolo della posizione del satellite rispetto al sistema IJK

In questo paragrafo si descrive come ricavare effettivamente il vettore di posizione del satellite a partire da una particolare TLE. Il calcolo è effettuato in funzione di una certa data di interesse, specificata come il lasso di tempo trascorso a partire dalla data di creazione della TLE (specificata nella stessa), e basandosi esclusivamente sulle teorie approssimative di Keplero (assenza di perturbazioni).

Il primo parametro ricavabile a partire da una specifica TLE è sicuramente il valore dell'anomalia vera, indicata con la lettera  $v$ . Questo termine, infatti, non è contemplato nel formato della TLE, ma è possibile ottenerlo a partire dai valori di eccentricità ed anomalia eccentrica (il valore di  $E$  è noto grazie all'algoritmo di soluzione dell'equazione di Keplero, presentato nel paragrafo precedente), utilizzando le relazioni riportate di seguito:

$$\cos(v) = (\cos(E) - e) / (1 - e \cos(E)) \quad (4.5)$$

$$\sin(v) = \sin(E) \sqrt{(1 - e^2) / (1 - e \cos(E))} \quad (4.6)$$

$$v = \text{atan2}(\sin(v), \cos(v)) \quad (4.7)$$

Nella relazione 4.7 precedente è stata introdotta la funzione  $\text{atan2}$ . Questa funzione converte una coppia di coordinate  $(x, y)$  nell'angolo corretto in tutti e quattro i quadranti ed è una funzione disponibile nelle librerie matematiche del C e di Matlab.

Il secondo parametro direttamente calcolabile a partire da una TLE è il valore del semiasse maggiore, indicato con la lettera  $a$ . Questa quantità è possibile valutarla utilizzando la relazione del moto medio seguente:

$$\bar{n} = \sqrt{\frac{\mu}{a^3}} \quad (4.8)$$

Dal momento che i termini  $\mu$  e  $\bar{n}$  sono noti, l'equazione 4.8 precedente è sufficiente a ricavare il valore del semiasse maggiore. Si riporta nell'espressione 4.9 il risultato finale:

$$a = \sqrt[3]{\frac{\mu}{\bar{n}^2}} \quad (4.9)$$

Si noti che il valore di  $\mu$  è espresso in  $[\text{km}^3 \cdot \text{sec}^{-2}]$ , mentre il valore di  $\bar{n}$ , presente nella TLE, è espresso in  $[\text{rev} / \text{giorno}]$  (rivoluzioni al giorno). Risulta quindi necessario trasformare questo valore in  $[\text{rad} / \text{s}]$  prima di utilizzare l'espressione 4.9 precedente. Così facendo si ottiene evidentemente il valore del semiasse maggiore espresso in  $[\text{Km}]$ .

Una volta quantificato il valore del semiasse maggiore è possibile ricavare un nuovo parametro, detto *semiparametro*, indicato con la lettera  $p$ , mediante la relazione 4.10 seguente:

$$p = a (1 - e^2) \quad (4.10)$$

Il semiparametro  $p$  rappresenta la distanza tra il fuoco primario (ovvero il centro della Terra) e l'orbita, misurata perpendicolarmente al semiasse maggiore. In figura 4.1 si rappresenta questo parametro:

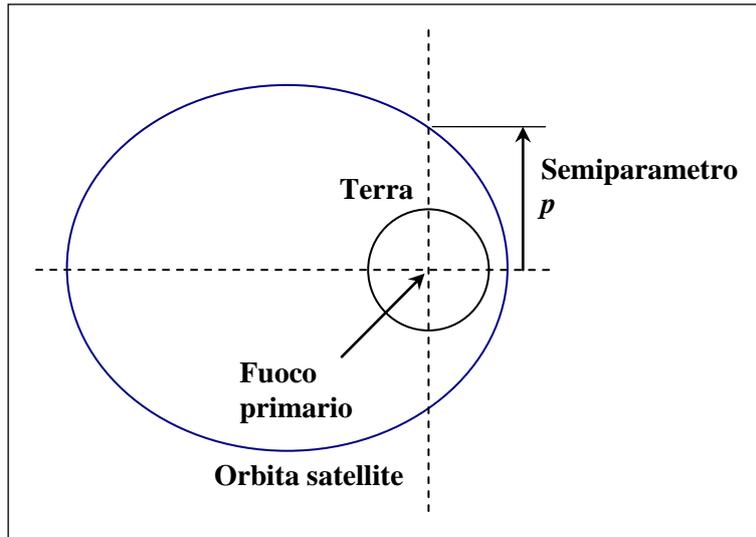


Figura 4.1 Semiparametro p.

A questo punto è possibile ricavare il vettore di posizione del satellite rispetto al sistema di riferimento orbitale PQW, indicato con il simbolo  $\vec{r}_{PQW}$ , sfruttando le relazioni seguenti:

$$\vec{r}_{PQW} = \begin{bmatrix} \frac{p \cos(v)}{1 + e \cos(v)} \\ \frac{p \sin(v)}{1 + e \cos(v)} \\ 0 \end{bmatrix} \quad (4.11)$$

Si noti che la componente in W del vettore appena calcolato risulta essere nulla. Questo è dovuto al fatto che il piano fondamentale del sistema PQW risulta essere coincidente con il piano dell'orbita, per definizione, e quindi la componente ortogonale al piano stesso è evidentemente sempre nulla, dal momento che il satellite si muove su di esso.

A questo punto è possibile specificare le componenti del medesimo vettore nel sistema di riferimento geocentrico inerziale IJK, effettuando una trasformazione di coordinate dal sistema PQW al sistema IJK. A tal scopo sono necessarie tre rotazioni successive, tutte in senso orario (ovvero rotazioni negative, per definizione): la prima attorno a W di un angolo pari all'argomento del perigeo,  $\omega$ , la seconda attorno a P di un angolo pari all'inclinazione dell'orbita,  $i$ , e la terza nuovamente attorno a W di un angolo pari all'ascensione retta del nodo ascendente,  $\Omega$ . Si riporta nella relazione 4.12 il risultato finale:

$$\frac{\vec{r}_{IJK}}{\vec{r}_{PQW}} = \begin{bmatrix} \cos(\Omega) \cos(\omega) - \sin(\Omega) \sin(\omega) \cos(i) & -\cos(\Omega) \sin(\omega) - \sin(\Omega) \cos(\omega) \cos(i) & \sin(\Omega) \sin(i) \\ \sin(\Omega) \cos(\omega) + \cos(\Omega) \sin(\omega) \cos(i) & -\sin(\Omega) \sin(\omega) + \cos(\Omega) \cos(\omega) \cos(i) & -\cos(\Omega) \sin(i) \\ \sin(\omega) \sin(i) & \cos(\omega) \sin(i) & \cos(i) \end{bmatrix} \quad (4.12)$$

Il calcolo appena presentato è riferito all'epoca specificata nella TLE. In altre parole, la posizione del satellite calcolata seguendo questo procedimento è valida soltanto nella particolare data cui sono associati gli elementi orbitali della TLE. Volendo ottenere la posizione del satellite per un tempo differente da quello specificato nella TLE è necessario valutare il valore corretto dell'anomalia

media ed effettuare nuovamente il computo. Il valore dell'anomalia media associato ad un tempo differente da quello specificato nella TLE (indicato con  $t_0$ ) è possibile valutarlo conoscendo il valore del moto medio del satellite, nel modo seguente:

$$M = M_0 + \bar{n} \cdot \Delta t \quad (4.13)$$

Dove  $\Delta t$  indica il lasso di tempo, espresso in secondi, trascorso dalla data specificata nella TLE. E' importante precisare che con questo tipo di approccio non si contempla affatto la presenza delle perturbazioni dell'orbita. E' quindi possibile utilizzare questo algoritmo soltanto se il valore di  $\Delta t$  risulta essere decisamente piccolo.

## 4.2 Le principali perturbazioni dell'orbita satellitare

Le orbite kepleriane introdotte nel capitolo 3 rappresentano un'eccellente approssimazione della realtà, tuttavia durante il moto del satellite ulteriori forze agiscono sullo stesso e causano una variazione dell'orbita kepleriana ideale. E' possibile classificare queste *perturbazioni*, o variazioni degli elementi orbitali, in base a quanto esse incidono sugli elementi orbitali. Si riportano in figura 4.2 le tipiche variazioni di uno degli elementi orbitali causate da forze perturbatrici:

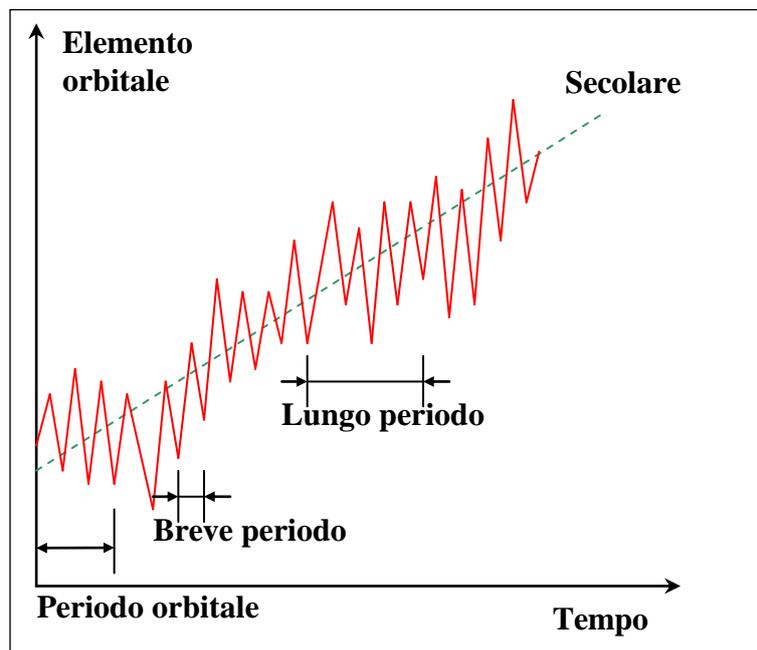


Figura 4.2 Variazioni secolari e periodiche di un elemento orbitale.

Nella figura precedente sono state rappresentate le *variazioni secolari*, le *variazioni a breve periodo* e le *variazioni a lungo periodo*. Le prime rappresentano una variazione lineare dell'elemento in funzione del tempo. Le seconde rappresentano delle variazioni di tipo periodico aventi un periodo minore o uguale al periodo orbitale. Le ultime, infine, rappresentano anch'esse delle variazioni periodiche, ma aventi un periodo maggiore del periodo orbitale. Le variazioni secolari sono sicuramente le più importanti, dal momento che causano degli effetti a lungo termine. Nelle

missioni in cui l'orbita deve essere stimata con molta accuratezza ricoprono un ruolo importante anche le variazioni di tipo periodico.

Gli elementi orbitali classici (si veda capitolo 3, paragrafo 2), quando sono prese in considerazione le forze perturbatrici, divengono dipendenti dal tempo e variano in funzione dello stesso. Al fine di stimare in modo accurato l'orbita satellitare risulta necessario determinare queste variazioni temporali, utilizzando delle tecniche apposite. Le due tecniche principali sono dette *perturbazioni speciali* e *perturbazioni generali*.

Nella tecnica delle perturbazioni speciali si sfrutta un'integrazione numerica diretta delle equazioni del moto.

Nella tecnica delle perturbazioni generali, invece, sono risolti analiticamente alcuni aspetti del moto del satellite soggetti all'azione delle forze perturbatrici. Sfortunatamente, le principali forze perturbatrici non conducono ad una soluzione analitica diretta ma ad espansioni in serie ed approssimazioni. Le tecniche delle perturbazioni generali sono più complicate ed approssimate, ma permettono di comprendere meglio come le perturbazioni influiscano sull'orbita. Inoltre, consentono di ottenere delle soluzioni in tempi più brevi, dal punto di vista del tempo computazionale necessario, rispetto alle tecniche delle perturbazioni speciali.

Le forze primarie che incidono sull'orbita del satellite provengono da:

- Presenza di un terzo corpo (come il Sole e la Luna).
- Distribuzione non sferica della massa della Terra.
- Attrito aerodinamico dell'atmosfera.
- Pressione causata dalla radiazione solare.

Nella discussione seguente si riporta una breve descrizione di queste perturbazioni.

### 4.2.1 Le perturbazioni introdotte dalla presenza del terzo corpo

Le forze gravitazionali esercitate dal Sole e dalla Luna causano delle variazioni periodiche su tutti gli elementi orbitali, ma soltanto i parametri  $\Omega$ ,  $\omega$  e  $M$ , ascensione retta del nodo ascendente, argomento del perigeo e anomalia media, rispettivamente, subiscono una variazione di tipo secolare. Questo tipo di variazione è dovuta alla precessione giroscopica dell'orbita attorno al polo eclittico. La variazione secolare dell'anomalia media è più contenuta rispetto a quella del moto medio ed influisce in modo minore sull'orbita del satellite. Le variazioni sull'ascensione retta del nodo ascendente e sull'anomalia media sono importanti, soprattutto per le orbite alte.

Per le orbite quasi circolari, le variazioni secolari dovute a Sole e Luna di  $\Omega$  e  $\omega$  sono date da:

- Variazione dell'ascensione retta del nodo ascendente:

$$\dot{\Omega}_{Luna} = -0.00338 \frac{\cos(i)}{n} \quad (4.14)$$

$$\dot{\Omega}_{Sole} = -0.00154 \frac{\cos(i)}{n} \quad (4.15)$$

- Variazione dell'argomento del perigeo:

$$\dot{\omega}_{Luna} = 0.00169 \frac{(4 - 5 \sin^2(i))}{n} \quad (4.16)$$

$$\dot{\omega}_{Sole} = 0.00077 \frac{(4 - 5 \sin^2(i))}{n} \quad (4.17)$$

Dove  $i$  rappresenta l'inclinazione dell'orbita,  $n$  rappresenta il numero di rivoluzioni compiute al giorno, mentre  $\dot{\Omega}$  e  $\dot{\omega}$  rappresentano le variazioni secolari espresse in [gradi / giorno]. Queste equazioni sono un'approssimazione. Entrambe trascurano il cambiamento dell'orientamento del piano orbitale del satellite rispetto al piano orbitale della Luna ed al piano dell'eclittica.

## 4.2.2 Le perturbazioni dovute alla non sfericit  della Terra

Nella derivazione dell'equazione del moto dei due corpi, presentata nel capitolo 3, la Terra   stata modellata come una massa sferica, omogenea e simmetrica. In realt , il globo terrestre, non risulta essere n  sferico, n  omogeneo. I fattori dominanti del fenomeno in questione sono dovuti alla protuberanza dell'equatore, alla lieve forma a pera della Terra ed allo schiacciamento dei poli.

E' possibile trovare l'accelerazione del satellite a partire da una funzione potenziale della Terra, indicata con il simbolo  $\Phi$ , e calcolando il gradiente di tale funzione. Una funzione geopotenziale largamente impiegata in quest'ambito   riportata nella relazione seguente:

$$\Phi = \frac{\mu}{r} [1 - \sum J_n (R_E / r)^n P_n \sin(lat)] \quad (4.18)$$

Dove  $\mu \equiv GM$  rappresenta la costante gravitazionale della Terra,  $R_E$  rappresenta il raggio della Terra misurato sull'equatore,  $P_n$  sono i polinomi di Legendre,  $lat$  rappresenta la latitudine geocentrica e  $J_n$  sono i coefficienti geopotenziali, privi di dimensioni, dei quali i primi sono riportati di seguito:

$$J_2 = 0.00108263$$

$$J_3 = -0.00000254$$

$$J_4 = -0.00000161$$

La funzione  $\Phi$ , come si pu  osservare facilmente nella relazione 4.18, risulta essere dipendente dalla latitudine, ed i coefficienti geopotenziali,  $J_n$ , sono chiamati *coefficienti zonali*.

Il potenziale generato dalla non sfericit  della Terra causa delle variazioni periodiche su tutti gli elementi orbitali. Gli effetti dominanti, comunque, sono dati dalle variazioni secolari dell'ascensione retta del nodo ascendente,  $\Omega$ , e dell'argomento del perigeo,  $\omega$ , causati dallo schiacciamento ai poli del globo terrestre. Questo fenomeno naturale   rappresentato dal termine  $J_2$  dell'espansione geopotenziale. Le variazioni di  $\Omega$  e  $\omega$  dovute a  $J_2$  sono date dalle espressioni seguenti:

$$\begin{aligned}\dot{\Omega}_{J_2} &= -1.5 \cdot n \cdot J_2 \cdot (R_E / a)^2 \cdot \cos(i) \cdot (1 - e^2)^{-2} \\ &\cong -2.006474 \cdot 10^{14} \cdot a^{-\frac{7}{2}} \cdot \cos(i) \cdot (1 - e^2)^{-2}\end{aligned}\quad (4.19)$$

$$\begin{aligned}\dot{\omega}_{J_2} &= 0.75 \cdot n \cdot J_2 \cdot (R_E / a)^2 \cdot (4 - 5 \cdot \sin^2(i)) \cdot (1 - e^2)^{-2} \\ &\cong 1.03237 \cdot 10^{14} \cdot a^{-\frac{7}{2}} \cdot (4 - 5 \cdot \sin^2(i)) \cdot (1 - e^2)^{-2}\end{aligned}\quad (4.20)$$

Dove  $n$  rappresenta il moto medio del satellite [gradi / giorno],  $R_E$  il raggio equatoriale della Terra [Km],  $a$  il semiasse maggiore dell'orbita [Km],  $e$  l'eccentricità,  $i$  l'inclinazione [gradi], e  $\dot{\Omega}$  e  $\dot{\omega}$  sono espressi in [gradi / giorno].

### 4.2.3 Le perturbazioni causate dall'attrito aerodinamico dell'atmosfera

Nelle missioni spaziali che prevedono un'orbita bassa la principale forza perturbatrice dell'orbita è data dall'attrito aerodinamico causato dall'atmosfera. La forza perturbatrice generata agisce in direzione opposta alla velocità del satellite e rimuove energia dall'orbita. Eventualmente questa riduzione di energia può far diminuire l'altitudine del satellite e farlo rientrare verso Terra. L'equazione dell'accelerazione causata da questo fenomeno è data dalla seguente relazione:

$$a_D = -\left(\frac{1}{2}\right)\rho\left(\frac{C_D A}{m}\right)V^2 \quad (4.21)$$

Dove  $\rho$  rappresenta la densità dell'atmosfera,  $C_D$  è il coefficiente di resistenza aerodinamica, approssimativamente pari a 2.2,  $A$  è l'area della sezione trasversale del satellite,  $m$  è la massa del satellite e  $V$  è la velocità del satellite rispetto all'atmosfera.

E' possibile approssimare la variazione del semiasse maggiore ( $\Delta a_{rev}$ ) e dell'eccentricità dell'orbita ( $\Delta e_{rev}$ ) per rivoluzione, ed il tempo di persistenza del satellite in un'orbita circolare, utilizzando le seguenti equazioni:

$$\Delta a_{rev} = -2\pi\left(\frac{C_D A}{m}\right)a^2 \rho_p \exp(-c) (I_0 + 2eI_1) \quad (4.22)$$

$$\Delta e_{rev} = -2\pi\left(\frac{C_D A}{m}\right)a\rho_p \exp(-c) \left(I_1 + \frac{e}{2(I_0 + I_2)}\right) \quad (4.23)$$

Dove  $\rho_p$  rappresenta la densità atmosferica in prossimità del perigeo,  $c \equiv (ae/H)$ ,  $H$  rappresenta una scala di valori di densità (valore tabulato) ed  $I_i$  sono le funzioni modificate di Bessel di ordine

$i$  ed argomento  $c$ . Il termine  $m/(C_D A)$ , detto *coefficiente balistico*, è sovente modellato come una costante per molti satelliti.

Nel particolare caso di orbite circolari, le equazioni appena riportate possono essere riscritte nella forma seguente, più semplice:

$$\Delta a_{rev} = -2\pi \left( \frac{C_D A}{m} \right) \rho \cdot a^2 \quad (4.24)$$

$$\Delta P_{rev} = -6\pi^2 \left( \frac{C_D A}{m} \right) \left( \frac{\rho \cdot a^2}{V} \right) \quad (4.25)$$

$$\Delta V_{rev} = \pi \left( \frac{C_D A}{m} \right) \rho \cdot a \cdot V \quad (4.26)$$

$$\Delta e_{rev} = 0 \quad (4.27)$$

Dove è stato indicato con il simbolo  $P$  il periodo dell'orbita.

Una stima grossolana della durata della vita del satellite nell'orbita, ovvero del tempo di persistenza del satellite nell'orbita prima di rientrare nell'atmosfera terrestre, indicato con il simbolo  $L$ , dovuta all'attrito aerodinamico è data da:

$$L \approx -\frac{H}{\Delta a_{rev}} \quad (4.28)$$

Una stima più accurata può essere ottenuta integrando l'equazione 4.24, prendendo in considerazione le variazioni della densità atmosferica dovute sia all'altitudine, sia all'attività solare.

#### 4.2.4 Le perturbazioni causate dalla pressione esercitata dalla radiazione solare

La pressione esercitata dalla radiazione solare provoca delle variazioni periodiche su tutti i parametri orbitali. Il suo effetto risulta maggiore nel caso di satelliti con coefficiente balistico basso, ovvero nel caso di satelliti leggeri con grande area frontale. Il valore dell'accelerazione dovuto alla pressione solare è dato da:

$$a_R \approx -4.5 \times 10^{-8} \left( \frac{A}{m} \right) \quad (4.29)$$

Dove  $A$  rappresenta l'area della sezione trasversale del satellite esposta al Sole e  $m$  rappresenta la massa del satellite. Nel caso di satelliti con altitudine inferiore a 800 Km il termine dovuto all'attrito aerodinamico è decisamente più importante; sopra gli 800 Km, invece, l'accelerazione dovuta alla pressione solare risulta preponderante.

### **4.3 Il calcolo della posizione del satellite rispetto al sistema geocentrico inerziale IJK mediante l'algoritmo SGP4**

L'algoritmo SGP4 (acronimo di Simplified General Perturbation, si vedano [9] e [10] per una descrizione dettagliata sull'argomento) è un software, rilasciato da NASA/NORAD, finalizzato al calcolo della posizione di satelliti in orbita bassa attorno alla Terra. L'algoritmo in questione si occupa di calcolare il vettore di posizione del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. E' possibile utilizzare questo algoritmo per calcolare la posizione di qualsiasi satellite avente un periodo orbitale inferiore a 225 minuti. Nel caso di satelliti con periodo orbitale maggiore è possibile utilizzare gli algoritmi SDP4 o SDP8. Di fatto, l'algoritmo rilasciato da NASA/NORAD contiene sia le procedure di calcolo della posizione dei satelliti in orbita bassa (SGP4), sia le procedure riguardanti le orbite più alte (SDP4/SDP8). Il lavoro di tesi si è concentrato sullo studio di applicazioni mirate verso missioni in orbita bassa. A tal scopo l'algoritmo appena citato è stato alleggerito notevolmente, eliminando opportunamente tutte le routine riguardanti le orbite con periodo orbitale maggiore o uguale a 225 minuti.

Gli elementi orbitali, nel formato della TLE (descritto nel capitolo 3, paragrafo 3), rappresentano i parametri d'ingresso dell'algoritmo SGP4. L'accuratezza di tale algoritmo è tipicamente di 1 Km sulla posizione. Ovviamente l'accuratezza è funzione dell'aggiornamento della TLE. In altre parole, risulta opportuno conoscere sempre la TLE più aggiornata disponibile al fine di ottenere una posizione, la più accurata possibile. Si noti che il vettore di posizione calcolato mediante SGP4 è specificato nel sistema di riferimento IJK denominato TEME (True Equator Mean Equinox). Nel capitolo 2 è stato definito il sistema geocentrico inerziale IJK ed è stata annunciata la variazione della direzione principale, dettata dalla direzione del punto vernale, a causa della precessione degli equinozi. Nello stesso capitolo è stato precisato che i sistemi di riferimento che sfruttano questo punto devono essere riferiti ad una particolare data di riferimento, al fine di aumentarne l'accuratezza. A tal scopo sono disponibili diversi sistemi di riferimento IJK. Uno particolare è dato dal sistema TEME. Sono disponibili due sistemi di riferimento TEME, il sistema TEME of Epoch ed il sistema TEME of Date. Nel primo l'asse I è diretto secondo l'equinozio vernale medio e l'asse K è diretto secondo l'asse di rotazione vero della Terra, definiti nella data specificata nella TLE. Nel secondo, invece, le due direzioni precedenti sono definite per la particolare data in cui si vuole conoscere la posizione del satellite. Dal momento che le TLE sono aggiornate con cadenza settimanale, in genere in questo piccolo lasso di tempo le differenze tra i due sistemi sono assolutamente trascurabili.

Si riporta in figura 4.3 il diagramma di flusso di massima dell'algoritmo in questione, al fine di evidenziarne i punti principali:

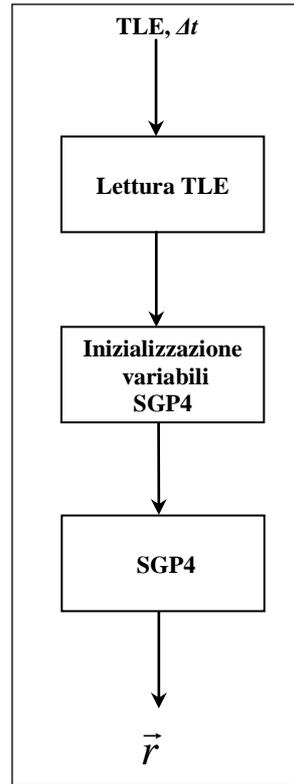


Figura 4.3 Diagramma di flusso di massima dell'algoritmo SGP4.

Come evidenziato nella figura precedente, le operazioni principali svolte dall'algoritmo SGP4 si possono riassumere nei seguenti punti principali:

- Estrazioni di tutte le informazioni necessarie dalle stringhe componenti la TLE in esame (si rimanda il lettore al capitolo 3, paragrafo 3, per la descrizione del formato della TLE).
- Inizializzazione di tutte le variabili necessarie all'algoritmo di calcolo della posizione del satellite.
- Routine SGP4 vera e propria; è questa parte dell'algoritmo che si occupa di calcolare il vettore di posizione del satellite, indicato con il simbolo  $\vec{r}$ , nel sistema di riferimento geocentrico inerziale IJK. L'algoritmo in questione riceve come parametro d'ingresso il tempo, indicato con il simbolo  $\Delta t$ , in cui si vuole conoscere l'informazione relativa alla posizione del satellite. Questo parametro raffigura il tempo trascorso (espresso in minuti) dalla data (compresa di ora) in cui è stata creata la TLE in esame.

Gli algoritmi presentati in questo capitolo permettono di ricavare i vettori di posizione e velocità del satellite, specificati nel sistema di riferimento geocentrico inerziale. Prendendo le stesse coordinate ottenute ed invertite di segno è possibile ricavare le coordinate del centro della Terra, come viste dal satellite.

Indicando con  $\vec{r}_e$  il vettore di posizione del centro della Terra, visto dal satellite, si ottiene quindi:

$$\vec{r}_e = -\vec{r} \quad (4.30)$$

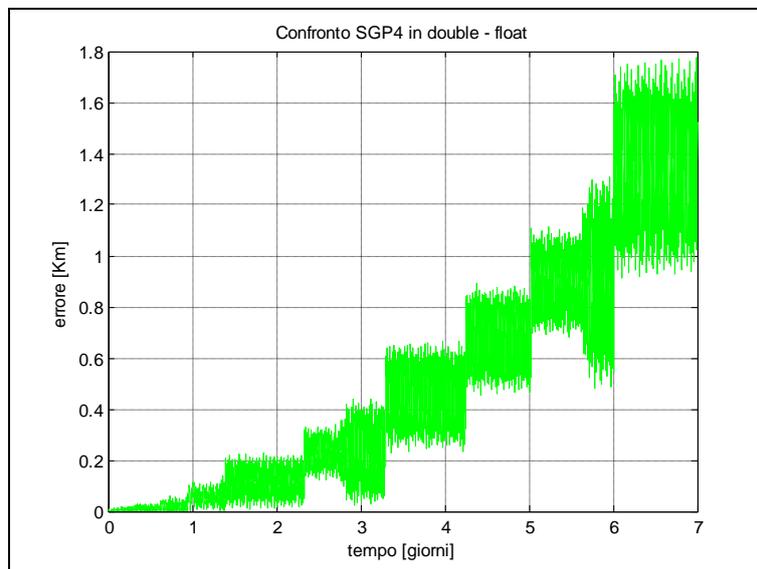
Quest'ultima operazione risulta utile per ottenere un modello della posizione della Terra vista dal satellite e risulta un'applicazione aggiuntiva molto semplice da inserire negli algoritmi già analizzati.

Si riporta in conclusione il confronto tra l'algoritmo SGP4 rilasciato dal NORAD ed una versione simile in cui tutte le variabili di tipo double sono state opportunamente modificate in float (al fine di risparmiare memoria su satellite). Al fine di effettuare una prova delle due versioni dell'algoritmo, è stata presa una TLE di test ed è stato ripetuto il calcolo della posizione del satellite in esame per una distanza temporale di sette giorni a partire dalla data specificata nella TLE, con un passo pari ad 1 minuto. Si riporta di seguito il formato della TLE di test utilizzato:

DTUSAT

```
1 27842U 03031C 09166.20608466 .00000024 00000-0 31026-4 0 5778
2 27842 98.7036 176.3713 0009831 124.5029 235.7083 14.20982944309069
```

Nella figura 4.4 seguente è possibile osservare le differenze tra i valori ottenuti mediante le due versioni del medesimo algoritmo:



**Figura 4.4 Errori algoritmo SGP4 in double e float.**

Nella figura precedente si riporta in ascissa il tempo trascorso a partire dalla data della TLE ed in ordinata la radice quadrata della somma al quadrato delle differenze delle componenti del vettore di posizione ottenute con le due versioni dell'algoritmo. Come risulta semplice osservare l'errore tra le due versioni non risulta essere eccessivamente elevato.

Pur non conoscendo quale sarà effettivamente il processore di bordo del satellite AraMiS, la versione in float dell'algoritmo SGP4 è stata implementata su di un processore MSP430FG439, con lo scopo di valutare approssimativamente le tempistiche di calcolo. Questo processore rappresenta un possibile candidato a divenire il processore di bordo di AraMiS ed ha un clock a 8 MHz. Nella prova effettuata sono stati compiuti 50 calcoli della posizione del satellite ed il processore in analisi ha impiegato 30 secondi per eseguire i vari computi. Questo equivale ad affermare che questo tipo di processore, con le sue peculiari caratteristiche, impiega circa 0.6 secondi ad eseguire il calcolo circa la posizione del satellite nell'orbita. Le prestazioni non risultano quindi del tutto disprezzabili, soprattutto se si considera la possibilità di utilizzare degli MSP con prestazioni migliori, aventi una frequenza di clock anche tre volte maggiore.

## 4.4 Il confronto dei risultati ottenuti mediante l'ambiente STK

In questo paragrafo si riporta il confronto tra i valori ottenuti mediante l'algoritmo SGP4 utilizzato ed i valori ottenuti sfruttando l'ambiente STK.

Al fine di effettuare una prova dell'algoritmo SGP4 è stata presa una TLE di test, la medesima descritta nel paragrafo precedente. I risultati ottenuti sono stati confrontati con i valori calcolati mediante l'ambiente STK. STK è un pacchetto software commerciale mediante il quale è possibile effettuare delle simulazioni delle orbite satellitari. Il software in questione permette di inserire come ingresso una TLE (è stata inserita la stessa TLE riportata in precedenza) e di scegliere il tipo di propagatore dell'orbita. Con lo scopo di confrontare i risultati ottenuti mediante l'algoritmo SGP4 sviluppato è stato impostato anche in STK il propagatore SGP4. Si noti che, pur avendo lo stesso nome, i due algoritmi messi a confronto potrebbero avere delle differenze, dal momento che l'algoritmo SGP4 utilizzato da STK potrebbe avere delle differenze rispetto all'algoritmo rilasciato al pubblico dal NORAD. Si riporta in figura 4.5 una rappresentazione 3D dell'orbita in questione, definita mediante la TLE di test precedente, ottenuta durante la simulazione effettuata in STK:

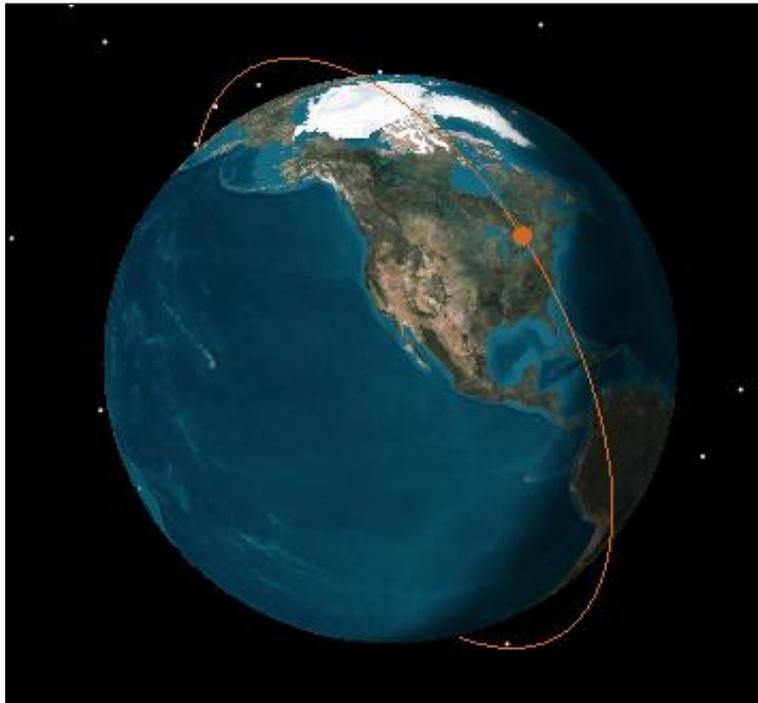
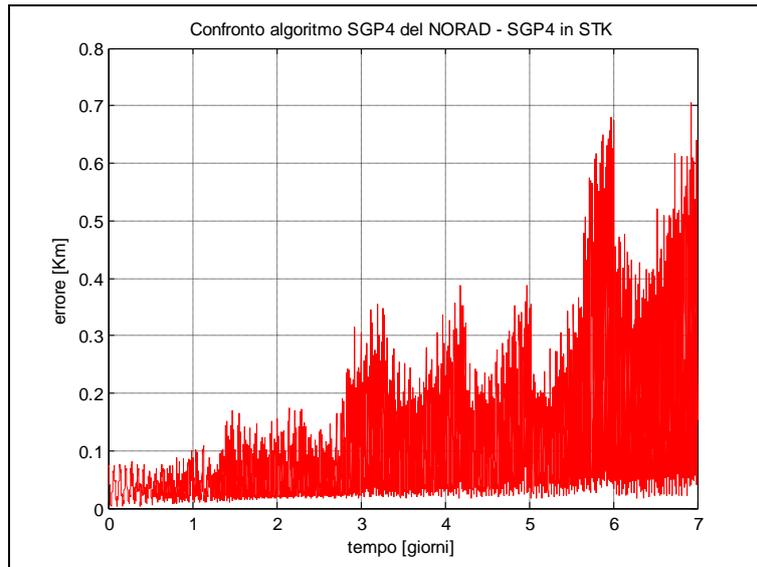


Figura 4.5 Simulazione di un'orbita mediante l'ambiente STK.

Si riporta in conclusione, nella figura 4.6, un grafico relativo agli errori dell'algoritmo SGP4 rispetto ai valori ottenuti mediante STK:



**Figura 4.6 Errori SGP4 fornito dal NORAD e SGP4 di STK.**

Nella figura precedente è riportato in ascissa il tempo trascorso a partire dalla data specificata nella TLE in esame ed in ordinata la radice quadrata della somma al quadrato delle differenze delle componenti del vettore di posizione ottenute mediante i due algoritmi in questione. E' possibile notare che, dopo una settimana dalla data specificata nella TLE, l'errore risulta essere inferiore ad 1 Km. Ricordando infine che l'accuratezza dell'algoritmo SGP4 è tipicamente di 1 Km è possibile confermare l'attendibilità dell'algoritmo utilizzato.

# Capitolo 5

## 5 Il problema della determinazione dell'assetto nello spazio

In questo capitolo si presenta il problema, da un punto di vista generale, riguardante la determinazione dell'orientamento (*assetto*) del satellite nello spazio tridimensionale. In questo ambito ricoprono una notevole importanza i parametri utilizzati nella rappresentazione dell'assetto. A tal scopo sono analizzati i principali fra essi, cercando di evidenziarne pregi e difetti (*parametrizzazione dell'assetto*).

### 5.1 Il problema generale della determinazione dell'assetto del satellite e la matrice dei coseni direttori

Si assuma che il satellite in esame sia provvisto di una terna destrorsa ortogonale, indicata con i simboli UVW, rappresentata dai vettori unitari  $\hat{u}$ ,  $\hat{v}$ ,  $\hat{w}$ , solidali con la struttura meccanica dello stesso (*sistema di riferimento del satellite*), definiti in modo che risulti soddisfatta la relazione seguente:

$$\hat{u} \times \hat{v} = \hat{w} \quad (5.1)$$

Dove il simbolo  $\times$  rappresenta l'operazione di prodotto vettoriale. Il problema di base della determinazione dell'assetto del satellite consiste nello specificare l'orientamento della terna UVW, e quindi del satellite, nello spazio tridimensionale rispetto a qualche sistema di coordinate di riferimento prefissato, indicato con i simboli 123 (*sistema di riferimento assoluto*). In questo capitolo si indica con 123 un sistema di riferimento generico, nei capitoli successivi si farà sempre riferimento al sistema di riferimento geocentrico inerziale IJK e tutti gli orientamenti saranno specificati rispetto ad esso. Si riporta in figura 5.1 lo schema di principio del problema che si vuole affrontare:

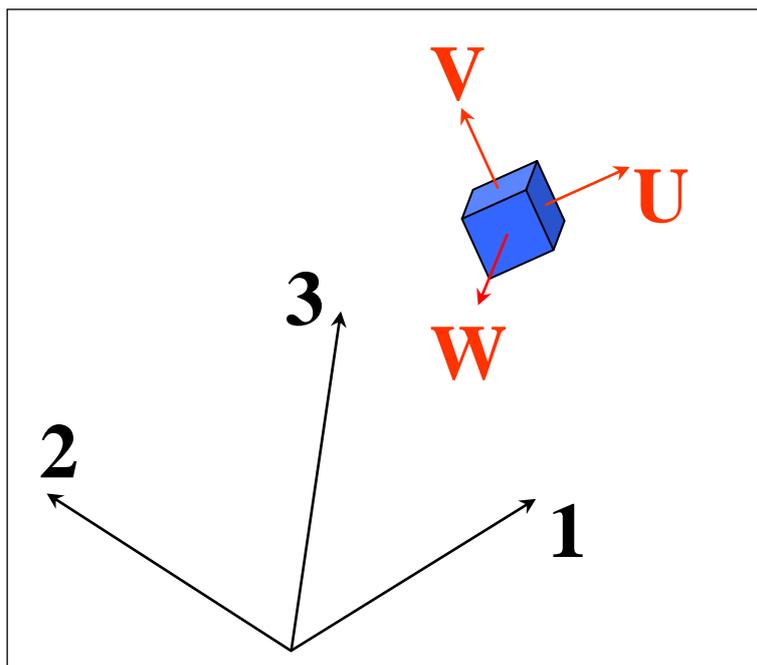


Figura 5.1 Problema della determinazione dell'assetto nello spazio.

Evidentemente specificando le componenti di  $\hat{u}$ ,  $\hat{v}$  e  $\hat{w}$  lungo i tre assi del sistema di riferimento assoluto risulta possibile fissare l'orientamento del satellite in modo completo. Questo procedimento richiede nove parametri (tre parametri per ogni asse del satellite), i quali possono essere definiti come gli elementi di una matrice di dimensioni  $(3 \times 3)$ . In genere questa matrice è chiamata *matrice d'assetto*, o *matrice di rotazione*, ed è indicata con la lettera  $A$ . Si riporta nella relazione 5.2 seguente la definizione della matrice appena citata:

$$A \equiv \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix} \quad (5.2)$$

Dove  $\hat{u} = (u_1, u_2, u_3)^T$ ,  $\hat{v} = (v_1, v_2, v_3)^T$  e  $\hat{w} = (w_1, w_2, w_3)^T$ . Ognuno di questi elementi rappresenta il coseno dell'angolo compreso tra il vettore unitario e l'asse di riferimento del sistema assoluto prefissato;  $u_1$ , per esempio, raffigura il coseno dell'angolo compreso tra  $\hat{u}$  e l'asse di riferimento 1. Per questo motivo la matrice  $A$  è sovente indicata con il nome di *matrice dei coseni direttori*. Gli elementi della matrice dei coseni direttori, come risulta semplice intuire, non possono essere tutti indipendenti. Per esempio, dal momento che  $\hat{u}$  è un vettore unitario, deve essere considerato il vincolo sul modulo dello stesso, come mostra la relazione seguente:

$$u_1^2 + u_2^2 + u_3^2 = 1 \quad (5.3)$$

Inoltre, la definizione dell'ortogonalità tra  $\hat{u}$  e  $\hat{v}$  implica la relazione seguente:

$$u_1 v_1 + u_2 v_2 + u_3 v_3 = 0 \quad (5.4)$$

Queste relazioni appena riportate possono essere riassunte dall'affermazione che il prodotto di  $A$  per la sua trasposta corrisponde alla matrice identità:

$$AA^T = \mathbf{1} \quad (5.5)$$

Questo risulta essere equivalente a dichiarare che  $A$  è una matrice *ortogonale*. La definizione di determinante della stessa matrice è riportata nella relazione 5.6 seguente:

$$\det A = \hat{u} (\hat{v} \times \hat{w}) \quad (5.6)$$

Dal momento che  $\hat{u}$ ,  $\hat{v}$ ,  $\hat{w}$  formano una terna destrorsa, il determinante della matrice  $A$  è sicuramente uguale a 1. Questa relazione può essere utilizzata come una rapida verifica della correttezza della matrice d'assetto ottenuta durante i calcoli.

La matrice dei coseni direttori è fisicamente interpretabile come una trasformazione di coordinate che mappa i vettori specificati rispetto al sistema di riferimento assoluto 123 nel sistema di riferimento del satellite UVW. Quindi, indicando con  $\bar{a}$  un vettore con componenti  $a_1, a_2, a_3$  lungo gli assi di riferimento 123, si ottiene la relazione seguente:

$$A\bar{a} = \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} u \cdot a \\ v \cdot a \\ w \cdot a \end{bmatrix} \equiv \begin{bmatrix} a_u \\ a_v \\ a_w \end{bmatrix} \quad (5.7)$$

Le componenti di  $A\bar{a}$  rappresentano le componenti del vettore  $\bar{a}$  rispetto alla terna solidale con la struttura del satellite, definita dai versori  $\hat{u}$ ,  $\hat{v}$ ,  $\hat{w}$ . Una trasformazione di coordinate di questo tipo, ovvero eseguita con una matrice ortogonale, preserva le lunghezze dei vettori e gli angoli compresi tra gli stessi, e quindi rappresenta una *rotazione*. Questo giustifica il nome di matrice di rotazione, introdotto all'inizio del paragrafo. È possibile anche raggruppare delle rotazioni successive. Il prodotto di due matrici ortogonali,  $A'' = A'A$ , rappresenta il risultato di due successive rotazioni, di  $A$  e  $A'$ , rispettivamente. Dal momento che la trasposta e l'inversa di una matrice ortogonale risultano essere coincidenti, la matrice  $A^T$  risulta essere la trasformazione di coordinate che mappa i vettori specificati rispetto al sistema di riferimento del satellite UVW nel sistema di riferimento assoluto 123 prefissato.

Dall'algebra delle matrici è possibile ricavare che ogni matrice ortogonale possiede almeno un autovalore unitario cui è associato un autovettore. Questo implica l'esistenza di un vettore unitario, indicato con il simbolo  $\bar{e}$ , che non è ruotato dalla matrice  $A$ , come mostra la relazione seguente:

$$A\bar{e} = \bar{e} \quad (5.8)$$

Il vettore  $\bar{e}$  possiede quindi le stesse componenti lungo gli assi del sistema di riferimento del satellite UVW e lungo gli assi del sistema di riferimento assoluto 123 prefissato. Quindi, è possibile affermare che  $\bar{e}$  è un vettore orientato secondo l'asse di rotazione in questione. L'esistenza del vettore  $\bar{e}$  dimostra il Teorema di Eulero, che afferma:

*Lo spostamento più generale di un corpo rigido con un punto fisso è una rotazione attorno ad un asse.*

Si noti che l'espressione 5.8 non è nient'altro che il problema agli autovalori associato alla matrice d'assetto  $A$ . Calcolando gli autovalori di  $A$  e ricavando, di conseguenza, l'autovettore associato all'autovalore unitario si ottiene l'asse su cui la rotazione avviene.

La matrice dei coseni direttori risulta essere l'entità fondamentale utilizzata nella descrizione dell'orientamento di un corpo nello spazio ed è stata impiegata durante lo sviluppo dell'algoritmo indirizzato alla determinazione dell'assetto del satellite. Sono, tuttavia, utilizzabili altri parametri d'assetto che possono risultare più o meno convenienti e saranno descritti nel seguito del capitolo.

## 5.2 La rappresentazione dell'assetto del satellite mediante l'asse e l'angolo di Eulero

Nel paragrafo precedente è stata dichiarata l'esistenza del vettore  $\vec{e}$ , orientato secondo l'asse di rotazione, per qualunque matrice dei coseni direttori. L'asse individuato da questo vettore è detto *asse di Eulero* e rappresenta l'asse attorno al quale avviene la rotazione. L'angolo di rotazione attorno a quest'asse è invece indicato con il nome di *angolo di Eulero*. Asse ed angolo di Eulero sono due parametri utilizzabili nella rappresentazione dell'orientamento del satellite, conservano il significato fisico del fenomeno e sono utili nella descrizione di altri parametri molto importanti analizzati nel seguito della discussione (quaternioni unitari). La rappresentazione dell'assetto del satellite mediante l'asse e l'angolo di Eulero necessita di quattro parametri; tre utilizzati per indicare l'asse di rotazione, ed uno utilizzato per indicare l'angolo di rotazione. Quindi, rispetto alla rappresentazione mediante la matrice dei coseni direttori, sono sufficienti cinque parametri in meno. Inoltre solo tre dei parametri risultano essere indipendenti, dal momento che deve valere il vincolo sul modulo, ovvero  $|\vec{e}|=1$ .

Una rotazione particolarmente semplice è sicuramente rappresentata da una rotazione attorno ad uno dei tre assi del sistema di riferimento interessato, indicato con il simbolo  $\Phi$ . Il senso positivo della rotazione è fissato, per convenzione, quello antiorario. Si supponga, per esempio, di effettuare una rotazione attorno all'asse 3 di un angolo  $\Phi$ . Si riporta in figura 5.2 la particolare geometria in questione (l'asse 3 in questione non è visibile in figura, dal momento che risulta uscente dal piano della figura stessa):

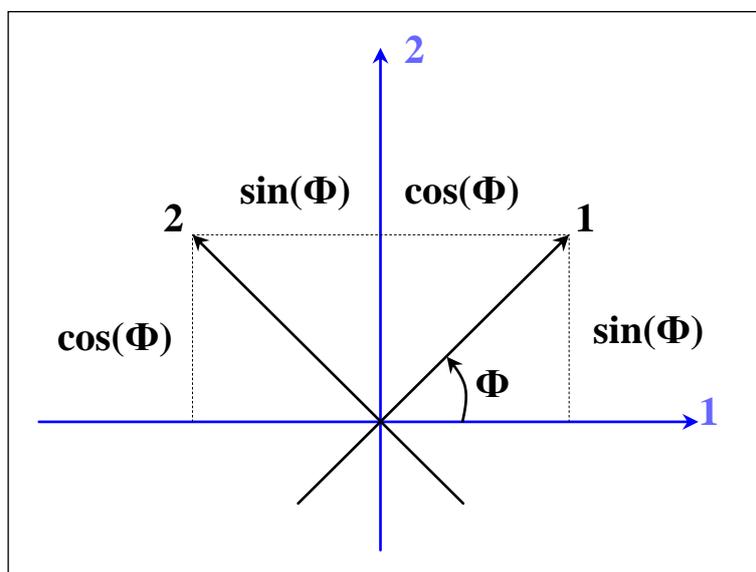


Figura 5.2 Rotazione attorno ad un asse.

La matrice dei coseni direttori, per questa semplice rotazione, indicata con il simbolo  $A_3(\Phi)$ , risulta avere la forma seguente:

$$A_3(\Phi) = \begin{bmatrix} \cos(\Phi) & \sin(\Phi) & 0 \\ -\sin(\Phi) & \cos(\Phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \vec{e} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (5.9a)$$

Allo stesso modo è possibile definire una rotazione di un angolo  $\Phi$  attorno agli assi 1 e 2, indicata con i simboli  $A_1(\Phi)$  e  $A_2(\Phi)$ , rispettivamente, nel modo seguente:

$$A_1(\Phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \quad \vec{e} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (5.9b)$$

$$A_2(\Phi) = \begin{bmatrix} \cos(\Phi) & 0 & -\sin(\Phi) \\ 0 & 1 & 0 \\ \sin(\Phi) & 0 & \cos(\Phi) \end{bmatrix} \quad \vec{e} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (5.9c)$$

Osservando le matrici  $A_1(\Phi)$ ,  $A_2(\Phi)$  e  $A_3(\Phi)$  si può notare facilmente che tutte e tre hanno la stessa traccia, data dalla relazione seguente:

$$\text{tr}(A(\Phi)) = 1 + 2\cos(\Phi) \quad (5.10)$$

Dalla relazione 5.10 è quindi facilmente ricavabile l'angolo di Eulero. Infatti, invertendo la relazione si ottiene:

$$\cos(\Phi) = \frac{1}{2}(\text{tr}(A) - 1) \quad (5.11)$$

Conoscendo la matrice  $A$ , quindi, è possibile calcolare sia l'asse, sia l'angolo di Eulero mediante semplici operazioni. La matrice dei coseni direttori, espressa in funzione dell'asse e dell'angolo di Eulero ottiene la forma specificata nell'espressione 5.12 seguente:

$$A = \begin{bmatrix} \cos(\Phi) + e_1^2(1 - \cos(\Phi)) & e_1e_2(1 - \cos(\Phi)) + e_3\sin(\Phi) & e_1e_3(1 - \cos(\Phi)) - e_2\sin(\Phi) \\ e_1e_2(1 - \cos(\Phi)) - e_3\sin(\Phi) & \cos(\Phi) + e_2^2(1 - \cos(\Phi)) & e_2e_3(1 - \cos(\Phi)) + e_1\sin(\Phi) \\ e_1e_3(1 - \cos(\Phi)) + e_2\sin(\Phi) & e_2e_3(1 - \cos(\Phi)) - e_1\sin(\Phi) & \cos(\Phi) + e_3^2(1 - \cos(\Phi)) \end{bmatrix} \quad (5.12)$$

Quindi, noti i parametri d'assetto consistenti nell'asse e nell'angolo di Eulero, è possibile ottenere la matrice d'assetto  $A$ . E' possibile effettuare anche l'operazione inversa, ovvero ricavare  $\vec{e}$  e  $\Phi$  a partire dalla matrice dei coseni direttori, sfruttando le relazioni seguenti:

$$\Phi = \cos^{-1} \left[ \frac{1}{2}(\text{tr}(A) - 1) \right] \quad (5.13)$$

$$\begin{cases} e_1 = \frac{(A_{23} - A_{32})}{2\sin(\Phi)} \\ e_2 = \frac{(A_{31} - A_{13})}{2\sin(\Phi)} \\ e_3 = \frac{(A_{12} - A_{21})}{2\sin(\Phi)} \end{cases} \quad (5.14)$$

La matrice  $A$  è sempre possibile ricavarla a partire dall'asse e dall'angolo di Eulero. Il passaggio inverso, invece, non risulta essere sempre consentito, infatti, quando il seno di  $\Phi$  si annulla le forme 5.14 diventano indeterminate. Questo avviene per valori di  $\Phi = n\pi$ , che fisicamente si può interpretare come il fatto che l'asse di rotazione, in questa situazione, non è univocamente determinato, ovvero è possibile raggiungere la stessa posizione mediante due rotazioni aventi lo stesso angolo ma asse diverso. Quando non risulta possibile invertire la parametrizzazione dell'assetto si parla di singolarità della parametrizzazione d'assetto. Inoltre, la funzione arcoseno fornisce due soluzioni per l'angolo di Eulero ( $\pm \Phi$ ) che portano a due assi ( $\pm \bar{e}$ ). Questo non è però un problema, infatti, si tratta dello stesso asse che visto dalla direzione negativa rappresenta la stessa rotazione con l'opposto dell'angolo.

### 5.3 La rappresentazione dell'assetto del satellite mediante i quaternioni unitari

Una parametrizzazione della matrice dei coseni direttori particolarmente conveniente nell'ambito spaziale risulta essere quella fornita dai *parametri simmetrici di Eulero*, detti anche *quaternioni unitari*. I quaternioni sono vettori algebrici e non fisici, indicati con il simbolo  $\vec{q}$ , e sono legati alla parametrizzazione dell'assetto mediante l'asse e l'angolo di Eulero per mezzo delle relazioni riportate di seguito:

$$\begin{cases} q_1 \equiv e_1 \sin\left(\frac{\Phi}{2}\right) \\ q_2 \equiv e_2 \sin\left(\frac{\Phi}{2}\right) \\ q_3 \equiv e_3 \sin\left(\frac{\Phi}{2}\right) \\ q_0 \equiv \cos\left(\frac{\Phi}{2}\right) \end{cases} \quad (5.15)$$

Questi quattro parametri non sono indipendenti, infatti, deve valere il vincolo di normalizzazione seguente:

$$q_1^2 + q_2^2 + q_3^2 + q_0^2 = 1 \quad (5.16)$$

Per questo motivo si parla di parametrizzazione dell'assetto mediante *quaternioni unitari*.

Sovente il quaternione è rappresentato dividendolo in due parti distinte; una parte vettoriale ed una parte scalare, nel seguente modo:

$$\vec{q} = \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \end{Bmatrix}, \quad q_0 \quad (5.17)$$

La matrice dei coseni direttori, scritta in funzione dei quaternioni unitari, acquisisce la forma seguente:

$$A = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_0^2 & 2(q_1q_2 + q_3q_0) & 2(q_1q_3 - q_2q_0) \\ 2(q_1q_2 - q_3q_0) & -q_1^2 + q_2^2 - q_3^2 + q_0^2 & 2(q_2q_3 + q_1q_0) \\ 2(q_1q_3 + q_2q_0) & 2(q_2q_3 - q_1q_0) & -q_1^2 - q_2^2 + q_3^2 + q_0^2 \end{bmatrix} \quad (5.18)$$

Anche in questo caso è possibile effettuare la trasformazione inversa, ovvero ricavare i parametri del quaternione a partire dalla matrice  $A$ , sfruttando le seguenti formule di conversione:

$$\left\{ \begin{array}{l} q_1 = \frac{1}{4q_0}(A_{23} - A_{32}) \\ q_2 = \frac{1}{4q_0}(A_{31} - A_{13}) \\ q_3 = \frac{1}{4q_0}(A_{12} - A_{21}) \\ q_0 = \pm \frac{1}{2}(1 + A_{11} + A_{22} + A_{33})^{\frac{1}{2}} \end{array} \right. \quad (5.19)$$

E' possibile spiegare, anche per questi parametri, l'ambiguità di segno, come dichiarato nel caso dell'asse e dell'angolo di Eulero. La parametrizzazione mediante quaternioni risulta però molto più conveniente. Di fatto, nella relazione 5.18 non sono presenti funzioni trigonometriche le quali richiedono lunghe operazioni computazionali da effettuare a bordo del satellite. Inoltre, con questi parametri è possibile eliminare il problema della singolarità della parametrizzazione, dal momento che almeno uno dei quattro parametri risulta essere sempre diverso da zero. Nella relazione 5.19, quindi, se  $q_0$  risulta nullo è sempre possibile cercare di calcolare uno degli altri tre, che sicuramente non sarà nullo, e ricavare gli altri di conseguenza. Per la trasformazione inversa sono quindi valide le seguenti alternative:

$$\left\{ \begin{array}{l} q_0 = \frac{1}{4q_1}(A_{23} - A_{32}) \\ q_2 = \frac{1}{4q_1}(A_{12} + A_{21}) \\ q_3 = \frac{1}{4q_1}(A_{13} + A_{31}) \\ q_1 = \pm \frac{1}{2}(1 + A_{11} - A_{22} - A_{33})^{\frac{1}{2}} \end{array} \right\} \left\{ \begin{array}{l} q_1 = \frac{1}{4q_2}(A_{12} + A_{21}) \\ q_3 = \frac{1}{4q_2}(A_{23} + A_{32}) \\ q_0 = \frac{1}{4q_2}(A_{31} - A_{13}) \\ q_2 = \pm \frac{1}{2}(1 - A_{11} + A_{22} - A_{33})^{\frac{1}{2}} \end{array} \right\} \left\{ \begin{array}{l} q_1 = \frac{1}{4q_3}(A_{13} + A_{31}) \\ q_2 = \frac{1}{4q_3}(A_{23} + A_{32}) \\ q_0 = \frac{1}{4q_3}(A_{12} - A_{21}) \\ q_3 = \pm \frac{1}{2}(1 - A_{11} - A_{22} + A_{33})^{\frac{1}{2}} \end{array} \right. \quad (5.20)$$

Siccome i quaternioni ricoprono un ruolo importante in ambito spaziale, se ne riportano, infine, le proprietà fondamentali. Si consideri il quaternione unitario seguente:

$$\bar{q} = q_0 + iq_1 + jq_2 + kq_3 \quad (5.21)$$

Dove  $i, j, e k$  rappresentano i numeri iper immaginari che soddisfano le seguenti condizioni:

$$\begin{cases} i^2 = j^2 = k^2 = -1 \\ ij = -ji = k \\ jk = -kj = i \\ ki = -ik = j \end{cases} \quad (5.22)$$

L'operazione di somma di due quaternioni,  $\vec{q}'$  e  $\vec{q}''$ , è definita nel modo seguente:

$$\vec{q}' + \vec{q}'' = (q_0' + q_0'') + i(q_1' + q_1'') + j(q_2' + q_2'') + k(q_3' + q_3'') \quad (5.23)$$

Il quaternioni coniugato, o inverso, di  $\vec{q}$  è definito nel modo seguente:

$$\vec{q}^* = q_0 - iq_1 - jq_2 - kq_3 \quad (5.24)$$

La moltiplicazione tra quaternioni è effettuata in modo simile all'operazione di moltiplicazione tra numeri complessi, con l'eccezione che l'ordine delle operazioni deve essere preso in considerazione. Infatti, le equazioni 5.22 non risultano essere commutative.

Il prodotto di due quaternioni,  $\vec{q}'$  e  $\vec{q}''$ , è definito nel modo seguente:

$$\begin{aligned} \vec{q}' \vec{q}'' &= (-q_1' q_1'' - q_2' q_2'' - q_3' q_3'' + q_0' q_0'') \\ &+ i(q_1' q_0'' + q_2' q_3'' - q_3' q_2'' + q_0' q_1'') \\ &+ j(-q_1' q_3'' + q_2' q_0'' + q_3' q_1'' + q_0' q_2'') \\ &+ k(q_1' q_2'' - q_2' q_1'' + q_3' q_4'' + q_4' q_3'') \end{aligned} \quad (5.25)$$

In conclusione, la lunghezza o norma del quaternioni  $\vec{q}$  è definita nel modo seguente:

$$|\vec{q}| \equiv \sqrt{\vec{q}\vec{q}^*} = \sqrt{\vec{q}^*\vec{q}} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \quad (5.26)$$

## 5.4 La rappresentazione dell'assetto del satellite mediante gli angoli di Eulero

Gli angoli di Eulero rappresentano tre parametri d'assetto facilmente interpretabili fisicamente. Questi parametri non sono convenienti dal punto di vista computazionale rispetto ai parametri simmetrici di Eulero, sono tuttavia molto utilizzati nelle applicazioni software come formato di input ed output, in virtù della loro facile interpretazione.

L'idea cardine si basa sul fatto che dati due qualsiasi sistemi di riferimento è sempre possibile farli coincidere con una serie di tre rotazioni successive rispetto agli assi di uno dei due.

Per fissare le idee si considerino quattro terne di versori ortogonali, indicate dai simboli seguenti:

- $\hat{x}, \hat{y}, \hat{z}$
- $\hat{x}', \hat{y}', \hat{z}'$
- $\hat{x}'', \hat{y}'', \hat{z}''$
- $\hat{x}''', \hat{y}''', \hat{z}'''$

La terna iniziale  $\hat{x}, \hat{y}, \hat{z}$  è parallela al sistema di riferimento assoluto 123. La terna  $\hat{x}', \hat{y}', \hat{z}'$  differisce da  $\hat{x}, \hat{y}, \hat{z}$  a causa di una rotazione attorno all'asse  $i$  ( $i = \hat{x}, \hat{y},$  o  $\hat{z}$  secondo la particolare trasformazione) di un angolo  $\Phi$ . Quindi l'orientamento di  $\hat{x}', \hat{y}', \hat{z}'$  rispetto alla terna  $\hat{x}, \hat{y}, \hat{z}$  è dettato dalla matrice di rotazione  $A_i(\Phi)$  con  $i = 1, 2$  o  $3$ , rappresentante una delle semplici matrici dei coseni direttori date dalle equazioni 5.9 precedenti. Allo stesso modo l'orientamento della terna  $\hat{x}'', \hat{y}'', \hat{z}''$  rispetto a  $\hat{x}', \hat{y}', \hat{z}'$  è dettato da una rotazione attorno ad un asse coordinato di  $\hat{x}', \hat{y}', \hat{z}'$  di un angolo  $\theta$ , specificata dalla matrice dei coseni direttori  $A_j(\theta)$ , con  $j = 1, 2,$  o  $3$  e  $j \neq i$ . Infine, l'orientamento della terna  $\hat{x}''', \hat{y}''', \hat{z}'''$  rispetto a  $\hat{x}'', \hat{y}'', \hat{z}''$  rappresenta una terza rotazione, di un angolo  $\Psi$ , specificata dalla matrice dei coseni direttori  $A_k(\Psi)$ , con  $k = 1, 2,$  o  $3$  e  $k \neq j$ . Quindi, in seguito alle tre operazioni eseguite, è stato possibile far coincidere le terne  $\hat{x}, \hat{y}, \hat{z}$  e  $\hat{x}'', \hat{y}'', \hat{z}''$  e le tre rotazioni successive rappresentano l'orientamento della terna  $\hat{x}''', \hat{y}''', \hat{z}'''$  rispetto a  $\hat{x}, \hat{y}, \hat{z}$ .

E' necessario tenere presente che i prodotti delle matrici di rotazione vanno sempre eseguiti nell'ordine inverso a quello delle rotazioni in questione.

Come esempio, si riporta la sequenza di angoli di Eulero 3-1-3. I simboli 3-1-3 stanno ad indicare che le rotazioni sono eseguite nell'ordine seguente:

1. Rotazione attorno all'asse Z di un angolo  $\Phi$ .
2. Rotazione attorno all'asse X di un angolo  $\theta$ .
3. Rotazione attorno all'asse Z di un angolo  $\Psi$ .

La matrice dei coseni direttori, in questa particolare situazione, è definita dalla seguente espressione:

$$A_{313}(\Phi, \theta, \Psi) = A_3(\Psi)A_1(\theta)A_3(\Phi) =$$

$$\begin{bmatrix} \cos(\Psi)\cos(\Phi) - \cos(\theta)\sin(\Psi)\sin(\Phi) & \cos(\Psi)\sin(\Phi) + \cos(\theta)\sin(\Psi)\cos(\Phi) & \sin(\theta)\sin(\Psi) \\ -\sin(\Psi)\cos(\Phi) - \cos(\theta)\cos(\Psi)\sin(\Phi) & -\sin(\Psi)\sin(\Phi) + \cos(\theta)\cos(\Psi)\cos(\Phi) & \sin(\theta)\cos(\Psi) \\ \sin(\theta)\sin(\Phi) & -\sin(\theta)\cos(\Phi) & \cos(\theta) \end{bmatrix}$$

(5.27)

Gli angoli di Eulero 3-1-3 sono ricavabili a partire dalla matrice dei coseni direttori 5.27 nel modo seguente:

$$\begin{cases} \theta = \cos^{-1}(A_{33}) \\ \Phi = -\tan^{-1}\left(\frac{A_{31}}{A_{32}}\right) \\ \Psi = \tan^{-1}\left(\frac{A_{13}}{A_{23}}\right) \end{cases} \quad (5.28)$$

Mediante la rappresentazione dell'assetto con gli angoli di Eulero si ottiene una diversa matrice  $A$  per ogni sequenza di rotazione. Due rotazioni successive non possono avvenire sullo stesso asse (risultano equivalenti ad una rotazione data dalla somma delle singole rotazioni), quindi le possibili sequenze di rotazioni si riducono a dodici, sei con indici tutti diversi e sei con due indici non consecutivi uguali, come riportato di seguito:

- 3-1-2, 2-1-3, 1-2-3, 3-2-1, 2-3-1, 1-3-2      (indici diversi)
- 3-1-3, 3-2-3, 2-1-2, 2-3-2, 1-3-1, 1-2-1      ( $1^\circ$  e  $3^\circ$  indice uguale)

Le diverse famiglie di sequenze di angoli di Eulero risultano singolari per valori diversi dell'angolo  $\theta$ , in funzione degli indici delle rotazioni. Se gli indici sono tutti diversi la singolarità si ottiene per  $\theta = (2n+1)\pi/2$ , mentre se due indici non consecutivi sono uguali la singolarità si ottiene per  $\theta = n\pi$ . La rappresentazione dell'assetto mediante gli angoli di Eulero richiede molti conti coinvolgenti funzioni trigonometriche e risulta, di conseguenza, sconsigliato da effettuare a bordo del satellite.

# Capitolo 6

## 6 I sistemi di riferimento del satellite e delle tiles ed i sensori d'assetto del satellite

In questo capitolo si descrivono i sistemi di riferimento del satellite e delle tiles ed i principali sensori d'assetto cui è fornita la Power Management tile del satellite AraMiS. I sensori d'assetto risultano essere molto utili nella soluzione del problema riguardante la determinazione dell'orientamento del satellite. L'algoritmo di determinazione dell'assetto è, infatti, basato sulla lettura delle informazioni fornite dai vari sensori d'assetto disposti diversamente sulla struttura del satellite.

### 6.1 I sistemi di riferimento del satellite e delle tiles e le configurazioni del satellite

In questo paragrafo si descrivono i sistemi di riferimento UVW del satellite e XYZ delle tiles componenti il satellite e si mostrano le possibili configurazioni del satellite AraMiS. Come già annunciato nel capitolo 1, secondo la particolare missione spaziale, è possibile scegliere tra diverse configurazioni del satellite.

#### 6.1.1 I sistemi di riferimento del satellite e delle tiles

Il satellite AraMiS è provvisto di un opportuno sistema di riferimento, indicato con i simboli UVW (detto anche sistema degli *assi geometrici* del satellite), definito in fase di costruzione e statico, ovvero il suo orientamento rispetto alla struttura del satellite non varia durante la vita operativa dello stesso. L'origine del sistema di riferimento può essere ogni punto in linea di principio, ma in genere è fissata in modo che essa coincida con il centro geometrico della struttura del satellite, o con il suo centro di gravità. I versori  $\hat{u}$ ,  $\hat{v}$  e  $\hat{w}$ , che definiscono l'orientamento del sistema di riferimento UVW, sono fissati in modo che  $\hat{u} \times \hat{v} = \hat{w}$ , dove il simbolo  $\times$  indica l'operazione di prodotto vettoriale.

Ogni tile costituente la struttura del satellite possiede, a sua volta, il proprio sistema di riferimento, indicato con i simboli XYZ ed identificato dai versori  $\hat{x}$ ,  $\hat{y}$  e  $\hat{z}$ . Il sistema di riferimento XYZ è fissato in modo che  $\hat{x} \times \hat{y} = \hat{z}$  e che il versore  $\hat{z}$  sia ortogonale al piano della tile e diretto in modo uscente dallo stesso. L'origine del sistema di riferimento XYZ è fissata nell'esatto centro geometrico della tile. L'orientamento di ogni sistema XYZ rispetto al sistema di riferimento del

satellite UVW è considerato noto e statico. Il particolare orientamento della tile  $m$  è descritto dalla matrice dei coseni direttori,  $A_m$ , in cui le righe della matrice rappresentano le componenti di  $\hat{x}$ ,  $\hat{y}$  e  $\hat{z}$  rispetto al sistema di riferimento del satellite UVW. Una struttura dati apposita si occupa di memorizzare le matrici dei coseni direttori di tutte le tiles rispetto al sistema di riferimento UVW del satellite. Si riportano nei paragrafi successivi due esempi di geometrie nel caso di un satellite a forma cubica e prismatica, rispettivamente.

Com'è stato presentato nel capitolo 5, l'obiettivo della determinazione dell'assetto è specificare l'orientamento della terna UVW rispetto al sistema di riferimento geocentrico inerziale IJK, ovvero ricavare la matrice d'assetto  $A$  del satellite rispetto al sistema di riferimento IJK.

### 6.1.2 Il satellite a configurazione cubica

Il satellite organizzato in configurazione cubica, avente una tile per ogni faccia, risulta particolarmente semplice da analizzare e da presentare e rappresenta la struttura cubica più piccola realizzabile (configurazione a piccolo cubo). In questa configurazione, chiaramente, il satellite è composto di sei tiles. Si rappresenta in figura 6.1 la geometria in questione. In particolare si notino il sistema di riferimento geocentrico inerziale IJK, il sistema di riferimento del satellite UVW ed il sistema di riferimento XYZ delle tiles:

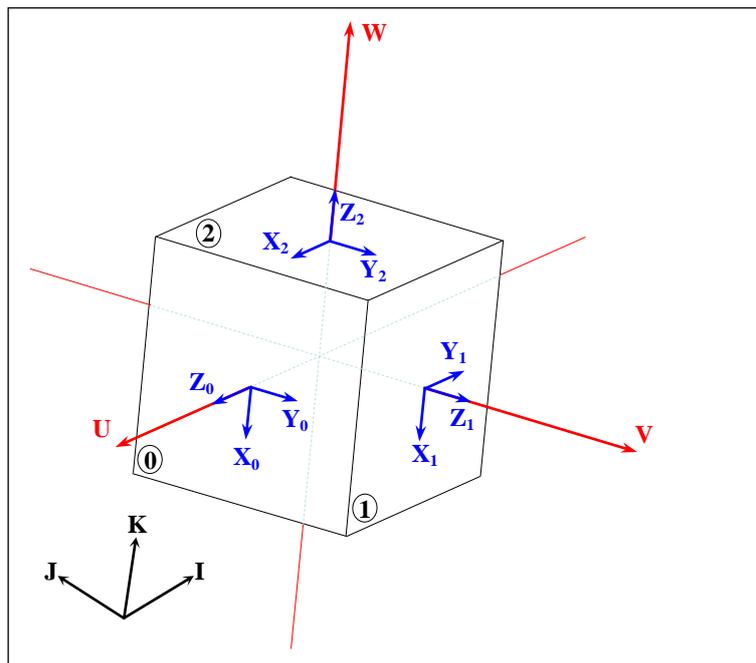


Figura 6.1 Satellite a configurazione cubica.

Nella figura precedente le tiles sono state numerate (la numerazione inizia da 0 per convenzione) al fine di poterle evidenziare separatamente e sono chiaramente visibili le prime tre. Il centro del sistema UVW è definito in modo da coincidere esattamente con il centro geometrico del cubo. L'asse U è fissato in modo che risulti coincidente con la direzione passante per il centro geometrico della tile 0, l'asse V è fissato in modo che risulti coincidente con la direzione passante per il centro geometrico della tile 1 e l'asse W, infine, è fissato in modo che risulti coincidente con la direzione passante per il centro geometrico della tile 2.

Si riporta di seguito l'orientamento, descritto mediante la matrice dei coseni direttori, delle tre tiles rappresentate in figura 6.1, rispetto al sistema di riferimento UVW del satellite (si indica con  $A_i$  la matrice di rotazione della i-esima tile in UVW):

$$A_0 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

Il sistema di riferimento della tile 0, per esempio, è ottenuto ruotando il sistema di riferimento UVW attorno all'asse V di  $90^\circ$ , mentre quello della tile 2 è coincidente con il sistema di riferimento UVW, come risulta semplice osservare nella figura 6.1 precedente.

### 6.1.3 I satelliti a configurazione esagonale ed ottagonale

Per concludere il paragrafo si riportano, nelle figure seguenti, due esempi del satellite a configurazione prismatica, rappresentati dal satellite a configurazione esagonale e dal satellite a configurazione ottagonale, con un esempio di sistema di riferimento UVW definito sulla struttura degli stessi:

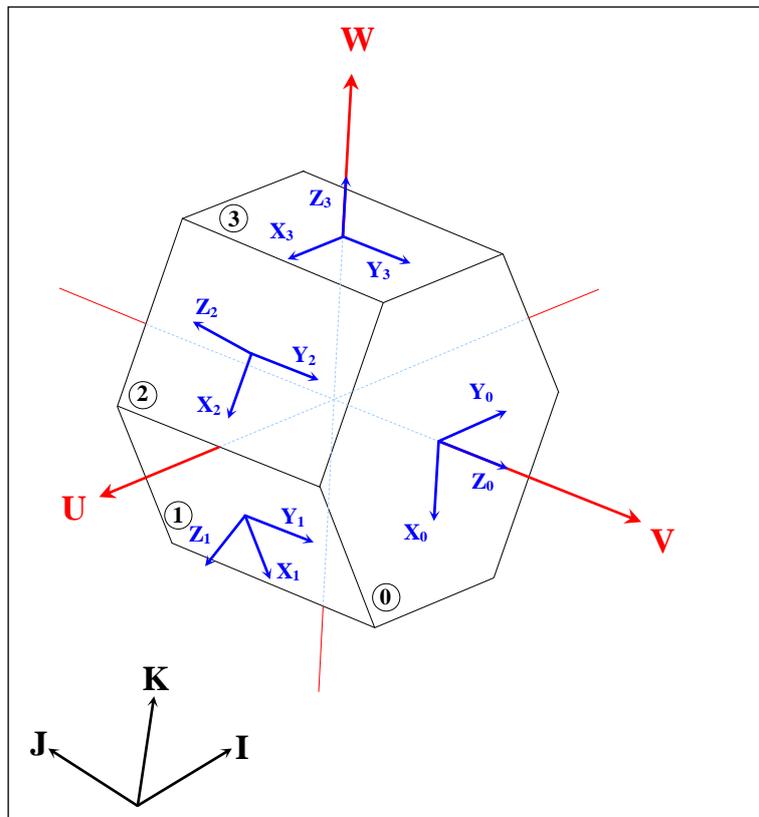


Figura 6.2 Satellite a configurazione esagonale.

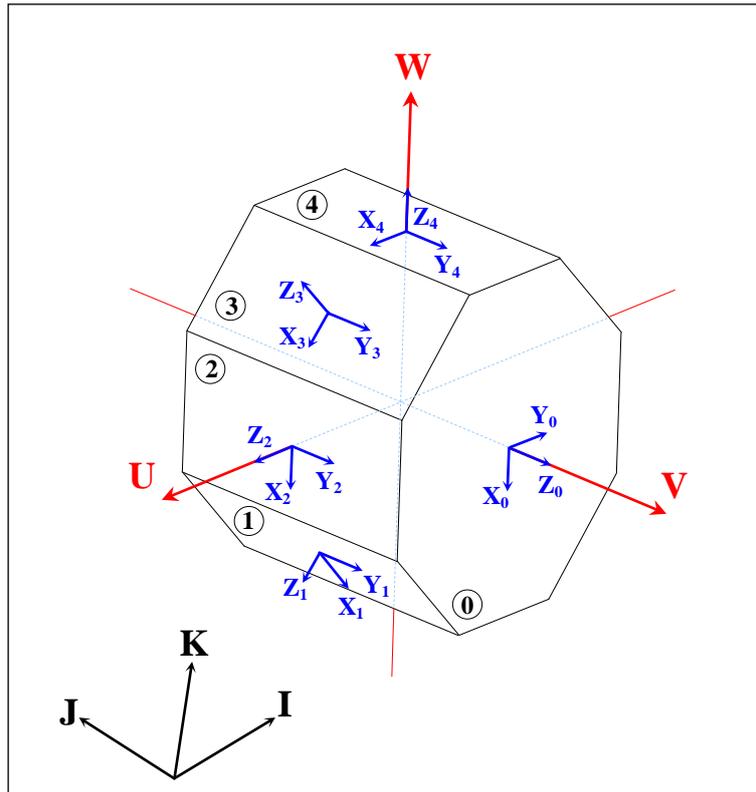


Figura 6.3 Satellite a configurazione ottagonale.

Data la particolare configurazione, questi satelliti a sezione prismatica consentono di ospitare al loro interno strumenti scientifici piuttosto ingombranti, come per esempio dei telescopi.

Si riporta in figura 6.4 un prototipo del satellite AraMiS a configurazione esagonale, in cui è visibile il telescopio montato al suo interno:

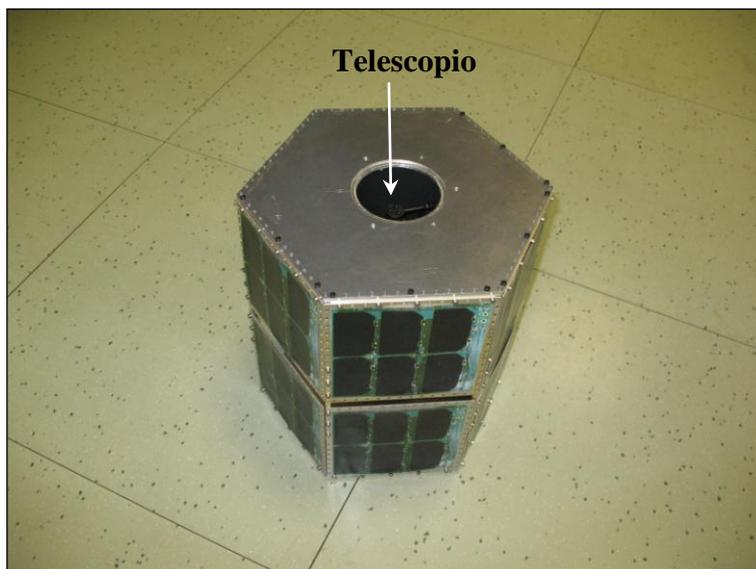


Figura 6.4 Prototipo del satellite AraMiS a configurazione esagonale.

## 6.2 I sensori d'assetto

In questo paragrafo si descrivono i principali sensori d'assetto disposti diversamente sulla struttura del satellite AraMiS ed interrogati durante la risoluzione del problema della determinazione dell'orientamento del satellite. I sensori d'assetto costituiscono una parte fondamentale del sottosistema dedicato alla determinazione ed al controllo dell'assetto del satellite. Questi sensori forniscono una serie di misure relative a qualche direzione nello spazio tridimensionale circostante rispetto al sistema di riferimento XYZ delle tiles su cui sono montati. Grazie a queste misure è possibile affrontare, e risolvere, il problema della determinazione dell'orientamento del satellite.

### 6.2.1 Il sensore di posizione del Sole

In questo paragrafo si descrive il principio di funzionamento del sensore di posizione del Sole presente sulla Power Management tile del satellite AraMiS.

Il sensore di posizione del Sole è sicuramente il tipo di sensore d'assetto più utilizzato in ambito aerospaziale. Questo tipo di sensore specifica la posizione del Sole ed aiuta a risolvere il problema della determinazione dell'assetto del satellite. Il sensore di Sole è composto di una maschera avente un piccolo foro al centro e dal sensore vero e proprio. Ci sono sostanzialmente due tipi di sensori disponibili, il *PSD* (Position-Sensitive-Detector) ed il *CMOS*. La maschera è posizionata ad una certa distanza, indicata con la lettera  $h$ , dal sensore. La luce del Sole passa attraverso il foro e colpisce il sensore, generando un piccolo pallino luminoso sulla superficie del sensore. Si riporta nella figura 6.5 seguente la geometria del sensore in questione:

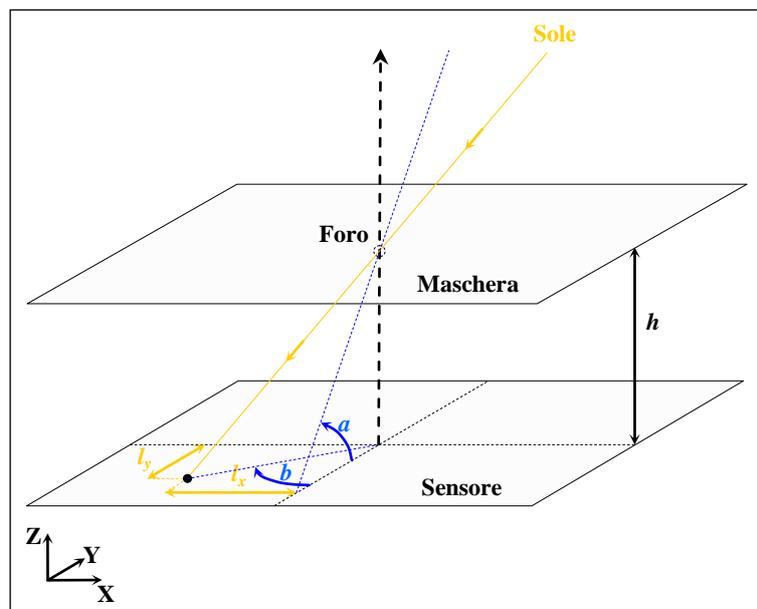


Figura 6.5 Geometria del sensore di posizione del Sole.

Un'apposita circuiteria si occupa di calcolare le due componenti, indicate con i simboli  $l_x$  e  $l_y$ , del pallino luminoso, in X e Y, rispettivamente, rispetto al sistema di riferimento della tile XYZ, e di

ricavare le due informazioni angolari, indicate con i simboli  $a$  e  $b$ . Si noti che l'angolo  $a$  rappresenta lo spostamento angolare compreso tra l'asse  $Z$  della tile e la proiezione della direzione del Sole nel piano  $Z$ - $Y$ , mentre l'angolo  $b$  rappresenta lo spostamento angolare, misurato nel piano  $X$ - $Y$ , compreso tra il piano  $Z$ - $Y$  ed il pallino luminoso generato sul sensore. Dalla lettura del pallino luminoso focalizzato sul sensore è quindi possibile ricavare la posizione del Sole, rispetto al particolare sistema  $XYZ$  della tile in esame. Come risulta semplice comprendere, questo tipo di sensore è usufruibile soltanto se la tile in analisi è esposta alla luce solare. In caso contrario non è possibile evincere alcuna informazione mediante questo tipo di sensore.

I due angoli letti dal sensore di posizione del Sole sono espressi in unità di 0.0001 rad e sono restituiti come due interi di 16 bit con segno.

## 6.2.2 Il sensore di campo magnetico

In questo paragrafo si descrive il principio di funzionamento del sensore di campo magnetico presente sulla Power Management tile del satellite AraMiS.

Il sensore di campo magnetico, detto anche *magnetometro*, rappresenta il blocco indirizzato all'acquisizione ed alla misurazione del campo magnetico biassiale. Il blocco in questione viene montato sulla Power Management tile e si occupa di fornire le componenti misurate del vettore campo magnetico terrestre, rispetto agli assi  $X$  e  $Y$  della tile in questione.

Il magnetometro è composto di un sensore magnetoresistivo ed un opportuno circuito di condizionamento del segnale. Il principio di funzionamento si basa sull'effetto dettato dalla magnetoresistività. Quest'ultima è una proprietà di alcuni materiali e consiste nella variazione della propria resistenza elettrica in funzione del campo magnetico applicato esternamente. Il materiale avente questa particolare caratteristica raffigura quindi il sensore utilizzato nel magnetometro. Questo tipo di sensore, grazie alle sue ridotte dimensioni ed ai suoi bassi costi, si presta molto bene all'integrazione all'interno di opportuni circuiti integrati.

Il sensore in esame si basa sulla *forza di Lorentz*, indicata con il simbolo  $\vec{F}_l$ . La forza di Lorentz rappresenta la forza che agisce su un oggetto elettricamente carico che si muove all'interno di un campo magnetico. Si consideri una carica puntiforme  $q$  in moto con velocità  $\vec{v}$  in una regione caratterizzata dalla presenza di un campo magnetico  $\vec{B}$ . Sulla carica  $q$  allora agisce una forza di Lorentz data dalla relazione 6.1 seguente:

$$\vec{F}_l = q \cdot \vec{v} \times \vec{B} \quad (6.1)$$

Questa forza devia il moto dei portatori di carica, allungandone il percorso, e quindi aumentando la resistenza elettrica vista dall'esterno. Si riporta nella figura 6.6 seguente l'idea di base sul principio di funzionamento del sensore di campo magnetico:

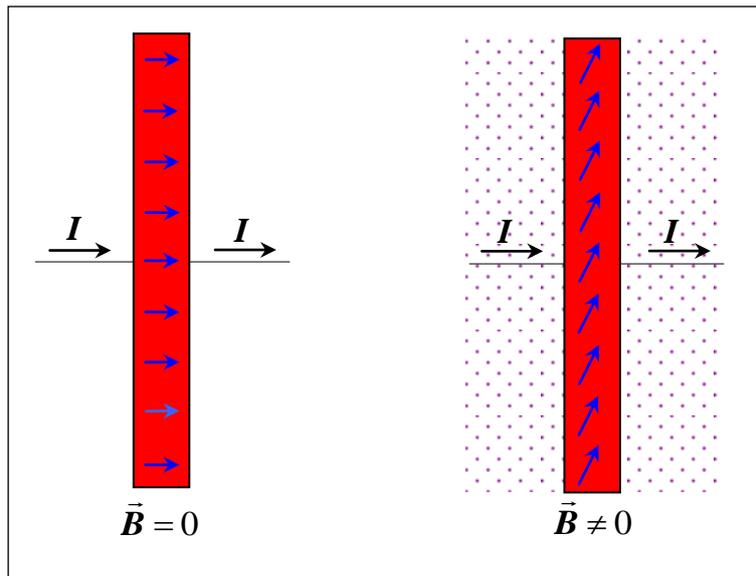


Figura 6.6 Magnetoresistenza a film sottile.

Nella figura precedente è schematizzata una resistenza realizzata con tecnologia a film sottile ed è stata indicata con  $I$  la corrente che attraversa la regione interessata. Nella parte a sinistra della figura è possibile osservare il moto dei portatori di carica (rappresentato con le frecce blu) nel caso in cui esternamente non è applicato alcun campo magnetico. Nella figura a destra, invece, è possibile osservare la deviazione del moto dei portatori in seguito all'applicazione di un campo magnetico  $\vec{B}$  esterno. Il campo magnetico è stato rappresentato con dei punti in quanto risulta essere uscente dal piano della figura. Come risulta semplice osservare, la deviazione del moto dei portatori causa un allungamento del percorso degli stessi, ovvero un aumento della resistenza elettrica dello strato costituente la resistenza. Risulta semplice notare che il massimo valore di resistenza è raggiunto nella situazione descritta nella figura precedente a destra, ovvero quando la direzione del moto dei portatori risulta essere ortogonale al campo magnetico. Nel caso in cui il campo magnetico ed il moto dei portatori sono paralleli, invece, la forza di Lorentz si annulla (perché si annulla il prodotto vettoriale) e ci si riporta nella situazione descritta nella figura precedente a sinistra.

La magnetoresistenza del sensore di campo magnetico presente sul satellite AraMiS è costituita da un film sottile di una lega di Nickel-Ferro (detta *Permalloy*) depositato su un wafer di Silicio e modellato come una striscia resistiva. Il sensore è montato all'interno di un apposito circuito di condizionamento del segnale rappresentato da un ponte di Wheatstone. L'uscita in tensione del ponte rappresenta quindi la misura del campo magnetico sull'asse su cui è montata la magnetoresistenza. Infatti, in presenza di un campo magnetico esterno si ottiene un cambiamento nella resistenza del ponte che provoca una corrispondente variazione della tensione di uscita. Dal punto di vista esterno il dispositivo costituente il sensore di campo magnetico necessita soltanto di una tensione di alimentazione del ponte. Il magnetometro, una volta alimentato il ponte presente al suo interno, inizia la misurazione del campo magnetico esterno.

Le componenti del vettore campo magnetico terrestre misurate dal magnetometro, rispetto agli assi X e Y del sistema di riferimento XYZ della tile, sono espresse in unità di 10 nT e sono restituite come due interi di 16 bit con segno.

### 6.2.3 Il giroscopio

In questo paragrafo si descrive il principio di funzionamento del sensore giroscopico presente sulla Power Management tile del satellite AraMiS.

Il sensore giroscopico, detto anche *giroscopio*, rappresenta il blocco indirizzato all'acquisizione ed alla misurazione della velocità angolare assiale. Il blocco in questione viene montato sulla Power Management tile e si occupa di fornire la misura della velocità angolare attorno all'asse Z della stessa, indicata con il simbolo  $\omega_z$ . Il giroscopio è composto di un sensore ed un opportuno circuito di condizionamento del segnale. Nella figura seguente si rappresenta la velocità angolare che il sensore in esame si preoccupa di leggere:

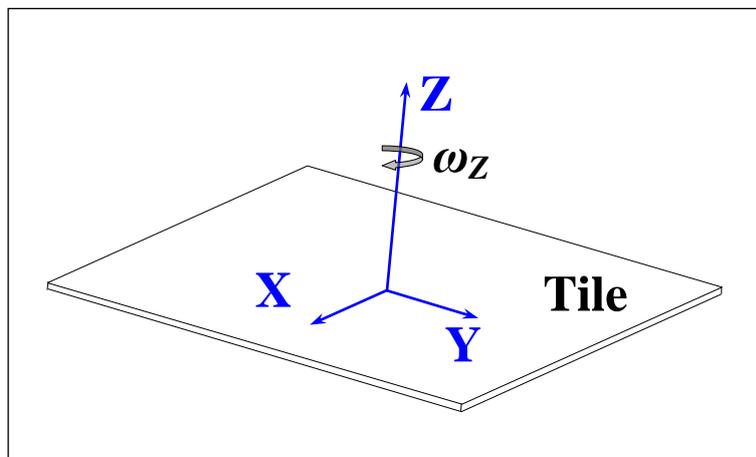
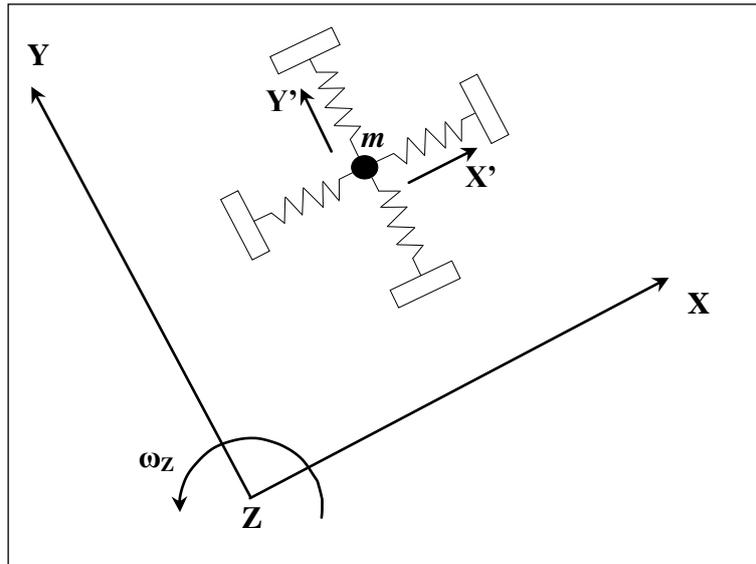


Figura 6.7 Lettura della velocità angolare attorno all'asse Z della tile.

Il sensore presente all'interno del giroscopio in esame si basa sulla *forza di Coriolis*, indicata con il simbolo  $\vec{F}_C$ . La forza di Coriolis è una forza di tipo apparente che risulta essere applicata ad un corpo, quando si osserva il moto da un sistema di riferimento che sia in moto circolare rispetto ad un sistema di riferimento inerziale. Si consideri un corpo dotato di una massa  $m$  che si muove con velocità  $\vec{v}$  rispetto ad un sistema di riferimento non inerziale rotante. Indicando con  $\vec{\omega}$  la velocità angolare del sistema di riferimento non inerziale, rispetto al sistema di riferimento inerziale, la forza di Coriolis è espressa nel modo seguente:

$$\vec{F}_C = -2 \cdot m \cdot \vec{\omega} \times \vec{v} \quad (6.2)$$

Dalla relazione precedente risulta evidente che la forza di Coriolis raggiunge il suo massimo valore nel caso in cui  $\vec{v}$  e  $\vec{\omega}$  risultano essere ortogonali. L'idea di base consiste nell'utilizzare una massa di prova, indicata con la lettera  $m$ , posta in oscillazione rispetto ad una direzione prestabilita. Si riporta nella figura 6.8 seguente il principio su cui è basato il giroscopio:



**Figura 6.8** Principio di funzionamento del giroscopio a vibrazione.

La massa di prova è posta in oscillazione, con ampiezza costante, lungo l'asse X (in figura è stata rappresentata l'oscillazione mediante delle molle). Quest'oscillazione rappresenta il moto primario. Se il giroscopio è posto in rotazione, la massa  $m$  risente della forza di Coriolis. Questa forza induce una vibrazione della massa lungo l'asse Y. Quest'oscillazione aggiuntiva rappresenta il moto secondario. La misura della sua ampiezza consente di stimare la velocità angolare del sistema di riferimento XYZ della tile attorno all'asse Z.

La massa di prova è costruita mediante una struttura a pettine in polisilicio. Le dita di questa struttura fungono da piatti mobili di un condensatore equivalente. Le variazioni di oscillazione del pettine causano quindi una variazione della capacità del condensatore equivalente. Il sensore giroscopico così costruito necessita di una circuiteria elettronica in grado di rilevare le variazioni di capacità.

Il valore di velocità angolare attorno all'asse Z della tile misurato dal giroscopio è espresso in unità di  $0.0001 \text{ rad / s}$  ( $100 \mu\text{rad / s}$ ) ed è restituito come un intero di 16 bit con segno.

# Capitolo 7

## 7 I modelli

In questo capitolo si descrivono i due modelli utilizzati durante la soluzione del problema riguardante la determinazione dell'assetto del satellite. I modelli in questione consistono in due algoritmi indirizzati al calcolo della posizione del Sole e del vettore campo magnetico terrestre, rispetto al sistema di riferimento geocentrico inerziale IJK. Queste due grandezze rappresentano le direzioni di riferimento principali su cui si basa l'algoritmo di determinazione dell'orientamento del satellite implementato.

### 7.1 Il modello della posizione del Sole rispetto al sistema geocentrico inerziale IJK

In questo paragrafo si presenta un metodo di calcolo della posizione del Sole rispetto al sistema di riferimento geocentrico inerziale IJK (si veda [8]). Nello sviluppo dell'algoritmo seguente si considera il moto apparente del Sole attorno alla Terra, sebbene sia il pianeta a compiere la rivoluzione attorno al Sole. Pertanto si considerano noti gli elementi orbitali del Sole.

#### 7.1.1 L'algoritmo di calcolo della posizione del Sole

La conoscenza di posizioni note (i riferimenti) nello spazio tridimensionale ricopre un ruolo importante durante la soluzione del problema della determinazione dell'orientamento del satellite. L'algoritmo di calcolo presentato risulta valido per qualsiasi data nel corso del XX e XXI secolo e ha un'accuratezza pari ad una frazione di minuto d'arco (1 minuto d'arco = 1/60 gradi). In fondo al capitolo si riporta un test dell'algoritmo per una data assegnata ed il confronto con le effemeridi del Sole tabulate dall'Almanacco astronomico 2009 (documento annuale che si occupa della tabulazione delle effemeridi dei principali corpi celesti presenti nel sistema solare e dei principali eventi astronomici sfruttando le teorie più evolute disponibili). L'algoritmo è stato implementato in ambiente Matlab e, successivamente, è stato tradotto in linguaggio di programmazione C.

Preliminarmente è opportuno definire una scala del tempo e fissare una data di riferimento, il 1° Gennaio 2000, alle ore 00:00 (UT). A tal scopo si calcola il parametro  $d$ , detto *day number*, che rappresenta il numero di giorni trascorsi dalla data di riferimento. Si noti che le ore sono espresse in UT, *universal time*, che rappresenta il tempo solare medio in prossimità del Royal Observatory (Greenwich). La relazione utilizzata è riportata nella formula 7.1:

$$d = 367 \cdot Y - (7 \cdot (Y + ((M+9) / 12))) / 4 + (275 \cdot M) / 9 + D - 730530 \quad (7.1)$$

Dove  $Y$  rappresenta l'anno (quattro cifre),  $M$  il mese (1-12) e  $D$  il particolare giorno del mese. Nell'utilizzare questa formula è necessario assicurarsi che tutte le divisioni avvengano tra numeri interi, al fine di ottenere come risultato un numero intero. In linguaggio C è semplice ottenere questo risultato, infatti, è sufficiente dichiarare ogni variabile di tipo intero (int). Per tener conto anche dell'ora del particolare giorno in esame si applica infine l'evidente relazione 7.2:

$$d = d + UT / 24.0 \quad (7.2)$$

La relazione precedente, a differenza della 7.1, è una divisione in floating-point dal momento che si vuole considerare anche la parte frazionale.

I parametri orbitali del Sole utilizzati sono riportati di seguito (si noti che sono espressi in funzione del parametro  $d$ ):

$$\begin{aligned} \Omega &= 0.0 \\ i &= 0.0 \\ \omega &= 282.9404 + 4.70935 \cdot 10^{-5} \cdot d \\ a &= 1.000000 \text{ (A.U.)} \\ e &= 0.016709 - 1.151 \cdot 10^{-9} \cdot d \\ M &= 356.0470 + 0.9856002585 \cdot d \end{aligned}$$

Dove  $\Omega$  rappresenta la longitudine del nodo ascendente,  $i$  l'inclinazione dell'orbita,  $\omega$  l'argomento del *perielio* (il perielio rappresenta il punto dell'orbita terrestre più vicino al Sole),  $a$  il semiasse maggiore,  $e$  l'eccentricità e  $M$  l'anomalia media.

La Terra, nella sua rivoluzione attorno al Sole, si muove sul piano dell'eclittica. Risulta quindi conveniente scegliere questo piano come piano fondamentale di questa geometria spaziale e, al limite, effettuare una trasformazione di coordinate in un secondo momento. Il sistema di riferimento fissato, indicato con i simboli  $xyz$ , è quindi determinato da: origine coincidente con il centro della Terra, asse  $x$  ed asse  $y$  situati sul piano dell'eclittica, con asse  $x$  in direzione del punto vernale ed asse  $y$  ruotato di  $90^\circ$  in senso antiorario rispetto a  $x$  ed asse  $z$  ortogonale al piano fondamentale. L'inclinazione dell'orbita, sotto queste condizioni, risulta nulla e la longitudine del nodo ascendente risulterebbe, di conseguenza, indeterminata ed è fissata, per comodità, pari a zero.

Si noti che il semiasse maggiore è espresso in *unità astronomiche* (A.U., astronomical unit). Un A.U. è pari a  $149.6 \cdot 10^6$  Km e rappresenta la distanza media del segmento congiungente il Sole e la Terra.

Una volta acquisiti i parametri orbitali è possibile impostare il calcolo delle grandezze fondamentali che permettono di risalire alla posizione del Sole nel sistema di riferimento prefissato.

Si calcola dapprima l'anomalia eccentrica,  $E$ , utilizzando la relazione approssimata 7.3 (le due anomalie,  $E$  e  $M$ , sono espresse in gradi):

$$E = M + e \cdot (180 / \pi) \cdot \sin(M) \cdot (1.0 + e \cdot \cos(M)) \quad (7.3)$$

Come già introdotto nel capitolo 4, quest'espressione non è esatta, dal momento che non esiste una forma chiusa per ricavare  $E$  dall'equazione di Keplero. La relazione 7.3, essendo l'eccentricità dell'orbita molto piccola, risulta tuttavia sufficientemente accurata per gli obiettivi prefissati.

Conoscendo l'anomalia eccentrica è possibile sfruttare la trigonometria e calcolare le componenti dell'anomalia vera proiettate sul piano dell'eclittica. Le relazioni utilizzate sono riportate di seguito:

$$x_v = r \cdot \cos(v) = \cos(E) - e \quad (7.4)$$

$$y_v = r \cdot \sin(v) = \sqrt{1.0 - e^2} \cdot \sin(E) \quad (7.5)$$

Da queste due grandezze è quindi possibile ricavare la distanza del Sole,  $r$ , e la sua anomalia vera,  $v$ :

$$v = \text{atan2}(y_v, x_v) \quad (7.6)$$

$$r = \sqrt{x_v^2 + y_v^2} \quad (7.7)$$

Si noti che nella formula 7.6 è stata introdotta la funzione  $\text{atan2}()$ . Questa funzione, già descritta nel capitolo 4, converte una coppia di coordinate  $(x, y)$  nell'angolo corretto in tutti e quattro i quadranti. Inoltre, la funzione  $\text{atan2}()$  è disponibile nelle librerie matematiche standard di Matlab e del linguaggio C.

A questo punto è possibile calcolare la *longitudine vera*, indicata dal simbolo  $lon_{\text{sun}}$ , definita dalla somma dell'anomalia vera e dell'argomento del perielio, come mostrato nell'espressione 7.8:

$$lon_{\text{sun}} = v + \omega \quad (7.8)$$

Il parametro appena trovato è possibile sfruttarlo per calcolare le coordinate eclittiche rettangolari, indicate dai simboli  $x_s$  e  $y_s$ , mediante le relazioni seguenti:

$$x_s = r \cdot \cos(lon_{\text{sun}}) \quad (7.9)$$

$$y_s = r \cdot \sin(lon_{\text{sun}}) \quad (7.10)$$

La coordinata in  $z$ ,  $z_s$ , non è stata indicata, dal momento che il Sole è sempre situato sul piano dell'eclittica e di conseguenza tale coordinata risulta essere sempre nulla.

Si introduce ora il parametro *obliquità dell'eclittica*, indicato dal simbolo  $ecl$ . Questo parametro indica l'inclinazione del piano equatoriale rispetto al piano dell'eclittica, in altre parole, indica di quanto è inclinato l'asse di rotazione terrestre rispetto alla normale al piano dell'eclittica e ha la forma seguente:

$$ecl = 23.4393 - 3.563 \cdot 10^{-7} \cdot d \quad (7.11)$$

E' dunque possibile, sfruttando il parametro appena calcolato, ricavare le coordinate del Sole geocentriche equatoriali come trasformazione di coordinate dal piano dell'eclittica al piano equatoriale, indicate con i simboli  $x_e$ ,  $y_e$  e  $z_e$ . Si riporta nelle relazioni seguenti le operazioni necessarie:

$$x_e = x_s \quad (7.12)$$

$$y_e = y_s \cdot \cos(ecl) + z_s \cdot \sin(ecl) = y_s \cdot \cos(ecl) \quad (7.13)$$

$$z_e = y_s \cdot \sin(ecl) + z_s \cdot \cos(ecl) = y_s \cdot \sin(ecl) \quad (7.14)$$

Si noti che la trasformazione di coordinate appena effettuata non è nient'altro che una rotazione attorno all'asse  $x_s$  di un angolo pari al valore dell'obliquità dell'eclittica. Infatti, la coordinata in  $x$  è rimasta la medesima e, dopo la trasformazione, continua ad indicare il punto vernale, direzione di riferimento nello spazio e comune al piano equatoriale ed al piano dell'eclittica. Il nuovo sistema di riferimento ottenuto utilizza quindi il piano equatoriale come piano fondamentale e risulta pertanto essere il sistema di riferimento geocentrico inerziale IJK (si veda capitolo 2).

Infine è possibile ricavare l'ascensione retta e la declinazione del Sole, indicate con i simboli  $RA$  e  $Dec$ , rispettivamente, dalle espressioni 7.15 e 7.16:

$$RA = \text{atan2}(y_e, x_e) \quad (7.15)$$

$$Dec = \text{atan2}(z_e, \sqrt{x_e^2 + y_e^2}) \quad (7.16)$$

Queste grandezze risultano utili per fare il confronto con i valori tabulati da opportuni documenti astronomici.

## 7.1.2 Verifica dell'algoritmo di calcolo implementato

Nella tabella seguente si riporta una prova dell'algoritmo implementato. E' stata presa una data di test, il 14 agosto 2009 alle ore 00:00 (UT), sono state calcolate le coordinate  $RA$  e  $Dec$  del Sole per tale riferimento temporale e sono stati messi a confronto i risultati ottenuti con quelli forniti dall'Almanacco astronomico 2009:

14/08/2009 00:00	$RA$	$Dec$
Almanacco astronomico 2009	9h; 34min; 56.6s	14°; 23'; 2.5''
Algoritmo sviluppato	9h; 34min; 56.76s	14°; 22'; 57.1''

Si noti che i valori di  $RA$  tabulati nell'Almanacco sono espressi in ore, minuti e secondi. Dalla formula 7.15, invece, i valori sono espressi in gradi. E' necessario quindi trasformare tale grandezza in ore dividendo semplicemente per 15, al fine di poter confrontare i valori. La divisione per 15 è giustificata dal fatto che una rotazione completa della Terra si conclude in 24 ore, questo equivale ad affermare che il pianeta impiega 1 ora per spostarsi di 15 gradi.

I valori ottenuti mostrano evidentemente l'efficacia dell'algoritmo di calcolo sviluppato. Indicando con  $\Delta_{RA}$  il modulo dell'errore di  $RA$  ( $\Delta_{RA} = |RA_{\text{Almanacco}} - RA_{\text{algoritmo}}|$ ) e  $\Delta_{Dec}$  il modulo dell'errore di  $Dec$  ( $\Delta_{Dec} = |Dec_{\text{Almanacco}} - Dec_{\text{algoritmo}}|$ ) si ottiene:

TEST	
$\Delta_{RA}$	2.4''
$\Delta_{Dec}$	5.4''

L'algoritmo sviluppato risulta essere versatile e può essere utilizzato per calcolare la posizione di qualsiasi corpo celeste, noti i suoi parametri orbitali ed, eventualmente, le principali perturbazioni che influiscono sugli stessi. E' stato così possibile implementare un'applicazione aggiuntiva che permette di rappresentare le orbite dei principali pianeti del sistema solare: Mercurio, Venere, Marte, Giove, Saturno, Urano e Nettuno. In questo caso il conto è stato ripetuto per un vettore temporale formato da molti punti, al fine di rappresentare l'intera orbita di tutti i pianeti elencati.

Si riporta in figura 7.1 una simulazione 3D, eseguita in ambiente Matlab, dell'algoritmo che implementa questa applicazione aggiuntiva (nella figura sono visibili le coordinate equatoriali delle orbite elencate):

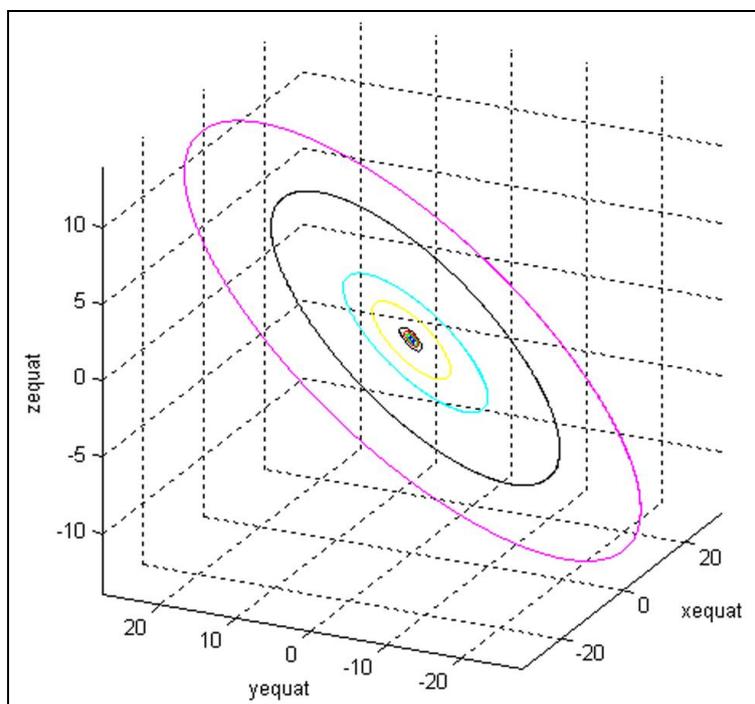


Figura 7.1 Orbite dei pianeti del sistema solare.

## 7.2 Il modello del campo magnetico terrestre

In questo paragrafo si presenta il modello del campo magnetico terrestre, una grandezza vettoriale che risulta molto utile durante la soluzione del problema della determinazione dell'orientamento del satellite. Conoscendo la posizione del satellite nel sistema di riferimento geocentrico inerziale IJK (si veda capitolo 4) è possibile calcolare, servendosi di opportuni modelli, il vettore campo magnetico, indicato con il simbolo  $\vec{\varphi}$ , in quel punto particolare. Questo vettore rappresenta il modello cui riferirsi durante il confronto con i vettori misurati dai magnetometri presenti a bordo del satellite.

### 7.2.1 Il campo magnetico terrestre

La Terra è possibile modellarla come un enorme magnete di forma sferoidale, completamente circondato da un campo magnetico, detto campo *geomagnetico*. Il campo magnetico lontano dalla Terra è assimilabile al campo generato da un dipolo magnetico centrato nel centro della Terra ed

avente asse non allineato con l'asse di rotazione terrestre. Il campo geomagnetico varia sia in funzione del tempo, sia in funzione della posizione. L'angolo compreso tra l'asse di rotazione terrestre e l'asse del dipolo è approssimativamente pari a  $10^\circ$ . Questo equivale ad affermare che i poli Nord e Sud geografici della Terra non coincidono con i rispettivi poli magnetici. Si riportano in figura 7.2 le linee di forza del campo magnetico terrestre, al fine di evidenziare lo spostamento angolare compreso tra l'asse di rotazione della Terra e l'asse del dipolo equivalente:

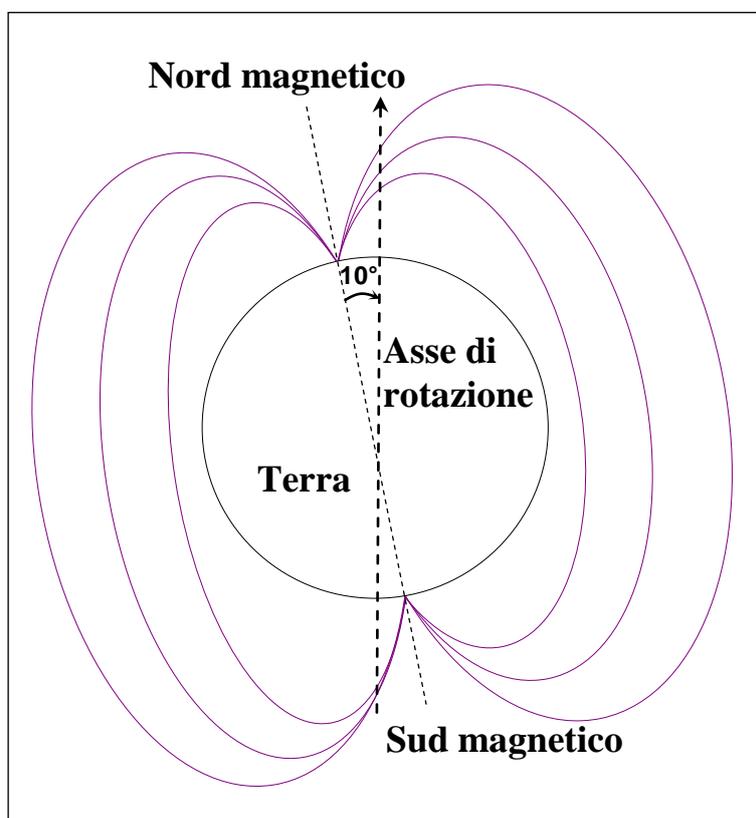


Figura 7.2 Campo magnetico terrestre.

Per definizione, un polo magnetico è un punto in cui il campo magnetico generato da un dipolo magnetico, posto nel nucleo esterno della Terra, ha una direzione coincidente con l'asse del dipolo equivalente, cioè verticale. Si noti che la nomenclatura che si occupa di definire i poli Nord e Sud magnetici è soltanto una convenzione. Le linee di forza del campo geomagnetico, infatti, risultano entranti nell'emisfero boreale (Nord terrestre) e uscenti dall'emisfero australe (Sud terrestre). Pertanto il polo Nord magnetico è quello posto in prossimità del Sud geografico ed il polo Sud magnetico è quello posto in prossimità del Nord geografico. Di fatto l'ago Nord della bussola non potrebbe indicare un polo di uguale polarità, ma soltanto uno di polarità opposta. E' stato quindi fissato, per comodità, che il polo magnetico Sud della Terra sia chiamato polo magnetico Nord, al fine di avere un'associazione con il rispettivo polo Nord geografico. In modo analogo è stato chiamato polo Sud magnetico quello situato nell'emisfero australe, in associazione con il polo Sud geografico.

Il campo magnetico terrestre è caratterizzato, per qualsiasi punto ed in qualsiasi istante di tempo, da una direzione e da un'intensità, entrambe misurabili. Il campo risulta essere di intensità maggiore ai poli rispetto all'equatore, dove vale soltanto la metà del valore raggiunto ai poli. Allontanandosi dalla Terra l'intensità del campo magnetico diminuisce come il cubo della distanza.

Al fine di conoscere il campo magnetico terrestre,  $\vec{\phi}$ , in un punto qualsiasi attorno al pianeta, è necessario misurare sia la direzione, sia l'intensità del campo magnetico stesso.

## 7.2.2 I parametri utilizzati nella descrizione del campo magnetico terrestre

L'International Geomagnetic Reference Field (IGRF) ed il World Magnetic Field (WMM) risultano essere i due principali modelli di campo magnetico terrestre cui riferirsi (si veda [11]). Per entrambi i modelli citati il campo geomagnetico è descritto mediante sette parametri, come mostrato in figura 7.3 (si noti che sono stati rappresentati i vari parametri mantenendo la stessa nomenclatura fornita dai modelli sopra citati, pertanto, nella figura, il campo magnetico  $\vec{\varphi}$  è rappresentato dal vettore  $F$ ):

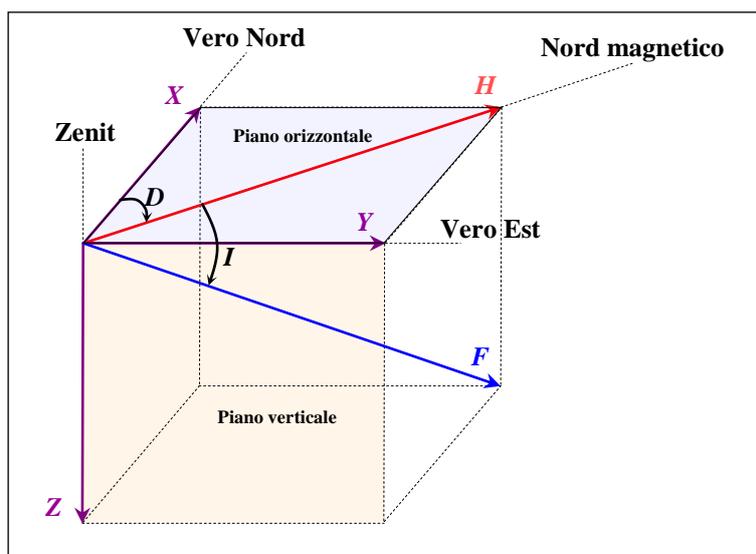


Figura 7.3 Gli elementi magnetici.

Nella figura precedente sono stati riportati i seguenti parametri:

- *Declinazione magnetica (D).*
- *Inclinazione magnetica (I).*
- *Intensità orizzontale del campo (H).*
- *Intensità totale del campo (F).*
- *Componente verso il vero Nord di H (X).*
- *Componente verso il vero Est di H (Y).*
- *Componente verticale (verso il Nadir) di F (Z).*

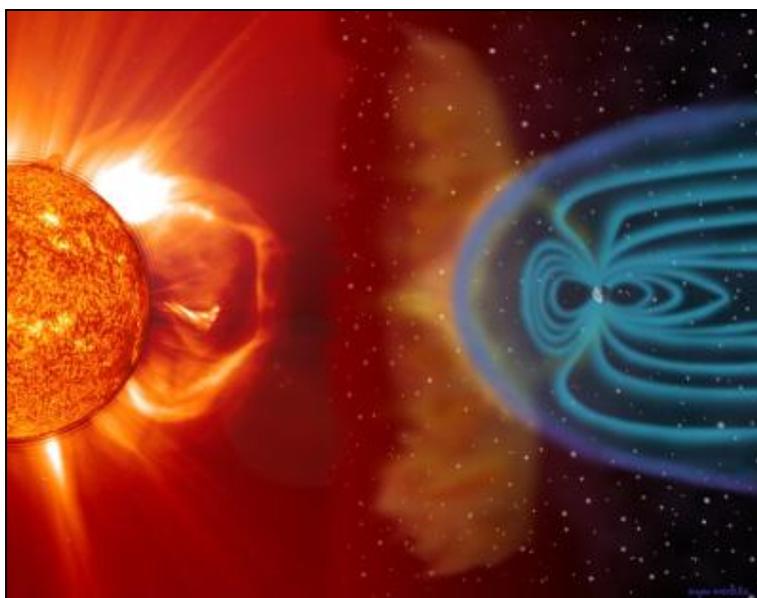
I parametri che descrivono la direzione del campo magnetico terrestre risultano pertanto essere la declinazione magnetica ( $D$ ) e l'inclinazione magnetica ( $I$ ). Questi parametri vengono entrambi misurati in gradi. Si fissa come positivo lo spostamento angolare che porta  $D$  a coincidere con la direzione indicata del vero Est e lo spostamento angolare che porta  $I$  a coincidere con la direzione indicata dal Nadir.

L'intensità del campo totale, invece, è descritta mediante la componente del campo orizzontale ( $H$ ), la componente verticale ( $Z$ ), e le componenti verso il vero Nord ( $X$ ) e verso il vero Est ( $Y$ ) dell'intensità orizzontale del campo. Tutte queste componenti elencate sono espresse in Gauss (G),

ma sono generalmente riportate in nano Tesla ( $1\text{nT} = 10^{-5}\text{G}$ ). L'intensità del campo magnetico terrestre è approssimativamente compresa tra 25,000 e 65,000 nT (0.25G - 0.65G).

La declinazione magnetica rappresenta l'angolo compreso tra la direzione fissata dal vero Nord e la direzione indicata dal polo Nord magnetico. L'inclinazione magnetica, invece, rappresenta l'angolo compreso tra il piano orizzontale ed il vettore campo magnetico totale ( $F$ ).

L'interazione tra il campo magnetico terrestre ed il *vento solare* (flusso di particelle elettricamente cariche emesso ininterrottamente dalla corona solare) genera una regione di spazio che avvolge totalmente la Terra e prende il nome di *magnetosfera*. La magnetosfera ha una forma simile a quella di una stella cometa. La parte esposta al Sole risulta più compressa e si estende per circa una decina di raggi terrestri. La parte non esposta alla pressione solare, invece, è decisamente più allungata e si estende per circa un centinaio di raggi terrestri. La magnetosfera, di fatto, si comporta come uno scudo protettivo dai raggi cosmici provenienti dallo spazio. Si riporta in figura 7.4 una rappresentazione della magnetosfera (figura tratta da: <http://www.historiasdaterra.com/wp-content/uploads/2009/04/magnestofera.png>):



**Figura 7.4** La magnetosfera.

Il campo magnetico terrestre, misurato in un punto qualsiasi attorno al pianeta, è dato dalla combinazione di diversi campi magnetici generati da varie sorgenti. Questi campi generati si sovrappongono ed interagiscono tra loro. La stragrande maggioranza del campo misurato (più del 90%) è generato all'interno del pianeta, nel nucleo esterno della Terra. Questa parte del campo magnetico è spesso indicata con il nome di *campo principale*. Il campo principale varia lentamente nel tempo ed è possibile descriverlo mediante modelli matematici come l'IGRF ed il WMM, citati in precedenza. L'accuratezza di questi modelli è di circa 30 minuti d'arco per la declinazione e l'inclinazione magnetica, ed approssimativamente di 200 nT per gli elementi rappresentanti l'intensità del campo.

### 7.2.3 I principi su cui si basano i modelli del campo magnetico terrestre

Si riporta in questo paragrafo l'idea di base su cui sono costruiti i modelli di campo magnetico terrestre. Sebbene l'esatta natura del campo geomagnetico,  $\vec{\varphi}$ , non sia nota, il fatto che esso sia generato internamente al pianeta suggerisce una sua descrizione attraverso una soluzione di un problema con valori al contorno. La mancanza di correnti elettriche implica che, nella parte esterna della Terra, il campo magnetico  $\vec{\varphi}$ , debba avere un rotore nullo, ovvero deve valere la relazione seguente:

$$\nabla \times \vec{\varphi} = 0 \quad (7.17)$$

La relazione precedente equivale ad affermare che il campo magnetico terrestre può essere modellato in modo preciso utilizzando il gradiente di un potenziale scalare  $V$ , nel modo seguente:

$$\vec{\varphi} = -\nabla V \quad (7.18)$$

Inoltre, l'assenza di monopoli magnetici implica che la divergenza del campo magnetico risulti nulla, ovvero:

$$\nabla \cdot \vec{\varphi} = 0 \quad (7.19)$$

Sostituendo, infine, la relazione 7.18 nella relazione 7.19 si ottiene l'equazione di Laplace:

$$\nabla^2 V = 0 \quad (7.20)$$

L'espressione precedente, a causa della natura sferica della condizione al contorno dettata dalla superficie della Terra, possiede una soluzione convenientemente rappresentabile mediante una serie di funzioni armoniche sferiche, come mostrato nella relazione seguente:

$$V(r, colat, lon) = R \sum_{n=1}^k \left( \frac{R}{r} \right)^{n+1} \sum_{m=0}^n (g_n^m \cos(m \cdot lon) + h_n^m \sin(m \cdot lon)) P_n^m(colat) \quad (7.21)$$

Dove  $R$  rappresenta il raggio terrestre relativo all'equatore,  $g_n^m$  e  $h_n^m$  sono dei coefficienti (tabulati dall'IGRF e dal WMM ed aggiornati con cadenza quinquennale), detti *Gaussiani*,  $r$ ,  $colat$  e  $lon$  rappresentano le coordinate sferiche della posizione del satellite (distanza dal centro della Terra, colatitudine e longitudine misurata in direzione Est da Greenwich, rispettivamente) e  $P_n^m$  sono i polinomi normalizzati secondo Schmidt, nel modo seguente:

$$\int_0^\pi [P_n^m(colat)]^2 \sin(colat) d(colat) = \frac{2(2 - \delta_m^0)}{2n + 1} \quad (7.22)$$

Dove  $\delta_m^0$  rappresenta la delta di Kronecker ( $\delta_i^j = 1$  se  $i = j$ ,  $\delta_i^j = 0$  se  $i \neq j$ ).

Le componenti di  $\vec{\varphi}$  si ricavano quindi dalle relazioni seguenti:

$$\begin{cases} \varphi_r = \frac{-\partial V}{\partial r} \\ \varphi_{colat} = \frac{-1}{r} \frac{\partial V}{\partial(colat)} \\ \varphi_{lon} = \frac{-1}{r \sin(colat)} \frac{\partial V}{\partial(lon)} \end{cases} \quad (7.23)$$

Dove  $\varphi_r$  rappresenta la componente radiale di  $\vec{\varphi}$ , positiva uscente,  $\varphi_{colat}$  rappresenta la componente di coelevazione di  $\vec{\varphi}$ , positiva se diretta verso Sud, e  $\varphi_{lon}$  la componente azimutale di  $\vec{\varphi}$ , positiva verso Est.

I modelli IGRF e WMM si occupano di risolvere questo sistema di equazioni e forniscono le componenti di  $\vec{\varphi}$  in funzione di una certa data ed una certa posizione attorno alla Terra. Si noti però che le componenti fornite da questi modelli, ovvero le componenti X, Y, Z, rappresentate in precedenza, sono definite in modo differente. In particolare esse sono legate alle componenti elencate nel sistema 7.23 mediante le relazioni seguenti:

$$\begin{cases} X = -\varphi_{colat} \cos(\varepsilon) - \varphi_r \sin(\varepsilon) \\ Y = \varphi_{lon} \\ Z = \varphi_{colat} \sin(\varepsilon) - \varphi_r \cos(\varepsilon) \end{cases} \quad (7.24)$$

Dove  $\varepsilon \equiv lat_g - Dec$ , in cui  $lat_g$  rappresenta la latitudine geodetica del satellite (ovvero la latitudine del satellite calcolata prendendo in considerazione la non sfericità della Terra) e  $Dec$  rappresenta la declinazione del satellite ( $Dec = \pi/2 - colat$ ). Il valore di  $\varepsilon$  risulta essere sempre inferiore a  $0.2^\circ$ . Ovviamente, se si trascura la non sfericità della Terra, latitudine e declinazione coincidono,  $\varepsilon$  si annulla ed il sistema 7.24 si riduce al seguente:

$$\begin{cases} X = -\varphi_{colat} \\ Y = \varphi_{lon} \\ Z = -\varphi_r \end{cases} \quad (7.25)$$

Le componenti di  $\vec{\varphi}$  specificate rispetto al sistema di riferimento geocentrico inerziale IJK sono, infine, ottenute mediante la seguente trasformazione di coordinate:

$$\begin{cases} \varphi_i = (\varphi_r \cos(Dec) + \varphi_{colat} \sin(Dec)) \cos(RA) - \varphi_{lon} \sin(RA) \\ \varphi_j = (\varphi_r \cos(Dec) + \varphi_{colat} \sin(Dec)) \sin(RA) + \varphi_{lon} \cos(RA) \\ \varphi_k = (\varphi_r \sin(Dec) - \varphi_{colat} \cos(Dec)) \end{cases} \quad (7.26)$$

Dove  $Dec$  rappresenta la declinazione del satellite e  $RA$  rappresenta l'ascensione retta del satellite, legata alla longitudine  $lon$  dalla relazione seguente:

$$RA = lon + RA_G \quad (7.27)$$

Dove  $RA_G$  rappresenta l'ascensione retta del meridiano di Greenwich, ovvero lo spostamento angolare compreso tra il punto vernale ed il meridiano di Greenwich. Si tengano presente le

relazioni 7.24 prima di effettuare le trasformazioni di coordinate elencate nel sistema di equazioni 7.26.

# Capitolo 8

## 8 La determinazione dell'assetto del satellite

In questo capitolo si descrive la soluzione del problema riguardante la determinazione dell'assetto del satellite da un punto di vista implementativo. I modelli descritti nel capitolo 7 fungono da riferimento dei metodi di calcolo presentati nel seguito della discussione. Gli algoritmi presentati in questo capitolo si basano sulla lettura dei sensori d'assetto presenti sulla Power Management tile del satellite (si veda capitolo 6). Per ogni sensore è descritto un apposito algoritmo utile nella soluzione del problema. In fondo al capitolo, infine, si descrive un algoritmo di determinazione dell'assetto del tutto generico ed indipendente dal tipo di sensori presenti sul satellite.

### 8.1 La determinazione dell'assetto mediante sensori di posizione del Sole

In questo paragrafo si descrive un procedimento utilizzabile nella determinazione dell'assetto del satellite basato esclusivamente sulla lettura delle informazioni fornite dai sensori di posizione del Sole. La Power Management tile del satellite AraMiS, come descritto nel capitolo 6, è provvista di un sensore di posizione del Sole usufruibile nella soluzione del problema della determinazione dell'assetto.

L'algoritmo di determinazione dell'assetto presentato nella discussione seguente si preoccupa di ricavare l'orientamento del sistema di riferimento UVW del satellite rispetto al sistema di riferimento geocentrico inerziale, rappresentato mediante la matrice dei coseni direttori. Questa matrice è indicata con il simbolo  $A$ .

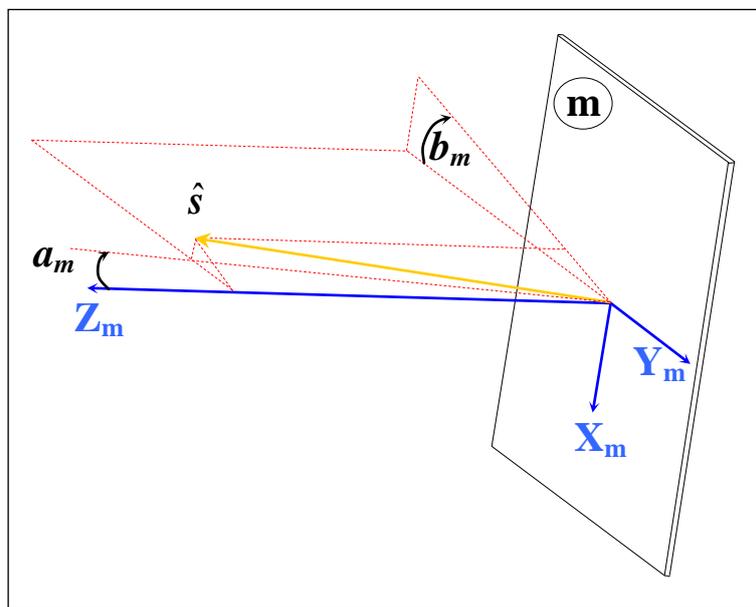
Il sensore di Sole (si veda capitolo 6, paragrafo 2.1) fornisce l'informazione circa la direzione del Sole rispetto all'asse  $Z$  della tile. In particolare sono fornite le coordinate sferiche del Sole rispetto alla direzione definita dal versore  $\hat{z}$  della tile, indicate con  $a$  e  $b$ . L'angolo  $a$ , si ricorda, è calcolato sul piano  $Z$ - $Y$  della tile e rappresenta lo spostamento angolare da compiere per portare l'asse  $Z$  della tile a coincidere con il vettore in  $XYZ$  che ha componente in  $X$  nulla, componente in  $Y$  pari alla componente in  $Y$  del versore indicante la direzione del Sole in  $XYZ$  e componente in  $Z$  pari alla componente in  $Z$  dello stesso. Si fissa come positivo lo spostamento angolare che porta l'asse  $Z$  a coincidere con l'asse  $Y$ . L'angolo appena definito è equivalente al concetto di longitudine. L'angolo  $b$ , invece, è calcolato sul piano  $X$ - $Y$  e rappresenta lo spostamento angolare necessario per portare a coincidere il vettore in  $Z$ - $Y$  creato con l'operazione precedente con l'esatta direzione del Sole in  $XYZ$  e risulta, pertanto, l'equivalente del concetto di latitudine. Si fissa come positivo lo spostamento angolare in direzione oraria.

Nel capitolo 7 è stato presentato un algoritmo di calcolo della posizione del Sole rispetto al sistema di riferimento IJK, in funzione di una certa data ed ora d'interesse. Nella descrizione seguente si considera nota tale posizione (coordinate  $RA$  e  $Dec$  del Sole) e l'algoritmo sviluppato nel capitolo 7 costituisce il modello cui riferirsi. Si noti che la posizione calcolata nel capitolo 7 è fissata come il

modello della posizione del Sole rispetto al sistema di riferimento UVW del satellite, si trascura cioè la parallasse dovuta alla distanza tra il centro della Terra ed il centro del satellite. Quest'approssimazione è giustificata dal fatto che la distanza del Sole, pari ad 1 A.U., è molto grande rispetto al segmento congiungente il centro della Terra con il centro del satellite. Per un satellite in orbita bassa la parallasse risulta essere pari a qualche secondo d'arco, quindi, tranquillamente trascurabile.

Durante il moto del satellite nella sua orbita attorno alla Terra alcune tiles risulteranno esposte al Sole, mentre altre non saranno esibite alla luce solare. Soltanto le tiles illuminate potranno fornire l'informazione sulla posizione del Sole e riusciranno, quindi, ad essere d'aiuto nel determinare l'assetto del satellite.

Si consideri la tile  $m$ , esposta al Sole, con le due informazioni angolari, indicate con i simboli  $a_m$  e  $b_m$ . Queste due informazioni rappresentano la direzione del Sole rispetto all'asse Z della tile  $m$ , come dichiarato in precedenza. Si riporta in figura 8.1 un esempio della geometria in questione:



**Figura 8.1** Coordinate sferiche della direzione del Sole rispetto al sistema di riferimento XYZ.

Nella figura precedente è stato indicato con il simbolo  $\hat{s}$  il versore diretto verso la direzione dettata dalla posizione del Sole.

Dal modello sviluppato nel capitolo 7, invece, è possibile conoscere, nello stesso istante temporale, la posizione del Sole rispetto al sistema di riferimento geocentrico inerziale IJK. Queste due informazioni sono quindi sufficienti per ricavare le coordinate sferiche dell'asse Z della tile  $m$ , indicate con i simboli  $\alpha_m$  e  $\beta_m$ , rispetto al sistema di riferimento geocentrico inerziale IJK. Si riporta nella relazione 8.1 la semplice operazione necessaria:

$$\begin{cases} \alpha_m = RA - a_m \\ \beta_m = Dec - b_m \end{cases} \quad (8.1)$$

Conoscendo le coordinate sferiche della normale alla tile  $m$  rispetto al sistema di riferimento geocentrico inerziale IJK, e quindi il suo orientamento, è possibile ricavare le coordinate rettangolari dello stesso asse, indicate con i simboli  $z_i, z_j, z_k$ , mediante le relazioni seguenti:

$$\begin{cases} z_i = \cos(\alpha_m) \cos(\beta_m) \\ z_j = \sin(\alpha_m) \cos(\beta_m) \\ z_k = \sin(\beta_m) \end{cases} \quad (8.2)$$

Le tre grandezze appena ricavate evidentemente costituiscono le componenti del versore  $\hat{z}_m$  della tile in esame rispetto al sistema di riferimento geocentrico inerziale IJK. Questo vettore appena calcolato è possibile definirlo come la prima riga di una matrice di rotazione, indicata con il simbolo  $A_{ijk}$ . Quest'ultima ha lo scopo di contenere l'orientamento dei versori  $\hat{z}$  delle tiles illuminate dal Sole rispetto al sistema di riferimento geocentrico inerziale IJK.

L'orientamento del sistema di riferimento delle tiles rispetto al sistema di riferimento UVW del satellite (si veda capitolo 6, paragrafo 1) è noto, ed è quindi possibile determinare anche la direzione dell'asse Z della tile  $m$  rispetto al sistema di riferimento UVW del satellite. Infatti, indicando con  $A_m$  la matrice dei coseni direttori della tile  $m$  in UVW, è possibile ricavare l'orientamento incognito dell'asse Z della stessa tile in UVW, indicato con i simboli  $z_u, z_v, z_w$ , nel modo seguente:

$$\begin{cases} z_u = A_m^T(1,3) \\ z_v = A_m^T(2,3) \\ z_w = A_m^T(3,3) \end{cases} \quad (8.3)$$

La relazione 8.3 è stata ricavata osservando che il generico versore  $\hat{z}$  nel sistema di riferimento XYZ della tile ha la forma  $\hat{z} = (0 \ 0 \ 1)$  e, di conseguenza, le sue componenti rispetto al sistema UVW del satellite sono ottenute dalla relazione seguente:

$$z_{uvw} = A_m^T \cdot z_{xyz} \quad (8.4)$$

A questo punto è possibile dichiarare una nuova matrice di rotazione, indicata con il simbolo  $A_{uvw}$ , come fatto nella relazione 8.2 precedente, e definire il vettore  $z_{uvw}$  precedente come la prima riga di tale matrice.

Con le operazioni svolte è stato quindi determinato l'orientamento dell'asse Z della tile  $m$ , sia in IJK, sia in UVW, ed è stato memorizzato il risultato nella prima riga delle matrici  $A_{ijk}$  e  $A_{uvw}$ , rispettivamente. Al fine di specificare in modo completo la matrice d'assetto è necessario determinare l'orientamento dei tre assi del satellite UVW rispetto al sistema IJK. In altre parole è necessario ottenere tre vettori di misura. Nel metodo di calcolo presentato nella discussione seguente si suppone tuttavia di ricavare l'orientamento di due assi soltanto e di ottenere il terzo con l'operazione di prodotto vettoriale. Nel caso di presenza di più informazioni dai sensori di Sole sarà possibile effettuare una verifica dei valori ottenuti.

L'algoritmo di determinazione dell'assetto si occupa quindi di interrogare tutte le tiles del satellite e di ricercare due vettori di misura. Siccome l'architettura del satellite AraMiS prevede una struttura di tipo modulare, ovvero che possa essere configurata nella forma appropriata alla particolare missione (si veda capitolo 6), l'algoritmo di determinazione dell'assetto deve essere in grado di operare qualsiasi sia la configurazione del satellite, indipendentemente dal numero di tiles presenti e dal loro orientamento rispetto al sistema di riferimento UVW del satellite.

Durante la ricerca del secondo vettore di misura si possono presentare i seguenti tre casi:

1. Il secondo vettore di misura è parallelo al primo.
2. Il secondo vettore di misura è ortogonale al primo.
3. Il secondo vettore di misura non è né parallelo, né ortogonale al primo.

Il primo caso si riscontra durante la lettura di due tiles giacenti sulla stessa faccia del satellite. Per esempio, si consideri un satellite a configurazione cubica avente quattro tiles per ogni faccia. Questa configurazione del satellite avrà sicuramente quattro tiles con asse Z parallelo per ogni faccia. In questo caso è possibile prendere in considerazione tutte le informazioni presenti sulla stessa faccia ed effettuare un'operazione di media dei valori ottenuti.

Il secondo caso si riscontra durante la lettura di due tiles ortogonali. In questo caso è possibile scrivere la seconda riga delle matrici  $A_{ijk}$  e  $A_{uvw}$  e ricavare, infine, anche l'ultima con l'operazione di prodotto vettoriale.

Il terzo caso si riscontra durante la lettura di due tiles non parallele e non ortogonali. Questa situazione è possibile riscontrarla nel caso di un satellite a configurazione prismatica. In questo caso si applicano alcuni semplici accorgimenti algebrici. Si indichi con  $v_1$  il primo vettore di misura ricavato e con  $v_2$  il secondo vettore di misura ricavato. Anche se i due vettori non sono ortogonali è possibile applicare l'operazione di prodotto vettoriale ed ottenere un terzo vettore, indicato con il simbolo  $v_3$ , perpendicolare al piano generato da  $v_1$  e  $v_2$ , nel modo seguente:

$$v_3 = v_1 \times v_2 \quad (8.5)$$

Con l'operazione 8.5 si ottiene chiaramente un vettore  $v_3$  ortogonale agli altri due e con modulo non unitario, per le ipotesi fatte. E' quindi opportuno effettuare l'operazione di normalizzazione, mediante la relazione 8.6:

$$v_3 = \frac{v_3}{|v_3|} \quad (8.6)$$

Si noti, infatti, che uno dei vincoli delle matrici di rotazione è proprio quello di avere i moduli dei vettori, componenti le righe delle matrici stesse, unitari.

A questo punto è possibile scartare il vettore  $v_2$  e calcolare un nuovo vettore  $v_2$  con l'operazione vettoriale seguente:

$$v_2 = v_3 \times v_1 \quad (8.7)$$

Così facendo si ha la garanzia di ottenere tre vettori ortogonali a partire da due tiles qualsiasi del satellite. In altre parole, dopo queste operazioni, è stato possibile determinare le matrici  $A_{ijk}$  e  $A_{uvw}$  in modo completo.

Infine è possibile risolvere il problema della determinazione dell'assetto del satellite ricavando la matrice incognita  $A$  mediante l'espressione 8.8 seguente:

$$A = A_{uvw}^T \cdot A_{ijk} \quad (8.8)$$

L'algoritmo presentato in questo paragrafo si occupa della ricerca di due vettori di misura soltanto. Nel seguito del capitolo si riporterà una generalizzazione del concetto, basata sull'acquisizione di un numero variabile di vettori di misura.

Con la configurazione a piccolo cubo si può notare che, in un dato istante, al massimo tre tiles potranno essere illuminate dal Sole, dalle altre tre, quindi, non sarà possibile reperire alcuna informazione dai sensori di Sole. Nella descrizione dell'algoritmo di determinazione d'assetto precedente è stato affermato che una sola informazione proveniente da un sensore di Sole non permette la determinazione completa della matrice  $A$ . Questa affermazione è possibile giustificarla osservando la figura 8.2 seguente:

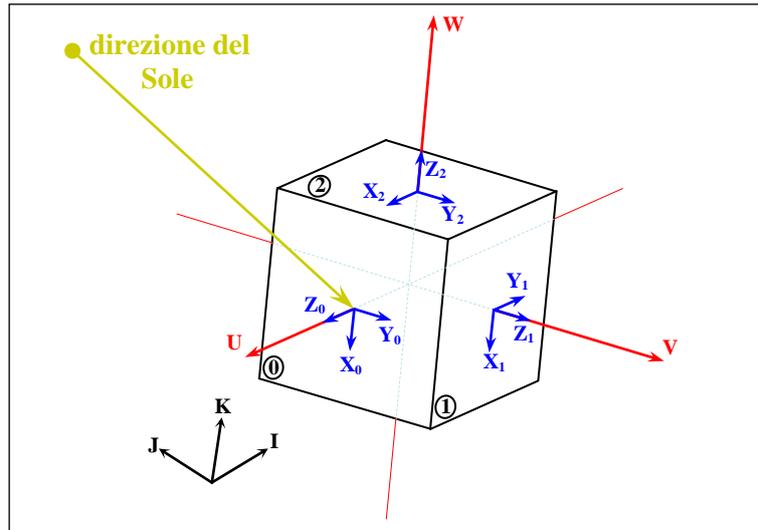


Figura 8.2 Indeterminazione dell'assetto.

Nella situazione indicata in figura si ipotizza che soltanto la tile 0 fornisca l'informazione circa la direzione del Sole. Una qualunque rotazione del satellite attorno all'asse coincidente con la direzione dettata dalla posizione Sole, a causa dell'architettura del sensore di Sole, non sarà letta dal sensore. In altre parole il pallino luminoso focalizzato sul sensore di posizione del Sole rimane immobile se il satellite ruota attorno all'asse in questione. In questo caso ci sarà un grado di libertà non fissato, ovvero, la non determinazione completa dell'assetto del satellite.

## 8.2 La determinazione dell'assetto mediante sensori di campo magnetico

In questo paragrafo si descrive un algoritmo utilizzabile durante la soluzione del problema della determinazione dell'assetto del satellite basato sulle informazioni fornite dai sensori di campo magnetico. Si rimanda il lettore al capitolo 6 per osservare la definizione dei sistemi di riferimento del satellite e delle tiles. È opportuno precisare fin dall'inizio che, mediante il procedimento che sarà descritto nel seguito, non risulterà possibile risolvere in modo completo il problema della determinazione dell'orientamento del satellite. Il campo magnetico terrestre, indicato con il simbolo  $\vec{\varphi}$ , infatti, definisce una singola direzione nello spazio e, di conseguenza, il problema della determinazione dell'assetto possiede ancora un grado di libertà. Questo porta ad affermare che qualsiasi rotazione attorno alla direzione designata da  $\vec{\varphi}$  non potrà essere specificata. Quindi, per riuscire a risolvere il problema della determinazione dell'orientamento del satellite sono necessarie altre informazioni, per esempio quelle fornite dai sensori di posizione del Sole.

La Power Management tile del satellite AraMiS, come annunciato nel capitolo 6, è provvista di un sensore di campo magnetico che si occupa di effettuare la misura del campo magnetico terrestre esterno biassiale. Il sensore di campo magnetico, a causa della sua particolare geometria di costruzione, fornisce soltanto due delle componenti del campo magnetico terrestre, quella su X, indicata con il simbolo  $\varphi_x$ , e quella su Y, indicata con il simbolo  $\varphi_y$ . Dalla lettura di diverse tiles risulta quindi possibile specificare il vettore  $\vec{\varphi}$  in modo completo. Si noti che due tiles non parallele

sono sufficienti, infatti, in questa situazione, sicuramente sono ottenibili tutte e tre le componenti di  $\vec{\varphi}$ .

### 8.2.1 L'algoritmo di determinazione del vettore campo magnetico terrestre rispetto al sistema UVW del satellite

In questo paragrafo si descrive un algoritmo finalizzato alla determinazione dell'orientamento del vettore campo magnetico terrestre  $\vec{\varphi}$  rispetto al sistema di riferimento UVW del satellite. Si indichino le componenti di tale vettore in UVW nel seguente modo:

$$\vec{\varphi} = (\varphi_u, \varphi_v, \varphi_w) \quad (8.9)$$

Questo vettore rappresenta il vettore di misura del problema della determinazione dell'assetto nella discussione seguente.

Nel capitolo 7 precedente è stato descritto il campo magnetico terrestre. Si indichino le componenti del vettore  $\vec{\varphi}$ , rispetto al sistema di riferimento geocentrico inerziale IJK nel seguente modo:

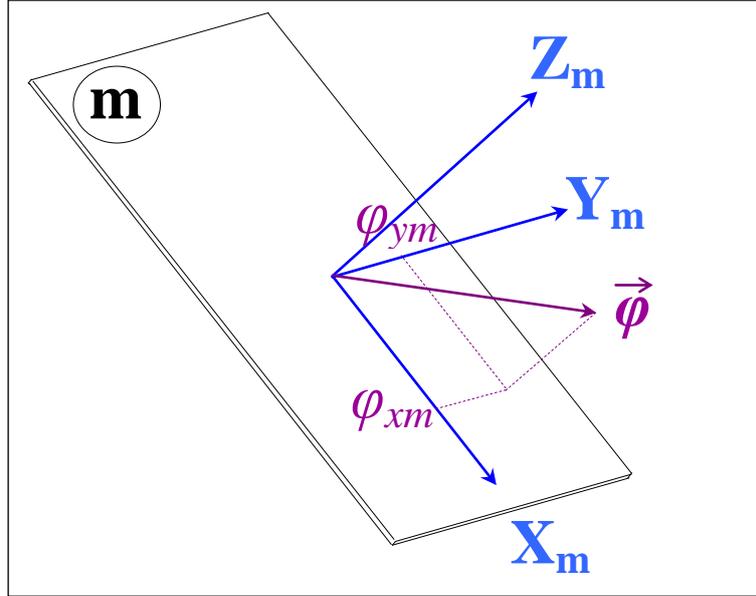
$$\vec{\varphi} = (\varphi_i, \varphi_j, \varphi_k) \quad (8.10)$$

Questo vettore rappresenta il vettore di riferimento e, nella discussione seguente, le sue componenti si considerano note.

Si consideri la tile  $m$ , e le componenti di campo magnetico associate ad essa,  $\varphi_{xm}$  e  $\varphi_{ym}$ . E' possibile scrivere la relazione 8.11 seguente:

$$\begin{cases} \hat{x}_m \cdot \vec{\varphi} = \varphi_{xm} \\ \hat{y}_m \cdot \vec{\varphi} = \varphi_{ym} \end{cases} \quad (8.11)$$

Dove il simbolo  $\cdot$  rappresenta l'operazione di prodotto scalare mentre  $\hat{x}_m$  e  $\hat{y}_m$  rappresentano i versori in direzione X e Y, rispettivamente. Si riporta in figura 8.3 la geometria del sensore di campo magnetico relativo alla tile  $m$ :



**Figura 8.3 Geometria del sensore di campo magnetico.**

Dal momento che l'orientamento del sistema XYZ della tile  $m$  rispetto al sistema UVW è noto, le componenti  $\hat{x}_m = (x_{mu}, x_{mv}, x_{mw})$  e  $\hat{y}_m = (y_{mu}, y_{mv}, y_{mw})$  risultano anch'esse note. Quindi è possibile scrivere il sistema di equazioni 8.12 seguente:

$$\begin{bmatrix} x_{mu} & x_{mv} & x_{mw} \\ y_{mu} & y_{mv} & y_{mw} \end{bmatrix} \begin{bmatrix} \varphi_u \\ \varphi_v \\ \varphi_w \end{bmatrix} = \begin{bmatrix} \varphi_{xm} \\ \varphi_{ym} \end{bmatrix} \quad (8.12)$$

Il sistema di equazioni 8.12 corrisponde a due equazioni nelle tre incognite  $\varphi_u$ ,  $\varphi_v$  e  $\varphi_w$ , quindi la sua soluzione risulta impossibile. Se si considera una seconda tile, indicata con il simbolo  $n$ , con le due misure di campo magnetico associate ad essa,  $\varphi_{xm}$  e  $\varphi_{ym}$ , è possibile aggiungere due nuove equazioni al sistema 8.12 ed ottenere il seguente sistema di equazioni:

$$\begin{bmatrix} x_{mu} & x_{mv} & x_{mw} \\ y_{mu} & y_{mv} & y_{mw} \\ x_{nu} & x_{nv} & x_{nw} \\ y_{nu} & y_{nv} & y_{nw} \end{bmatrix} \begin{bmatrix} \varphi_u \\ \varphi_v \\ \varphi_w \end{bmatrix} = \begin{bmatrix} \varphi_{xm} \\ \varphi_{ym} \\ \varphi_{xn} \\ \varphi_{yn} \end{bmatrix} \quad (8.13)$$

Nel sistema 8.13, come risulta facile osservare, sono presenti quattro equazioni nelle tre incognite  $\varphi_u$ ,  $\varphi_v$  e  $\varphi_w$ , quindi il sistema risulta sovra determinato. Si noti che questa affermazione è valida soltanto se le tiles  $m$  e  $n$  non sono parallele, infatti, nel caso di due tiles parallele si ritorna nella situazione descritta dal sistema 8.12, dal momento che soltanto due colonne del sistema 8.13 risulterebbero differenti.

Nel caso generale si considerano  $N$  tiles differenti, quindi il sistema di equazioni risultante è dato da:

$$\begin{bmatrix} x_{1u} & x_{1v} & x_{1w} \\ y_{1u} & y_{1v} & y_{1w} \\ & \vdots & \\ x_{Nu} & x_{Nv} & x_{Nw} \\ y_{Nu} & y_{Nv} & y_{Nw} \end{bmatrix} \begin{bmatrix} \varphi_u \\ \varphi_v \\ \varphi_w \end{bmatrix} = \begin{bmatrix} \varphi_{x1} \\ \varphi_{y1} \\ \vdots \\ \varphi_{xN} \\ \varphi_{yN} \end{bmatrix} \quad (8.14)$$

L'espressione 8.14 rappresenta la moltiplicazione di una matrice di dimensioni  $(2N \times 3)$ , indicata con il simbolo  $\underline{\underline{B}}$ , per un vettore di dimensioni  $(3 \times 1)$ , il quale rappresenta il trasposto di  $\vec{\varphi}$ . Il risultato è chiaramente un vettore di dimensioni  $(2N \times 1)$ , indicato con il simbolo  $\vec{\Phi}$ . Il sistema 8.14 può quindi essere riscritto nella forma 8.15 seguente:

$$\underline{\underline{B}} \cdot \vec{\varphi}^T = \vec{\Phi}^T \quad (8.15)$$

Il sistema di equazioni precedente non ammette una soluzione esatta, dal momento che risulta essere sovra determinato. E' possibile tuttavia cercare la soluzione che minimizzi l'errore quadratico medio, calcolando la matrice pseudoinversa di  $\underline{\underline{B}}$ , indicata con il simbolo  $\underline{\underline{B}}^+$ , e moltiplicandola per ambo i membri della relazione 8.15. La soluzione finale del problema risulta pertanto essere data da:

$$\vec{\varphi}^T = \underline{\underline{B}}^+ \cdot \vec{\Phi}^T \quad (8.16)$$

La matrice pseudoinversa in questione è possibile ricavarla utilizzando la decomposizione ai valori singolari (indicata con l'acronimo SVD, *Singular Value Decomposition*). Utilizzando la tecnica SVD una matrice di numeri reali  $M$ , di dimensioni  $(m \times n)$ , è scomposta nel prodotto di tre matrici, come riportato nella relazione 8.17 seguente:

$$M = U \cdot \Sigma \cdot V^T \quad (8.17)$$

Dove  $U$  rappresenta una matrice unitaria di dimensioni  $(m \times m)$ ,  $\Sigma$  rappresenta una matrice diagonale di dimensioni  $(m \times n)$  avente numeri reali positivi sulla diagonale e  $V^T$  rappresenta la matrice trasposta di  $V$ , una matrice unitaria di dimensioni  $(n \times n)$ .

Le colonne di  $V$  rappresentano gli autovettori di  $M^T M$  e, quindi, una base di vettori ortonormale di direzioni per  $M$ . Le colonne di  $U$ , invece, rappresentano gli autovettori di  $MM^T$ . Gli elementi diagonali della matrice  $\Sigma$ , infine, rappresentano gli *elementi singolari* della matrice  $M$ . Questi ultimi raffigurano le radici quadrate degli autovalori corrispondenti agli autovettori contenuti nelle colonne di  $U$  e  $V$ .

La tecnica SVD risulta utile nel calcolo della matrice pseudoinversa, indicata con il simbolo  $M^+$ . Infatti, tale matrice, risulta essere data dal prodotto seguente:

$$M^+ = V \cdot \Sigma^+ \cdot U^T \quad (8.18)$$

Dove  $U^T$  rappresenta la trasposta di  $U$  e  $\Sigma^+$  la matrice pseudoinversa di  $\Sigma$ . Quest'ultima, dal momento che  $\Sigma$  è diagonale, è ricavabile con due semplici operazioni:

- Calcolo della matrice trasposta di  $\Sigma$ .

- Calcolo del reciproco di ogni elemento non nullo della matrice calcolata con l'operazione precedente.

### 8.3 La determinazione dello spin del satellite

In questo paragrafo si presenta il problema della determinazione dello spin del satellite ed una sua possibile soluzione. Le forze esterne agenti sulla struttura del satellite (per una descrizione più approfondita si veda paragrafo 2, capitolo 4), quali la pressione esercitata dalla radiazione solare o l'attrito dovuto all'atmosfera, possono causare la rotazione del satellite attorno ad un asse incognito. Questo movimento è detto *spin*. Lo spin può essere espresso usando un vettore, indicato con il simbolo  $\vec{\omega}$ , la cui direzione rappresenta l'asse attorno al quale avviene la rotazione ed il cui modulo rappresenta la velocità angolare attorno allo stesso asse. Si fissa come positivo lo spin che produce una rotazione del satellite in senso antiorario, negativo nel caso contrario. Conoscere lo spin del satellite risulta molto utile anche nella prima fase di una missione spaziale in cui il satellite è espulso dal lanciatore ed acquisisce, di conseguenza, uno spin incognito.

La Power Management tile del satellite AraMiS, come annunciato nel capitolo 6, è fornita di un sensore giroscopico che si occupa della misurazione dello spin attorno all'asse Z della stessa. Si ricordi che il sistema di riferimento della singola tile, indicato con XYZ, è fissato in modo che l'asse Z sia disposto in modo normale al piano della tile (si veda capitolo 6).

#### 8.3.1 L'algoritmo di determinazione dello spin del satellite

Scopo della discussione seguente è quello di presentare un algoritmo finalizzato al calcolo del vettore di spin, indicato con il simbolo  $\vec{\omega}$ , e della sua rappresentazione nelle componenti rispetto agli assi geometrici UVW del satellite, nel seguente modo:

$$\vec{\omega} = (\omega_u \quad \omega_v \quad \omega_w) \quad (8.19)$$

L'algoritmo utilizzato nel seguito della discussione, di fatto, è del tutto equivalente a quello presentato nel paragrafo precedente relativo alla determinazione dell'assetto del satellite mediante i sensori di campo magnetico.

Si consideri la tile  $m$  e la corrispondente misura dello spin ad essa associata, indicata con il simbolo  $\omega_{zm}$ . È possibile quindi scrivere la relazione seguente:

$$\hat{z}_m \cdot \vec{\omega} = \omega_{zm} \quad (8.20)$$

Dove è stato indicato con il simbolo  $\cdot$  l'operazione di prodotto scalare e con  $\hat{z}_m$  il versore diretto verso l'asse Z della tile  $m$ . Siccome l'orientamento di  $\hat{z}_m$  rispetto al sistema UVW,  $\hat{z}_m = (z_{mu} \quad z_{mv} \quad z_{mw})$ , è noto e statico, è possibile scrivere anche la relazione 8.21 seguente:

$$\begin{bmatrix} z_{mu} & z_{mv} & z_{mw} \end{bmatrix} \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix} = \begin{bmatrix} \omega_{zm} \end{bmatrix} \quad (8.21)$$

L'espressione appena scritta corrisponde ad un'equazione nelle tre incognite  $\omega_u$ ,  $\omega_v$  e  $\omega_w$ , e, pertanto, risulta non risolvibile.

Se si considerano due tiles separate, al sistema 8.21 precedente si aggiunge un'equazione e, quindi, si ottiene un sistema di due equazioni in tre incognite, non ancora risolvibile.

Si consideri ora il caso di tre tiles, indicate con i simboli  $m$ ,  $n$  e  $p$ , con le relative misure di velocità angolari,  $\omega_{zm}$ ,  $\omega_{zn}$  e  $\omega_{zp}$ , rispettivamente. In questa situazione, il sistema di equazioni 8.21 diventa:

$$\begin{bmatrix} z_{mu} & z_{mv} & z_{mw} \\ z_{nu} & z_{nv} & z_{nw} \\ z_{pu} & z_{pv} & z_{pw} \end{bmatrix} \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix} = \begin{bmatrix} \omega_{zm} \\ \omega_{zn} \\ \omega_{zp} \end{bmatrix} \quad (8.22)$$

Il sistema di equazioni ottenuto è composto di tre equazioni in tre incognite e risulta, quindi, risolvibile. Si noti che, affinché il sistema 8.22 ammetta una soluzione, le tre tiles non devono risultare parallele. In questo caso, infatti, il numero di equazioni si riduce e si ritorna nelle situazioni descritte in precedenza.

Nel caso generale il satellite è variegato da  $N$  differenti tiles, quindi il sistema di equazioni generale è dato dal seguente sistema di equazioni:

$$\begin{bmatrix} z_{1u} & z_{1v} & z_{1w} \\ \vdots & \vdots & \vdots \\ z_{Nu} & z_{Nv} & z_{Nw} \end{bmatrix} \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix} = \begin{bmatrix} \omega_{z1} \\ \vdots \\ \omega_{zN} \end{bmatrix} \quad (8.23)$$

Il sistema 8.23 è equivalente al prodotto di una matrice di dimensioni  $(N \times 3)$ , indicata con il simbolo  $\underline{\underline{Z}}$ , per un vettore di dimensioni  $(3 \times 1)$ , indicato con il simbolo  $\vec{\omega}^T$ . Il risultato dell'operazione risulta quindi essere un vettore di dimensioni  $(N \times 1)$ , indicato con il simbolo  $\vec{\Omega}^T$ . Utilizzando questi simboli è possibile scrivere la relazione 8.23 nella forma seguente:

$$\underline{\underline{Z}} \cdot \vec{\omega}^T = \vec{\Omega}^T \quad (8.24)$$

Il sistema di equazioni ottenuto risulta essere sovra determinato e, quindi, non risolvibile in modo esatto. E' tuttavia possibile cercare la soluzione che minimizzi l'errore quadratico medio, calcolando la matrice pseudoinversa di  $\underline{\underline{Z}}$ , indicata con il simbolo  $\underline{\underline{Z}}^+$ , e moltiplicandola per ambo i membri della relazione 8.24. In questo modo la soluzione finale del problema è data dalla relazione 8.25 seguente:

$$\vec{\omega}^T = \underline{\underline{Z}}^+ \cdot \vec{\Omega}^T \quad (8.25)$$

## 8.4 La generalizzazione della soluzione del problema della determinazione dell'assetto

In questo paragrafo si riporta la generalizzazione della soluzione del problema della determinazione dell'assetto del satellite basato su di un numero variabile di informazioni fornite da qualsiasi sensore d'assetto presente. Nei capitoli precedenti sono stati analizzati gli algoritmi di determinazione dell'assetto mediante sensori di posizione del Sole e sensori di campo magnetico. Esistono anche altri sensori d'assetto, quali, ad esempio, i sensori di Terra (restituiscono l'informazione circa la direzione del centro della Terra) ed i sensori di stelle (restituiscono l'informazione circa la direzione di una particolare stella o pianeta). Nella discussione seguente non ci si interessa del particolare sensore utilizzato. In altre parole si presenta un algoritmo di determinazione dell'assetto totalmente indipendente dalla sensoristica a bordo del satellite.

### 8.4.1 L'algoritmo di determinazione dell'assetto del satellite

Si ricorda che lo scopo della determinazione dell'assetto è quello di specificare l'orientamento del sistema di riferimento UVW del satellite, mediante la matrice dei coseni direttori, rispetto al sistema geocentrico inerziale IJK (si vedano i capitoli 2 e 6 per trovare la descrizione dei vari sistemi di riferimento). Per risolvere il problema è sufficiente conoscere le direzioni in IJK di due assi non paralleli, le cui componenti sono note anche rispetto al sistema di riferimento UVW del satellite.

In generale, dati  $N$  versori, indicati con i simboli  $\hat{a}_1, \dots, \hat{a}_N$ , le cui coordinate sono note sia in IJK, sia in UVW, è possibile scrivere la relazione seguente:

$$\begin{cases} \hat{a}_m = (a_{mu} & a_{mv} & a_{mw}) \\ \hat{a}_m = (a_{mi} & a_{mj} & a_{mk}) \end{cases} \quad (8.26)$$

Dove  $m \in [1, \dots, N]$ . Risulta quindi possibile definire due matrici di dimensioni  $(N \times 3)$ , indicate con i simboli  $\underline{\underline{F}}$  e  $\underline{\underline{G}}$ , nel modo seguente:

$$\begin{cases} \underline{\underline{F}} = \begin{bmatrix} a_{1u} & a_{1v} & a_{1w} \\ \vdots & \vdots & \vdots \\ a_{Nu} & a_{Nv} & a_{Nw} \end{bmatrix} \\ \underline{\underline{G}} = \begin{bmatrix} a_{1i} & a_{1j} & a_{1k} \\ \vdots & \vdots & \vdots \\ a_{Ni} & a_{Nj} & a_{Nk} \end{bmatrix} \end{cases} \quad (8.27)$$

La matrice  $\underline{\underline{F}}$  permette di trasformare le coordinate di un vettore noto in UVW nelle coordinate rispetto alla base formata dagli  $N$  versori. Viceversa, la matrice  $\underline{\underline{G}}$  permette di effettuare la trasformazione di coordinate di un vettore noto in IJK nelle coordinate rispetto alla base formata dagli  $N$  versori. Si indichi con  $\underline{\underline{G}}^+$  la matrice di dimensioni  $(3 \times N)$ , pseudoinversa di  $\underline{\underline{G}}$ . Le coordinate in IJK di un vettore  $\vec{p} = (p_u \ p_v \ p_w)$  in UVW risultano quindi essere date dalla relazione 8.28 seguente:

$$\begin{bmatrix} p_i \\ p_j \\ p_k \end{bmatrix} = \underline{\underline{G}}^+ \cdot \underline{\underline{F}} \cdot \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (8.28)$$

La matrice  $\underline{\underline{A}}^{-1} = \underline{\underline{G}}^+ \cdot \underline{\underline{F}}$ , di dimensioni  $(3 \times 3)$ , rappresenta la trasformazione di coordinate dal sistema UVW al sistema di riferimento IJK. La trasformazione inversa risulta pertanto essere data dalla matrice  $\underline{\underline{A}}$  e rappresenta la matrice d'assetto ricercata, ovvero la trasformazione di coordinate dal sistema di riferimento inerziale IJK al sistema di riferimento UVW. Essendo le terne IJK e UVW ortogonali non è necessario invertire le matrici, è sufficiente effettuare l'operazione di trasposizione.

Nel caso particolare in cui  $N = 2$  è necessario calcolare un terzo versore al fine di ottenere una matrice  $\underline{\underline{A}}$  piena, altrimenti la terza riga della stessa risulterebbe tutta a zero. Il terzo versore è semplicemente calcolabile come prodotto vettoriale dei primi due, nel modo seguente:

$$\hat{a}_3 = \hat{a}_1 \times \hat{a}_2 \quad (8.29)$$

In questa situazione si ottiene quindi:

$$\left\{ \begin{array}{l} \underline{\underline{F}} = \begin{bmatrix} a_{1u} & a_{1v} & a_{1w} \\ a_{2u} & a_{2v} & a_{2w} \\ a_{1v}a_{2w} - a_{1w}a_{2v} & a_{1w}a_{2u} - a_{1u}a_{2w} & a_{1u}a_{2v} - a_{1v}a_{2u} \end{bmatrix} \\ \underline{\underline{G}} = \begin{bmatrix} a_{1i} & a_{1j} & a_{1k} \\ a_{2i} & a_{2j} & a_{2k} \\ a_{1j}a_{2k} - a_{1k}a_{2j} & a_{1k}a_{2i} - a_{1i}a_{2k} & a_{1i}a_{2j} - a_{1j}a_{2i} \end{bmatrix} \end{array} \right. \quad (8.30)$$

Si noti che nel caso particolare in cui le matrici  $\underline{\underline{F}}$  e  $\underline{\underline{G}}$  hanno dimensioni  $(3 \times 3)$ , ovvero nel caso particolare in cui  $N = 3$  (rientra chiaramente in questo caso anche la situazione descritta nella relazione 8.30 precedente), il problema è equivalente ad un sistema di equazioni di 3 equazioni in 3 incognite. Questo sistema di equazioni ammette, ovviamente, una soluzione esatta e la relazione 8.28 è possibile riscriverla nella forma seguente:

$$\begin{bmatrix} p_i \\ p_j \\ p_k \end{bmatrix} = \underline{\underline{G}}^{-1} \cdot \underline{\underline{F}} \cdot \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (8.31)$$

Si noti che nella relazione 8.31 precedente è stata indicata la matrice inversa di  $\underline{\underline{G}}$ , dal momento che, nella situazione analizzata, tale matrice ha dimensioni  $(3 \times 3)$  e, quindi, la matrice pseudoinversa risulta essere coincidente con la matrice inversa.

Nel caso generale, invece,  $N > 3$ , e, di conseguenza, il sistema di equazioni non ammette una soluzione esatta, dal momento che risulta essere sovra determinato. E' tuttavia possibile calcolare la matrice pseudoinversa di  $\underline{\underline{G}}$ , che si occupa di minimizzare l'errore quadratico medio della soluzione finale.

Quando si considerano le informazioni relative ai sensori di posizione del Sole i versori dell'algorithmo risultano pertanto essere:

- Gli orientamenti dei versori  $\hat{z}_m$  rispetto al sistema di riferimento geocentrico inerziale IJK di tutte le tiles illuminate dal Sole, ovvero le componenti  $\hat{z}_m = (z_{mi} \quad z_{mj} \quad z_{mk})$ .
- L'orientamento di ogni tile illuminata rispetto al sistema del satellite UVW, ovvero le matrici di rotazione di esse.

Così facendo le righe della matrice  $\underline{\underline{F}}$  rappresentano le componenti dei vari versori  $\hat{z}_m$  rispetto al sistema di riferimento UVW. Le righe di  $\underline{\underline{G}}$ , invece, rappresentano le componenti dei vari versori  $\hat{z}_m$  rispetto al sistema di riferimento IJK. Se almeno due tiles risultano illuminate dalla luce solare è dunque possibile specificare l'orientamento della terna UVW rispetto al sistema di riferimento IJK in modo completo. Altrimenti è necessario ricavare altre informazioni.

Quando si considerano le informazioni relative ai sensori di campo magnetico, invece, i versori dell'algorithmo risultano essere:

- L'orientamento attuale del vettore campo magnetico terrestre  $\vec{\varphi}$  rispetto al sistema di riferimento geocentrico IJK, ovvero le componenti  $\vec{\varphi} = (\varphi_i \quad \varphi_j \quad \varphi_k)$ . Si tenga presente che quest'orientamento è calcolato in funzione della posizione del satellite nell'orbita.
- Le misurazioni del campo magnetico terrestre di ogni tile avente il sensore di campo magnetico, ovvero le componenti  $\varphi_{xm}$  e  $\varphi_{ym}$  di ogni tile provvista di sensore di campo magnetico.
- L'orientamento di ogni tile rispetto al sistema di riferimento del satellite UVW, ovvero le matrici di rotazione di esse.

Si riporta, in conclusione, in figura 8.4 lo schema di flusso di massima dell'algorithmo di determinazione dell'assetto implementato, al fine di evidenziarne i punti principali (si noti che gli ingressi dell'algorithmo in analisi, non rappresentati in figura, risultano essere la data e l'ora, utili nel calcolo del modello della posizione del Sole e la posizione del satellite, utile nel calcolo del modello del vettore campo magnetico terrestre, si veda capitolo 7):

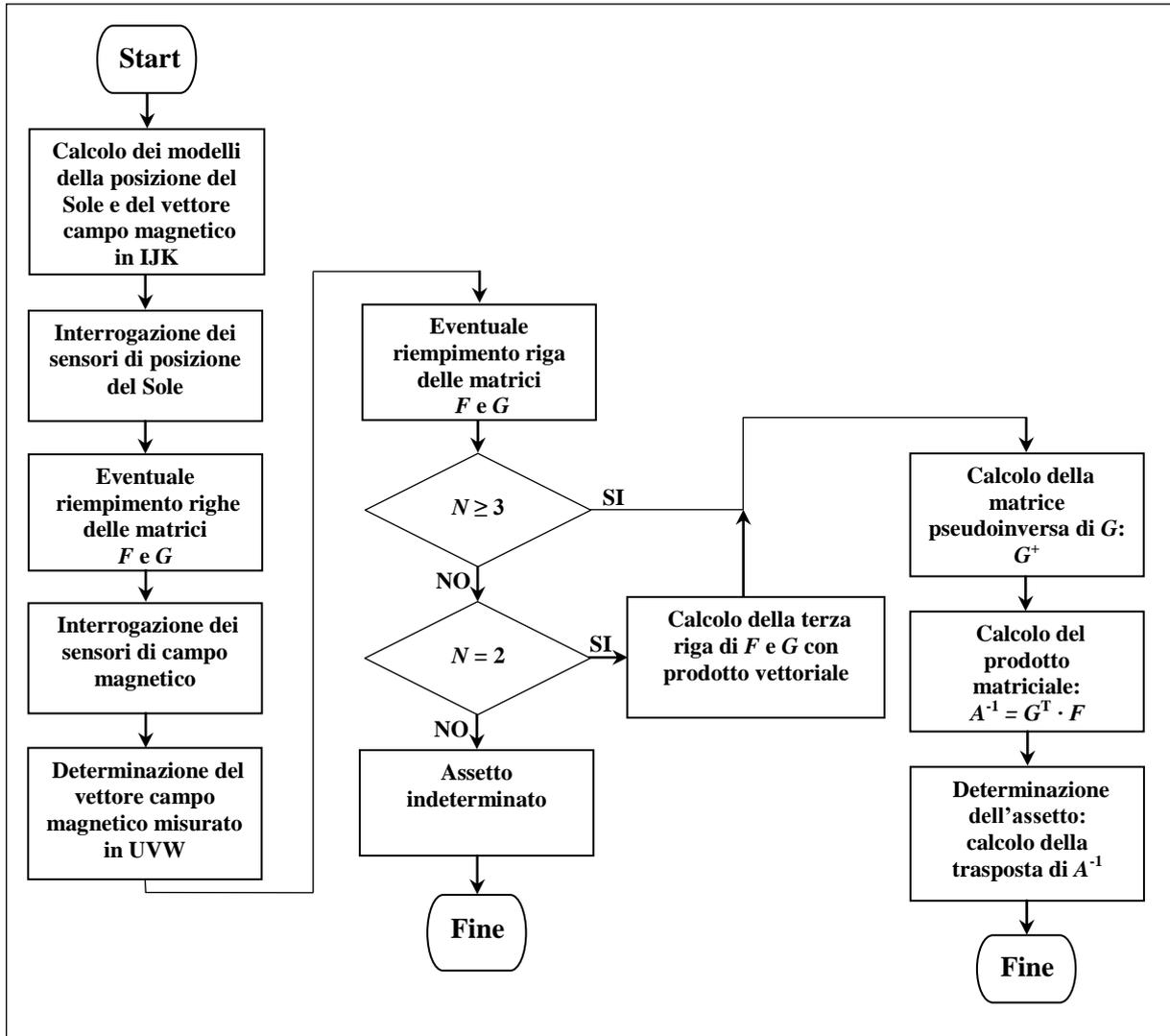


Figura 8.4 Diagramma di flusso dell'algoritmo di determinazione dell'assetto.

Si noti che, nel caso in cui  $N = 3$ , come annunciato in precedenza, il calcolo della matrice pseudoinversa è equivalente al calcolo della matrice inversa (non evidenziato nel diagramma precedente).

# Capitolo 9

## 9 La dinamica del satellite nello spazio

In questo capitolo si presenta la dinamica del satellite. Quest'ultima rappresenta un requisito fondamentale nello studio delle problematiche relative all'orientamento del satellite nello spazio e, nella discussione seguente, si introducono le definizioni fondamentali necessarie.

### 9.1 Le grandezze fondamentali

In questo paragrafo si introducono le grandezze fondamentali ed indispensabili nella descrizione della dinamica del satellite nello spazio tridimensionale. Nella discussione seguente il satellite è modellato come un corpo rigido perfetto, ovvero si fa l'assunzione che le mutue distanze tra due punti qualunque della struttura del satellite restano invariate in ogni istante. Inoltre, ci si sofferma sullo studio e la descrizione del moto di tipo rotazionale, tralasciando il caso generale della rototraslazione in quanto il satellite AraMiS non è provvisto di motori finalizzati al cambio d'orbita e, quindi, la traslazione del satellite non è contemplata nella discussione seguente.

La prima grandezza descritta, e, di fatto, la più importante nel caratterizzare il moto di tipo rotatorio, risulta essere il *momento della quantità di moto* o *momento angolare*, indicato con il simbolo  $\vec{L}$ .

Il momento della quantità di moto di un punto materiale di massa  $m$ , relativo all'origine  $O$  di un sistema di riferimento prefissato, indicato con il simbolo  $\vec{L}_O$ , risulta essere un vettore definito come il prodotto vettoriale tra il vettore di posizione ed il vettore della quantità di moto (definita come prodotto scalare tra la massa del punto materiale in questione ed il vettore della velocità), nel modo seguente:

$$\vec{L}_O = \vec{r} \times \vec{p} = \vec{r} \times m\vec{V} \quad (9.1)$$

Dove è stato indicato con il simbolo  $\times$  l'operazione di prodotto vettoriale,  $\vec{r}$  rappresenta il vettore di posizione,  $\vec{p}$  rappresenta il vettore della quantità di moto e  $\vec{V}$  rappresenta il vettore di velocità. La grandezza appena definita risulta essere, quindi, un vettore perpendicolare al piano generato dal vettore di posizione del punto materiale e dal vettore di velocità dello stesso. Il verso del momento angolare risulta essere individuato dalla regola della mano destra (posto il pollice della mano destra nella direzione di  $\vec{r}$ , e l'indice in quella di  $\vec{p}$ , il medio individuerà la direzione del momento angolare). È importante precisare che tale definizione di momento angolare ha senso soltanto se risulta specificata l'origine  $O$  del sistema di riferimento cui è associato il punto materiale in questione ( $O$  è detto *polo di riferimento*). Il modulo del momento angolare risulta pertanto essere dato dalla seguente relazione:

$$|\vec{L}_O| = r \cdot p \cdot \sin(\theta) \quad (9.2)$$

Dove  $\theta$  rappresenta l'angolo compreso tra il vettore di posizione ed il vettore della quantità di moto. Nel sistema internazionale, il momento della quantità di moto non possiede una sua unità di misura specifica, quindi risulta essere  $[\vec{L}] = [\text{m}^2 \cdot \text{Kg} \cdot \text{s}^{-1}]$ .

Si riporta in figura 9.1 la geometria del momento angolare di un punto materiale P relativo al polo di riferimento O prefissato:

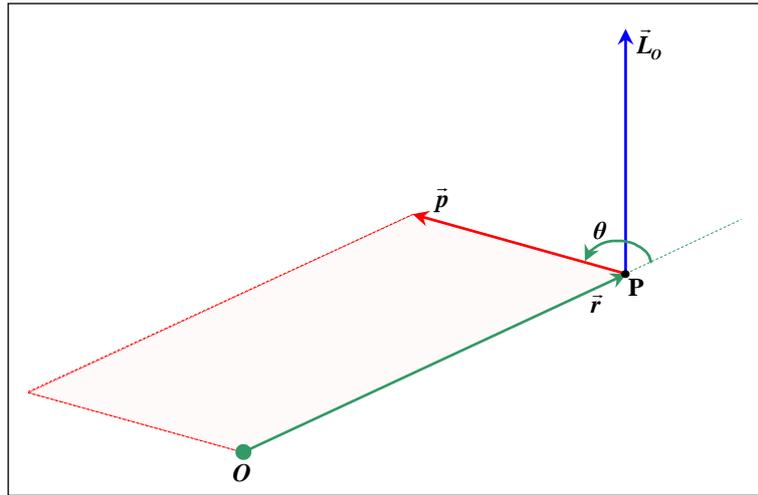


Figura 9.1 Momento angolare di un punto materiale.

Nel caso di un sistema di punti continuo, ovvero estendendo la definizione a tutto il corpo rigido (satellite) B, il momento angolare è definito come l'integrale dei momenti angolari dei singoli punti relativi all'origine O del sistema di riferimento solidale con il satellite prefissato, ovvero:

$$\vec{L}_O = \int_B \vec{r} \times \vec{V} dm \quad (9.3)$$

Dove  $dm$  rappresenta la massa infinitesimale dell'i-esimo punto materiale costituente il satellite.

Il vettore velocità  $\vec{V}$  assume la forma seguente:

$$\vec{V} = \vec{V}_O + \vec{\omega} \times \vec{r} + \vec{V}_{iB} \quad (9.4)$$

Dove  $\vec{V}_O$  rappresenta il vettore di velocità dell'origine del sistema di riferimento del satellite,  $\vec{\omega}$  rappresenta la velocità angolare del satellite e  $\vec{V}_{iB}$  rappresenta la velocità dell'i-esimo punto del corpo rigido rispetto al sistema di riferimento solidale con lo stesso che, per la definizione di corpo rigido, risulta essere nulla. La direzione del vettore  $\vec{\omega}$  specifica pertanto la direzione dell'asse di rotazione, il verso specifica il segno della rotazione, mentre il modulo specifica il valore della velocità espressa in  $[\text{rad} \cdot \text{s}^{-1}]$ .

Inserendo quindi la relazione 9.4 nella relazione 9.3, si ottiene l'espressione 9.5 seguente:

$$\vec{L}_O = \int_B \vec{r} \times (\vec{V}_O + \vec{\omega} \times \vec{r}) dm = -\vec{V}_O \times \int_B \vec{r} \cdot dm + \int_B \vec{r} \times (\vec{\omega} \times \vec{r}) dm \quad (9.5)$$

Si consideri ora il sistema di riferimento del satellite UVW (si veda il capitolo 6 per una descrizione del sistema di riferimento in questione) e si specifichino nello stesso le componenti dei vari vettori descritti in precedenza, nel modo seguente:

$$\vec{\omega} = \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix} \quad (9.6)$$

$$\vec{r} = \begin{bmatrix} r_u \\ r_v \\ r_w \end{bmatrix} \quad (9.7)$$

Svolgendo i prodotti vettoriali nel secondo termine dell'espressione 9.5 si ottiene:

$$\vec{r} \times (\vec{\omega} \times \vec{r}) = \begin{bmatrix} \omega_u r_v^2 - \omega_v r_u r_v + \omega_u r_w^2 - \omega_w r_u r_w \\ \omega_v r_w^2 - \omega_w r_v r_w + \omega_v r_u^2 - \omega_u r_u r_v \\ \omega_w r_u^2 - \omega_u r_u r_w + \omega_w r_v^2 - \omega_v r_v r_w \end{bmatrix} \quad (9.8)$$

Il primo termine dell'espressione 9.5 è possibile esplicitarlo in funzione del momento statico  $\vec{S}_O$ , il quale rappresenta una misura di quanto il polo di riferimento è spostato rispetto al centro di massa del corpo rigido, ovvero:

$$\vec{S}_O = \int_B \vec{r} \cdot dm \quad (9.9)$$

Il secondo termine dell'espressione 9.5, invece, risulta possibile esplicitarlo nel seguente modo, tenendo presente che le velocità angolari risultano essere costanti nel corpo rigido:

$$\int_B \vec{r} \times (\vec{\omega} \times \vec{r}) dm = \underline{\underline{I}} \cdot \vec{\omega} \quad (9.10)$$

Dove  $\underline{\underline{I}}$  rappresenta una matrice di dimensioni  $(3 \times 3)$ , detta *matrice d'inerzia* o *tensore d'inerzia* del satellite, rispetto alla terna di riferimento UVW, e definita nel modo seguente:

$$\underline{\underline{I}} = \begin{bmatrix} I_{uu} = \int_B (r_v^2 + r_w^2) dm & I_{uv} = \int_B -r_u r_v dm & I_{uw} = \int_B -r_u r_w dm \\ I_{vu} = \int_B -r_v r_u dm & I_{vv} = \int_B (r_u^2 + r_w^2) dm & I_{vw} = \int_B -r_v r_w dm \\ I_{wu} = \int_B -r_w r_u dm & I_{wv} = \int_B -r_w r_v dm & I_{ww} = \int_B (r_u^2 + r_v^2) dm \end{bmatrix} \quad (9.11)$$

In questo modo risulta possibile scrivere la relazione 9.5 nella forma seguente:

$$\vec{L}_O = -\vec{V}_O \times \vec{S}_O + \underline{\underline{I}} \cdot \vec{\omega} \quad (9.12)$$

La matrice d'inerzia  $\underline{\underline{I}}$  risulta essere una matrice simmetrica ed i termini posizionati sulla diagonale della stessa sono detti *momenti d'inerzia*, mentre gli altri sono detti *prodotti d'inerzia*. I momenti

d'inerzia risultano essere sicuramente positivi, mentre non è possibile conoscere a priori il segno dei prodotti d'inerzia. Gli elementi costituenti la matrice d'inerzia non risultano essere tutti indipendenti e devono soddisfare a precisi vincoli. E' possibile, infatti, scrivere sei disuguaglianze triangolari riguardanti gli elementi della matrice d'inerzia. Si hanno due tipi di vincoli, i primi tre riguardano i momenti d'inerzia e sono riportati nelle relazioni seguenti:

$$I_{uu} + I_{vv} = \int_B (r_u^2 + r_v^2 + 2r_w^2) dm \geq \int_B (r_u^2 + r_v^2) dm = I_{ww} \quad (9.13)$$

$$I_{uu} - I_{vv} = \int_B (r_v^2 - r_u^2) dm \leq \int_B (r_u^2 + r_v^2) dm = I_{ww} \quad (9.14)$$

Si hanno poi altre tre disuguaglianze, riguardanti anche i prodotti d'inerzia, e sono presentate nelle relazioni seguenti:

$$I_{uu} = \int_B (r_v^2 + r_w^2) dm = \int_B (r_v + r_w)^2 dm - \int_B 2r_w r_v dm = A + 2I_{vw} \quad (9.15)$$

Con  $A \geq 0$  e  $I_{uu} \geq 0$ , quindi, risulta essere:

$$I_{uu} \geq 2I_{vw} \quad (9.16)$$

Si noti che, fissando opportunamente il polo di riferimento  $O$  in modo che risulti coincidente con il centro di massa del satellite, il momento statico,  $\vec{S}_O$ , risulta essere nullo e, di conseguenza, la relazione 9.12 precedente si semplifica. Si riporta la forma semplificata della relazione 9.12 nella relazione 9.17 seguente:

$$\vec{L}_O = \underline{I} \cdot \vec{\omega} \quad (9.17)$$

Nella discussione seguente si assume che l'origine del sistema di riferimento del satellite UVW, definita in modo che risulti coincidente con l'esatto centro geometrico della struttura meccanica del satellite, coincida anche con il centro di massa del satellite. Se quest'ipotesi non sarà soddisfatta sarà necessario prendere in considerazione anche il termine relativo al momento statico.

## 9.2 Gli assi principali d'inerzia

Si supponga di osservare una rotazione attorno ad un asse arbitrario del corpo rigido. Nella situazione più generale possibile il momento angolare,  $\vec{L}$ , non risulta essere parallelo al vettore della velocità angolare  $\vec{\omega}$ .

Si consideri, ad esempio, una rotazione attorno all'asse W del satellite. In questo caso il vettore  $\vec{\omega}$ , specificandone opportunamente le componenti in UVW,  $\vec{\omega}_{uvw}$  in simbolo, risulta avere la forma seguente:

$$\vec{\omega}_{uvw} = (0 \quad 0 \quad \omega_w) \quad (9.18)$$

Dove è stata indicata con  $\omega_w$  la velocità angolare attorno all'asse W del satellite. Ora risulta possibile ricavare le componenti del momento angolare rispetto agli assi UVW. Utilizzando la relazione 9.17 si ottiene l'espressione seguente:

$$\begin{bmatrix} L_u \\ L_v \\ L_w \end{bmatrix} = \underline{\underline{I}} \cdot \vec{\omega} = \begin{bmatrix} I_{uw} \cdot \omega_w \\ I_{vw} \cdot \omega_w \\ I_{ww} \cdot \omega_w \end{bmatrix} \quad (9.19)$$

Nel caso generale  $I_{uw}$  e  $I_{vw}$  risultano essere diversi da zero e quindi il momento angolare risultante possiede le componenti lungo gli assi U e V. E' stato quindi dimostrato che, nel caso generale, il momento angolare non risulta essere parallelo all'asse di rotazione. Lo studio del moto del satellite, in queste condizioni, risulta essere piuttosto complicato.

Gli assi di rotazione, per i quali il momento angolare risulta essere parallelo all'asse di rotazione stesso, sono detti *assi principali d'inerzia*. E' possibile dimostrare che qualunque corpo rigido possiede almeno tre assi principali d'inerzia. Indichiamo con le lettere ABC il sistema di riferimento definito mediante gli assi appena citati. E' facile intuire che, nel sistema di riferimento ABC, la matrice d'inerzia deve avere una forma di tipo diagonale, in modo che si eliminino tutti i termini relativi ai prodotti d'inerzia. E' necessario quindi trovare un metodo per rendere diagonale la matrice d'inerzia,  $\underline{\underline{I}}$ . Il problema che si vuole risolvere è possibile formularlo nel modo seguente:

$$\underline{\underline{I}} = \underline{\underline{E}} \cdot \underline{\underline{I}}' \cdot \underline{\underline{E}}^T \quad (9.20)$$

La relazione 9.20 precedente risulta essere il problema agli autovalori associato alla matrice  $\underline{\underline{I}}$ . La soluzione di tale problema fornisce la matrice diagonale  $\underline{\underline{I}}'$  contenente i tre autovalori della matrice  $\underline{\underline{I}}$ . La matrice  $\underline{\underline{E}}$ , invece, rappresenta la matrice degli autovettori associati agli autovalori appena ricavati. Questa matrice può essere interpretata fisicamente come la trasformazione di coordinate, ovvero la matrice di rotazione, che porta la terna UVW del satellite a coincidere con la terna ABC degli assi principali d'inerzia. Le righe di  $\underline{\underline{E}}$  rappresentano dunque i coseni direttori, riferiti agli assi UVW, degli assi ABC, principali d'inerzia, rispetto ai quali i prodotti d'inerzia risultano essere nulli. Indicando con  $I_a$ ,  $I_b$  ed  $I_c$  i momenti d'inerzia nelle direzioni dettate dagli assi principali d'inerzia A, B e C, rispettivamente, la matrice  $\underline{\underline{I}}'$  assume la forma seguente:

$$\underline{\underline{I}}' = \begin{bmatrix} I_a & 0 & 0 \\ 0 & I_b & 0 \\ 0 & 0 & I_c \end{bmatrix} \quad (9.21)$$

Specificando l'espressione del momento angolare nel sistema degli assi principali, si ottiene la relazione seguente:

$$\vec{L}_O = \underline{\underline{I}}' \cdot \vec{\omega}_{abc} = \begin{bmatrix} I_a \cdot \omega_a \\ I_b \cdot \omega_b \\ I_c \cdot \omega_c \end{bmatrix} \quad (9.22)$$

La relazione 9.22 risulta essere decisamente più semplice della relazione 9.17 generale. A questo punto è possibile riprendere l'esempio fatto in precedenza ed applicarlo nel sistema di riferimento dettato dagli assi principali. A tal scopo si consideri una rotazione attorno all'asse principale C e si rappresenti anche il vettore della velocità angolare nel sistema ABC, principale d'inerzia, nel seguente modo:

$$\vec{\omega}_{abc} = (0 \quad 0 \quad \omega_c) \quad (9.23)$$

In questo caso il momento angolare risultante è dato dalla relazione seguente:

$$\vec{L}_O = \underline{\underline{I}} \cdot \vec{\omega}_{abc} = I_c \cdot \omega_c \quad (9.24)$$

E' stato quindi verificato che, ruotando attorno ad un asse principale, il vettore momento angolare e l'asse di rotazione risultano essere paralleli. Quest'importante proprietà è stata sfruttata durante l'implementazione dell'algoritmo di controllo dell'orientamento del satellite (capitolo 10).

Si noti che la direzione degli assi principali d'inerzia risulta essere dettata dalla simmetria della particolare struttura del satellite in questione. In altre parole, se la sezione del satellite presenta assi di simmetria, questi risultano essere gli assi principali d'inerzia del corpo. In genere si calcola la matrice d'inerzia rispetto ad una terna di assi geometricamente significativa fissata sulla struttura del satellite (*assi geometrici UVW*), in seguito si calcola la matrice principale d'inerzia con un problema agli autovalori associato alla prima.

### 9.3 Le equazioni dinamiche

In questo paragrafo si presenta l'equazione utilizzata nella descrizione della dinamica del satellite. L'equazione fondamentale che descrive la dinamica rotatoria del satellite, nota anche come *equazione di Eulero*, è riportata nella relazione seguente:

$$\vec{M}_O = \frac{d\vec{L}_O}{dt} \quad (9.25)$$

Dove il simbolo  $\vec{M}_O$  rappresenta una nuova grandezza detta *momento della forza* rispetto al polo di riferimento  $O$ , oppure *momento meccanico della forza* rispetto allo stesso.

Si consideri, per fissare le idee, un punto materiale situato in un punto  $P$  dello spazio cui è applicata una forza  $\vec{F}$ . La definizione classica del momento della forza applicata al punto materiale rispetto ad un polo  $O$  di riferimento è data dalla seguente relazione:

$$\vec{M}_O = \vec{r} \times \vec{F} \quad (9.26)$$

Dove  $\vec{r}$  rappresenta il vettore che specifica la posizione del punto materiale rispetto all'origine  $O$  prefissata. L'unità di misura di questa grandezza risulta pertanto essere  $[\vec{M}] = [N \cdot m]$ . Dalla relazione 9.26 è semplice osservare che la direzione di  $\vec{M}_O$  risulta essere ortogonale al piano

generato dai vettori  $\vec{r}$  ed  $\vec{F}$ . Il verso del momento della forza  $\vec{M}_O$ , invece, risulta essere dato dalla regola della mano destra, come annunciato per il momento angolare definito in precedenza. Si riporta in figura 9.2 la geometria del momento della forza in esame:

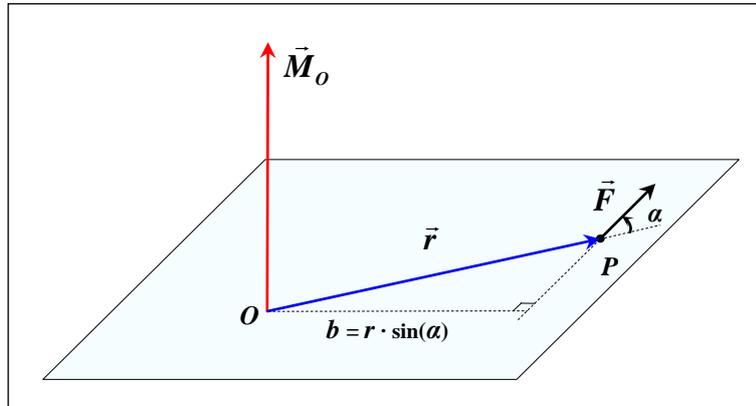


Figura 9.2 Momento della forza applicata ad un punto materiale.

Nella figura precedente è stato rappresentato l'angolo compreso tra il vettore  $\vec{r}$  ed il vettore  $\vec{F}$ , indicato con il simbolo  $\alpha$ , e la grandezza  $b$ , detta *braccio della forza*, definita dal prodotto tra il modulo del vettore  $\vec{r}$  ed il seno dell'angolo  $\alpha$ . Il modulo del momento della forza  $M_O$  risulta pertanto essere dato dalla relazione seguente:

$$M_O = r \cdot F \cdot \sin(\alpha) = F \cdot b \quad (9.27)$$

Nel caso particolare in cui  $\vec{r}$  e  $\vec{F}$  risultano ortogonali il modulo di  $\vec{r}$  diventa pari al braccio  $b$  ed il momento assume il suo massimo valore. Quest'affermazione è giustificata dal fatto che il seno dell'angolo  $\alpha$ , nella situazione in esame, diventa pari ad 1. Il momento risulta invece nullo nel caso in cui la forza applicata, oppure il braccio, risultano essere nulli, oppure nel caso in cui  $\vec{r}$  e  $\vec{F}$  sono paralleli.

La relazione 9.25 afferma d'altra parte che il momento della forza è equivalente alla derivata temporale del momento angolare. Quest'affermazione è nota anche come *teorema del momento angolare*. Per riuscire a dimostrare che effettivamente le relazioni 9.25 e 9.26 risultano essere equivalenti si riprenda la definizione di momento angolare rispetto al polo di riferimento  $O$  e si derivi rispetto al tempo, come riportato nelle relazioni seguenti:

$$\frac{d(\vec{L}_O)}{dt} = \frac{d(\vec{r} \times \vec{p})}{dt} = \frac{d(\vec{r})}{dt} \times \vec{p} + \vec{r} \times \frac{d(\vec{p})}{dt} = \vec{V} \times m \cdot \vec{V} + \vec{r} \times \vec{F} = 0 + \vec{r} \times \vec{F} = \vec{r} \times \vec{F} = \vec{M}_O \quad (9.28)$$

Si noti che il termine  $\vec{V} \times m \cdot \vec{V}$  si annulla, dal momento che il prodotto vettoriale tra vettori paralleli risulta essere nullo.

A questo punto risulta possibile estendere le definizioni precedenti nel caso del corpo rigido individuato dalla struttura del satellite. Si consideri l'equazione 9.17 del momento angolare ricavata precedentemente e si ricavi il momento della forza  $\vec{M}_O$ :

$$\vec{M}_O = \frac{d(\vec{L}_O)}{dt} = \frac{d(I \cdot \vec{\omega})}{dt} \quad (9.29)$$

Dalla relazione precedente è semplice osservare che nel caso in cui  $\vec{M}_O$  risulta essere nullo, ovvero nei casi elencati in precedenza, il momento angolare risulta essere di conseguenza una costante. Quest'importante affermazione è nota come *legge di conservazione del momento angolare* e specifica che il momento angolare, nel caso in cui la sua derivata temporale è nulla, risulta essere una costante del moto, ovvero si conserva.

Nell'eseguire l'operazione di derivazione riportata nella relazione precedente si tenga presente il sistema di riferimento su cui si sta lavorando. Nello studio di qualsiasi missione spaziale si ha a che fare con diversi sistemi di riferimento. Questo fa nascere i problemi di relativismo, ovvero un vettore specificato rispetto ad un sistema di riferimento è visto in modo differente da un altro sistema di riferimento. Le definizioni di momento angolare, tensore d'inerzia e velocità angolare sono state specificate rispetto al sistema di riferimento UVW. Se si vuole esprimere la relazione 9.29 anch'essa rispetto al sistema UVW, allora si ottiene l'evidente relazione seguente:

$$\vec{M}_{uvw} = \underline{\underline{I}} \cdot \vec{\omega}_{uvw} \quad (9.30)$$

Se, invece, si vuole esprimere il momento della forza rispetto al sistema di riferimento inerziale IJK si deve considerare sia la derivata relativa (ovvero quella riportata nella relazione precedente), sia la derivata assoluta (che tiene conto del movimento degli assi UVW rispetto ad IJK) del momento angolare. Utilizzando la *regola di derivazione di Poisson* è possibile dimostrare che si ottiene la relazione seguente:

$$\vec{M}_{ijk} = \dot{\vec{L}}_{uvw} + \vec{\omega} \times \vec{L}_{uvw} \quad (9.31)$$

# Capitolo 10

## 10 Il controllo dell'assetto del satellite

In questo capitolo si presenta l'algoritmo di controllo ad anello aperto dell'orientamento del satellite. Tale algoritmo si preoccupa di determinare il comando da fornire alle ruote di reazione, attuatori disposti diversamente sulla struttura del satellite, al fine di ottenere l'orientamento desiderato (spostamento di tipo punto-punto). Preliminarmente si riporta una descrizione delle ruote di reazione e del principio fisico su cui esse sono basate.

### 10.1 Le ruote di reazione

Le ruote di reazione rappresentano un importante esempio di attuatori molto utilizzati nell'ambito del controllo dell'assetto nello spazio. L'algoritmo di controllo dell'orientamento sviluppato è basato sull'attuazione di questi attuatori disposti diversamente sulla struttura del satellite. In altre parole l'algoritmo dedicato al controllo dell'assetto si occupa del calcolo del comando da fornire agli attuatori descritti nella discussione seguente.

Una ruota di reazione è composta di un disco con una certa massa che è messo in rotazione al fine di controllare la dinamica rotatoria del satellite e quindi il suo puntamento. Il disco è montato sulla Power Management tile in modo che il suo asse di rotazione sia coincidente con l'asse Z della tile (si veda capitolo 6 per la descrizione dei sistemi di riferimento del satellite e della tile). Si riporta in figura 10.1 una Power Management tile del satellite AraMiS con la ruota di reazione montata al suo interno:

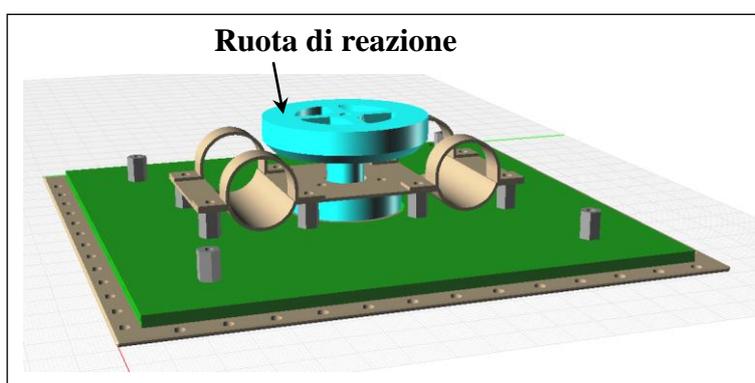


Figura 10.1 Power Management tile con ruota di reazione.

### 10.1.1 Il principio di funzionamento

Il principio di funzionamento della ruota di reazione si basa sulla legge di conservazione del momento angolare. Nel capitolo 9 è stato affermato che, nel caso in cui il momento delle forze applicate è nullo, il momento angolare risulta essere una costante. Questo equivale ad affermare che ponendo in rotazione una ruota di reazione si genera a sua volta una rotazione del satellite sullo stesso asse di rotazione su cui è montata la ruota, ma con verso opposto. Per fissare le idee si osservi la figura seguente in cui si riporta la configurazione cubica del satellite ed una sola ruota di reazione posizionata sulla tile 2 (si noti che nella figura seguente la ruota di reazione è disposta nella parte esterna della faccia del satellite, questo è fatto soltanto per comodità di rappresentazione, la ruota di reazione è chiaramente montata nella parte interna del satellite):

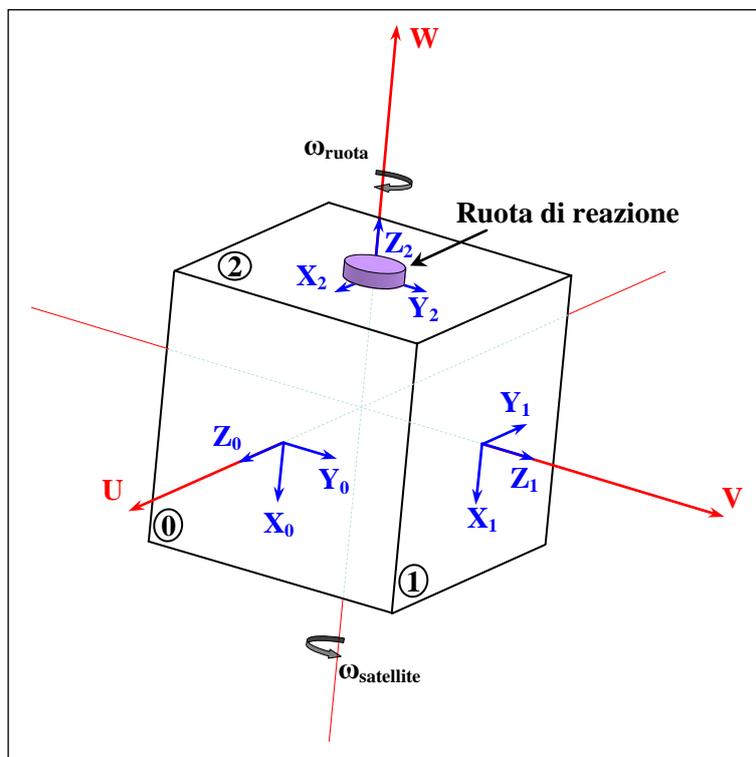


Figura 10.2 Principio di funzionamento della ruota di reazione.

Un opportuno motore elettrico si occupa di porre in rotazione le varie ruote di reazione presenti sul satellite. Siccome il motore elettrico applica una forza direttamente sul baricentro della ruota il braccio della forza applicata (distanza compresa tra il baricentro ed il punto di applicazione della forza) risulta essere nullo, ovvero il momento della forza generato è nullo.

La legge di conservazione del momento angolare è possibile esprimerla nella seguente relazione:

$$L_{iniziale} = L_{finale} \quad (10.1)$$

La relazione precedente afferma che, presi due istanti di tempo qualunque, nell'ipotesi di non applicare alcun momento della forza al satellite, il momento angolare deve risultare una costante. Si consideri quindi il caso in cui il satellite sia fermo inizialmente e si ponga in rotazione la ruota di reazione rappresentata nella figura precedente. Si indichi con i simboli  $I_{Z_2}$  e  $I_w$  i momenti d'inerzia della ruota di reazione rispetto alla direzione dettata dall'asse  $Z_2$  e del satellite rispetto alla

direzione dettata dall'asse W, rispettivamente. Risulta quindi possibile scrivere le relazioni per il momento angolare iniziale e finale:

$$L_{iniziale} = (I_{Z2} + I_W) \cdot \omega_{satellite} = (I_{Z2} + I_W) \cdot 0 \quad (10.2)$$

$$L_{finale} = I_{Z2} \cdot \omega_{ruota} + I_W \cdot \omega_{satellite} \quad (10.3)$$

Si noti che inizialmente sia la ruota, sia il satellite risultano essere fermi, per l'ipotesi fatta. A questo punto è possibile eguagliare i due termini ed ottenere la relazione seguente:

$$(I_{Z2} + I_W) \cdot 0 = I_{Z2} \cdot \omega_{ruota} + I_W \cdot \omega_{satellite} \quad (10.4)$$

Da cui si ricava la relazione seguente:

$$I_{Z2} \cdot \omega_{ruota} = -I_W \cdot \omega_{satellite} \quad (10.5)$$

Quest'espressione dimostra quindi il funzionamento descritto della ruota di reazione. Il satellite, infatti, in seguito alla rotazione della ruota acquisisce la velocità angolare seguente:

$$\omega_{satellite} = -\frac{I_{Z2} \cdot \omega_{ruota}}{I_W} \quad (10.6)$$

Questa relazione mostra chiaramente che il satellite è posto in rotazione con una velocità angolare di verso opposto a quella della ruota in esame e con un valore dettato dal rapporto tra il momento d'inerzia della ruota di reazione in analisi ed il momento d'inerzia del satellite.

## 10.2 L'algoritmo di controllo dell'orientamento del satellite

In questo paragrafo si descrive l'algoritmo di controllo ad anello aperto dell'assetto del satellite. L'algoritmo in questione è suddiviso, a sua volta, in altri algoritmi e sono descritti nei paragrafi seguenti.

### 10.2.1 Il calcolo della matrice di manovra

In questo paragrafo si presenta il problema della determinazione della manovra da compiere al fine di orientare il satellite secondo delle direzioni desiderate, specificate rispetto al sistema di riferimento geocentrico inerziale IJK. Queste direzioni desiderate costituiscono il comando che la stazione di Terra invia al computer di bordo del satellite ed il sistema software sviluppato si preoccupa di leggerlo (dopo essere stato opportunamente decodificato dal Command Interpreter, software presente sul computer di bordo del satellite) e di ricavare le grandezze opportune al fine di

risolvere i problemi riguardanti l'orientamento del satellite. Lo scopo dell'algoritmo di controllo dell'assetto risulta essere quello di prevedere il comando da fornire ad opportuni attuatori presenti a bordo del satellite al fine di ottenere l'orientamento desiderato. L'approccio utilizzato è un metodo ad anello aperto che si basa sull'inversione delle equazioni che descrivono il moto del satellite ed è descritto nei paragrafi successivi. In questa sede si vuole soltanto ricavare la rotazione necessaria al fine di orientare il satellite secondo le direzioni desiderate, ovvero si vuole determinare la matrice di rotazione equivalente.

Durante la trattazione del problema in questione è stata fissata la specifica riguardante il formato del comando previsto in ricezione. In particolare è stato fissato in modo che esso risulti composto di due campi distinti, ovvero:

- Un tempo.
- Una matrice dei coseni direttori.

Il primo rappresenta il tempo, espresso in secondi, in cui la manovra in questione deve essere portata a termine.

Il secondo rappresenta l'assetto desiderato del satellite (espresso sotto forma di matrice dei coseni direttori), ovvero l'orientamento desiderato della terna di riferimento UVW del satellite rispetto al sistema geocentrico inerziale IJK.

La questione del tempo necessario al completamento della manovra sarà chiarita meglio nel seguito della discussione. Ora ci si sofferma soltanto sul problema riguardante la valutazione della manovra necessaria, ovvero della rotazione necessaria, al fine di ottenere l'assetto desiderato del satellite. Quando è stato introdotto il momento angolare (si veda capitolo 9), è stata dichiarata l'esistenza, sulla struttura del satellite, della terna d'assi ABC, detta principale d'inerzia, rispetto alla quale la matrice d'inerzia risulta essere diagonale. Le direzioni dettate dagli assi ABC possono essere quindi considerate come delle direzioni privilegiate su cui agire, infatti, rispetto a questi assi il vettore della velocità angolare ed il momento angolare risultano essere paralleli, come annunciato nel capitolo precedente. Questo equivale ad affermare che un momento angolare applicato su di un asse principale interessa una rotazione soltanto attorno all'asse stesso (a patto che la rotazione interessi un asse alla volta ovviamente). E' stato pertanto deciso di sfruttare i benefici apportati da queste direzioni ed è stato deciso di agire su di esse durante la movimentazione del satellite. La manovra che ci si accinge quindi di determinare è espressa rispetto agli assi principali d'inerzia, ABC.

Si indichi con  $\underline{E}$  la matrice contenente gli autovettori della matrice d'inerzia  $\underline{I}$ , ovvero la matrice dei coseni direttori degli assi principali ABC rispetto al sistema di riferimento UVW del satellite (come annunciato nel capitolo precedente), con  $\underline{A}$  la matrice d'assetto attuale del satellite, ovvero l'orientamento della terna UVW rispetto al sistema di riferimento geocentrico inerziale IJK e con  $\underline{A}_d$  la matrice d'assetto desiderato, ovvero l'orientamento della terna UVW rispetto al sistema IJK che si desidera ottenere. Tutti questi parametri sono considerati noti nella discussione seguente.

Si indichino ora con  $\hat{u}'$ ,  $\hat{v}'$  e  $\hat{w}'$  i versori rappresentanti l'orientamento desiderato del satellite, inizialmente espressi rispetto al sistema di riferimento geocentrico inerziale IJK, e con  $\hat{a}'$ ,  $\hat{b}'$  e  $\hat{c}'$  i versori rappresentanti l'orientamento desiderato, equivalente, degli assi principali del satellite in seguito alla manovra.

La prima grandezza ricavabile risulta essere l'orientamento di  $\hat{u}'$ ,  $\hat{v}'$  e  $\hat{w}'$  rispetto al sistema di riferimento del satellite UVW. Indichiamo con il simbolo  $\underline{A}_{uvw}$  la matrice di rotazione associata a quest'orientamento. E' facile osservare che essa è data dalla relazione seguente:

$$\underline{A}_{uvw} = \underline{A}_d \cdot \underline{A}^T \quad (10.7)$$

A questo punto è facile ottenere l'orientamento di  $\hat{a}'$ ,  $\hat{b}'$  e  $\hat{c}'$  rispetto agli assi del satellite UVW, cui non è stato data un nome particolare, sfruttando la relazione seguente:

$$\underline{\underline{E}} \cdot \underline{\underline{A}}_{uvw} \quad (10.8)$$

In conclusione risulta possibile ricavare l'orientamento di  $\hat{a}'$ ,  $\hat{b}'$  e  $\hat{c}'$  rispetto agli assi principali ABC del satellite nel modo seguente:

$$\underline{\underline{M}}_{abc} = \underline{\underline{E}} \cdot \underline{\underline{A}}_{uvw} \cdot \underline{\underline{E}}^T \quad (10.9)$$

Dove  $\underline{\underline{M}}_{abc}$  rappresenta la matrice di manovra ricercata.

### 10.2.2 La scomposizione della matrice di manovra nella sequenza di tre singole rotazioni successive mediante la scomposizione dei quaternioni unitari

In questo paragrafo si presenta il problema della scomposizione di una matrice di rotazione nella sequenza di tre singole rotazioni successive. Nel caso in esame la matrice da scomporre è data dalla matrice di manovra  $\underline{\underline{M}}_{abc}$  ricavata precedentemente. La strategia di controllo adottata, infatti, prevede di utilizzare una sequenza di tre rotazioni successive, eseguite attorno agli assi principali del satellite ABC, con lo scopo di orientare il satellite qualsiasi sia l'assetto desiderato dello stesso. In questo ambito ci si basa sul teorema di Eulero, il quale afferma che qualsiasi rotazione nello spazio tridimensionale è scomponibile nella sequenza di tre rotazioni successive. Le tre rotazioni ricercate sono pertanto equivalenti agli angoli di Eulero associati alla particolare matrice di rotazione che si desidera scomporre.

Nel capitolo 5 sono stati introdotti i quaternioni unitari ed è stato affermato che una qualunque matrice di rotazione è possibile esprimerla, senza riscontrare problemi di singolarità, sotto forma di un quaternione unitario ad essa associato (si rimanda il lettore al capitolo 5, paragrafo 3, per una descrizione dei parametri in questione). Risulta quindi possibile esprimere la matrice di manovra  $\underline{\underline{M}}_{abc}$  sotto forma di un quaternione unitario, indicato con il simbolo  $\vec{q}_{abc}$ . Si noti che questi due parametri rappresentano la stessa ed identica rotazione nello spazio, infatti, sono entrambi specificati rispetto alla terna principale d'inerzia ABC. Si esprima il quaternione  $\vec{q}_{abc}$  nelle sue componenti rispetto ai versori  $\hat{a}$ ,  $\hat{b}$  e  $\hat{c}$ , nel modo seguente:

$$\vec{q}_{abc} = q_0 + \hat{a} \cdot q_1 + \hat{b} \cdot q_2 + \hat{c} \cdot q_3 \quad (10.10)$$

L'obiettivo in esame risulta essere quello di scomporre il quaternione  $\vec{q}_{abc}$  nel prodotto di tre quaternioni, indicati con i simboli  $\vec{q}'$ ,  $\vec{q}''$ ,  $\vec{q}'''$ , corrispondenti a tre singole rotazioni in sequenza attorno agli assi A, B e C, rispettivamente. In altre parole si vogliono trovare  $\vec{q}'$ ,  $\vec{q}''$ ,  $\vec{q}'''$  che soddisfino la relazione seguente:

$$\vec{q}_{abc} = \vec{q}''' \cdot \vec{q}'' \cdot \vec{q}' = (q_0''' + \hat{c} \cdot q_3''') \cdot (q_0'' + \hat{b} \cdot q_2'') \cdot (q_0' + \hat{a} \cdot q_1') \quad (10.11)$$

Si noti che nella relazione precedente il quaternione  $\vec{q}_{abc}$  è scomposto nella sequenza di tre rotazioni successive da compiere attorno agli assi principali d'inerzia nel seguente modo:

- Prima rotazione attorno all'asse principale C.
- Seconda rotazione attorno all'asse principale B.
- Terza rotazione attorno all'asse principale A.

Preliminarmente è possibile affrontare il problema più semplice consistente nella scomposizione del quaternione  $\vec{q}_{abc}$  nel prodotto di due quaternioni soltanto, indicati con i simboli  $\vec{q}''''$ ,  $\vec{q}'$ , definiti nel modo seguente:

$$\vec{q}_{abc} = \vec{q}'''' \cdot \vec{q}' = (q_0'''' + \hat{a} \cdot q_1'''' + \hat{b} \cdot q_2'''' + \hat{c} \cdot q_3''') \cdot (q_0' + \hat{a} \cdot q_1') \quad (10.12)$$

Si noti che il quaternione  $\vec{q}'$ , definito nella relazione 10.12 precedente è esattamente il medesimo presente nella relazione 10.11. Confrontando le relazioni 10.11 e 10.12 risulta semplice comprendere la necessità di scomporre il quaternione  $\vec{q}''''$  nel prodotto dei quaternioni  $\vec{q}''''$  e  $\vec{q}''$ , ovvero si desidera ottenere la condizione seguente:

$$\vec{q}'''' = \vec{q}'''' \cdot \vec{q}'' = (q_0'''' + \hat{c} \cdot q_3''') \cdot (q_0'' + \hat{b} \cdot q_2'') \quad (10.13)$$

L'operazione 10.13 precedente è permessa soltanto se la condizione seguente risulta essere soddisfatta:

$$q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0 \quad (10.14)$$

Si verifichi quindi la condizione precedente esplicitando i prodotti della relazione 10.13, nel modo seguente:

$$\begin{cases} q_0'''' = q_0'''' \cdot q_0'' \\ q_1'''' = -q_3'''' \cdot q_2'' \\ q_2'''' = q_0'''' \cdot q_2'' \\ q_3'''' = q_0'''' \cdot q_3'' \end{cases} \quad (10.15)$$

Dalla relazione precedente è semplice osservare che la condizione 10.14 risulta essere soddisfatta, infatti:

$$q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = (-q_0'''' \cdot q_0'' \cdot q_3'' \cdot q_2'') + (q_0'''' \cdot q_2'' \cdot q_0'' \cdot q_3'') = 0 \quad (10.16)$$

Si supponga quindi di conoscere un quaternione  $\vec{q}''''$  con il vincolo dettato dalla relazione 10.14 e si cerchi di determinare  $\vec{q}''''$  e  $\vec{q}''$ . Considerando le relazioni precedenti, e ricordando la definizione di quaternioni unitario, risulta possibile definire due angoli,  $\Psi$  e  $\theta$ , entrambi compresi nell'intervallo  $[0, 2\pi]$ , nel modo seguente:

$$\tan\left(\frac{\Psi}{2}\right) = \frac{q_3''''}{q_0''''} \quad (10.17)$$

$$\tan\left(\frac{\theta}{2}\right) = \frac{q_2''''}{q_0''''} \quad (10.18)$$

In questo modo risulta possibile determinare i rapporti  $q_3'''' / q_0''''$  e  $q_2'''' / q_0''''$ . Dal momento che si desidera ottenere  $\bar{q}''''$  e  $\bar{q}''$  con il vincolo che siano quaternioni unitari, i rapporti ottenuti risultano quindi sufficienti per definire:

$$\begin{cases} q_0'''' = \cos\left(\frac{\Psi}{2}\right) \\ q_3'''' = \sin\left(\frac{\Psi}{2}\right) \end{cases} \quad (10.19)$$

$$\begin{cases} q_0'' = \cos\left(\frac{\theta}{2}\right) \\ q_2'' = \sin\left(\frac{\theta}{2}\right) \end{cases} \quad (10.20)$$

Le definizioni di  $\Psi$  e  $\theta$  ottenute potrebbero risultare dipendenti dai rapporti tra  $q_0''''$ ,  $q_1''''$ ,  $q_2''''$ ,  $q_3''''$  scelti. Grazie all'ipotesi fatta ( $q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0$ ) queste definizioni risultano essere ben poste. Infatti, valgono le seguenti relazioni:

$$\begin{cases} \frac{q_3''''}{q_0''''} = \frac{q_3''''}{q_0''''} = -\frac{q_1''''}{q_2''''} = \frac{q_3''''}{q_0''''} \\ \frac{q_2''}{q_0''} = \frac{q_2''''}{q_0''''} = -\frac{q_1''''}{q_3''''} = \frac{q_2''}{q_0''} \end{cases} \quad (10.21)$$

In entrambe le identità, la prima uguaglianza è dettata dalla definizione di  $\Psi$  e  $\theta$ , la seconda uguaglianza è dettata dall'ipotesi fatta, ovvero la condizione  $q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0$  e la terza si ottiene calcolando i rapporti indicati mediante le equazioni 10.15. Quella appena presentata riguarda la scomposizione di un quaternioni rispetto agli assi C e B, principali d'inerzia. Ovviamente sono possibili altre scelte e per ognuna si otterranno delle relazioni differenti.

A questo punto è possibile affrontare il problema generale espresso nella relazione 10.11 precedente.

Il primo passo verso la soluzione generale consiste nel trovare  $\bar{q}'''' = (q_0'''' + \hat{a} \cdot q_1'''' + \hat{b} \cdot q_2'''' + \hat{c} \cdot q_3'''')$  e  $\bar{q}' = (q_0' + \hat{a} \cdot q_1')$  tali che  $\bar{q}_{abc} = \bar{q}'''' \cdot \bar{q}'$  e  $q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0$ . Sotto queste condizioni è possibile sfruttare il criterio appena presentato.

Dal momento che  $\bar{q}_{abc} = \bar{q}'''' \cdot \bar{q}'$ , è possibile scrivere la relazione che specifica  $\bar{q}''''$  semplicemente invertendo la relazione, nel modo seguente:

$$\begin{aligned} \bar{q}'''' &= (q_0'''' + \hat{a} \cdot q_1'''' + \hat{b} \cdot q_2'''' + \hat{c} \cdot q_3'''' ) = (q_0 + \hat{a} \cdot q_1 + \hat{b} \cdot q_2 + \hat{c} \cdot q_3) \cdot (q_0' - \hat{a} \cdot q_1') = \\ &= (q_0 \cdot q_0' + q_1 \cdot q_1') + \hat{a} \cdot (-q_0 \cdot q_1' + q_1 \cdot q_0') + \hat{b} \cdot (q_2 \cdot q_0' - q_3 \cdot q_1') + \hat{c} \cdot (q_3 \cdot q_0' + q_2 \cdot q_1') \end{aligned} \quad (10.22)$$

Si cerchi ora di definire  $q_0'$ ,  $q_1'$  tali che  $(q_0')^2 + (q_1')^2 = 1$  e tali che  $q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0$ . A tal scopo si scrivano le relazioni seguenti:

$$\begin{aligned} q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' &= (q_0 \cdot q_0' + q_1 \cdot q_1') \cdot (-q_0 \cdot q_1' + q_1 \cdot q_0') + (q_2 \cdot q_0' - q_3 \cdot q_1') \cdot (q_3 \cdot q_0' + q_2 \cdot q_1') = \\ &= (q_0')^2 \cdot (q_0 \cdot q_1 + q_2 \cdot q_3) + (q_0' \cdot q_1') \cdot (-q_0^2 + q_1^2 + q_2^2 - q_3^2) + (q_1')^2 \cdot (-q_0 \cdot q_1 - q_2 \cdot q_3) = \\ &= A \cdot [(q_0')^2 - (q_1')^2] + B \cdot (q_0' \cdot q_1') = 0 \end{aligned} \quad (10.23)$$

Dove  $A = q_0 \cdot q_1 + q_2 \cdot q_3$  e  $B = -q_0^2 + q_1^2 + q_2^2 - q_3^2$ .

A questo punto risulta possibile determinare l'angolo di rotazione attorno all'asse principale A, indicato con il simbolo  $\Phi$ , sfruttando l'espressione seguente:

$$A \cdot [(q_0')^2 - (q_1')^2] + B \cdot (q_0' \cdot q_1') = 0 \Rightarrow \frac{2 \cdot q_0' \cdot q_1'}{(q_0')^2 - (q_1')^2} = -\frac{2 \cdot A}{B} = \tan(\Phi) \quad (10.24)$$

In conclusione è possibile definire il quaternion  $\bar{q}'$  nel modo seguente:

$$\bar{q}' = (q_0' + \hat{a} \cdot q_1') = \left[ \cos\left(\frac{\Phi}{2}\right) + \hat{a} \cdot \sin\left(\frac{\Phi}{2}\right) \right] \quad (10.25)$$

Con questa definizione di  $q_0'$  e  $q_1'$  si ottiene un  $\bar{q}''''$  con  $q_0'''' \cdot q_1'''' + q_2'''' \cdot q_3'''' = 0$ , ovvero  $\bar{q}''''$  è possibile scomporlo nel prodotto di due quaternioni,  $\bar{q}'''$  e  $\bar{q}''$ , relativi ad una rotazione attorno a C ed a B, rispettivamente, come mostrato in precedenza.

Questo algoritmo di scomposizione sviluppato è quindi in grado di ottenere i tre angoli di Eulero relativi alla matrice di manovra calcolata nel paragrafo precedente. Queste tre grandezze saranno utilizzate nel calcolo del comando da fornire alle ruote di reazione presenti sul satellite, al fine di effettuare la manovra assegnata.

### 10.2.3 Il calcolo del comando da fornire alle ruote di reazione

Nel paragrafo precedente è stato presentato un metodo di scomposizione della matrice di manovra nel prodotto di tre singole rotazioni successive attorno agli assi principali d'inerzia, A, B e C. In questo paragrafo si affronta quindi il problema del calcolo del comando da fornire alle ruote di reazione al fine di generare effettivamente le rotazioni calcolate precedentemente.

La Power Management tile del satellite AraMiS, come annunciato nel paragrafo 10.1 precedente, è fornita di un'opportuna ruota di reazione pilotata da un apposito motore elettrico. La ruota è montata in modo che il suo asse di rotazione risulti coincidente con l'asse Z della tile. Quando questa è messa in movimento è generato del momento angolare dettato dal prodotto della massa della ruota per la sua velocità angolare attorno all'asse Z della tile in questione. Siccome il momento angolare deve essere conservato, la rotazione della ruota di reazione deve essere compensata da una rotazione, in senso contrario, del satellite, non essendoci ulteriori vincoli nello spazio. L'obiettivo in esame risulta essere quello di generare delle rotazioni del satellite e, quindi, i momenti angolari che saranno calcolati in questo ambito (i comandi) saranno forniti alle ruote di reazione preoccupandosi di invertire opportunamente il segno degli stessi. Nella discussione seguente si fa l'ipotesi che le ruote di reazione siano pilotate in modo che esse ruotino a velocità costante ed abbiano un'accelerazione infinita, ovvero che siano in grado di portarsi da velocità nulla a velocità arbitraria costante in un tempo infinitesimo.

A partire dagli angoli di Eulero,  $\Psi$ ,  $\theta$  e  $\Phi$ , ricavati nel paragrafo precedente, è possibile ricavare i tre vettori delle velocità angolari relativi alle singole rotazioni attorno agli assi principali ABC del satellite. In altre parole, risulta possibile invertire le equazioni del moto del satellite e prevedere quanto comando generare al fine di ottenere le rotazioni desiderate (controllo ad anello aperto). Si riporta, per comodità, la definizione del momento angolare rispetto al sistema degli assi principali ABC, nella relazione seguente:

$$\underline{\underline{\vec{L}}}_O = \underline{\underline{I}}' \cdot \underline{\underline{\vec{\omega}}}_{abc} \quad (10.26)$$

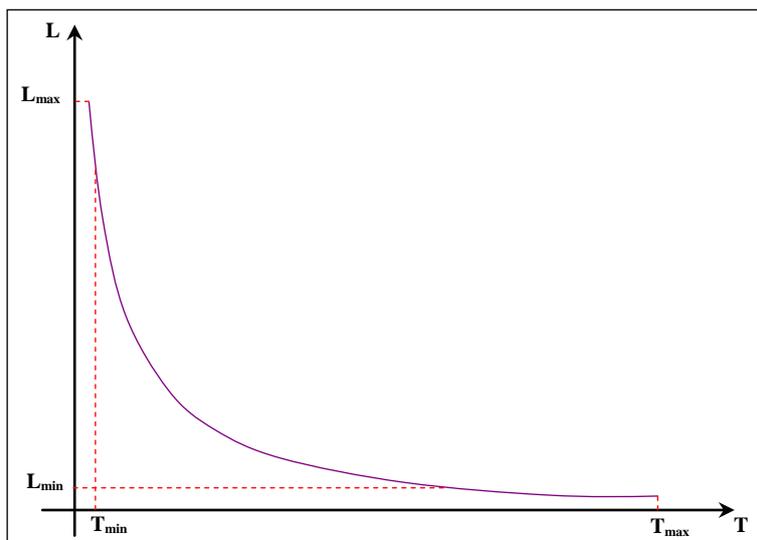
Dove  $\underline{\underline{I}}'$  rappresenta la matrice diagonale contenente gli autovalori della matrice d'inerzia definita rispetto agli assi UVW.

Si noti che il polo di riferimento risulta essere coincidente con l'origine del sistema di riferimento ABC e che, di fatto, è coincidente con l'origine del sistema UVW, dal momento che gli assi ABC sono semplicemente ruotati rispetto agli assi UVW. A questo punto è evidente che risulta possibile ricavare le componenti del vettore della velocità angolare fissando un tempo arbitrario in cui la rotazione in questione deve essere eseguita. Conoscendo  $\Psi$ ,  $\theta$ ,  $\Phi$ , ovvero le singole rotazioni da compiere sul sistema ABC e fissando il tempo desiderato per eseguire le tre rotazioni si ricavano i tre vettori delle velocità angolari, indicati con i simboli  $\underline{\underline{\vec{\omega}}}_c$ ,  $\underline{\underline{\vec{\omega}}}_b$  e  $\underline{\underline{\vec{\omega}}}_a$ :

$$\begin{cases} \underline{\underline{\vec{\omega}}}_c = \begin{pmatrix} 0 & 0 & \frac{\Psi}{t_3} \\ 0 & \frac{\theta}{t_2} & 0 \\ \frac{\Phi}{t_1} & 0 & 0 \end{pmatrix} \\ \underline{\underline{\vec{\omega}}}_b = \begin{pmatrix} 0 & \frac{\theta}{t_2} & 0 \\ 0 & \frac{\theta}{t_2} & 0 \\ \frac{\Phi}{t_1} & 0 & 0 \end{pmatrix} \\ \underline{\underline{\vec{\omega}}}_a = \begin{pmatrix} 0 & 0 & \frac{\Psi}{t_3} \\ 0 & \frac{\theta}{t_2} & 0 \\ \frac{\Phi}{t_1} & 0 & 0 \end{pmatrix} \end{cases} \quad (10.27)$$

Dove  $t_3, t_2, t_1$ , rappresentano i tempi fissati per eseguire le tre rotazioni attorno agli assi C, B ed A rispettivamente. Note le velocità angolari è possibile ricavare i rispettivi momenti angolari, moltiplicandole per l'opportuno autovalore contenuto nella matrice  $\underline{I}'$ .

Risulta tuttavia possibile adottare un approccio differente. E', infatti, possibile fissare un valore arbitrario di momento angolare e ricavare di conseguenza i vettori delle velocità angolari. Esprimendo il momento angolare in funzione del tempo si ottiene chiaramente un grafico di tipo iperbolico, dal momento che i momenti d'inerzia non dipendono dal tempo e la velocità angolare risulta essere inversamente proporzionale ad esso. Si riporta tale grafico nella figura seguente:



**Figura 10.3 Momento angolare in funzione del tempo.**

Nella figura precedente è stato evidenziato il momento angolare massimo e minimo ammissibile ed il tempo massimo e minimo della manovra. I valori massimi e minimi del momento angolare risultano essere fissati dalle caratteristiche delle ruote di reazione impiegate. D'altra parte anche il tempo minimo risulta essere fissato dalle caratteristiche delle ruote. In precedenza è stata fatta l'ipotesi di accelerazione infinita delle ruote, questo ovviamente non è ottenibile nella realtà. E' possibile quindi fissare un tempo minimo di attivazione delle ruote. Questo vincolo se non è soddisfatto porta all'impossibilità d'esecuzione della manovra. Il tempo massimo, infine, è specificato nel formato del comando, come dichiarato in precedenza, e rappresenta il tempo in cui l'intera manovra (somma delle tre singole rotazioni successive) deve essere eseguita. La somma dei tempi delle singole rotazioni deve quindi risultare inferiore, o al limite uguale, a questa grandezza di comando.

E' stato quindi deciso di fissare un momento angolare di default, pari al 15% in meno del valore massimo che una singola ruota di reazione può generare. La singola rotazione sarà quindi eseguita utilizzando questo valore di momento angolare e calcolando di conseguenza il tempo necessario. Il procedimento si ripete per tutte e tre le singole rotazioni della sequenza calcolata. Durante il calcolo del tempo relativo alle singole rotazioni potrebbe accadere di violare i vincoli imposti sullo stesso. In questo caso si dovrà fissare un tempo consentito e ricavare un nuovo momento angolare. Si può procedere nel seguente modo (approccio utilizzato nell'algoritmo sviluppato):

- Si fissa il momento angolare di default pari al 15% in meno del valore massimo ammissibile.
- Si calcolano i 3 vettori della velocità angolare relativi alle tre manovre.

- Si calcolano i rispettivi tempi necessari alle 3 rotazioni successive.
- Nel caso in cui un tempo risulti essere minore di  $T_{\min}$  è possibile fissare  $T_{\min}$  come tempo della rotazione in questione e ricavare un nuovo valore di momento angolare. A questo punto, se il valore di momento ricavato risulta essere inferiore a  $L_{\min}$  la rotazione in questione non potrà essere eseguita e si invia un messaggio di errore.
- Nel caso in cui la somma dei tre tempi risulta essere superiore a  $T_{\max}$  è possibile effettuare nuovamente tutti i calcoli precedenti partendo da un valore di momento angolare pari a  $L_{\max}$ . Nel caso in cui al termine dell'operazione il vincolo sul tempo massimo della manovra non risulti ancora soddisfatto, la manovra in questione non potrà essere eseguita e si invia un messaggio di errore.

Questo algoritmo permette di ricavare le velocità angolari definite nel sistema 10.27 precedente. Si noti che le velocità angolari così ricavate sono relative agli assi principali d'inerzia ABC. Questi, nel caso generale, non possiedono nessuna ruota di reazione avente l'asse Z parallelo ad uno di essi. In altre parole non risulta possibile generare un momento angolare sugli assi principali in modo diretto, dal momento che essi risultano ruotati rispetto agli assi del satellite UVW, nel caso generale. Risulta quindi necessario effettuare la scomposizione della velocità angolare trovata, oppure del momento angolare in modo indifferente, sugli assi in cui effettivamente sono presenti le ruote di reazione. Si noti che, nel caso in cui fosse presente almeno una ruota di reazione su ogni asse principale d'inerzia, le grandezze appena calcolate risultano sufficienti ad effettuare la manovra desiderata. La disposizione delle ruote di reazione rispetto alla struttura del satellite è funzione della particolare configurazione dello stesso. Si noti che la presenza di più di una ruota di reazione su un asse fornisce la possibilità di generare un momento angolare più elevato attorno allo stesso, dato dalla somma dei contributi delle singole ruote di reazione presenti. Si riporta in figura 10.4 un esempio di configurazione cubica del satellite AraMiS in cui sono chiaramente visibili numerose ruote di reazione disposte sugli assi UVW:

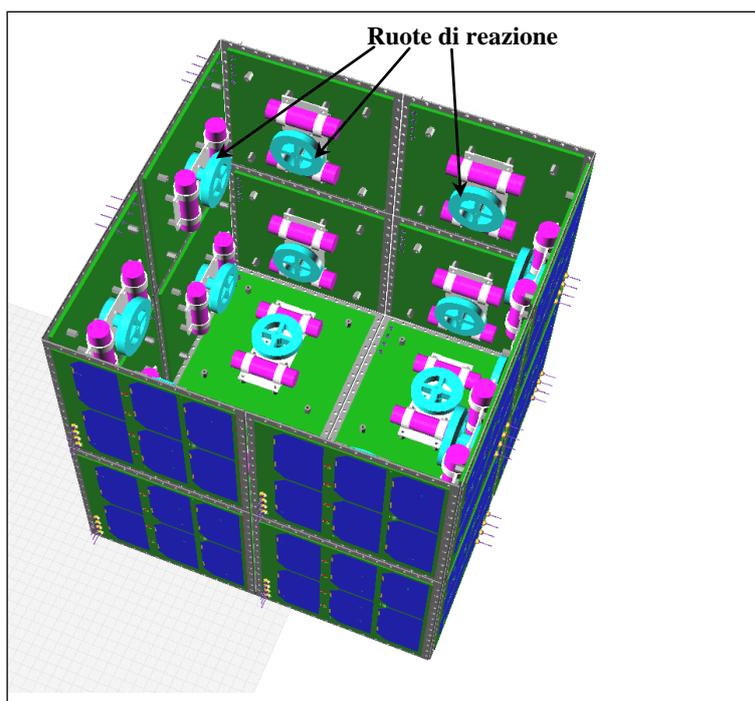


Figura 10.4 Satellite con numerose ruote di reazione.

Le strutture realmente ottenibili, combinando insieme un certo numero di tiles, sono sostanzialmente di tre tipi, ovvero le strutture cubiche, le strutture a sezione ottagonale e le strutture a sezione esagonale (si veda il capitolo 6). Per le prime due il procedimento risulta più semplice, in quanto ogni asse geometrico del satellite possiede almeno una ruota di reazione. Per la struttura esagonale, invece, esiste un asse geometrico sprovvisto di ruote di reazione. E' stato quindi deciso di sviluppare due algoritmi a hoc, in grado di gestire separatamente i due casi.

Si consideri il primo caso, più semplice, ovvero la configurazione cubica oppure ottagonale. In questa particolare struttura del satellite ogni asse UVW dello stesso è fornito di almeno una ruota di reazione. Il valore massimo del momento angolare ottenibile attorno ad un asse geometrico è dettato dalla somma dei contributi delle singole tiles aventi la ruota di reazione ed aventi l'asse Z parallelo all'asse geometrico in questione. L'algoritmo di calcolo del comando, ovvero dei momenti angolari, è stato implementato in modo che preliminarmente fossero calcolate le ruote disponibili attorno agli assi geometrici, al fine di contemplare la possibilità di guasti su alcune ruote di reazione. In questo modo il valore di momento angolare di default, citato in precedenza, è moltiplicato per il numero di ruote di reazione presenti, e funzionanti, attorno all'asse in questione. Si tenga presente che sono prese in considerazione sia le tiles aventi il verso dell'asse Z coincidenti con il versore diretto secondo il particolare asse in esame, sia quelle aventi verso opposto. In una struttura a piccolo cubo, per esempio, sull'asse U contribuiscono due ruote di reazione. La prima,  $a$ , avente asse Z orientato secondo l'asse U, la seconda,  $b$ , orientata secondo l'asse  $-U$ . Volendo quindi ottenere una rotazione positiva (in senso antiorario) del satellite a configurazione di piccolo cubo attorno all'asse U è possibile generare un momento angolare di default negativo mediante una rotazione oraria della ruota  $a$  ed una rotazione antioraria della ruota  $b$  in contemporanea. Il risultato risulta pertanto essere un momento angolare doppio (a patto di generare esattamente lo stesso momento angolare con le due ruote) attorno all'asse U.

Il primo calcolo effettuabile risulta essere la determinazione dei tre vettori delle velocità angolari corrispondenti sugli assi geometrici, indicati con i simboli  $\vec{\omega}_{1uvw}$ ,  $\vec{\omega}_{2uvw}$ ,  $\vec{\omega}_{3uvw}$ , a partire dai vettori delle velocità angolari ricavate sugli assi principali d'inerzia, sfruttando la matrice contenente gli autovettori della matrice d'inerzia,  $\underline{\underline{E}}$ , nel modo seguente:

$$\begin{cases} \vec{\omega}_{1uvw} = \underline{\underline{E}}^T \cdot \vec{\omega}_c \\ \vec{\omega}_{2uvw} = \underline{\underline{E}}^T \cdot \vec{\omega}_b \\ \vec{\omega}_{3uvw} = \underline{\underline{E}}^T \cdot \vec{\omega}_a \end{cases} \quad (10.28)$$

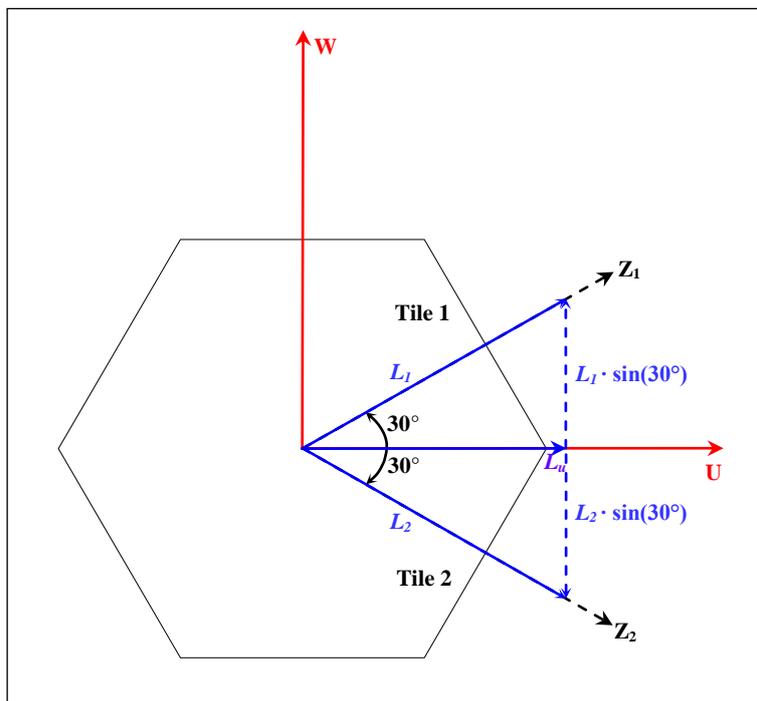
E' possibile dunque ricavare i tre vettori del momento angolare corrispondenti, indicati con  $\vec{L}_{1uvw}$ ,  $\vec{L}_{2uvw}$ ,  $\vec{L}_{3uvw}$ , mediante le relazioni seguenti:

$$\begin{cases} \vec{L}_{1uvw} = \underline{\underline{I}} \cdot \vec{\omega}_{1uvw} \\ \vec{L}_{2uvw} = \underline{\underline{I}} \cdot \vec{\omega}_{2uvw} \\ \vec{L}_{3uvw} = \underline{\underline{I}} \cdot \vec{\omega}_{3uvw} \end{cases} \quad (10.29)$$

A questo punto, conoscendo i tre tempi necessari alle tre singole rotazioni ed i tre vettori del momento angolare, il comando risulta essere completamente determinato. E' sufficiente, infatti, fornire tali grandezze alle varie ruote di reazione presenti sugli assi geometrici UVW.

Si consideri ora il caso della configurazione esagonale del satellite. In questa configurazione è possibile osservare che un asse geometrico non è provvisto di ruote di reazione. Considerando il sistema di riferimento UVW definito sul satellite esagonale (si veda il capitolo 6) è semplice notare

che l'asse in questione risulta essere quello indicato dal simbolo U. Si riporta nella figura seguente la sezione del satellite esagonale al fine di evidenziare la situazione in esame:



**Figura 10.5 Mancanza di ruote di reazione su un asse geometrico.**

L'asse V del satellite non è stato rappresentato in quanto entrante nel piano della figura precedente. Nella figura sono stati rappresentati i due spostamenti angolari compresi tra l'asse U del satellite e gli assi Z delle tiles adiacenti all'asse U stesso. Essendo l'esagono in esame regolare è semplice convincersi del fatto che tali angoli siano entrambi pari a 30°. Questa situazione non complica eccessivamente l'algoritmo di calcolo del comando da fornire alle ruote di reazione. Si supponga, infatti, di avere due ruote di reazione con asse coincidente con le due direzioni dettate dagli assi Z<sub>1</sub> e Z<sub>2</sub>, delle tiles 1 e 2, riportate in figura. Generando lo stesso momento angolare mediante entrambe le ruote si genera un momento angolare equivalente attorno all'asse U. Volendo generare un certo momento angolare attorno all'asse U, indicato con  $\vec{L}_u$ , è possibile generare metà momento angolare mediante la ruota posizionata sulla tile 1 e metà mediante quella posizionata sulla tile 2.

Si consideri il caso in cui si desideri generare un momento attorno all'asse U,  $\vec{L}_u$ , pari ad 1. Si noti che in figura è stato riportato il modulo di  $\vec{L}_u$ , ovvero  $L_u$ . Come primo calcolo si divida a metà il momento angolare che si vuole generare, ovvero  $L_u / 2 = 0.5$ . A questo punto è possibile ricavare i momenti angolari nelle direzioni fissate dagli assi Z<sub>1</sub> e Z<sub>2</sub>, che, data la particolare geometria simmetrica, risultano essere identici:

$$L_1 = L_2 = \frac{(L_u / 2)}{\cos(30)} = \frac{0.5}{\cos(30)} \quad (10.30)$$

Queste due grandezze risultano pertanto essere i momenti angolari che le ruote di reazione presenti sulle tiles 1 e 2 devono generare al fine di ottenere come risultante un momento angolare sull'asse U pari a  $L_u$ . Per dimostrare quest'affermazione si sommino le componenti in U e W di  $L_1$  e  $L_2$ :

$$\begin{aligned}
 L_u &= (L_{1u} + L_{1w}) + (L_{2u} + L_{2v}) = (0.5 + 0.5) \cdot \hat{u} + (\sin(30) \cdot L_1 - \sin(30) \cdot L_2) \cdot \hat{v} = \\
 &= 1 \cdot \hat{u} + (\sin(30) \cdot L_1 - \sin(30) \cdot L_1) \cdot \hat{v} = 1 \cdot \hat{u}
 \end{aligned}
 \tag{10.31}$$

La relazione precedente conferma quindi la correttezza del metodo utilizzato.

Si riporta, in figura 10.6, il diagramma di flusso di massima dell'algoritmo di controllo dell'assetto sviluppato, al fine di evidenziarne i punti principali:

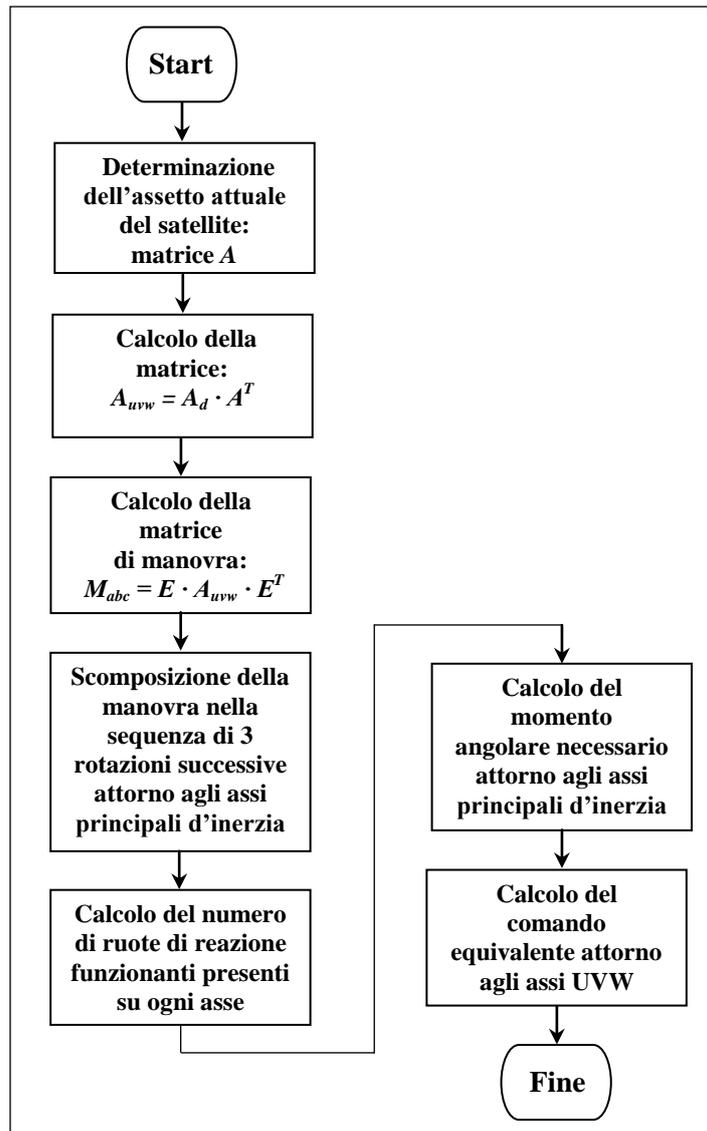


Figura 10.6 Diagramma di flusso dell'algoritmo di controllo dell'assetto.

### 10.3 La simulazione dell'algoritmo mediante Simulink

In questo paragrafo si riporta una prova dell'algoritmo implementato sfruttando Matlab e l'ambiente Simulink ad esso associato. Simulink è un programma indirizzato alla simulazione, alla modellazione ed all'analisi di sistemi dinamici descritti nei modi più differenti, quali equazioni differenziali, funzioni di trasferimento, sistemi in forma di stato. La descrizione del sistema che si vuole analizzare avviene mediante l'utilizzo degli schemi a blocchi. Simulink contiene una libreria di blocchi che descrivono elementi algebrici e dinamici elementari. Inoltre fornisce una serie di blocchi indirizzati alla descrizione di particolari sistemi appartenenti ad un particolare contesto. Per esempio è disponibile la libreria contenente i blocchi relativi all'ambiente aerospaziale. Questa libreria è stata utilizzata durante la simulazione dell'algoritmo in questione. L'utente può descrivere il proprio sistema dinamico mediante l'interconnessione dei vari blocchi elementari, o dei blocchi già disponibili in Simulink. A questo punto risulta possibile effettuare la simulazione del sistema mediante l'integrazione numerica delle equazioni differenziali, in un certo intervallo di tempo.

La libreria aerospaziale contiene il blocco 6DoF (Six Degrees of Freedom). Questo blocco si occupa di integrare le equazioni del moto in sei gradi di libertà ed è stato utilizzato per simulare la dinamica rotatoria del satellite. Inoltre, nella stessa libreria, sono disponibili altri blocchi indirizzati alle trasformazioni tra i vari parametri d'assetto, quali matrici dei coseni direttori, angoli di Eulero e quaternioni unitari. Il blocco 6DoF possiede due ingressi di comando. Il primo risulta essere un vettore contenente le tre componenti della forza applicata al satellite rispetto al suo sistema di riferimento UVW. Questo primo ingresso non è stato considerato, dal momento che il satellite AraMiS non possiede motori finalizzati al cambio d'orbita, ovvero al moto di traslazione. Il secondo ingresso invece risulta essere un vettore contenente le componenti dei momenti delle forze applicate al satellite, rispetto al suo sistema di riferimento UVW. Questo ingresso di comando è stato utilizzato per inserire i valori dei momenti calcolati con l'algoritmo sviluppato. Si noti che nell'algoritmo sviluppato sono stati calcolati i vettori contenenti le componenti dei momenti angolari che le ruote di reazione devono generare, mentre nella simulazione di Simulink sono stati calcolati i momenti delle forze relativi, ovvero le derivate rispetto al tempo dei momenti angolari ottenuti dall'algoritmo (rappresentate dalle funzioni delta di Dirac, dal momento che i momenti angolari sono delle costanti). I risultati ottenuti sono stati visualizzati sfruttando sostanzialmente tre delle otto uscite fornite dal blocco 6DoF, ovvero l'uscita contenente gli angoli di Eulero del satellite (orientamento degli assi del satellite UVW rispetto al sistema geocentrico inerziale IJK), l'uscita contenente il vettore delle velocità angolari espresse nel sistema di riferimento del satellite UVW e l'uscita contenente la matrice dei coseni direttori degli assi UVW rispetto al sistema inerziale IJK. Inoltre, il blocco 6DoF permette di fissare il tensore d'inerzia del satellite in esame e l'assetto iniziale dello stesso.

Con lo scopo di effettuare una prova dell'algoritmo implementato è stato fissato un tensore d'inerzia di prova del satellite (espresso in  $[Kg \cdot m^2]$ ), definito relativo agli assi geometrici UVW, nel modo seguente:

$$\underline{\underline{I}} = \begin{bmatrix} 0.007252141800000 & -0.000011600796000 & 0.000002868117000 \\ -0.000011600796000 & 0.002740675100000 & -0.000056058867000 \\ 0.000002868117000 & -0.000056058867000 & 0.007292028300000 \end{bmatrix} \quad (10.32)$$

A questo punto è stato risolto il problema agli autovalori associato alla matrice  $\underline{\underline{I}}$ . Si riportano di seguito la matrice  $\underline{\underline{E}}$ , contenente gli autovettori di  $\underline{\underline{I}}$  (ovvero i coseni direttori degli assi principali

d'inerzia ABC rispetto agli assi del satellite UVW), e la matrice diagonale  $\underline{\underline{I}}'$ , contenente gli autovalori di  $\underline{\underline{I}}$ :

$$\underline{\underline{E}} = \begin{bmatrix} 0.997281076129401 & -0.001649028612697 & -0.073673169463634 \\ 0.002562962438925 & 0.999920914518460 & 0.012312429983664 \\ 0.073647039436216 & -0.012467774989968 & 0.997206432073661 \end{bmatrix} \quad (10.33)$$

$$\underline{\underline{I}}' = \begin{bmatrix} 0.007251949102846 & 0 & 0 \\ 0 & 0.002739955089778 & 0 \\ 0 & 0 & 0.007292941007376 \end{bmatrix} \quad (10.34)$$

Si noti che risulta importante l'ordine con cui gli autovettori sono inseriti nella matrice  $\underline{\underline{E}}$ . Un ordinamento non corretto porta all'ottenimento di risultati completamente errati. Effettuando la scomposizione agli autovalori sfruttando la funzione eig() presente nelle librerie di Matlab, per esempio, l'ordine degli autovettori non è fornito correttamente ed inoltre, tale funzione, fornisce gli autovettori disposti in colonna e non in riga. Al fine di verificare la correttezza della matrice  $\underline{\underline{E}}$  calcolata è possibile effettuare l'operazione di prodotto vettoriale tra la prima e la seconda riga della stessa e verificare, infine, che il vettore ottenuto sia identico all'ultima riga della stessa matrice.

E' stata fissata una matrice di comando di prova, indicata con  $\underline{\underline{A}}_d$ , nel seguente modo (durante questa prova dell'algorithm non è stato imposto il vincolo sul tempo in cui la manovra in questione deve essere eseguita):

$$\underline{\underline{A}}_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (10.35)$$

Questa matrice di comando, si ricorda, rappresenta l'orientamento desiderato degli assi del satellite UVW rispetto al sistema geocentrico inerziale IJK. Si noti che la matrice di comando 10.35 equivale ad una rotazione di 90° attorno all'asse inerziale I. Inoltre è stato fissato l'assetto iniziale del satellite, rappresentato dalla matrice  $\underline{\underline{A}}$ , in modo che coincidesse esattamente con gli assi inerziali IJK, ovvero nel seguente modo:

$$\underline{\underline{A}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10.36)$$

A questo punto risulta possibile ricavare la matrice  $\underline{\underline{A}}_{uvw}$  mediante la relazione 10.37:

$$\underline{\underline{A}}_{uvw} = \underline{\underline{A}}_d \cdot \underline{\underline{A}}^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (10.37)$$

Quindi risulta possibile calcolare la matrice di manovra,  $\underline{\underline{M}}_{abc}$ , sfruttando la relazione 10.38, nel modo seguente:

$$\underline{\underline{M}}_{abc} = \underline{\underline{E}} \cdot \underline{\underline{A}}_{uvw} \cdot \underline{\underline{E}}^T = \begin{bmatrix} 0.994569544805816 & 0.076203033375386 & 0.070883836303769 \\ -0.071091045497045 & 0.000006568776463 & 0.997469830725215 \\ 0.076009761181619 & -0.997092321533589 & 0.005423886417720 \end{bmatrix} \quad (10.38)$$

Utilizzando l'algoritmo di scomposizione della matrice di manovra nella sequenza di tre rotazioni successive, mediante la scomposizione dei quaternioni unitari associati alla matrice  $\underline{\underline{M}}_{abc}$ , sono state ricavate le tre rotazioni da compiere, indicate con i simboli  $\Psi$ ,  $\theta$ ,  $\Phi$ , attorno agli assi principali C, B ed A rispettivamente:

$$\begin{cases} \Psi = 4.3813913795535^\circ \\ \theta = -4.0647534189236^\circ \\ \Phi = 89.688448985754^\circ \end{cases} \quad (10.39)$$

Ovviamente i valori ottenuti risultano essere composti di numeri con molte cifre dopo la virgola e, quindi, ottenere un risultato perfetto risulta impossibile, a causa degli errori di troncamento numerico. Si verifichino quindi le tre rotazioni calcolate mediante il prodotto delle matrici di rotazioni associate a tali angoli di rotazione, indicate con i simboli  $\underline{\underline{A}}_3$ ,  $\underline{\underline{A}}_2$  ed  $\underline{\underline{A}}_1$  rispettivamente:

$$\underline{\underline{A}}_3 = \begin{bmatrix} \cos(4.381391379553500) & \sin(4.381391379553500) & 0 \\ -\sin(4.381391379553500) & \cos(4.381391379553500) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10.40)$$

$$\underline{\underline{A}}_2 = \begin{bmatrix} \cos(-4.064753418923600) & 0 & -\sin(-4.064753418923600) \\ 0 & 1 & 0 \\ \sin(-4.064753418923600) & 0 & \cos(-4.064753418923600) \end{bmatrix} \quad (10.41)$$

$$\underline{\underline{A}}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(89.688448985753993) & \sin(89.688448985753993) \\ 0 & -\sin(89.688448985753993) & \cos(89.688448985753993) \end{bmatrix} \quad (10.42)$$

Si esegua quindi il prodotto delle tre matrici, ricordando che le matrici dei coseni direttori devono essere inserite nell'ordine inverso, come riportato nella relazione seguente:

$$\underline{\underline{A}}_1 \cdot \underline{\underline{A}}_2 \cdot \underline{\underline{A}}_3 = \begin{bmatrix} 0.994569544805817 & 0.076203033375386 & 0.070883836303768 \\ -0.071091045497045 & 0.000006568776459 & 0.997469830725215 \\ 0.076009761181619 & -0.997092321533588 & 0.005423886417715 \end{bmatrix} \quad (10.43)$$

Si verifichi quindi l'errore della matrice appena ricavata rispetto alla matrice di manovra desiderata,  $\underline{M}_{abc}$ , facendo la differenza seguente:

$$\begin{aligned} & (\underline{A}_1 \cdot \underline{A}_2 \cdot \underline{A}_3) - \underline{M}_{abc} = \\ & = \begin{bmatrix} 0.122124532708767 \cdot 10^{-14} & -0.008326672684689 \cdot 10^{-14} & -0.027755575615629 \cdot 10^{-14} \\ 0.048572257327351 \cdot 10^{-14} & -0.440879971419506 \cdot 10^{-14} & -0.022204460492503 \cdot 10^{-14} \\ -0.058286708792821 \cdot 10^{-14} & 0.022204460492503 \cdot 10^{-14} & -0.450420950537378 \cdot 10^{-14} \end{bmatrix} \end{aligned} \quad (10.44)$$

Come si può facilmente notare, l'errore risulta essere decisamente piccolo, soltanto dovuto ai problemi di arrotondamento numerico durante i vari computi.

A questo punto è stato fissato, per comodità, un tempo per eseguire le tre rotazioni successive, pari a 5 secondi. Si noti che in sede di simulazione è stato fissato il tempo, mentre nell'implementazione dell'algoritmo è stato fissato il momento angolare, come descritto in precedenza. I tre vettori delle velocità angolari attorno agli assi principali ABC risultano pertanto essere pari a:

$$\begin{cases} \vec{\omega}_c = (0 \quad 0 \quad (4.3813913795535 \cdot \pi/180)/5) = (0 \quad 0 \quad 0.015293941078341) \\ \vec{\omega}_b = (0 \quad -4.0647534189236 \cdot \pi/180)/5 \quad 0) = (0 \quad -0.014188666088383 \quad 0) \\ \vec{\omega}_a = ((89.688448985754 \cdot \pi/180)/5 \quad 0 \quad 0) = (0.313071747161675 \quad 0 \quad 0) \end{cases} \quad (10.45)$$

Ora è possibile specificare i tre vettori delle velocità angolari precedenti rispetto al sistema di riferimento del satellite UVW, mediante le relazioni seguenti:

$$\begin{cases} \vec{\omega}_{1uvw} = \underline{E}^T \cdot \vec{\omega}_c = (0.001126353481732 \quad -0.000190681416075 \quad 0.015251216415077) \\ \vec{\omega}_{2uvw} = \underline{E}^T \cdot \vec{\omega}_b = (-0.000036365018243 \quad -0.014187543970893 \quad -0.000174696957775) \\ \vec{\omega}_{3uvw} = \underline{E}^T \cdot \vec{\omega}_a = (0.312220528915107 \quad -0.000516264268897 \quad -0.023064987882918) \end{cases} \quad (10.46)$$

Si ricavano quindi i tre vettori dei momenti angolari rispetto agli assi UVW,  $\vec{L}_{1uvw}$ ,  $\vec{L}_{2uvw}$ ,  $\vec{L}_{3uvw}$ , sfruttando le relazioni 10.47 seguenti:

$$\begin{cases} \vec{L}_{1uvw} = \underline{I} \cdot \vec{\omega}_{1uvw} = \\ = (0.008214429495722 \cdot 10^{-3} \quad -0.001390628318635 \cdot 10^{-3} \quad 0.111226221605880 \cdot 10^{-3}) \\ \vec{L}_{2uvw} = \underline{I} \cdot \vec{\omega}_{2uvw} = \\ = (-0.000996385168247 \cdot 10^{-4} \quad -0.388732333145012 \cdot 10^{-4} \quad -0.004786618186238 \cdot 10^{-4}) \\ \vec{L}_{3uvw} = \underline{I} \cdot \vec{\omega}_{3uvw} = \\ = (0.002264207384556 \quad -0.000003743922202 \quad -0.000167266118185) \end{cases} \quad (10.47)$$

Si noti che questi valori di momento angolare risultano essere costanti nell'intervallo di manovra. Inoltre, per comodità di visualizzazione, i tre momenti angolari sono stati distanziati nel tempo di un intervallo pari a 5 secondi.

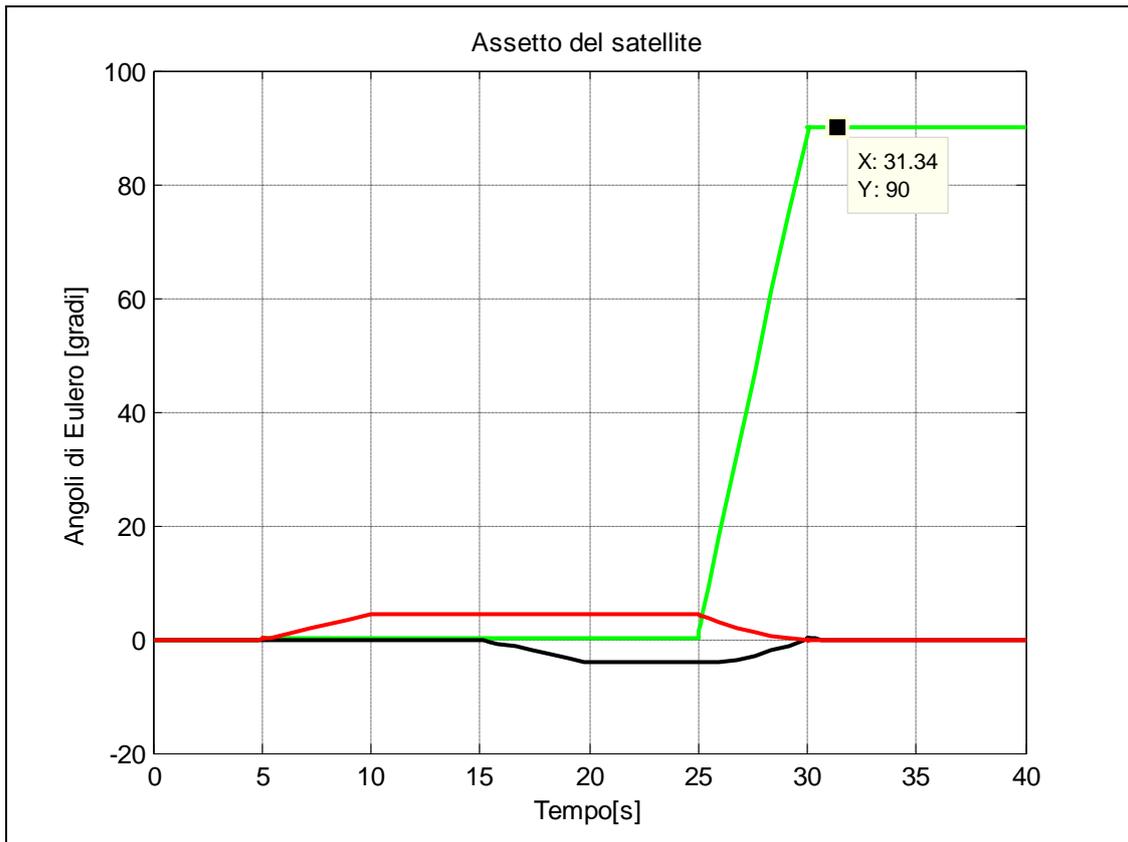
Come annunciato in precedenza, al fine di effettuare la simulazione, risulta necessario calcolare i momenti delle forze equivalenti, calcolando le derivate temporali dei momenti angolari appena ricavati. Ovviamente la derivata di una funzione a gradino risulta essere la funzione delta di Dirac. Una funzione del genere è caratterizzata dall'avere una base infinitesima ed un'altezza infinita con il vincolo che l'area sottesa alla funzione sia pari al valore del momento angolare appena calcolato. Si noti che per ogni intervallo di manovra si ottengono due momenti della forza, il primo si ottiene durante la transizione del momento angolare dal valore nullo al valore calcolato, il secondo (uguale, ma di segno opposto) si ottiene durante la transizione del momento angolare dal valore calcolato al valore nullo. I due momenti della forza possono essere interpretati come il comando di partenza ed il comando di arresto delle ruote di reazione e, quindi, delle rotazioni del satellite. Infatti, nel caso in cui il secondo momento non fosse applicato il satellite continuerebbe a ruotare sull'asse attorno al quale è stato applicato il momento di partenza. Evidentemente per effettuare una manovra completa sul modello Simulink si devono applicare in successione sei momenti, ovvero due per ogni momento angolare. Dal momento che una funzione delta di Dirac è soltanto un'astrazione matematica, non realizzabile fisicamente, è stato deciso di fissare la base della delta ad un valore pari a 0.0001 secondi. Così facendo si ottengono i seguenti risultati (si riportano soltanto i tre momenti della forza di partenza):

$$\begin{cases} \vec{M}_{1uvw} = \vec{L}_{1uvw} / 0.0001 = \\ = (0.082144294957220 \quad -0.013906283186348 \quad 1.112262216058800) \\ \vec{M}_{2uvw} = \vec{L}_{2uvw} / 0.0001 = \\ = (-9.963851682471834 \cdot 10^{-4} \quad -0.388732333145012 \quad -0.004786618186238) \\ \vec{M}_{3uvw} = \vec{L}_{3uvw} / 0.0001 = \\ = (22.642073845559715 \quad -0.037439222016559 \quad -1.672661181846784) \end{cases} \quad (10.48)$$

Dove  $\vec{M}_{1uvw}$  rappresenta il momento della forza applicato per effettuare la prima rotazione,  $\vec{M}_{2uvw}$  quello relativo alla seconda rotazione ed  $\vec{M}_{3uvw}$  quello relativo all'ultima rotazione.

Si riportano, in conclusione, i risultati ottenuti dalla simulazione. In particolare sono riportati i grafici relativi all'assetto (rappresentato mediante gli angoli di Eulero), ed alle velocità angolari del satellite.

Si riporta, nella figura 10.7 seguente, l'assetto del satellite rappresentato mediante gli angoli di Eulero (in rosso è rappresentata l'angolo della rotazione attorno all'asse W, in nero l'angolo della rotazione attorno all'asse V ed in verde l'angolo della rotazione attorno all'asse U):



**Figura 10.7 Assetto del satellite (angoli di Eulero).**

Nella figura precedente è possibile apprezzare le rotazioni effettuate dal satellite, rispetto al sistema geocentrico inerziale IJK. Si noti che al termine della manovra, ovvero dopo i 30 secondi nella figura precedente, l'assetto desiderato è stato correttamente raggiunto. L'effettiva manovra totale, infatti, risulta essere una rotazione di  $90^\circ$  attorno all'asse U del satellite, come risulta chiaramente visibile nella figura precedente.

Nella figura 10.8 seguente si riporta un ingrandimento della figura 10.7, effettuato in prossimità del punto in cui è applicato il momento di arresto del satellite, ovvero al termine della manovra (30 secondi):

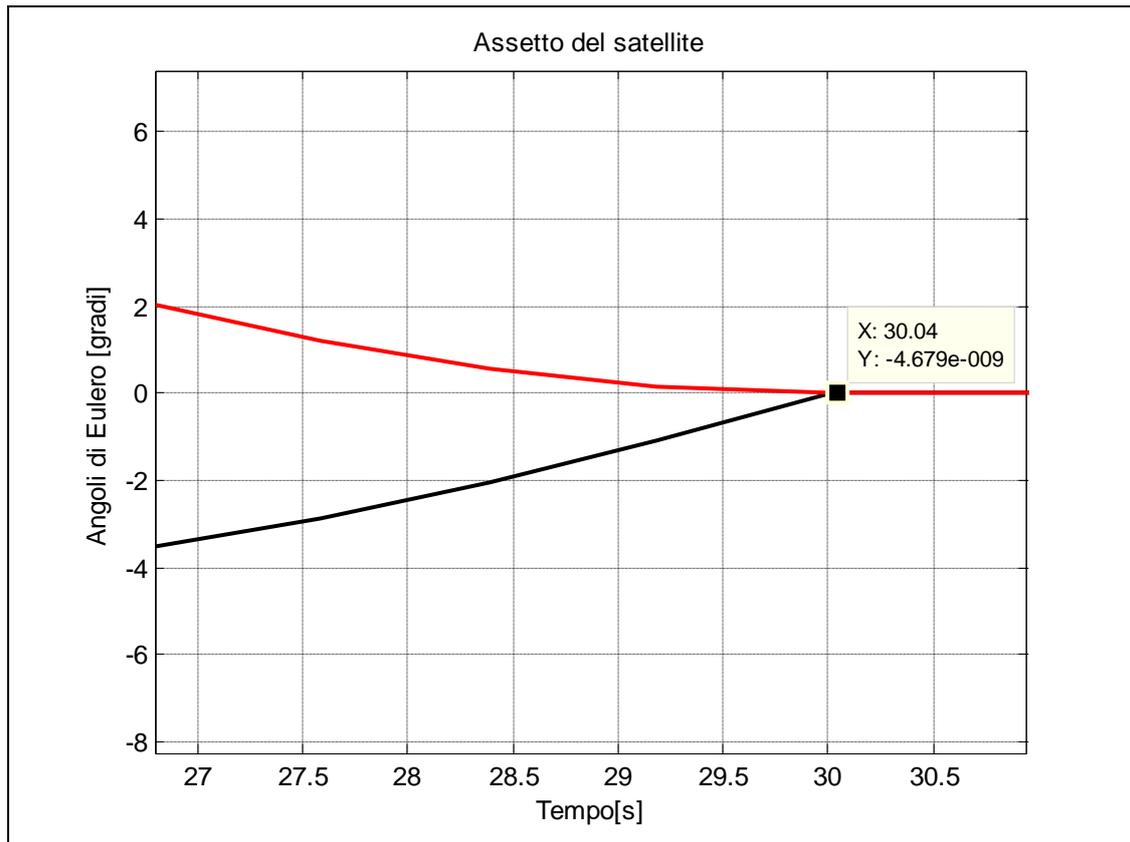


Figura 10.8 Zoom assetto del satellite (angoli di Eulero).

Si noti la presenza dell'errore residuo sugli assi V e W, causato dalla presenza dei problemi di troncamento numerico durante i vari computi. L'ordine di grandezza di questo errore risulta essere pari a  $10^{-9}$  e, quindi, è assolutamente trascurabile.

Nella figura 10.9 seguente si riporta il grafico delle velocità angolari del satellite relative agli assi principali d'inerzia ABC dello stesso (in rosso è rappresentata la componente della velocità angolare relativa all'asse principale C, in nero la componente della velocità angolare relativa all'asse principale B ed in verde la componente della velocità angolare relativa all'asse principale A):

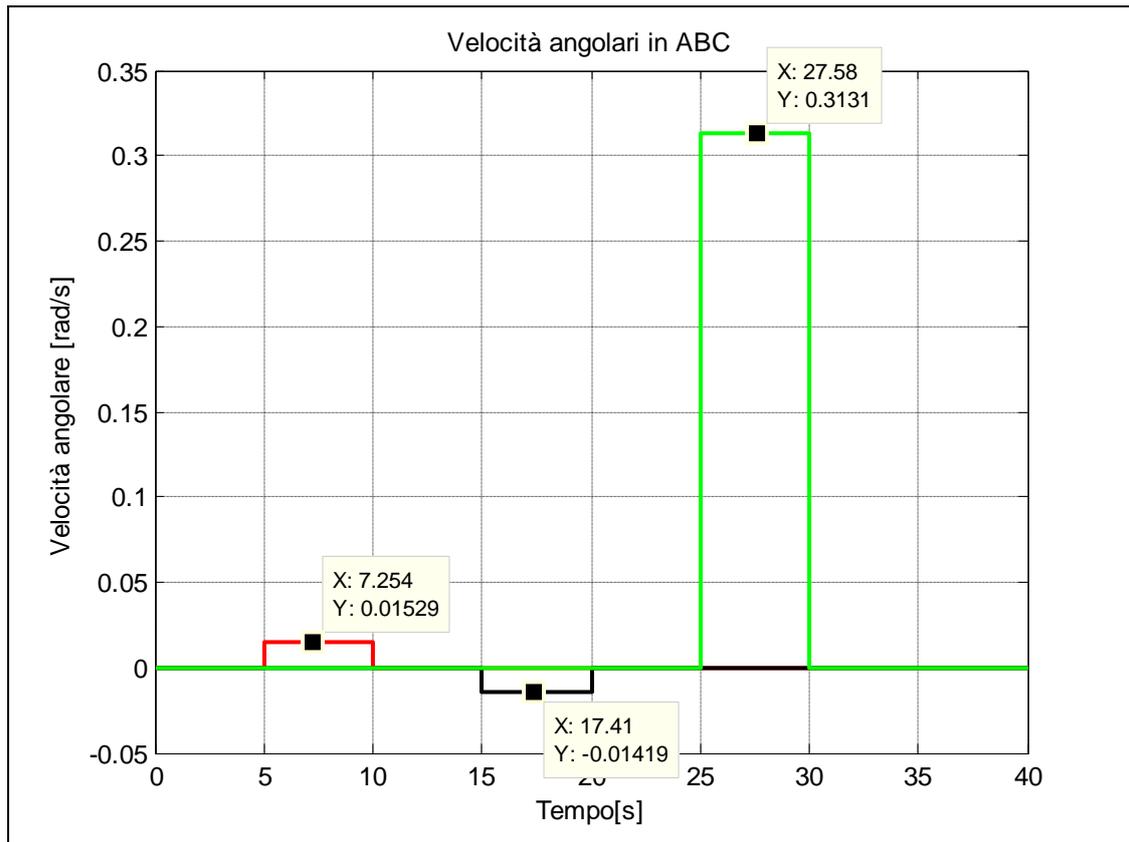
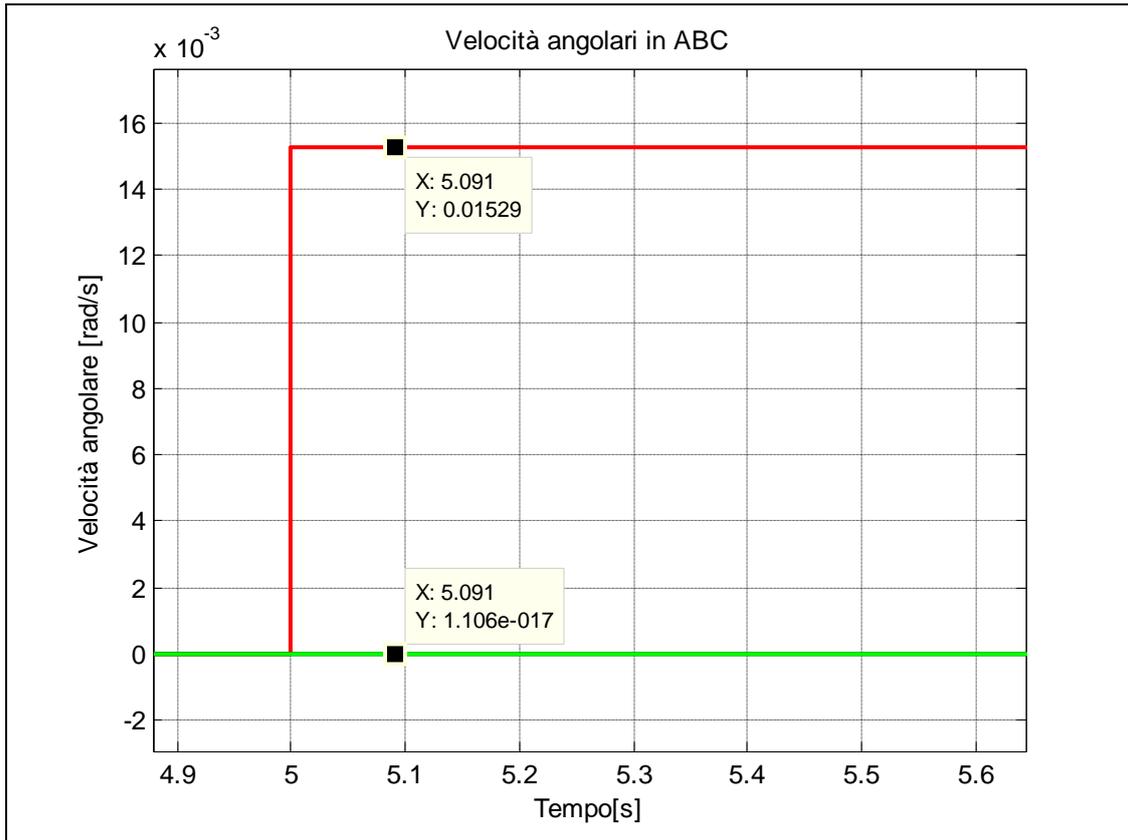


Figura 10.9 Velocità angolari del satellite relative agli assi ABC.

Le velocità angolari relative agli assi principali ABC risultano essere quelle effettivamente calcolate in precedenza. In particolare si noti che, durante le singole rotazioni (la prima è visualizzabile tra 5-10 secondi, la seconda tra 15-20, la terza tra 25-30 secondi), soltanto una componente della velocità angolare risulta essere non nulla e questo dimostra l'approccio prefissato in fase di sviluppo dell'algoritmo. Al fine di visualizzare meglio il risultato ottenuto si riporta nella figura 10.10 seguente un ingrandimento della velocità angolare relativa alla prima rotazione:

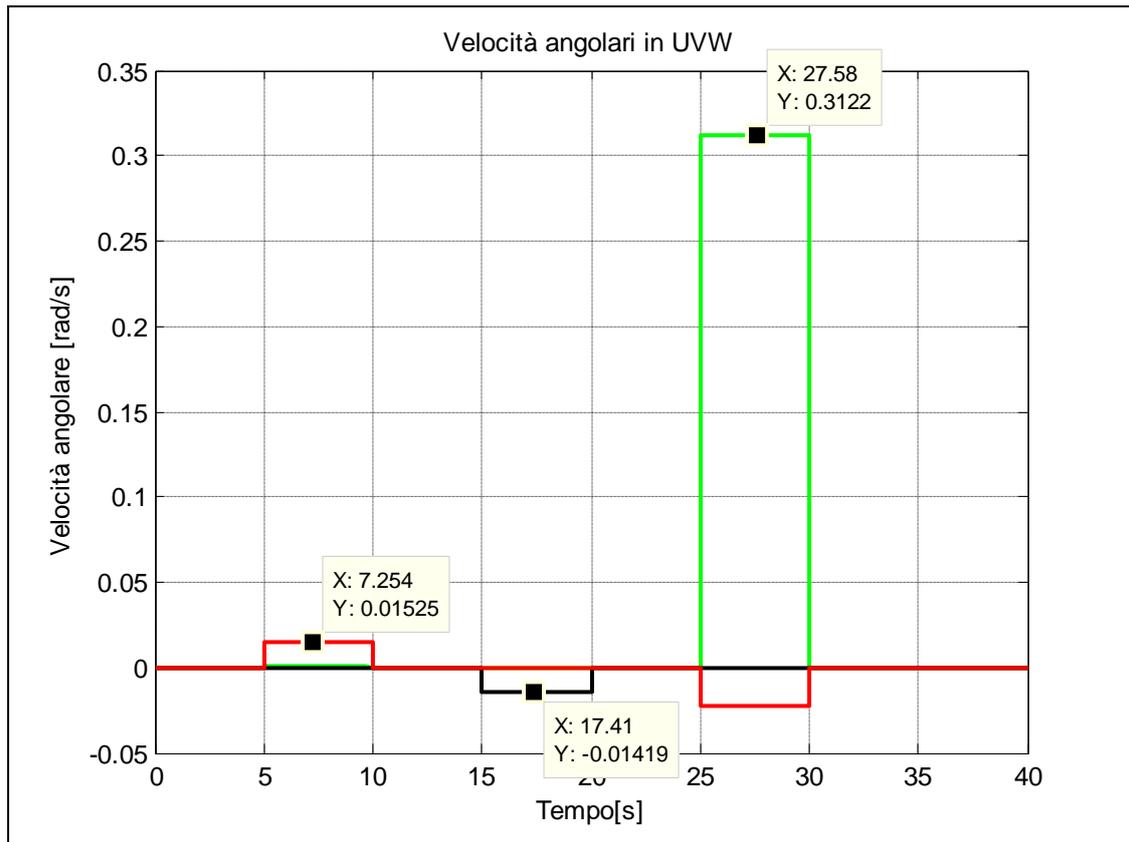


**Figura 10.10 Zoom velocità angolare relativa alla prima rotazione (asse principale).**

Nella figura precedente si vede chiaramente che, durante la prima rotazione, soltanto la componente della velocità angolare relativa all'asse principale C è diversa da zero. Le altre due componenti possiedono un errore dell'ordine di grandezza di  $10^{-17}$  rad / s, decisamente trascurabile e causato dai problemi di troncamento numerico.

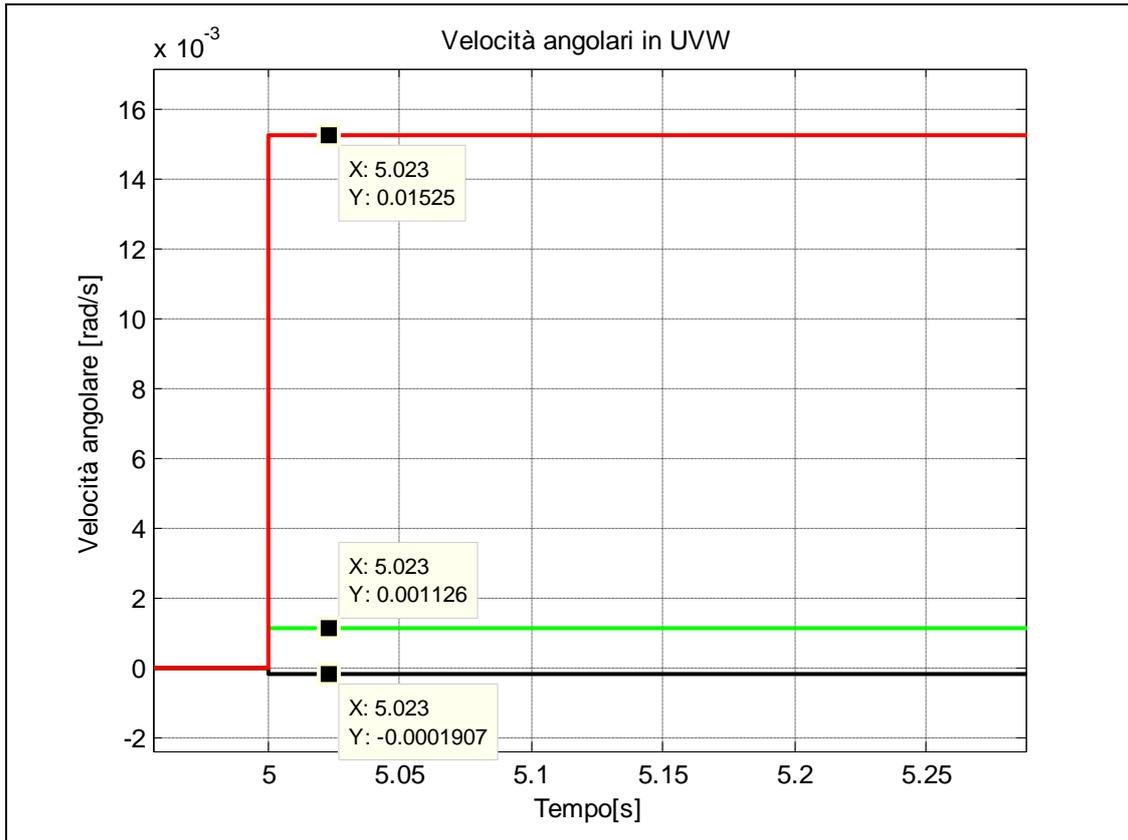
La soluzione adottata, si ricorda, consiste nel ruotare il satellite attorno ad un singolo asse principale alla volta e l'assetto finale è raggiunto in seguito alla sequenza di tre rotazioni singole attorno agli assi principali d'inerzia ABC.

Nella figura 10.11 seguente si riportano, infine, le velocità angolari del satellite relative agli assi UVW dello stesso (in rosso è rappresentata la componente della velocità angolare relativa all'asse W, in nero la componente della velocità angolare relativa all'asse V ed in verde la componente della velocità angolare relativa all'asse U):



**Figura 10.11** Velocità angolari del satellite rispetto agli assi UVW.

Le velocità angolari espresse rispetto agli assi del satellite UVW risultano essere quelle calcolate in precedenza. Si noti che nel grafico 10.11, a differenza del grafico 10.9 precedente, per ogni intervallo di tempo relativo ad una singola rotazione tutte e tre le componenti della velocità angolare risultano essere diverse da zero, come d'altronde risulta essere evidente. Si riporta, anche in questo caso, un ingrandimento della velocità angolare relativa alla prima rotazione:



**Figura 10.12 Zoom velocità angolare relativa alla prima rotazione (asse del satellite).**

In ultimo si riporta in figura 10.13 la descrizione della dinamica del satellite effettuata mediante lo schema a blocchi, sviluppato in ambiente Simulink, utilizzato durante la simulazione eseguita:

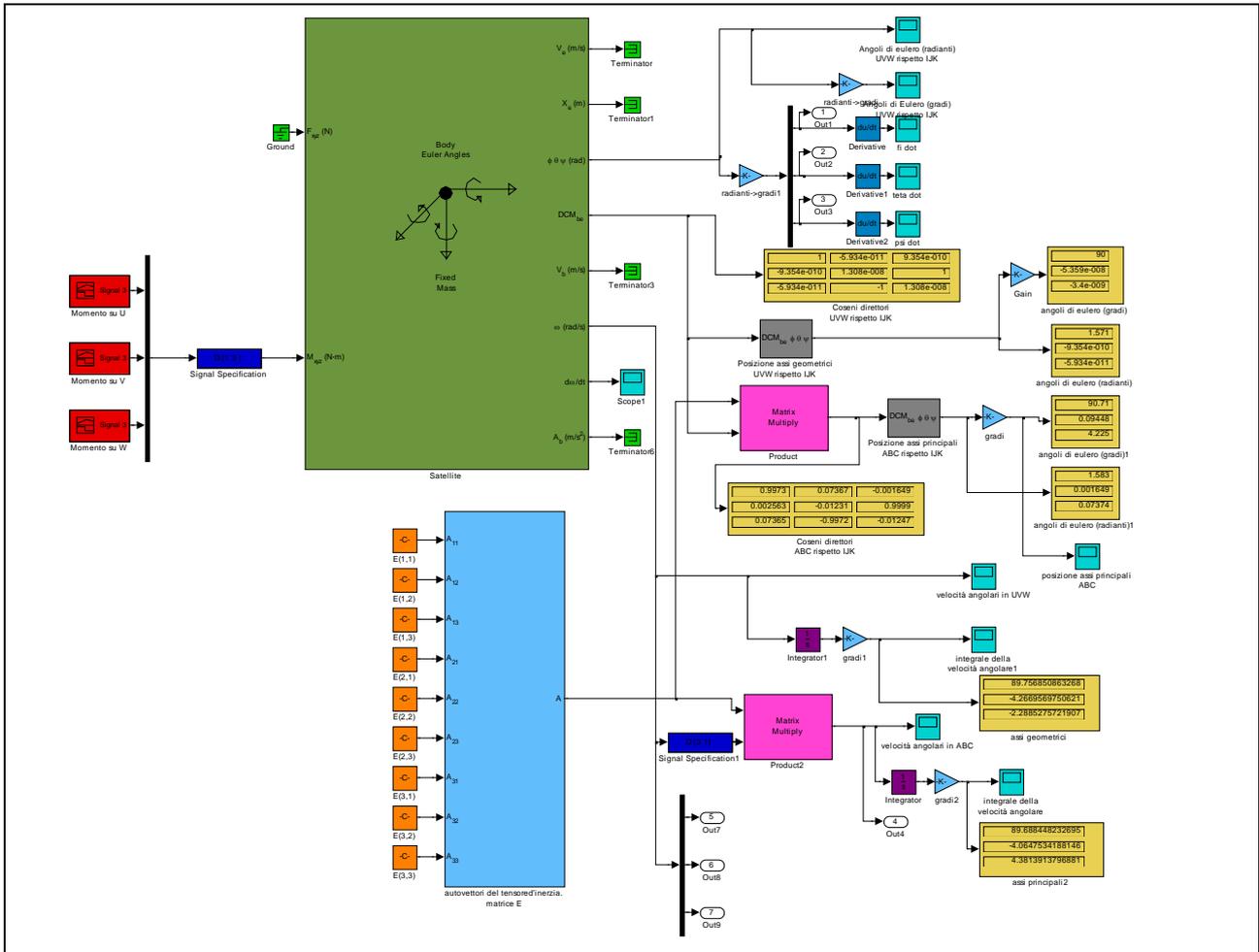


Figura 10.13 Descrizione della dinamica del satellite mediante lo schema a blocchi in Simulink.

Si notino nella figura precedente i tre blocchi in rosso a sinistra. Questi rappresentano gli ingressi di comando al sistema, ovvero le componenti dei momenti delle forze calcolati in precedenza. Il blocco 6DoF rappresenta la dinamica del satellite ed è raffigurato chiaramente in figura dal blocco di colore verde. Infine il blocco in azzurro rappresenta la matrice degli autovettori  $\underline{E}$ , utilizzata per effettuare le trasformazioni di coordinate tra il sistema di riferimento del satellite UUV ed il sistema di riferimento dettato dagli assi principali d'inerzia ABC. Gli altri blocchi visualizzabili nella stessa figura sono sostanzialmente dei blocchi contenenti funzioni algebriche e dei blocchi di output (utilizzati al fine di visualizzare i risultati ottenuti in simulazione).

# Capitolo 11

## 11 La descrizione del sistema ACS mediante il linguaggio UML

In questo capitolo si riporta la descrizione del sistema ACS sviluppato mediante il linguaggio UML (Unified Modeling Language, “linguaggio di modellazione unificato”). Il software del sistema, implementato in principio in linguaggio di programmazione C, è stato opportunamente suddiviso in classi di dati, descritto in UML e tradotto in linguaggio di programmazione C++, al fine di ottenere tutti i benefici derivanti dal paradigma della programmazione orientata agli oggetti. Durante la descrizione UML è stato utilizzato l’ambiente Visual Paradigm. Quest’ambiente software fornisce gli strumenti utili nella descrizione UML di un sistema qualunque. L’intero progetto AraMiS è descritto utilizzando l’UML e l’ambiente Visual Paradigm ad esso associato. Si riportano preliminarmente i concetti fondamentali su cui si basa il linguaggio UML.

### 11.1 Il paradigma Object-Oriented

L’UML è un linguaggio utilizzabile nella descrizione (modello) di un sistema di natura qualunque basato sul paradigma *Object-Oriented*. Nel caso in esame il modello che si vuole descrivere risulta essere il software del sistema ACS sviluppato nel corso del lavoro e si parla, quindi, di modello del software. Quando si parla di paradigma Object-Oriented, ovvero paradigma orientato agli oggetti, ci si riferisce ad un insieme di concetti introdotti dai linguaggi di programmazione orientati agli oggetti. I medesimi concetti sono stati estesi, in seguito, a numerosi altri contesti della tecnologia dell’informazione e della comunicazione. Nella descrizione seguente si riportano i concetti fondamentali su cui si basa il paradigma orientato agli oggetti e quindi il linguaggio UML stesso. I principali concetti sfruttati dall’UML possono essere riassunti in:

- Astrazione dei dati.
- Incapsulamento.
- Polimorfismo.
- Ereditarietà.

Il concetto principale su cui si fonda il paradigma orientato agli oggetti risulta essere l’*astrazione dei dati*. Per categoria astratta di dati si intende un insieme di dati e di operazioni che devono essere svolte su tali dati. Lo schema utilizzato nel rappresentare una categoria astratta di dati prende il nome di *classe*. Una classe rappresenta quindi un’astrazione di un concetto qualunque del mondo reale (una persona, un luogo, una parte di un sistema software, per fare alcuni esempi). Quando una particolare classe è inclusa in un programma si genera un *oggetto*. La classe costituisce quindi il modello degli oggetti che risulta possibile creare. Una classe è caratterizzata sostanzialmente da un *nome*, dai suoi *attributi* e dai suoi *metodi*. Il nome della classe identifica la particolare categoria di

dati astratti descritti nella classe stessa. Gli attributi della classe rappresentano l'insieme dei dati utilizzati nella descrizione della particolare classe. I metodi della classe, infine, rappresentano le operazioni svolte su tali dati, ovvero le operazioni svolte sugli attributi della classe in esame. Per esempio, una persona è possibile descriverla mediante una classe di nome *Persona* in cui sono presenti gli attributi *nome*, *cognome*, *data di nascita* ed il metodo *età()* che restituisce l'età della particolare persona in esame. Grazie all'astrazione dei dati è possibile affrontare un problema seguendo i concetti utilizzati dalla mente umana ed allontanandosi dal modo di pensare delle macchine (programmazione basata su altri paradigmi).

La suddivisione del problema in classi di dati permette di ottenere un altro importante beneficio che rappresenta un ulteriore concetto cardine del paradigma orientato agli oggetti, ovvero l'*information hiding* (l'informazione nascosta). L'*information hiding* rappresenta il principio teorico su cui si basa la tecnica dell'incapsulamento, largamente utilizzata in informatica. La tecnica dell'incapsulamento consiste nel nascondere il funzionamento interno delle classi in modo da proteggere le altre parti del sistema da funzionamenti difettosi della classe stessa, oppure nel caso in cui si decide di implementarla in modo differente. Quando un sistema utilizzando le classi deve essere modificato, quindi, tutte le modifiche tendono ad essere limitate alle classi stesse. Questo importante concetto permette la realizzazione di sistemi semplici da mantenere.

Il concetto di *polimorfismo* o *genericità*, invece, consiste nello sfruttare la definizione di una particolare classe nella generazione di numerosi oggetti. Quest'importante proprietà dei linguaggi di programmazione orientati agli oggetti, e di tutti i sistemi basati sul paradigma Object-Oriented, consente di sviluppare una libreria di categorie astratte di dati di base che potrà essere utilizzata da altre future applicazioni.

Il concetto di *ereditarietà*, infine, consiste nella possibilità di definizione di una classe come una *sottoclasse*, o *classe derivata*, di una classe preesistente detta *superclasse*, o *classe base*. La sottoclasse così definita eredita tutti gli attributi e tutti i metodi della classe base e li integra con opportune modifiche od estensioni.

I linguaggi di programmazione orientati agli oggetti consentono di ottenere due vantaggi importanti. In primo luogo permettono, grazie all'astrazione dei dati, lo sviluppo di sistemi software che per essere modificati richiedono soltanto delle modifiche a parti relativamente piccole del sistema. In secondo luogo, i concetti di polimorfismo ed ereditarietà consentono ad i progettisti del software di sviluppare una libreria di classi di base che possono essere riutilizzate, con relativa semplicità, in nuove applicazioni. L'approccio con cui un problema è affrontato, inoltre, non risulta essere basato su ciò che il sistema dovrà fare. Bensì l'interesse maggiore è focalizzato sui dati che stanno alla base del problema che si vuole affrontare e questo rappresenta la sostanziale differenza di un linguaggio di programmazione orientato agli oggetti rispetto ad un linguaggio di programmazione basato su di un paradigma differente quale, per esempio, la programmazione procedurale (come il linguaggio C). I linguaggi di programmazione orientati agli oggetti rappresentano oggi il massimo livello di astrazione raggiungibile durante il ciclo di progetto di un sistema software.

## 11.2 Il linguaggio UML

Il linguaggio UML sfrutta i concetti descritti nel paragrafo precedente. In UML la descrizione del sistema in esame avviene sfruttando una serie di diagrammi. Questi diagrammi sono costruiti componendo elementi grafici, elementi di testo formali ed elementi di testo libero. Il linguaggio UML permette la costruzione di modelli basati sul paradigma Object-Oriented indirizzati alla descrizione di un qualunque sistema. Il modello permette di mostrare numerose viste (struttura, comportamento, funzionamento, per fare alcuni esempi) del sistema interessato, grazie ai

diagrammi introdotti sopra, e risulta essere utile sia nella fase di analisi, sia nella fase di progetto del sistema in esame. L'UML è stato progettato con l'obiettivo di facilitare il supporto software alla costruzione di modelli e l'integrazione di questo supporto con gli IDE (Integrated Development Environment, "ambienti integrati di sviluppo"). Nella discussione seguente si descrivono i due diagrammi UML utilizzati nella descrizione del sistema ACS sviluppato, ovvero gli *Use Case Diagrams* (i "diagrammi dei casi d'uso") ed i *Class Diagrams* (i "diagrammi di classe").

### 11.2.1 Gli Use Case Diagrams

Gli Use Case Diagrams sono dei diagrammi UML dedicati alla descrizione delle funzionalità offerte dal sistema che si sta descrivendo, come viste dal mondo esterno che interagisce con il sistema stesso. In altre parole un caso d'uso rappresenta un particolare comportamento di un sistema dal punto di vista dell'utente. Gli utenti interagenti con il sistema analizzato, che possono essere sia delle persone, sia degli altri sistemi (hardware, software o misti), sono detti *attori*. Un attore è rappresentato mediante un'icona raffigurante un uomo stilizzato. Un particolare caso d'uso invece è rappresentato graficamente mediante un'ellisse contenente il nome sintetico del caso d'uso. Mediante il diagramma dei casi d'uso è possibile avere una chiara visione dei requisiti del sistema dal punto di vista dell'utenza finale. I casi d'uso rappresentano quindi i modi con cui il sistema in analisi può essere utilizzato. Grazie a queste caratteristiche, i casi d'uso risultano quindi essere un ottimo strumento di valutazione della corretta risposta dei requisiti di un sistema. Si riporta nella figura seguente un esempio di Use Case Diagram composto di un attore ed un caso d'uso:

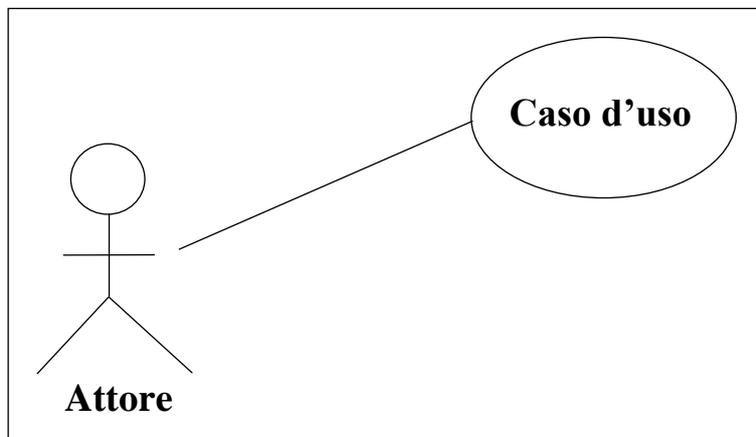


Figura 11.1 Diagramma dei casi d'uso.

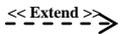
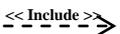
Ovviamente il numero di attori e di casi d'uso presenti in uno Use Case Diagram è funzione del particolare sistema che si desidera descrivere.

All'interno di un diagramma dei casi d'uso le relazioni tra i vari elementi avvengono mediante opportune linee di collegamento. Si riportano di seguito i diversi tipi di relazione con l'opportuna notazione:

- Associazione: rappresenta il percorso della comunicazione tra un attore ed il caso d'uso cui partecipa.

- Estensione: rappresenta una relazione di estensione dal caso d'uso A al caso d'uso B. In particolare indica che il secondo può includere, sotto specifiche condizioni dettagliate nell'estensione, il comportamento del caso d'uso A.
- Generalizzazione o specializzazione: rappresenta la relazione che intercorre tra un caso d'uso più generale ed uno più specifico che eredita dal primo alcune funzionalità e ne aggiunge altre.
- Inclusione: rappresenta un'inclusione in un caso d'uso di base di funzionalità illustrate in altri casi d'uso.

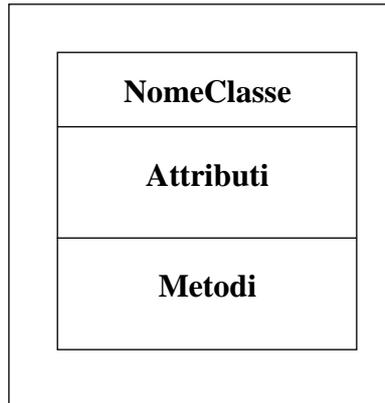
Si riporta nella tabella seguente la notazione utilizzata per i diversi tipi di relazione:

<b>Associazione</b>	
<b>Estensione</b>	
<b>Generalizzazione</b>	
<b>Inclusione</b>	

**Figura 11.2 Tipi di relazione.**

## 11.2.2 I Class Diagrams

I Class Diagrams sono dei diagrammi UML dedicati alla descrizione delle classi, il cui concetto è stato introdotto precedentemente, componenti il sistema che si vuole descrivere. Le classi del sistema descritto in UML sono rappresentate mediante un rettangolo. Nel rettangolo sono presenti tre campi distinti in cui è possibile inserire il nome, gli attributi ed i metodi della classe interessata. Si riporta nella figura seguente la rappresentazione UML di una classe:



**Figura 11.3 Rappresentazione di una classe in UML.**

Per convenzione il nome di una classe è indicato con la lettera maiuscola. Nel caso in cui la parola del nome è composta di più parole si utilizza la convenzione di scrivere tutte le parole attaccate con l'iniziale maiuscola, come è stato rappresentato nella figura precedente.

Come risulta semplice osservare nella figura precedente, gli attributi della classe NomeClasse sono separati dal nome da una linea orizzontale. Per ogni attributo è specificato il tipo (int, float, bool, per fare alcuni esempi) ed eventualmente il valore di default dell'attributo. Si riportano due esempi di definizione di attributi per fissare le idee:

```
attributo1: tipo1 = "Valore di default"
attributo2: tipo2
```

Anche per gli attributi si utilizza la stessa convenzione di scrittura adottata per il nome della classe. La lista delle operazioni eseguibili sugli attributi di una particolare classe, ovvero la lista dei metodi della classe, è riportata nella parte inferiore del rettangolo rappresentante la classe. Anche in questo caso una linea orizzontale divide la regione contenente gli attributi da quella contenente i metodi. La definizione del metodo riporta i parametri eventualmente necessari al metodo stesso con il tipo di ogni parametro (tra parentesi tonde) e, se il metodo in questione rappresenta una funzione, il tipo di ritorno. Nel caso del sistema ACS in esame, tutti i metodi rappresentano delle funzioni e, quindi, per ognuno di essi è necessario indicare il tipo di ritorno (void se non ritorna niente). Si riportano alcuni esempi di definizione permesse di metodi per fissare le idee:

```
metodo1()
metodo2(lista di parametri)
metodo3(lista di parametri): tipo restituito
```

La convenzione di scrittura risulta essere la medesima adottata per gli attributi e per il nome della classe.

## 11.3 Il diagramma dei casi d'uso del sistema ACS

In questo paragrafo si descrivono i casi d'uso del sistema ACS sviluppato, ovvero le funzionalità del sistema viste dal punto di vista esterno. Si riporta nella figura seguente il diagramma UML in questione, costruito mediante Visual paradigm:

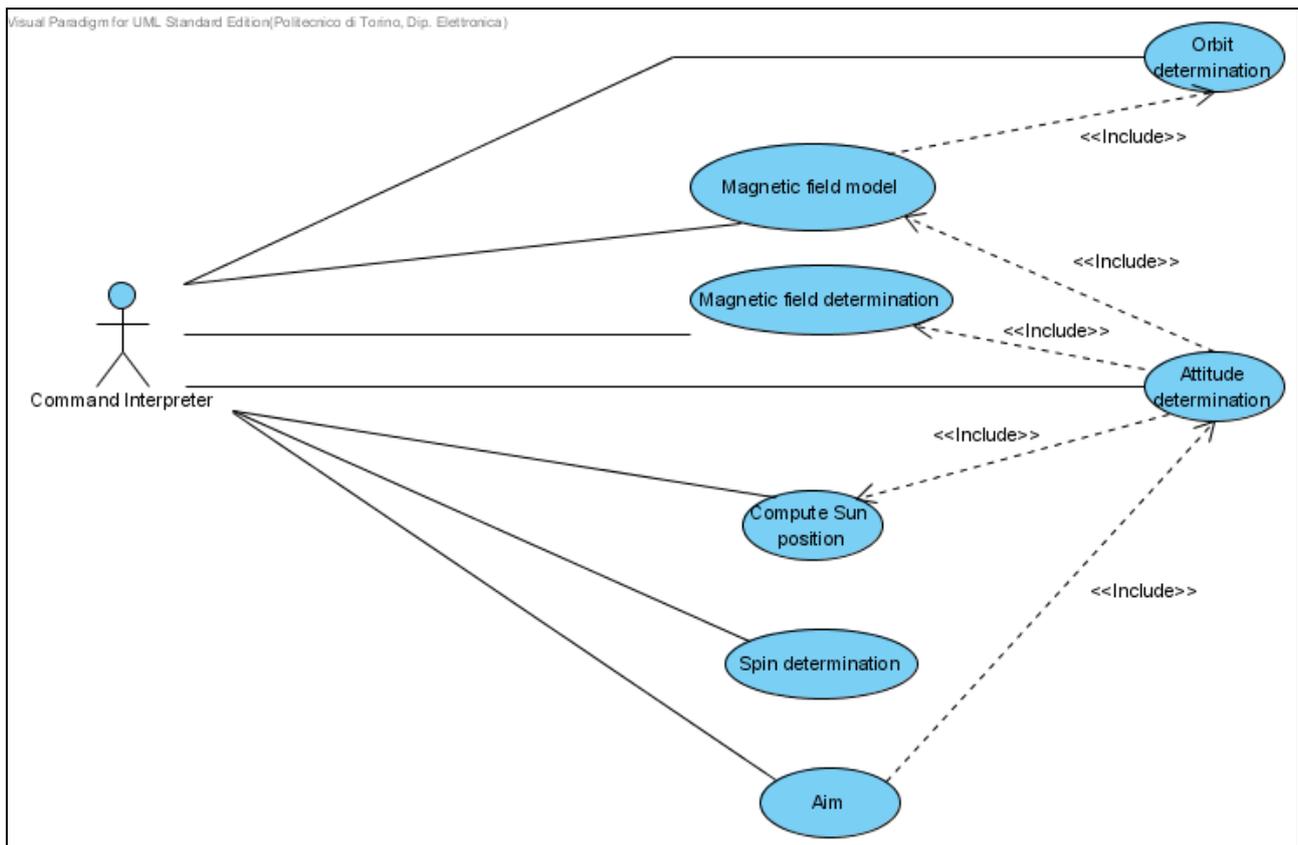


Figura 11.4 Diagramma dei casi d'uso del sistema ACS.

Nel diagramma precedente è possibile osservare l'attore interagente con il sistema ACS, raffigurato dal Command Interpreter. Quest'ultimo rappresenta il software dell'OBC (On Board Computer) indirizzato alla decodifica dei comandi provenienti dalla stazione di Terra e ricevuti dall'OBC, mediante il modulo di antenne apposito posizionato sulla Telecommunication tile.

Si descrivono nel seguito i singoli casi d'uso componenti il diagramma precedente:

- **Orbit determination:** calcolo del vettore di posizione del satellite specificando le tre componenti rispetto al sistema di riferimento geocentrico inerziale IJK. Le componenti del vettore di posizione sono espresse in Km.
- **Compute Sun position:** calcolo delle coordinate equatoriali del Sole rispetto al sistema di riferimento geocentrico inerziale IJK, ovvero l'ascensione retta (*RA*) e la declinazione (*Dec*) del Sole, entrambe espresse in gradi.
- **Magnetic field model:** calcolo del modello del campo magnetico terrestre in funzione di una particolare data e di una particolare posizione attorno alla Terra, ovvero le componenti

del vettore  $\vec{\varphi}$  rispetto al sistema di riferimento geocentrico inerziale IJK. Le componenti del vettore in questione sono espresse in unità di 10 nT. Si noti che questo caso d'uso è collegato al caso d'uso Orbit determination mediante un'inclusione. Infatti, per calcolare la particolare posizione attorno alla Terra, dettata dalla posizione del satellite nell'orbita, si sfrutta il caso d'uso apposito (Orbit determination).

- **Magnetic field determination:** calcolo, interrogando tutti i magnetometri a bordo satellite, del vettore campo magnetico terrestre  $\vec{\varphi}$  misurato sul satellite, specificando le componenti rispetto al sistema di riferimento UVW del satellite. Le componenti del vettore sono espresse in unità di 10 nT.
- **Attitude determination:** calcolo dell'assetto del satellite specificando la matrice dei coseni direttori, di dimensioni  $(3 \times 3)$ , della terna di riferimento UVW del satellite rispetto alla terna geocentrica inerziale IJK. Gli elementi della matrice sono rappresentati in float. Si noti che questo caso d'uso è collegato, mediante delle inclusioni, ai casi d'uso Compute Sun position, Magnetic field model e Magnetic field determination. Infatti, la determinazione dell'assetto è basata sulla conoscenza della posizione del Sole e del vettore campo magnetico terrestre rispetto al sistema geocentrico inerziale IJK e sulla lettura dei sensori di posizione del Sole e dei sensori di campo magnetico.
- **Spin determination:** calcolo del vettore di spin del satellite, ovvero del vettore della velocità angolare dello stesso, specificando le componenti rispetto al sistema di riferimento UVW del satellite. Le componenti del vettore sono espresse in rad / s.
- **Aim:** calcolo dei comandi (momenti angolari) da fornire alle ruote di reazione al fine di orientare, in un certo lasso di tempo specificato in secondi, il satellite secondo un assetto desiderato e specificato rispetto al sistema di riferimento geocentrico inerziale IJK. Si noti che questo caso d'uso è collegato, mediante un'inclusione, al caso d'uso Attitude determination. Infatti, il controllo dell'orientamento del satellite è basato sulla conoscenza dell'orientamento attuale del satellite.

## 11.4 Il diagramma delle classi del sistema ACS

In questo paragrafo si descrive il diagramma delle classi del sistema ACS. Tutti gli algoritmi sviluppati durante l'attività di tesi sono stati analizzati da un punto di vista complessivo, con lo scopo di suddividerli opportunamente in apposite classi di dati. Durante questa fase l'intero codice del software ACS è stato modellato come un sistema software composto di sette classi distinte. In particolare è stata individuata la classe *Orbits*, la classe *elements\_s*, la classe *Magnet*, la classe *Position*, la classe *elsetrec*, la classe *Satellite* e la classe *Tile*.

La classe *Orbits* è stata modellata come la categoria astratta di dati indirizzata alla descrizione del software finalizzato al calcolo della posizione del Sole, rispetto al sistema geocentrico inerziale IJK (il modello della posizione del Sole).

La classe *elements\_s* è stata modellata come la categoria astratta di dati indirizzata alla memorizzazione dei principali parametri descrittivi l'orbita della Terra attorno al Sole. Questa classe è, quindi, utilizzata dalla classe *Orbits* per effettuare i vari computi riguardanti il calcolo della posizione del Sole (il modello).

La classe *Magnet* è stata modellata come la categoria astratta di dati indirizzata alla descrizione del software finalizzato al calcolo del vettore campo magnetico terrestre, rispetto al sistema di riferimento geocentrico inerziale IJK (il modello del vettore campo magnetico terrestre), in funzione di una posizione attorno alla Terra.

La classe *Position* è stata modellata come la categoria astratta di dati indirizzata alla descrizione del software finalizzato al calcolo della posizione del satellite, rispetto al sistema geocentrico inerziale IJK. Sostanzialmente questa classe raggruppa insieme tutto il software SGP4. Questa classe è utilizzata dalla classe *Magnet* (si noti, infatti, che la classe *Magnet* si occupa del calcolo del modello del vettore campo magnetico in funzione della posizione del satellite) e dalla classe *Satellite*.

La classe *elsetrec* è stata modellata come la categoria astratta di dati indirizzata alla memorizzazione dei principali parametri descriventi l'orbita del satellite ed utilizzati dall'algoritmo SGP4. Per questo motivo questa classe è utilizzata dalla classe *Position* per effettuare i vari computi.

La classe *Satellite* è stata modellata come la categoria astratta di dati indirizzata alla descrizione dell'intero software finalizzato alla determinazione ed al controllo dell'assetto del satellite sviluppato durante il lavoro di tesi. La classe utilizza la classe *Orbits* per ottenere il modello della posizione del Sole, utilizza la classe *Magnet* per ottenere il modello del vettore campo magnetico ed utilizza la classe *Position* per ottenere la posizione del satellite.

La classe *Tile*, infine, è stata modellata come la categoria astratta di dati indirizzata alla descrizione di una singola tile componente il satellite. Sostanzialmente questa classe contiene i principali parametri identificativi di una tile ed i metodi di lettura dei sensori d'assetto e di attuazione delle ruote di reazione. Questa classe è utilizzata dalla classe *Satellite*. Quest'ultima, infatti, istanzia un certo numero di classi *Tile*, in funzione della particolare configurazione del satellite considerata.

Il sistema ACS, suddiviso mediante queste classi menzionate, risulta essere completo. Tuttavia, sono state opportunamente aggiunte ancora due classi. La prima, nominata *ACS*, funge da interfaccia del sistema, come visto dal mondo esterno. Questa classe non possiede alcun attributo e si occupa soltanto di chiamare opportunamente i metodi interni del sistema. La seconda, nominata *Test*, è stata modellata come la categoria astratta di dati indirizzata alla descrizione del software di test del sistema ACS. Questa classe raffigura, quindi, il mondo esterno, interagente con il sistema ed utilizza la classe *ACS*.

Si riporta nella figura seguente la suddivisione delle classi appena descritta (per motivi di chiarezza si riporta soltanto il nome delle classi coinvolte nel sistema ACS in analisi):

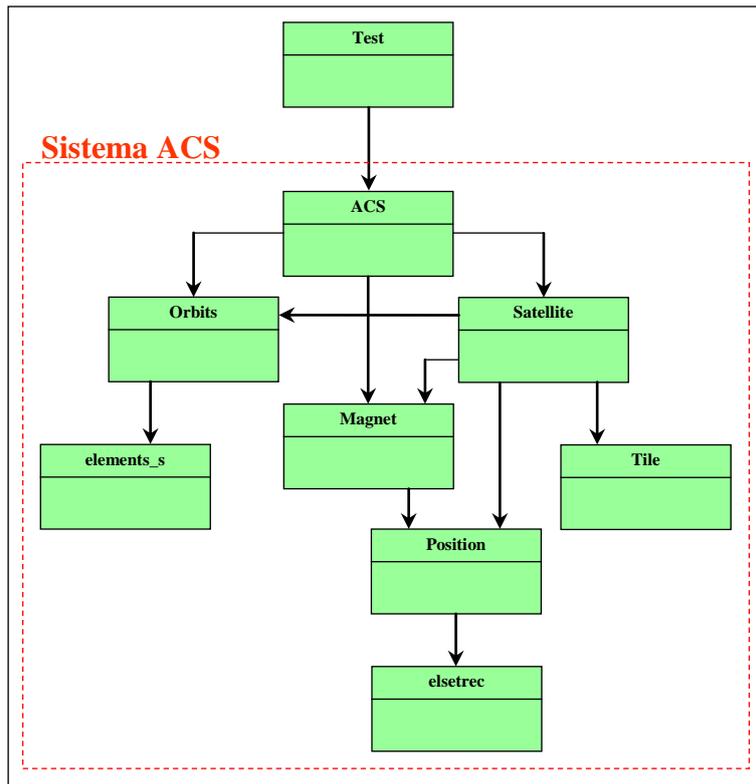


Figura 11.5 Suddivisione del sistema ACS in classi.

Si riporta nella figura seguente il diagramma delle classi complessivo costruito mediante Visual Paradigm (in questo diagramma, a differenza di quello presentato in figura 11.5, sono mostrate tutte le classi del sistema ACS, con i rispettivi attributi e metodi):



### 11.4.1 Test

Questa classe rappresenta il software di test del sistema ACS.

Gli attributi della classe Test sono descritti di seguito:

- *year* : *int const*, costante di tipo *int* destinata a rappresentare l'anno di interesse (4 cifre).
- *month* : *int const*, costante di tipo *int* destinata a rappresentare il mese dell'anno di interesse (1-12).
- *day* : *int const*, costante di tipo *int* destinata a rappresentare il giorno del mese di interesse (1-31).
- *hours* : *int const*, costante di tipo *int* destinata a rappresentare l'ora del giorno di interesse (0-23). L'ora in questione è da intendere come l'ora di Greenwich.
- *minutes* : *int const*, costante di tipo *int* destinata a rappresentare i minuti dell'ora di interesse.
- *seconds* : *float const*, costante di tipo *float* destinata a rappresentare i secondi dell'ora di interesse.

- *ro* : *float[3]*, array di tre elementi di tipo *float* destinato a contenere le componenti del vettore di posizione del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. Tali componenti sono espresse in [Km]. Le tre componenti sono rappresentate nel seguente modo:

*ro[0]* = componente in I del vettore di posizione del satellite.

*ro[1]* = componente in J del vettore di posizione del satellite.

*ro[2]* = componente in K del vettore di posizione del satellite.

- *vo* : *float[3]*, array di tre elementi di tipo *float* destinato a contenere le componenti del vettore di velocità del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. Tali componenti sono espresse in [Km / s]. Le tre componenti sono rappresentate nel seguente modo:

*vo[0]* = componente in I del vettore di velocità del satellite.

*vo[1]* = componente in J del vettore di velocità del satellite.

*vo[2]* = componente in K del vettore di velocità del satellite.

- *re* : *float[3]*, array di tre elementi di tipo *float* destinato a contenere le componenti del vettore di posizione del centro della Terra, come visto dal satellite. Tali componenti sono espresse in [Km] e sono identiche alle componenti contenute in *ro* ma invertite di segno.
- *RA* : *float*, variabile di tipo *float* destinata a contenere l'ascensione retta del Sole nel sistema di riferimento geocentrico inerziale IJK. Questa coordinata sferica è espressa in gradi.

- *Dec* : *float*, variabile di tipo float destinata a rappresentare la declinazione del Sole nel sistema di riferimento geocentrico inerziale IJK. Questa coordinata sferica è espressa in gradi.
- *Ad* : *float[3][3]*, matrice di dimensioni (3 × 3) di tipo float destinata a rappresentare la matrice dei coseni direttori desiderata degli assi geometrici UVW del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. La prima riga rappresenta le coordinate desiderate dell'asse U specificate nel sistema IJK (I è la prima colonna, J è la seconda colonna, K è la terza colonna). La seconda e la terza riga rappresentano le coordinate desiderate degli assi V e W in IJK, rispettivamente. Si ricorda che ogni riga deve possedere norma unitaria.
- *spin* : *float[3]*, array di tre elementi di tipo float destinato a rappresentare le componenti del vettore della velocità angolare del satellite (spin) rispetto al sistema di riferimento UVW del satellite. Tali componenti sono espresse in [rad / s]. Le tre componenti sono rappresentate nel seguente modo:

*spin[0]* = componente in U del vettore della velocità angolare del satellite.

*spin[1]* = componente in V del vettore della velocità angolare del satellite.

*spin[2]* = componente in W del vettore della velocità angolare del satellite.

- *Buvw* : *float[3]*, array di tre elementi di tipo float destinato a rappresentare le componenti del vettore campo magnetico terrestre misurato rispetto al sistema di riferimento UVW del satellite. Tali componenti sono espresse in unità di 10 nT. Le tre componenti sono rappresentate nel seguente modo:

*Buvw[0]* = componente in U del vettore campo magnetico terrestre misurato su satellite.

*Buvw[1]* = componente in V del vettore campo magnetico terrestre misurato su satellite.

*Buvw[2]* = componente in W del vettore campo magnetico terrestre misurato su satellite.

- *Bi* : *float*, variabile di tipo float destinata a rappresentare la componente in I del vettore campo magnetico (modello) rispetto al sistema di riferimento geocentrico inerziale IJK. Tale componente è espressa in unità di 10 nT.
- *Bj* : *float*, variabile di tipo float destinata a rappresentare la componente in J del vettore campo magnetico (modello) rispetto al sistema di riferimento geocentrico inerziale IJK. Tale componente è espressa in unità di 10 nT.
- *Bk* : *float*, variabile di tipo float destinata a rappresentare la componente in K del vettore campo magnetico (modello) rispetto al sistema di riferimento geocentrico inerziale IJK. Tale componente è espressa in unità di 10 nT.
- *A* : *float[3][3]*, matrice di dimensioni (3 × 3) di tipo float destinata a rappresentare la matrice dei coseni direttori determinata degli assi geometrici UVW del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. La prima riga rappresenta le coordinate dell'asse U specificate nel sistema IJK (I è la prima colonna, J è la seconda colonna, K è la terza colonna). La seconda e la terza riga rappresentano le coordinate degli assi V e W in IJK, rispettivamente. Si ricorda che ogni riga deve possedere norma unitaria.

- $t\_m$  : *float*, variabile di tipo float destinata a rappresentare il tempo desiderato per effettuare una manovra completa. Questo tempo è espresso in secondi.
- $m$  : *ACS*, rappresenta un oggetto di tipo ACS (si veda classe ACS).

La classe Test possiede un solo metodo:

- $main(void)$  : *int*, metodo rappresentante la funzione di test del sistema ACS. La funzione contiene una serie di chiamate ai metodi interni del sistema e ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.

## 11.4.2 ACS

Questa classe rappresenta l'interfaccia del sistema ACS, come visto dal mondo esterno. Gli attributi della classe ACS sono descritti di seguito:

- $O$  : *Orbits*, rappresenta un oggetto di tipo Orbits (si veda classe Orbits).
- $M$  : *Magnet*, rappresenta un oggetto di tipo Magnet (si veda classe Magnet).
- $S$  : *Satellite*, rappresenta un oggetto di tipo Satellite (si veda classe Satellite).

I metodi della classe ACS, che richiamano a loro volta i rispettivi metodi interni del sistema, sono descritti di seguito:

- $computeSatellitePosition(ro : float[3], vo : float[3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : void$ , metodo descritto nella classe Position.
- $computeEarthPosition(re : float[3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : void$ , metodo descritto nella classe Position.
- $getSpin(spin : float[3]) : bool$ , metodo descritto nella classe Satellite.
- $getMagnetic(Buvw : float[3]) : bool$ , metodo descritto nella classe Satellite.
- $computeMagneticField(Bi : float*, Bj : float*, Bk : float*, year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool$ , metodo descritto nella classe Magnet.
- $getAttitude(A : float[3][3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool$ , metodo descritto nella classe Satellite.
- $aim(Ad : float[3][3], t\_m : float, A : float[3][3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool$ , metodo descritto nella classe Satellite.

- *computeSunPosition(RA : float\*, Dec : float\*, year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool*, metodo descritto nella classe Orbits.
- *caricaDati(void) : void*, metodo descritto nella classe Satellite.

### 11.4.3 Orbits

Questa classe rappresenta il software indirizzato al calcolo della posizione del Sole rispetto al sistema geocentrico inerziale IJK. La teoria associata a tale classe è descritta nel dettaglio nel capitolo 7, paragrafo 1.

Gli attributi dell'oggetto Orbits sono descritti di seguito:

- *d : float*, variabile di tipo float destinata a rappresentare il parametro *d*, detto *day number*, ovvero i giorni trascorsi dal 1° gennaio 2000 alle ore 00:00 (ora di Greenwich). La parte frazionaria della variabile indica la frazione del particolare giorno, prende quindi in considerazione anche l'ora del giorno in esame.
- *e : elements\_s*, rappresenta un oggetto di tipo *elements\_s* (si veda classe *elements\_s*).

I metodi della classe Orbits sono descritti di seguito:

- *computeSunPosition(RA : float\*, Dec : float\*, year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool*, metodo di calcolo della posizione del Sole rispetto al sistema di riferimento geocentrico inerziale IJK. Il metodo restituisce le coordinate equatoriali del Sole, ovvero la sua ascensione retta e la sua declinazione. Le coordinate sono ritornate nelle variabili *RA* e *Dec*, rispettivamente (si veda la classe Test per una descrizione di tutti i parametri). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.
- *computeDay( year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : float*, metodo di assegnazione del parametro *d* (attributo della classe). Tutti i parametri del metodo sono descritti nella classe Test.

### 11.4.4 elements\_s

Questa classe rappresenta la struttura dati contenente i parametri utili nella descrizione dell'orbita della Terra attorno al Sole.

Gli attributi della classe *elements\_s* sono descritti di seguito:

- $Md$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'anomalia media, rappresentata dall'attributo  $M$ .
- $MO$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'anomalia media, rappresentata dall'attributo  $M$ .
- $M$  : *float*, variabile di tipo float destinata a rappresentare l'anomalia media dell'orbita della Terra. Tale grandezza è espressa in gradi.
- $ed$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'eccentricità dell'orbita della Terra, rappresentata dall'attributo  $e$ .
- $e0$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'eccentricità dell'orbita della Terra, rappresentata dall'attributo  $e$ .
- $e$  : *float*, variabile di tipo float destinata a rappresentare l'eccentricità dell'orbita della Terra (numero privo di dimensioni).
- $ad$  : *float const*, costante di tipo float utilizzata durante il calcolo del semiasse maggiore dell'orbita della Terra, rappresentata dall'attributo  $a$ .
- $a0$  : *float const*, costante di tipo float utilizzata durante il calcolo del semiasse maggiore dell'orbita della Terra, rappresentata dall'attributo  $a$ .
- $a$  : *float*, variabile di tipo float destinata a rappresentare il semiasse maggiore dell'orbita della Terra. Tale grandezza è espressa in A.U. (Astronomical Unit, 1 A.U. raffigura la distanza media della Terra dal Sole, pari a 149 597 870,691 Km).
- $wd$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'argomento del perielio, rappresentato dall'attributo  $w$ .
- $w0$ : *float const*, costante di tipo float utilizzata durante il calcolo dell'argomento del perielio, rappresentato dall'attributo  $w$ .
- $w$  : *float*, variabile di tipo float destinata a rappresentare l'argomento del perielio. Tale grandezza è espressa in gradi.
- $id$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'inclinazione dell'orbita della Terra, rappresentata dall'attributo  $i$ .
- $i0$  : *float const*, costante di tipo float utilizzata durante il calcolo dell'inclinazione dell'orbita della Terra, rappresentata dall'attributo  $i$ .
- $i$  : *float*, variabile di tipo float destinata a rappresentare l'inclinazione dell'orbita terrestre. Tale grandezza è espressa in gradi.
- $Nd$  : *float const*, costante di tipo float utilizzata durante il calcolo della longitudine del nodo ascendente, rappresentata dall'attributo  $N$ .

- *NO* : *float const*, costante di tipo float utilizzata durante il calcolo della longitudine del nodo ascendente, rappresentata dall'attributo *N*.
- *N* : *float*, variabile di tipo float destinata a rappresentare la longitudine del nodo ascendente. Tale grandezza è espressa in gradi.

La classe *elements\_s* possiede soltanto il seguente metodo:

- *computeElement(d : float) : void*, metodo di assegnazione degli attributi *M, e, a, w, i, N*, descritti precedentemente. Il parametro *d*, parametro d'ingresso al metodo, è descritto nella classe *Orbits*.

## 11.4.5 Magnet

Questa classe rappresenta il software indirizzato al calcolo (modello) del vettore campo magnetico terrestre, basato sui modelli IGRF o WMM. La teoria associata a tale classe è descritta nel dettaglio nel capitolo 7, paragrafo 2.

Gli attributi dell'oggetto *Magnet* sono descritti di seguito:

- *longitude* : *float*, variabile di tipo float destinata a rappresentare la longitudine del punto attorno alla Terra (dettato dalla posizione del satellite nell'orbita) in cui si desidera conoscere l'orientamento del vettore campo magnetico terrestre. Tale grandezza è espressa in gradi decimali e, per convenzione, risulta positiva verso Est e negativa verso Ovest.
- *latitude* : *float*, variabile di tipo float destinata a rappresentare la latitudine del punto attorno alla Terra (dettato dalla posizione del satellite nell'orbita) in cui si desidera conoscere l'orientamento del vettore campo magnetico terrestre. Tale grandezza è espressa in gradi decimali e, per convenzione, risulta positiva verso Nord e negativa verso Sud.
- *alt* : *float*, variabile di tipo float destinata a rappresentare l'altitudine (misurata sopra il livello del mare) del punto attorno alla Terra in cui si desidera conoscere l'orientamento del vettore campo magnetico terrestre (dove si trova il satellite nell'istante interessato). Tale grandezza è espressa in [Km] (i valori permessi sono compresi tra 0 - 601 Km).
- *IEXT* : *int const*, costante di tipo int destinata a rappresentare un flag per i coefficienti esterni.
- *MAXMOD* : *int const*, costante di tipo int destinata a rappresentare la grandezza di alcuni array utilizzati all'interno dei metodi.
- *EXT\_COEFF1* : *float const*, costante di tipo float destinata a rappresentare un coefficiente esterno del primo ordine (non utilizzato se *IEXT* = 0).

- *EXT\_COEFF2* : *float const* costante di tipo float destinata a rappresentare un coefficiente esterno del primo ordine (non utilizzato se *IEXT* = 0).
- *EXT\_COEFF3* : *float const*, costante di tipo float destinata a rappresentare un coefficiente esterno del primo ordine (non utilizzato se *IEXT* = 0).
- *MAXDEG* : *int const*, costante di tipo int utilizzata nella definizione dell'attributo *MAXCOEFF*.
- *MAXCOEFF* : *int const*, costante di tipo int destinata a definire la dimensione degli array *gh1*, *gh2*, *gha* e *ghb*.
- *gh1* : *float[MAXCOEFF]*, array contenente i coefficienti delle armoniche sferiche normalizzati secondo Schmidt.
- *gh2* : *float[MAXCOEFF]*, array contenente i coefficienti delle armoniche sferiche normalizzati secondo Schmidt.
- *gha* : *float[MAXCOEFF]*, array contenente i coefficienti delle armoniche sferiche normalizzati secondo Schmidt.
- *ghb* : *float[MAXCOEFF]*, array contenente i coefficienti delle armoniche sferiche normalizzati secondo Schmidt.
- *d* : *float*, variabile di tipo float destinata a rappresentare la declinazione del vettore campo magnetico terrestre, misurata dal polo Nord geografico. Tale grandezza è espressa in gradi.
- *f* : *float*, variabile di tipo float destinata a rappresentare l'intensità totale del campo magnetico terrestre. Tale grandezza è espressa in unità di 10 nT.
- *h* : *float*, variabile di tipo float destinata a rappresentare l'intensità orizzontale del campo magnetico terrestre espressa in unità di nT.
- *i* : *float*, variabile di tipo float destinata a rappresentare l'inclinazione del vettore campo magnetico terrestre espressa in gradi.
- *dtemp* : *float*, variabile temporanea di tipo float.
- *ftemp* : *float*, variabile temporanea di tipo float.
- *htemp* : *float*, variabile temporanea di tipo float.
- *itemp* : *float*, variabile temporanea di tipo float.
- *x* : *float*, variabile di tipo float destinata a rappresentare la componente verso Nord del vettore campo magnetico terrestre espressa in unità di 10 nT.
- *y* : *float*, variabile di tipo float destinata a rappresentare la componente verso Est del vettore campo magnetico terrestre espressa in unità di 10 nT.

- *z* : *float*, variabile di tipo float destinata a rappresentare la componente verticale (verso il Nadir) del vettore campo magnetico terrestre espressa in unità di 10 nT.
- *xtemp* : *float*, variabile temporanea di tipo float.
- *ytemp* : *float*, variabile temporanea di tipo float.
- *ztemp* : *float*, variabile temporanea di tipo float.
- *sdate* : *float*, variabile di tipo float destinata a rappresentare l'anno di interesse (4 cifre + parte frazionaria).
- *RA\_G* : *float*, variabile di tipo float destinata a rappresentare l'ascensione retta del meridiano di Greenwich, espressa in gradi decimali.
- *M* : *Position*, rappresenta un oggetto di tipo Position (si veda classe Position).

I metodi della classe Magnet sono descritti di seguito:

- *computeMagneticField(Bi : float\*, Bj : float\*, Bk : float\*, year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : bool*, metodo di calcolo del vettore campo magnetico terrestre (modello) rispetto al sistema di riferimento geocentrico inerziale IJK. Le componenti del vettore sono ritornate nelle variabili *Bi*, *Bj* e *Bk* (si veda la classe Test per una descrizione di tutti i parametri). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.
- *extrapsh(date : float, dte1 : float, nmax1 : int, nmax2 : int, gh : int) : int*, metodo che si occupa dell'extrapolazione lineare di un modello sferico armonico, mediante un modello contenente le variazioni. I parametri del metodo sono descritti di seguito:

Ingressi:

*date*: rappresenta l'attributo *sdate*.

*dte1*: rappresenta la data del modello di base (validità 5 anni).

*nmax1*: rappresenta il massimo grado ed ordine del modello di base.

*gh1*: rappresenta i coefficienti delle armoniche sferiche del modello di base normalizzati secondo Schmidt.

*nmax2*: rappresenta il massimo grado ed ordine del modello delle variazioni.

*gh2*: rappresenta i coefficienti delle armoniche sferiche del modello delle variazioni normalizzati secondo Schmidt.

Uscite:

*gha* o *b*: rappresentano i coefficienti delle armoniche sferiche normalizzati secondo Schmidt.

*nmax*: massimo grado ed ordine del modello utilizzato.

- *interpsh(date : float, dte1 : float, nmax1 : int, dte2 : float, nmax2 : int, gh : int) : int*, metodo che si occupa di effettuare l'interpolazione lineare, nel tempo, tra due modelli di armoniche sferiche. I parametri del metodo sono descritti di seguito:

Ingressi:

*date*: rappresenta la data del conseguente modello (anno in decimale).

*dte1*: rappresenta la data del modello precedente.

*nmax1*: rappresenta il massimo grado ed ordine del modello precedente.

*gh1*: rappresenta i coefficienti delle armoniche sferiche del modello precedente..

*dte2*: rappresenta la data del modello successivo.

*nmax2*: rappresenta il massimo grado ed ordine del modello successivo.

*gh2*: rappresenta i coefficienti delle armoniche sferiche del modello interno.

Uscite:

*gha* o *b*: rappresentano i coefficienti del modello risultante.

*nmax*: rappresenta il massimo grado ed ordine del conseguente modello.

- *shval3(igdgc : int, flat : float, flon : float, elev : float, nmax : int, gh : int, iext : int, ext1 : float, ext2 : float, ext3 : float) : int*, metodo di calcolo delle componenti del campo magnetico terrestre. Il metodo sfrutta, durante i vari computi, i modelli con le armoniche sferiche. I parametri del metodo sono descritti di seguito:

Ingressi:

*igdgc*: indica il sistema di coordinate utilizzato; impostato pari ad 1 se geodetico, 2 se geocentrico.

*flat*: rappresenta la latitudine nord, espressa in gradi.

*flon*: rappresenta la longitudine est, espressa in gradi.

*elev*: rappresenta l'altitudine WGS84 (World Geodetic System 1984, modello matematico della Terra) sopra il livello medio del mare (se *igdgc* = 1), oppure la distanza radiale dal centro della Terra (se *igdgc* = 2).

*nmax*: rappresenta il massimo grado ed ordine dei coefficienti.

*iext*: rappresenta un flag dei coefficienti esterni.

*ext1, 2, 3*: rappresentano 3 coefficienti esterni del primo grado (non utilizzato se *iext* = 0).

Uscite:

Assegnazione degli attributi *x*, *y* e *z*.

- *dihf(gh : int) : int*, metodo di assegnazione degli attributi *d*, *i* e *h* a partire dalle componenti *x*, *y* e *z* del campo magnetico terrestre ricavate precedentemente.
- *computeDay(year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : float*, metodo che restituisce il numero di giorni (più parte frazionaria) trascorsi a partire dal 1° gennaio 2000 00:00 (metodo già descritto nella classe *Orbits*).

## 11.4.6 Position

Questa classe rappresenta il software indirizzato al calcolo della posizione del satellite rispetto al sistema di riferimento geocentrico inerziale IJK (algoritmo SGP4). La teoria associata a tale classe è descritta nel dettaglio nel capitolo 4.

Gli attributi della classe Position sono descritti di seguito:

- *tsince* : *float*, variabile di tipo float destinata a rappresentare il lasso di tempo, espresso in minuti, trascorso dalla data (compresa di ora) specificata nella TLE, in cui si vuole conoscere la posizione del satellite.
- *tle1* : *char[130]*, array di 130 elementi di tipo char utilizzato per memorizzare la linea 1 di una TLE (si veda capitolo 3, paragrafo 3).
- *tle2* : *char[130]*, array di 130 elementi di tipo char utilizzato per memorizzare la linea 2 di una TLE (si veda capitolo 3, paragrafo 3).
- *satrec* : *elsetrec*, rappresenta un oggetto di tipo elsetrec (si veda classe elsetrec).

I metodi della classe Position sono descritti di seguito:

- *computeSatellitePosition(ro : float[3], vo : float[3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : void*, metodo di calcolo della posizione e della velocità del satellite rispetto al sistema di riferimento geocentrico inerziale IJK. Le componenti del vettore di posizione sono ritornate nell'array *ro* (descritto nella classe Test) ed espresse in [Km], mentre le componenti del vettore di velocità sono ritornate nell'array *vo* (descritto nella classe Test) ed espresse in [Km / s]. Tutti i parametri del metodo sono definiti nella classe Test.
- *computeEarthPosition(re : float[3], year : int, month : int, day : int, hours : int, minutes : int, seconds : float) : void*, metodo di calcolo della posizione del centro della Terra, come visto dal satellite. Il metodo richiama a sua volta il metodo di calcolo della posizione del satellite e ricava le componenti del centro della Terra semplicemente invertendo il segno delle componenti del vettore di posizione del satellite. In altre parole l'origine del sistema di riferimento IJK è traslata dal centro della Terra al centro del satellite. Le componenti del vettore sono ritornate nell'array *re* (descritto nella classe Test) e sono espresse in [Km] (come nel caso di *ro*). Tutti i parametri del metodo sono stati definiti nella classe Test.
- *sgp4(satrec : elsetrec\*, tsince : float, r : float\*, v : float\*) : int*, algoritmo di calcolo della posizione e della velocità del satellite rispetto al sistema di riferimento geocentrico inerziale IJK vero e proprio (algoritmo SGP4, si veda capitolo 4, paragrafo 3). I vettori di posizione e di velocità sono ritornati attraverso i parametri *r* e *v*, che sono gli equivalenti di *ro* e *vo* (descritti nella classe Test). Il metodo restituisce inoltre 0 se non sono stati riscontrati dei problemi ed i seguenti codici di errore nel caso contrario:

1: Eccentricità dell'orbita  $\geq 1.0$ , oppure eccentricità dell'orbita  $< -0.001$ , oppure semiasse maggiore  $< 0.95$  E.R. (Earth Radius).

- 2: Moto medio  $< 0.0$ .
- 3: Eccentricità  $< 0.0$ , oppure eccentricità  $< 1.0$ .
- 4: Semiparametro  $p < 0.0$ .
- 5: Gli elementi dell'epoca sono sub-orbitali.
- 6: Satellite decaduto.
- *getgravconst(tumin : float\*, mu : float\*, radiusearthkm : float\*, xke : float\*, j2 : float\*, j3 : float\*, j4 : float\*, j3oj2 : float\*) : void*, metodo di assegnazione delle costanti utilizzate dall'algoritmo SGP4. Il parametro *tumin* rappresenta i minuti in un'unità di tempo, *mu* rappresenta la costante gravitazionale della Terra [ $\text{Km}^3 \cdot \text{s}^{-2}$ ], *radiusearthkm* rappresenta il raggio della Terra espresso in Km, *xke* rappresenta l'inverso di *tumin*, *j2*, *j3* e *j4* rappresentano i coefficienti armonici zonali, *j3oj2* rappresenta il rapporto tra *j3* e *j2*.
  - *twoline2rv(longstr1 : char[130], longstr2 : char[130], satrec : elsetrec\*) : void*, metodo di conversione delle stringhe componenti una TLE (*longstr1* contiene la prima linea della TLE e *longstr2* contiene la seconda linea della TLE) nelle variabili necessarie all'algoritmo SGP4.
  - *sgp4init(satn : int, epoch : float, xbstar : float, xecco : float, xargpo : float, xinclo : float, xmo : float, xno : float, xnodeo : float, satrec : elsetrec\*) : int*, metodo di inizializzazione delle variabili dell'algoritmo SGP4. Il parametro *satn* rappresenta il numero identificativo del satellite, *epoch* rappresenta il numero di giorni (e frazione di giorno) trascorsi dal 1° gennaio 1950 00:00, *xbstar* rappresenta il coefficiente di resistenza aerodinamica, *xecco* rappresenta l'eccentricità, *xargpo* rappresenta l'argomento del perigeo, *xinclo* rappresenta l'inclinazione dell'orbita, *xmo* rappresenta l'anomalia media, *xno* rappresenta il moto medio, *xnodeo* rappresenta l'ascensione retta del nodo ascendente. Tutti questi parametri sono descritti nella classe *elsetrec* (cambiano soltanto i nomi). Il metodo restituisce 0 se non sono stati riscontrati problemi ed un codice d'errore analogo a quello restituito dal metodo *sgp4()* nel caso contrario.
  - *initl(satn : int, ecco : float, epoch : float, inclo : float, no : float\*, ainv : float\*, ao : float\*, con41 : float\*, con42 : float\*, cosio : float\*, cosio2 : float\*, eccsq : float\*, omeosq : float\*, posq : float\*, rp : float\*, rteosq : float\*, sinio : float\*, gsto : float\*) : void*, metodo di inizializzazione del propagatore SGP4. Il parametro *satn* rappresenta il numero identificativo del satellite, *ecco* rappresenta l'eccentricità dell'orbita, *epoch* rappresenta il numero di giorni (e frazione di giorno) trascorsi dal 1° gennaio 1950 00:00, *inclo* rappresenta l'inclinazione dell'orbita, *no* rappresenta il moto medio del satellite, *ainv* rappresenta l'inverso del semiasse maggiore, *ao* rappresenta il semiasse maggiore, *con41* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $-con42 - cosio2 - cosio2$ ), *con42* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $1.0 - 5.0 \cdot cosio2$ ), *cosio* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $\cos(inclo)$ ), *cosio2* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $cosio^2$ ), *eccsq* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $ecco^2$ ), *omeosq* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $1 - eccsq$ ), *posq* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $(ao \cdot omeosq)^2$ ), *rp* rappresenta il raggio del perigeo (assegnato a:  $ao \cdot (1.0 - ecco)$ ), *rteosq* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $omeosq^{1/2}$ ), *sinio* rappresenta una variabile utilizzata da SGP4 (assegnata a:  $\sin(inclo)$ ), e *gsto* rappresenta il tempo siderale di Greenwich.

- *days2mdhms(year : int, days : float, mon : int\*, day : int\*, hr : int\*, minute : int\*, sec : float\*) : void*, metodo di conversione del giorno dell'anno nell'equivalente mese, giorno, ora, minuti e secondi. Questi parametri sono equivalenti ai parametri *year, month, day, hours, minutes, seconds*, rispettivamente, descritti nella classe Test.
- *jday(year : int, mon : int, day : int, hr : int, minute : int, sec : float, jd : float\*) : void*, metodo di calcolo della data giuliana noto l'anno, il mese, il giorno, e l'ora. La data giuliana è definita come i giorni trascorsi dal 1° gennaio, 4713 a.C. 12:00 ed è restituita attraverso il parametro *jd*. I parametri *year, mon, day, hr, minute, sec* sono equivalenti ai parametri *year, month, day, hours, minutes, seconds*, rispettivamente, descritti nella classe Test.
- *caricaTle(void) : void*, metodo di assegnazione degli attributi *t1e1* e *t1e2*.

### 11.4.7 elsetrec

Questa classe rappresenta la struttura dati contenente i parametri utili nella descrizione dell'orbita del satellite attorno alla Terra.

Gli attributi della classe *elsetrec* sono descritti di seguito:

- *satnum : long int*, variabile di tipo *long int* destinata a rappresentare il numero identificativo del satellite.
- *epochyr : int*, variabile di tipo *int* destinata a rappresentare l'anno (ultime due cifre).
- *error : int*, variabile di tipo *int* destinata a rappresentare l'eventuale codice di errore. Nel caso in cui non sono riscontrati errori questa variabile è settata a 0.
- *method : char*, flag di tipo *char* destinato ad indicare quale routine devono essere utilizzate. Se devono essere utilizzate le routine di calcolo indirizzate alle orbite alte deve essere settato pari a 'd', mentre se devono essere utilizzate le routine di calcolo indirizzate alle orbite vicine alla Terra deve essere settato pari a 'n'. Di fatto, è settato in modo da lavorare con le routine delle orbite basse ('n'). Per orbite basse (le routine indirizzate alle orbite alte sono state opportunamente eliminate), si ricorda, si intendono le orbite con periodo minore di 220 minuti (si veda capitolo 4, paragrafo 3).
- *isimp : int*, variabile di tipo *float* definita internamente all'algoritmo SGP4. Nel caso di orbite vicine alla Terra è settata a 0.
- *aycof : float*, variabile di tipo *float* definita internamente all'algoritmo SGP4. Questa variabile è assegnata ad un valore pari a:  $-0.5 \cdot j3oj2 \cdot sinio$ . La variabile *j3oj2* è definita dal metodo *getgravconst()* della classe *Position*, mentre *sinio* è definita dal metodo *initl()* della stessa classe.

- *con41* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata all'interno del metodo *initl()* della classe Position ad un valore pari a:  $-con42 - cosio2 - cosio2$ .
- *cc1* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $bstar \cdot cc2$ . La variabile *cc2* è definita internamente al metodo *sgp4init()*.
- *cc4* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $2.0 \cdot no \cdot coef1 \cdot ao \cdot omeosq \cdot (eta \cdot (2.0 + 0.5 \cdot etasq) + ecco \cdot (0.5 + 2.0 \cdot etasq) - j2 \cdot tsi / (ao \cdot psisq) \cdot (-3.0 \cdot con41 \cdot (1.0 - 2.0 \cdot eeta + etasq \cdot (1.5 - 0.5 \cdot eeta))) + 0.75 \cdot x1mth2 \cdot (2.0 \cdot etasq - eeta \cdot (1.0 + etasq)) \cdot \cos(2.0 \cdot argpo))$ . Le variabile presenti nella formula sono in parte degli attributi della classe ed in parte delle variabili definite internamente al metodo *sgp4init()*.
- *cc5* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $2.0 \cdot coef1 \cdot ao \cdot omeosq \cdot (1.0 + 2.75 \cdot (etasq + eeta) + eeta \cdot etasq)$ .
- *d2* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $4.0 \cdot ao \cdot tsi \cdot cc1sq$ . Queste variabili sono definite internamente al metodo *sgp4init()*.
- *d3* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $(17.0 \cdot ao + sfour) \cdot temp$ . Queste variabili sono definite internamente al metodo *sgp4init()*.
- *d4* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $0.5 \cdot temp \cdot ao \cdot tsi \cdot (221.0 \cdot ao + 31.0 \cdot sfour) \cdot cc1$ . A parte *cc1* (attributo della classe), tutte le altre variabili sono definite internamente al metodo *sgp4init()*.
- *delmo* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $(1.0 + eta \cdot \cos(mo))^3$ . Tutte le variabili presenti sono degli attributi della classe.
- *eta* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $ao \cdot ecco \cdot tsi$ . A parte *ecco* (attributo della classe), le altre variabili sono definite internamente al metodo *sgp4init()*.
- *argpdot* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $-0.5 \cdot temp1 \cdot con42 + 0.0625 \cdot temp2 \cdot (7.0 - 114.0 \cdot cosio2 + 395.0 \cdot cosio4) + temp3 \cdot (3.0 - 36.0 \cdot cosio2 + 49.0 \cdot cosio4)$ . Queste variabili sono definite internamente al metodo *sgp4init()*.
- *omgcof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $bstar \cdot cc3 \cdot \cos(argpo)$ . La variabile *cc3* è definita internamente al metodo *sgp4init()*, mentre le altre due sono degli attributi della classe.

- *sinmao* : *float*, variabile di tipo float che rappresenta il seno dell'attributo *mo*.
- *t* : *float*, variabile di tipo float destinata a rappresentare il tempo, espresso in minuti, trascorso dalla data specificata nella TLE.
- *t2cof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $1.5 \cdot cc1$ .
- *t3cof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $d2 + 2.0 \cdot cc1sq$ . La variabile *cc1sq* è definita internamente al metodo *sgp4init()*.
- *t4cof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $0.25 \cdot (3.0 \cdot d3 + cc1 \cdot (12.0 \cdot d2 + 10.0 \cdot cc1sq))$ . La variabile *cc1sq* è definita internamente al metodo *sgp4init()*.
- *t5cof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $0.2 \cdot (3.0 \cdot d4 + 12.0 \cdot cc1 \cdot d3 + 6.0 \cdot d2 \cdot d2 + 15.0 \cdot cc1sq \cdot (2.0 \cdot d2 + cc1sq))$ . La variabile *cc1sq* è definita internamente al metodo *sgp4init()*.
- *x1mth2* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $1.0 - cosio2$ . La variabile *cosio2* è definita internamente al metodo *sgp4init()*.
- *x7thm1* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $7.0 \cdot cosio2 - 1.0$ . La variabile *cosio2* è definita internamente al metodo *sgp4init()*.
- *mdot* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $no + 0.5 \cdot temp1 \cdot rteosq \cdot con41 + 0.0625 \cdot temp2 \cdot rteosq \cdot (13.0 - 78.0 \cdot cosio2 + 137.0 \cdot cosio4)$ . A parte le variabili *no* e *con41* (attributi della classe), tutte le variabili sono definite internamente al metodo *sgp4init()*.
- *nodedot* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $xhdot1 + (0.5 \cdot temp2 \cdot (4.0 - 19.0 \cdot cosio2) + 2.0 \cdot temp3 \cdot (3.0 - 7.0 \cdot cosio2)) \cdot cosio$ . Tutte le variabili sono definite internamente al metodo *sgp4init()*.
- *xlcof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $-0.25 \cdot j3oj2 \cdot sinio \cdot (3.0 + 5.0 \cdot cosio) / (1.0 + cosio)$ . Tutte le variabili sono definite internamente al metodo *sgp4init()*.
- *xmcof* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $-x2o3 \cdot coef \cdot bstar / eeta$  se l'attributo *ecco* risulta maggiore di  $10^{-4}$ , altrimenti è assegnata a 0. A

parte la variabile *bstar* (attributo della classe), tutte le variabili sono definite internamente al metodo *sgp4init()*.

- *nodecf* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *sgp4init()* della classe Position ad un valore pari a:  $3.5 \cdot omeosq \cdot xhdot1 \cdot cc1$ . A parte la variabile *cc1* (attributo della classe), tutte le variabili sono definite internamente al metodo *sgp4init()*.
- *a* : *float*, variabile di tipo float destinata a rappresentare il semiasse maggiore dell'orbita del satellite.
- *altp* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *twoline2rv()* della classe Position ad un valore pari a:  $a \cdot (1.0 - ecco) - 1.0$ .
- *alta* : *float*, variabile di tipo float definita internamente all'algoritmo SGP4. Questa variabile è assegnata dal metodo *twoline2rv()* della classe Position ad un valore pari a:  $a \cdot (1.0 + ecco) - 1.0$ .
- *epochdays* : *float*, variabile di tipo int destinata a rappresentare il giorno dell'anno e la porzione di giorno.
- *jdsatepoch* : *float*, variabile di tipo float destinata a rappresentare la data giuliana equivalente alla data specificata nella TLE.
- *nddot* : *float*, variabile di tipo float destinata a rappresentare la derivata seconda temporale del moto medio.
- *ndot* : *float*, variabile di tipo float destinata a rappresentare la derivata prima temporale del moto medio.
- *bstar* : *float*, variabile di tipo float destinata a rappresentare il coefficiente di resistenza aerodinamica.
- *inclo* : *float*, variabile di tipo float destinata a rappresentare l'inclinazione dell'orbita del satellite, espressa in radianti.
- *nodeo* : *float*, variabile di tipo float destinata a rappresentare l'ascensione retta del nodo ascendente, espressa in radianti.
- *ecco* : *float*, variabile di tipo float destinata a rappresentare l'eccentricità dell'orbita del satellite, espressa in radianti.
- *argpo* : *float*, variabile di tipo float destinata a rappresentare l'argomento del perigeo, espresso in radianti.
- *mo* : *float*, variabile di tipo float destinata a rappresentare l'anomalia media dell'orbita del satellite, espressa in radianti.

- *no* : *float*, variabile di tipo float destinata a rappresentare il moto medio del satellite espresso in rivoluzioni al giorno.
- *gsto* : *float*, variabile di tipo float destinata a rappresentare il tempo siderale di Greenwich al tempo dell'osservazione, espresso in radianti.

La classe *elsetrec* non possiede alcun metodo.

## 11.4.8 Satellite

Questa classe rappresenta il software dedicato alla determinazione ed al controllo dell'assetto del satellite. La teoria associata a questa classe è descritta nel dettaglio nei capitoli 7 e 9.

Gli attributi della classe *Satellite* sono descritti di seguito:

- *NUMTILES* : *int const*, costante di tipo int destinata a definire il numero di tiles componenti il satellite (funzione della particolare configurazione del satellite).
- *sat* : *Tile[ NUMTILES ]*, array di *NUMTILES* elementi di tipo *Tile* (si veda classe *Tile*).
- *inertia* : *float[3][3]*, matrice di dimensioni  $(3 \times 3)$  di elementi di tipo float destinata a rappresentare la matrice d'inerzia del satellite, definita rispetto al sistema di riferimento UVW del satellite. I valori dei momenti di inerzia sono espressi in  $[Kg \cdot m^2]$ .
- *E* : *float[3][3]*, matrice di dimensioni  $(3 \times 3)$  di elementi di tipo float destinata a rappresentare la matrice degli autovettori della matrice *inertia*. Questa matrice rappresenta la matrice dei coseni direttori degli assi principali d'inerzia, ABC, rispetto agli assi geometrici UVW del satellite.
- *E\_t* : *float[3][3]*, matrice di dimensioni  $(3 \times 3)$  di elementi di tipo float destinata a rappresentare la matrice trasposta di *E*.
- *autovalori* : *float[3]*, array di 3 elementi di tipo float contenente gli autovalori della matrice *inertia*.
- *Tmin* : *float*, variabile di tipo float destinata a rappresentare il tempo minimo richiesto per effettuare una singola manovra (rotazione attorno ad un asse principale). Questo tempo è espresso in secondi.
- *time* : *float[3]*, array di 3 elementi di tipo float destinato a rappresentare il tempo richiesto, espresso in secondi, per effettuare le singole rotazioni attorno agli assi principali:

*time[0]* = tempo richiesto per effettuare la rotazione attorno all'asse principale A.

*time[1]* = tempo richiesto per effettuare la rotazione attorno all'asse principale B.

*time[2]* = tempo richiesto per effettuare la rotazione attorno all'asse principale C.

- *Lmin* : *float*, variabile di tipo float destinata a rappresentare il minimo momento angolare che una ruota di reazione è in grado di fornire. Questa grandezza è espressa in  $[\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$ .
- *Lmax* : *float*, variabile di tipo float destinata a rappresentare il massimo momento angolare che una ruota di reazione è in grado di fornire. Questa grandezza è espressa in  $[\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$ .
- *L* : *float*, variabile di tipo float destinata a rappresentare il valore di default fissato del momento angolare. Questa grandezza è fissata in modo che risulti essere pari al 15% in meno del valore dettato da *Lmax*.
- *angoli* : *float[3]*, array di 3 elementi di tipo float destinato a rappresentare il valore, espresso in gradi, delle tre rotazioni attorno agli assi principali ABC:

*angoli[0]* = rotazione attorno all'asse principale A.

*angoli[1]* = rotazione attorno all'asse principale B.

*angoli[2]* = rotazione attorno all'asse principale C.

- *L3* : *float[3]*, array di 3 elementi di tipo float destinato a rappresentare il primo comando, ovvero le componenti, rispetto al sistema di riferimento UVW del satellite, del vettore del momento angolare da fornire attorno all'asse principale C. Tali componenti sono espresse in  $[\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$ .
- *L2* : *float[3]*, array di 3 elementi di tipo float destinato a rappresentare il secondo comando, ovvero le componenti, rispetto al sistema di riferimento UVW del satellite, del vettore del momento angolare da fornire attorno all'asse principale B. Tali componenti sono espresse in  $[\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$ .
- *L1*: *float[3]*, array di 3 elementi di tipo float destinato a rappresentare il terzo comando, ovvero le componenti, rispetto al sistema di riferimento UVW del satellite, del vettore del momento angolare da fornire attorno all'asse principale A. Tali componenti sono espresse in  $[\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$ .
- *O* : *Orbits*, definisce un oggetto di tipo *Orbits* (si veda classe *Orbits*).
- *T* : *Tile*, definisce un oggetto di tipo *Tile* (si veda classe *Tile*).
- *M* : *Magnet*, definisce un oggetto di tipo *Magnet* (si veda classe *Magnet*).
- *P* : *Position*, definisce un oggetto di tipo *Position* (si veda classe *Position*).

I metodi della classe *Satellite* sono descritti di seguito:

- *getSpin*(*spin* : *float[3]*) : *bool*, metodo di calcolo del vettore della velocità angolare del satellite rispetto al sistema di riferimento UVW dello stesso. Le componenti sono espresse in  $[\text{rad} / \text{s}]$  e sono ritornate nell'array *spin* (descritto nella classe *Test*). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.

- *getAttitude*(*A* : *float*[3][3], *year* : *int*, *month* : *int*, *day* : *int*, *hours* : *int*, *minutes* : *int*, *seconds* : *float*) : *bool*, metodo di determinazione dell'assetto del satellite. L'assetto è ritornato nella matrice *A* (si veda la classe *Test*). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.
- *get\_magnetic*(*Buvw* : *float*[3]) : *bool*, metodo di calcolo del vettore campo magnetico terrestre, misurato a bordo satellite e specificato nel sistema di riferimento UVW del satellite. Le componenti del vettore sono ritornate nell'array *Buvw* (descritto nella classe *Test*). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.
- *aim*(*Ad* : *float*[3][3], *t\_m* : *float*, *A* : *float*[3][3], *year* : *int*, *month* : *int*, *day* : *int*, *hours* : *int*, *minutes* : *int*, *seconds* : *float*) : *bool*, metodo di calcolo del comando da fornire alle ruote di reazione al fine di orientare il satellite secondo le direzioni specificate nella matrice *Ad* (tutti i parametri sono descritti nella classe *Test*). Il metodo ritorna 0 nel caso in cui non sono stati riscontrati dei problemi.
- *computeSatellitePosition*(*ro* : *float*[3], *vo* : *float*[3], *year* : *int*, *month* : *int*, *day* : *int*, *hours* : *int*, *minutes* : *int*, *seconds* : *float*) : *void*, metodo descritto nella classe *Position*.
- *computeEarthPosition*(*re* : *float*[3], *year* : *int*, *month* : *int*, *day* : *int*, *hours* : *int*, *minutes* : *int*, *seconds* : *float*) : *void*, metodo descritto nella classe *Position*.
- *matrix\_product*(*m1* : *int*, *n1* : *int*, *n2* : *int*, *mat1* : *float*[3][3], *mat2* : *float*[3][3], *outmat* : *float*[3][3]) : *void*, metodo di calcolo del prodotto matriciale. La matrice *outmat* contiene il prodotto tra le matrici *mat1* e *mat2* in ingresso la metodo di calcolo. I parametri *m1*, *n1*, *n2* rappresentano rispettivamente il numero di righe di *mat1*, il numero di colonne di *mat1* ed il numero di colonne di *mat2*.
- *transpose*(*m* : *int*, *n* : *int*, *mat* : *float*[3][3], *outmat* : *float*[3][3]) : *void*, metodo di calcolo della matrice trasposta. Il metodo riceve in ingresso la matrice *mat*, di *m* righe e *n* colonne, e ricava la sua trasposta. La matrice trasposta risultante è ritornata nella matrice *outmat*.
- *cross*(*vec1* : *float*[3], *vec2* : *float*[3], *outvec* : *float*[3]) : *void*, metodo di calcolo del prodotto vettoriale tra due vettori. Il metodo riceve in ingresso due vettori, *vec1* e *vec2*, e ne calcola il prodotto vettoriale. Il risultato è restituito nel vettore *outvec*.
- *normalize*(*vec* : *float*[3]) : *void*, metodo di normalizzazione del modulo di un vettore. Il metodo riceve in ingresso il vettore *vec* e normalizza ad 1 il suo modulo. Il vettore normalizzato è restituito nuovamente nello stesso vettore *vec*.
- *get\_manoeuvre*(*A* : *float*[3][3], *Ad* : *float*[3][3]) : *void*, metodo di calcolo della matrice di manovra (si veda paragrafo 10.2.1, capitolo 10), ovvero della rotazione necessaria ad orientare il satellite secondo le direzioni specificate nella matrice *Ad* (si veda classe *Test*).
- *quaternion\_inverse*(*q1* : *float*[4], *q2* : *float*[4]) : *void*, metodo di calcolo del quaternione inverso. Il metodo riceve in ingresso il quaternione *q1*, ne calcola l'inverso e lo restituisce nel quaternione *q2*.

- *quaternion\_product*(*q1* : float[4], *q2* : float[4], *q3* : float[4]) : void, metodo di calcolo del prodotto tra quaternioni. Il metodo riceve in ingresso due quaternioni, *q1* e *q2*, ne calcola il prodotto e restituisce il risultato nel quaternione *q3*.
- *quaternion\_factorization*(*A* : float[3][3]) : void, metodo di scomposizione di una matrice di rotazione (matrice *A* in ingresso) nella sequenza di tre singole rotazioni successive (si veda paragrafo 10.2.2, capitolo 10). La matrice *A* in ingresso rappresenta la matrice di manovra che si desidera scomporre nella sequenza delle tre singole rotazioni attorno agli assi principali d'inerzia ABC.
- *pinv*(*m* : int, *n*, int, *a* : float\*\*, *q* : float\*, *u* : float\*\*, *v* : float\*\*, *Apinv* : float\*\*) : int, metodo di calcolo della matrice pseudoinversa. La matrice di ingresso *a*, con *m* righe e *n* colonne, viene scomposta mediante la scomposizione ai valori singolari (*u*, *q*, *v*) e viene calcolata la sua pseudoinversa (*Apinv*). La scomposizione ai valori singolari permette di scrivere la matrice *a* come il seguente prodotto:

$a = u \cdot \Sigma \cdot v^T$ , dove *u* è una matrice di dimensioni (*m* × *m*),  $\Sigma$  è una matrice diagonale di dimensioni (*m* × *n*) avente elementi diversi da zero solo sulla diagonale (elementi dell'array *q*) e *v* è una matrice di dimensioni (*n* × *n*).

- *caricaDati*(void) : void, metodo di assegnazione degli attributi *E*, *E\_t*, *autovalori*, *inertia* ed alcuni campi dell'oggetto *sat* (*num*, *tileO*, *sun*, *magnetometer*, *gyroscope*, *RW*, si veda la classe *Tile* per una descrizione di questi attributi).
- *compute\_angular\_momemtum*(*angle* : float[3], *time\_m* : float) : int, metodo di calcolo dei comandi (momenti angolari) da fornire alle ruote di reazione. I valori di momento angolare calcolati vengono assegnati agli attributi *L1*, *L2* e *L3*, mentre i tempi relativi alle singole rotazioni vengono assegnati all'attributo *time*. Il parametro *time\_m* rappresenta l'attributo *t\_m* descritto nella classe *Test*, mentre *angle* rappresenta l'attributo *angoli* opportunamente trasformato in radianti.

## 11.4.9 Tile

La classe *Tile* rappresenta il software dedicato alla descrizione delle tiles costituenti il satellite. Gli attributi della classe sono descritti di seguito:

- *num* : int, variabile di tipo int destinata a rappresentare il numero identificativo della particolare tile componente il satellite. La numerazione parte da 0, per convenzione.
- *tileO* : float[3][3], matrice di dimensioni (3 × 3) di elementi di tipo float destinata a rappresentare la matrice dei coseni direttori del sistema di riferimento XYZ della tile rispetto al sistema di riferimento del satellite UVW.
- *sun* : bool, flag destinato ad indicare se la tile possiede il sensore di Sole:

*sun* = 1: sensore di Sole presente.  
*sun* = 0: sensore di Sole non presente.

- *magnetometer* : *bool*, flag destinato ad indicare se la tile possiede il sensore di campo magnetico:

*magnetometer* = 1: magnetometro presente.  
*magnetometer* = 0: magnetometro non presente.

- *gyroscope* : *bool*, flag destinato ad indicare se la tile possiede il sensore giroscopico:

*gyroscope* = 1: giroscopio presente.  
*gyroscope* = 0: giroscopio non presente.

- *RW* : *bool*, flag destinato ad indicare se la tile possiede la ruota di reazione:

*RW* = 1: ruota di reazione presente.  
*RW* = 0: ruota di reazione non presente.

I metodi della classe *Tile* sono descritti di seguito:

- *readSunSensor(posizione : float[2])* : *bool*, metodo di interrogazione del sensore di Sole. I due angoli indicanti la posizione del Sole sono ritornati nell'array *posizione* nel seguente modo:

*posizione[0]* = angolo, espresso in gradi, misurato sul piano Z-Y della tile a partire dall'asse Z e positivo verso Y.

*posizione[1]* = angolo, espresso in gradi, misurato dal piano Z-Y verso la direzione del Sole, positivo verso X.

Il metodo ritorna 0 se i due angoli sono stati letti correttamente, 1 viceversa. Si noti che, come annunciato nel paragrafo 6.2.1 del capitolo 6, il sensore di posizione del Sole restituisce le coordinate sferiche del Sole espresse in unità di 0.0001 rad e come due interi su 16 bit con segno. Il sistema ACS, invece, si aspetta due dati espressi in gradi nella notazione in floating-point, come descritto. Il metodo in esame si preoccupa quindi di effettuare il cast delle variabili, di dividere opportunamente il valore letto per  $10^4$  e di effettuare la trasformazione del dato da radianti in gradi.

- *sunSensorIsLit()* : *bool*, il valore ritornato da questo metodo indica se il sensore di Sole è illuminato. Il metodo restituisce 0 se la tile non possiede il sensore di Sole.
- *readMagnetometer(Bxy : float[2])* : *bool*, metodo di interrogazione del sensore di campo magnetico. Le due componenti del vettore campo magnetico terrestre misurato vengono ritornate nell'array *Bxy* nel seguente modo:

*Bxy[0]* = componente in X del campo magnetico terrestre misurato, espressa in unità di 10 nT.

*Bxy[1]* = componente in Y del campo magnetico terrestre misurato, espressa in unità di 10 nT.

Il metodo ritorna 0 se le due componenti sono state lette correttamente, 1 viceversa. Si noti che, come annunciato nel paragrafo 6.2.2 del capitolo 6, il sensore di campo magnetico restituisce le componenti del vettore campo magnetico terrestre espresse in unità di 10 nT e come due interi su 16 bit con segno. Il sistema ACS, invece, si aspetta due dati espressi in unità di 10 nT nella notazione in floating-point, come descritto. Il metodo in esame si preoccupa quindi di effettuare il cast delle variabili.

- *readSpin(spinletto : float\*) : bool*, metodo di interrogazione del sensore giroscopico. Il valore misurato, ovvero la velocità angolare attorno all'asse Z della tile espressa in [rad / s], è ritornato nella variabile *spinletto*. Il metodo ritorna 0 se il valore dello spin è stato letto correttamente, 1 viceversa. Si noti che, come annunciato nel paragrafo 6.2.3 del capitolo 6, il sensore giroscopico restituisce la velocità angolare attorno all'asse Z della tile espressa in unità di 0.0001 rad / s e come un intero su 16 bit con segno. Il sistema ACS, invece, si aspetta un dato espresso in [rad / s] nella notazione in floating-point, come descritto. Il metodo in esame si preoccupa quindi di effettuare il cast delle variabili e di dividere opportunamente il valore letto per  $10^4$ .
- *actuateReactionWheel() : bool*, metodo di attuazione delle ruote di reazione. I valori di momento angolare calcolati sono passati alle varie ruote di reazione disposte diversamente sulla struttura del satellite. Il metodo restituisce 0 se non sono stati riscontrati dei problemi.
- *hasSunSensor() : bool*, metodo che restituisce il flag *sun*.
- *hasMagnetometer() : bool*, metodo che restituisce il flag *magnetometer*.
- *hasGyroscope() : bool*, metodo che restituisce il flag *gyroscope*.
- *hasReactionWheel() : bool*, metodo che restituisce il flag *RW*.

# Capitolo 12

## 12 I test effettuati sul sistema ACS

In questo capitolo si riportano alcune prove effettuate sul sistema ACS da un punto di vista complessivo al fine di mostrare il suo effettivo funzionamento. Nei test seguenti si prende in esame la semplice configurazione del satellite a piccolo cubo, formata da sei tiles. Nella fase di test è stato sviluppato un programma `main()` in modo che interagisse con tutti gli algoritmi facenti parte del sistema ACS (classe `Test` descritta nel capitolo precedente). Il `main()` sviluppato, di fatto, può essere interpretato come il mondo esterno che interagisce con il sistema ACS. Durante le prove effettuate è stata fissata una data di test, una TLE di test (di un nano-satellite reale), un assetto attuale del satellite, un assetto desiderato del satellite, un tensore d'inerzia del satellite e l'orientamento delle sei tiles, ovvero dei sistemi di riferimento XYZ, componenti la struttura del piccolo satellite cubico rispetto al sistema di riferimento UVW del satellite. Una volta fissate tutte queste grandezze è possibile calcolare, interrogando i metodi interni del sistema ACS, la posizione del satellite in IJK, la posizione del Sole in IJK, il vettore campo magnetico in IJK. A questo punto, conoscendo l'assetto del satellite (fissato) risulta possibile ricavare i valori attesi dai sensori magnetici e di posizione del Sole. In questo modo si può valutare l'assetto mediante gli algoritmi appositi e valutare se l'orientamento ottenuto si avvicina a quello fissato in principio. In seguito si può valutare se il controllo dell'orientamento è efficace, osservando l'assetto raggiunto dopo la manovra imposta (simulazione con Simulink).

### 12.1 Le grandezze di test

I valori fissati nelle prove effettuate sono elencati di seguito:

**TLE di test:**

DTUSAT

```
1 27842U 03031C 09166.20608466 .00000024 00000-0 31026-4 0 5778
2 27842 98.7036 176.3713 0009831 124.5029 235.7083 14.20982944309069
```

**Data specificata nella TLE:** 15 Jun 2009 04:56:45.71

**Data di test fissata:** 16 Jun 2009 00:00:0.0

**Tensore d'inerzia di test del satellite:**

$$\underline{\underline{I}} = \begin{bmatrix} 0.007252141800000 & -0.000011600796000 & 0.000002868117000 \\ -0.000011600796000 & 0.002740675100000 & -0.000056058867000 \\ 0.000002868117000 & -0.000056058867000 & 0.007292028300000 \end{bmatrix} [\text{Kg} \cdot \text{m}^2]$$

**Assetto del satellite fissato:**

Matrice dei coseni direttori:

$$\underline{\underline{A}} = \begin{bmatrix} 0.612372435695794 & -0.612372435695795 & 0.500000000000000 \\ 0.707106781186548 & 0.707106781186547 & 0 \\ -0.353553390593274 & 0.353553390593274 & 0.866025403784439 \end{bmatrix}$$

Angoli di Eulero corrispondenti alla matrice precedente:

$$\begin{cases} \Phi = 0^\circ \\ \theta = -30^\circ \\ \Psi = -45^\circ \end{cases}$$

**Assetto del satellite desiderato:**

Matrice dei coseni direttori:

$$\underline{\underline{A}}_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

Come risulta semplice osservare l'assetto desiderato del satellite, specificato rispetto al sistema di riferimento geocentrico inerziale IJK, è dettato dall'orientamento che si ottiene ruotando di 90° attorno all'asse inerziale I.

Gli angoli di Eulero corrispondenti a tale assetto desiderato, infatti, sono dati da:

$$\begin{cases} \Phi = 90^\circ \\ \theta = 0^\circ \\ \Psi = 0^\circ \end{cases}$$

Si riporta nella figura seguente l'orientamento dei sistemi di riferimento XYZ delle tiles fissato rispetto al sistema UVW (in figura le parti non in vista sono rappresentate a tratto discontinuo ed in colore celeste, mentre le parti in vista sono riportate a tratto continuo ed in colore blu):

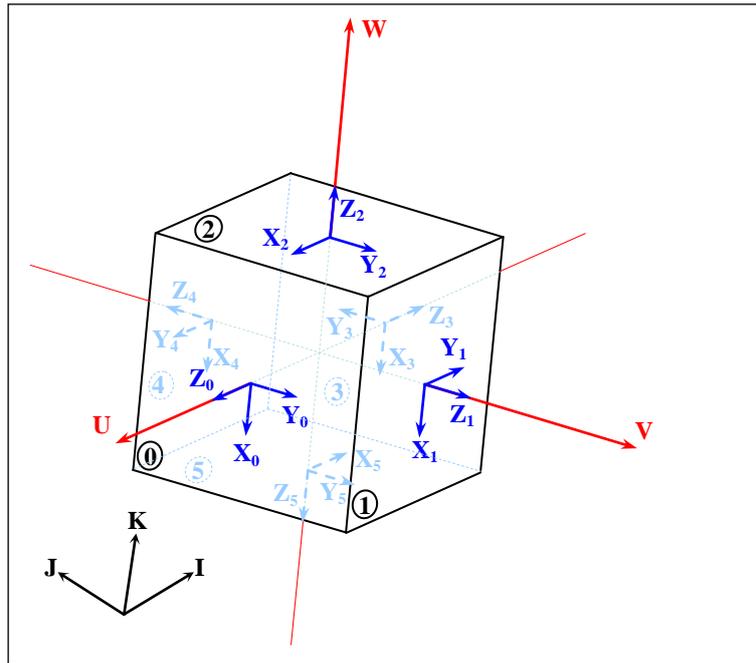


Figura 12.1 Orientamento delle tiles fissato rispetto al sistema del satellite UVW.

Le singole matrici di rotazione dei sistemi di riferimento delle sei tiles componenti il satellite in esame rispetto al sistema UVW vengono riportate di seguito (si indica con  $\underline{A}_i$  la matrice di rotazione della i-esima tile rispetto al sistema UVW):

$$A_0 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ -1 & 0 & 0 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix},$$

$$A_5 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

In fase di test è stato imposto che la tile 0 sia l'unica ad essere di tipo Telecommunication (non possiede né sensori, né attuatori d'assetto e quindi ai fini dei vari algoritmi del sistema ACS non viene presa in considerazione).

Si riportano di seguito i valori ottenuti interrogando i metodi di calcolo della posizione del Sole, del satellite e del campo magnetico del sistema ACS (i modelli):

**Posizione del satellite in IJK calcolata,  $\vec{r}$  :**

$$\vec{r} = [4097.62841796875 \ -1100.9205322265625 \ -5829.1435546875] \text{ [Km]}$$

**Posizione del Sole in IJK calcolata:**

$$RA = 84.553993225097656000^\circ$$

$$Dec = 23.343618392944336000^\circ$$

Indicando con  $\hat{S}$  il versore (vettore normalizzato ad 1) contenente le componenti cartesiane associate si ottiene:

$$\hat{S} = [0.087139019396207 \quad 0.914000557233585 \quad 0.396244586935094]$$

### Vettore campo magnetico in IJK calcolato:

$$\vec{B}_{ijk} = [8972.353515625 \quad -2366.843017578125 \quad 18176.728515625] \text{ [unità di 10 nT]}$$

Ottenuti questi valori risulta possibile ricavare le grandezze che dovranno essere misurate (grandezze attese) dai sensori di campo magnetico e dai sensori di posizione del Sole, in modo da poter effettuare i vari test sulla determinazione dell'assetto e sulla manovra del satellite.

### Vettore campo magnetico in UVW determinato:

$$\vec{B}_{uvw} = \underline{\underline{A}} \cdot \vec{B}_{ijk} = [16032.17565768340 \quad 4670.80126636788 \quad 11732.49727130680] \text{ [unità di 10 nT]}$$

### Vettore $\vec{\Phi}^T$ ricavato:

Si ricorda che il vettore  $\vec{\Phi}^T$  contiene le componenti, in X e Y del sistema di riferimento delle tiles, del vettore campo magnetico misurate dai vari magnetometri presenti sul satellite. In fase di test sono stati ricavati i valori corretti di tale vettore, in funzione della configurazione fissata del satellite:

$$\vec{\Phi}^T = [-11732.49727130680 \\ -16032.17565768340 \\ 16032.17565768340 \\ 4670.80126636788 \\ -11732.49727130680 \\ -4670.80126636788 \\ -11732.49727130680 \\ 16032.17565768340 \\ -16032.17565768340 \\ 4670.80126636788] \text{ [unità di 10 nT]}$$

### Tiles illuminate:

Dato l'assetto del satellite fissato e la posizione del Sole ottenuta per la particolare data in esame, tre tiles del satellite risultano essere illuminate dalla luce solare (le tiles indicate con i numeri 1, 2 e 3). Si riportano di seguito gli angoli attesi forniti dai sensori di posizione del Sole (valori ideali, calcolati in funzione della particolare situazione in analisi):

- **Tile 1: posizione\_sole** =  $[a_1 \ b_1] = [39.553993225097692^\circ \ 23.343618392944336^\circ]$
- **Tile 2: posizione\_sole** =  $[a_2 \ b_2] = [-50.446006774902344^\circ \ -36.656381607055657^\circ]$
- **Tile 3: posizione\_sole** =  $[a_3 \ b_3] = [-50.446006774902287^\circ \ 53.343618392944336^\circ]$

## 12.2 Test 1: valori esatti forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 5 magnetometri funzionanti

Il primo test effettuato verifica l'algoritmo di calcolo della matrice d'assetto e della manovra del satellite avendo posto come informazione provenienti dai vari sensori i valori ideali calcolati precedentemente (5 magnetometri funzionanti e tre tiles illuminate con rispettivi sensori di Sole funzionanti):

**Matrice d'assetto d'errore** ( $\underline{\underline{A}}_{\text{ottenuta}} \cdot \underline{\underline{A}}^T$ ):

$$\underline{\underline{A}}_{eo} = \begin{bmatrix} 0.999999785187973 & 0.000000042146849 & -0.000000013629249 \\ -0.000000037991276 & 0.999999898592032 & 0.000000102784575 \\ -0.000000073361598 & 0 & 1.000000024407108 \end{bmatrix}$$

Questa matrice d'errore (ottenuta dal prodotto tra la matrice d'assetto ottenuta mediante l'algoritmo di calcolo del sistema ACS e la trasposta della matrice d'assetto fissata in principio) mostra di quanto l'assetto ottenuto si discosta dall'assetto ideale fissato in precedenza. Nel caso ideale questa matrice d'errore dovrebbe avere la forma seguente:

$$\underline{\underline{A}}_{ei} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Come risulta semplice osservare la matrice d'errore ottenuta si discosta di molto poco dalla matrice d'errore ideale. Al fine di quantificare meglio queste piccole differenze, dovute sostanzialmente agli arrotondamenti numerici presenti nei vari computi, si ricavano gli angoli di Eulero associati alla matrice d'errore ottenuta.

**Angoli di Eulero di errore corrispondenti alla matrice d'assetto d'errore ottenuta:**

$$\begin{cases} \Phi = 5.889 \cdot 10^{-6} \\ \theta = 7.809 \cdot 10^{-6} \\ \Psi = 2.415 \cdot 10^{-6} \end{cases}$$

L'ordine di grandezza di questi angoli d'errore risulta essere decisamente piccolo e questo dimostra la correttezza dell'algoritmo di determinazione dell'assetto sviluppato.

Al fine di quantificare l'errore di puntamento espresso in Km, causato dalla non perfetta determinazione dell'assetto è stata fissata una direzione di test, indicata dal simbolo  $\vec{d}$ , specificata rispetto al sistema UVW del satellite. In particolare è stata fissata una direzione di test situata ad una distanza di 500 Km sull'asse del satellite U.

**Direzione di test rispetto al sistema UVW:**

$$\vec{d} = [500 \ 0 \ 0] \text{ [Km]}$$

A causa dell'errore presente sulla determinazione dell'assetto, il vettore  $\vec{d}$  viene sporcato a sua volta da errori:

**Errore sulle componenti:**

$$\vec{d}_e = \underline{\underline{A}}_e \cdot \vec{d}^T = [1.074060133987587 \cdot 10^{-4} \quad -0.189956378 \cdot 10^{-4} \quad -0.366807992 \cdot 10^{-4}] \text{ [Km]}$$

**Radice quadrata della somma delle differenze al quadrato:**

$$d_e = \sqrt{(\vec{d}_e(1))^2 + (\vec{d}_e(2))^2 + (\vec{d}_e(3))^2} = 1.150754839207458 \cdot 10^{-4} \text{ Km}$$

Si riporta anche la matrice d'errore valutata come la differenza tra la matrice d'assetto fissata e la matrice d'assetto ottenuta:

**Matrice d'assetto di errore ( $\underline{\underline{A}} - \underline{\underline{A}}_{ottenuta}$ ):**

$$\underline{\underline{A}}_{e2} = \begin{bmatrix} 0.096923973957885 \cdot 10^{-6} & -0.156528619732477 \cdot 10^{-6} & 0.119209289550781 \cdot 10^{-6} \\ 0.131310907147686 \cdot 10^{-6} & 0.012101616597704 \cdot 10^{-6} & -0.070018415954110 \cdot 10^{-6} \\ 0.053553835976938 \cdot 10^{-6} & -0.053553835976938 \cdot 10^{-6} & 0.015543624831693 \cdot 10^{-6} \end{bmatrix}$$

A questo punto è possibile testare gli algoritmi indirizzati al controllo dell'orientamento del satellite. Si riportano le grandezze ottenute:

**Vettore time:**

time = [0.300000011920928960 0.300000011920928960 0.300000011920928960] [s]

Questo vettore contiene i tempi necessari per eseguire le singole rotazioni attorno agli assi principali del satellite ABC.

**Vettore angle:**

angle = [-1.215568900108337400 2.842236995697021500 -2.231189250946044900] [rad]

Questo vettore contiene le singole rotazioni, espresse in radianti, da effettuare attorno agli assi principali ABC del satellite.

**Vettore L1:**

[-0.029304251074790955 0.000048455294745509 0.002164823003113270] [ $\text{Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}$ ]

Questo vettore contiene le componenti in UVW del momento angolare relativo alla terza rotazione, attorno ad A.

**Vettore L2:**

[0.000066531109041534 0.025956617668271065 0.000319614395266399]

Questo vettore contiene le componenti in UVW del momento angolare relativo alla seconda rotazione, attorno a B.

**Vettore L3:**

[-0.003994598519057035 0.000676249270327389 -0.054088253527879715]

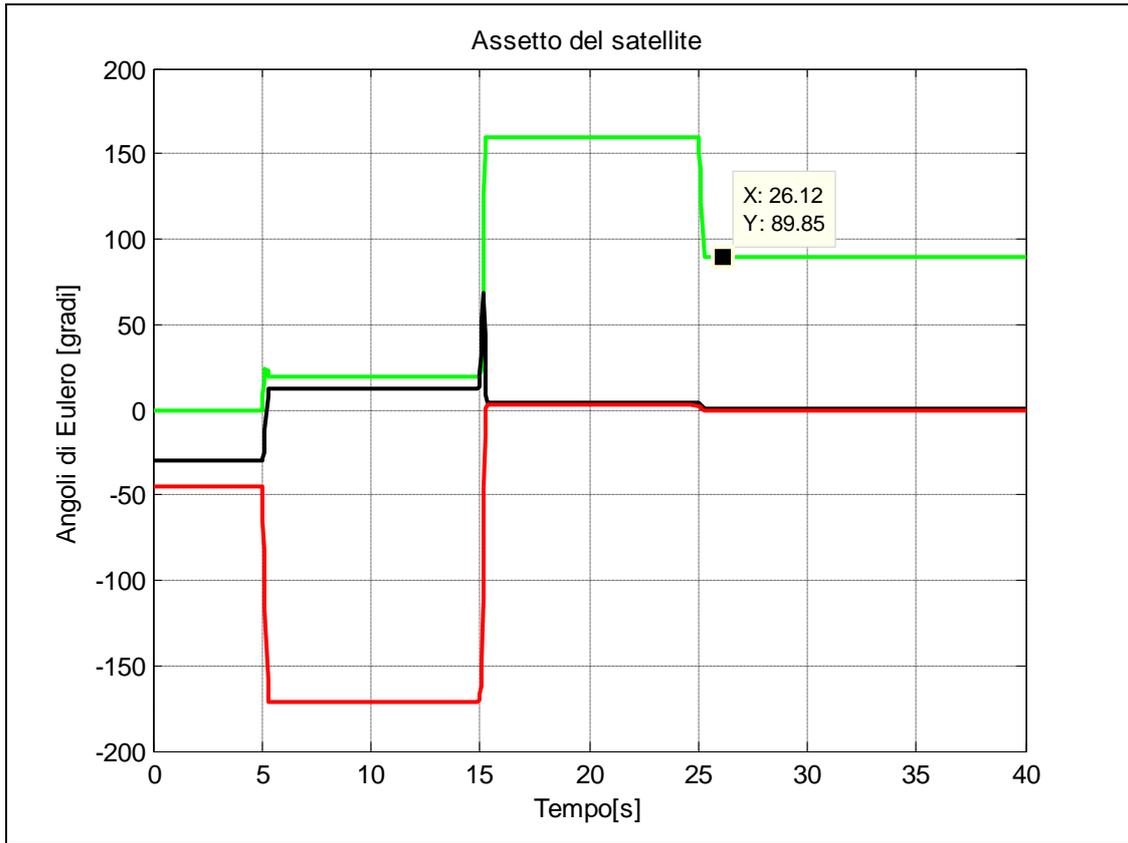
Questo vettore contiene le componenti in UVW del momento angolare relativo alla prima rotazione, attorno a C.

**Risultati simulazioni con Simulink:**

Con lo scopo di verificare il funzionamento dell'algoritmo di controllo dell'assetto, i valori di momento ricavati sono stati inseriti nel modello Simulink del satellite ed è stato osservato il comportamento dell'assetto del satellite ottenuto.

Si riporta l'assetto del satellite ottenuto in IJK in due casi:

- Assetto iniziale imposto nel blocco 6DoF uguale all'assetto fissato in principio:



**Figura 12.2 Assetto del satellite in IJK dopo la manovra.**

Nel grafico precedente le singole rotazioni sono separate molto sulla scala dei tempi, al fine di migliorare la lettura del grafico stesso.

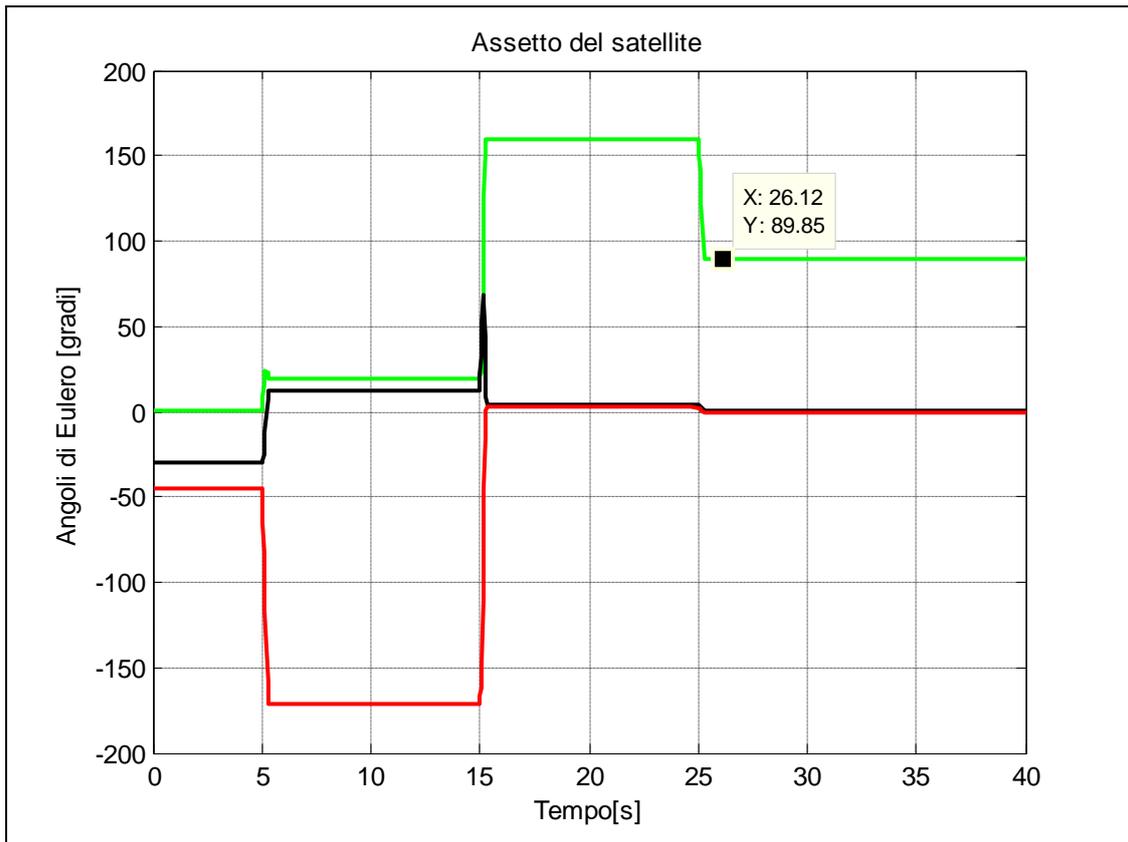
L'assetto ottenuto dopo la manovra risulta essere:

$$\begin{cases} \Phi = 89.851617867464^\circ \\ \theta = 0.13726697150576^\circ \\ \Psi = -0.048789357641226^\circ \end{cases}$$

Errore sull'assetto ottenuto:

$$\begin{cases} \Phi = 0.148382132536000^\circ \\ \theta = 0.13726697150576^\circ \\ \Psi = -0.048789357641226^\circ \end{cases}$$

- Assetto iniziale imposto nel blocco 6DoF uguale all'assetto ottenuto con il metodo apposito del sistema ACS:



**Figura 12.3 Assetto del satellite in IJK dopo la manovra.**

Assetto ottenuto dopo la manovra:

$$\begin{cases} \Phi = 89.851410129385^\circ \\ \theta = 0.137199114423830^\circ \\ \Psi = -0.049154401133131^\circ \end{cases}$$

Errore sull'assetto ottenuto:

$$\begin{cases} \Phi = 0.148589870614998^\circ \\ \theta = -0.137199114423830^\circ \\ \Psi = 0.049154401133131^\circ \end{cases}$$

## 12.3 Test 2: rumore sui valori forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 5 magnetometri funzionanti

In questa prova è stata contemplata la presenza del rumore sulle informazioni fornite dai sensori d'assetto del satellite. I valori calcolati in precedenza sono stati sporcati con un errore medio dell'1% al fine di valutare le risposte del sistema ad un peggioramento dell'accuratezza delle informazioni note.

**Vettore  $\vec{\Phi}^T$ :**

$$\vec{\Phi}^T = \begin{bmatrix} -11732.49727130680 \\ -16112.33653597182 \\ 15871.85390110657 \\ 4740.86328536340 \\ -11145.87240774146 \\ -4670.80126636788 \\ -11849.82224401987 \\ 15952.01477939499 \\ -16112.33653597182 \\ 4600.73924737236 \end{bmatrix} \text{ [unità di 10 nT]}$$

**Tiles illuminate:**

- **Tile 1: posizione\_sole** =  $[a_1 \ b_1] = [39.751763191223183^\circ \ 23.110182209014894^\circ]$
- **Tile 2: posizione\_sole** =  $[a_2 \ b_2] = [-49.689316673278810^\circ \ -36.839663515090933^\circ]$
- **Tile 3: posizione\_sole** =  $[a_3 \ b_3] = [-50.193776741027776^\circ \ 52.543464117050171^\circ]$

**Matrice d'assetto di errore ( $\underline{A}_{\text{ottenuta}} \cdot \underline{A}^T$ ):**

$$\underline{A}_{\text{eo}} = \begin{bmatrix} 1.002426189268299 & -0.004349217591536 & -0.001451863972351 \\ 0.004548433785015 & 1.000232001286781 & -0.005993505252486 \\ 0.007938882085486 & -0.003566929936326 & 0.995575797410828 \end{bmatrix}$$

**Angoli di Eulero di errore corrispondenti:**

$$\begin{cases} \Phi = -0.34492442252936^\circ \\ \theta = 0.083185707267478^\circ \\ \Psi = -0.24858712915619^\circ \end{cases}$$

**Direzione di prova in UVW:**

$$\vec{d} = [500 \ 0 \ 0] \text{ [Km]}$$

**Errore sulle componenti:**

$$\vec{d}_e = [-1.213094634149513 \quad 2.2742168925075 \quad 3.9694410427430] \text{ [Km]}$$

**Radice quadrata della somma delle differenze al quadrato:**

$$d_e = \sqrt{(\vec{d}_e(1))^2 + (\vec{d}_e(2))^2 + (\vec{d}_e(3))^2} = 4.732876847899337 \text{ Km}$$

Si nota chiaramente che l'errore sulla distanza di prova cresce in modo sostanziale a causa della determinazione peggiore dell'assetto del satellite.

**Matrice d'assetto di errore ( $\underline{\underline{A}} - \underline{\underline{A}}_{ottenuta}$ ):**

$$\underline{\underline{A}}_{e2} = \begin{bmatrix} 0.001076318390038 & 0.005074404113623 & 0.000044256448746 \\ -0.005068409262213 & 0.004740309895959 & 0.002916310913861 \\ -0.003903544039472 & 0.008947944731458 & -0.000137969209030 \end{bmatrix}$$

**Vettore time:**

$$\text{time} = [0.300000011920928960 \ 0.300000011920928960 \ 0.300000011920928960] \text{ [s]}$$

**Vettore angle:**

$$\text{angle} = [-1.215823054313659700 \ 2.843599319458007800 \ -2.222789764404296900] \text{ [rad]}$$

**Vettore L1:**

$$L_1 = [-0.029310379177331924 \ 0.000048465426516486 \ 0.002165275625884533] \text{ [Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}]$$

**Vettore L2:**

$$L_2 = [0.000066562999563757 \ 0.025969062000513077 \ 0.000319767539622262] \text{ [Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}\text{]}$$

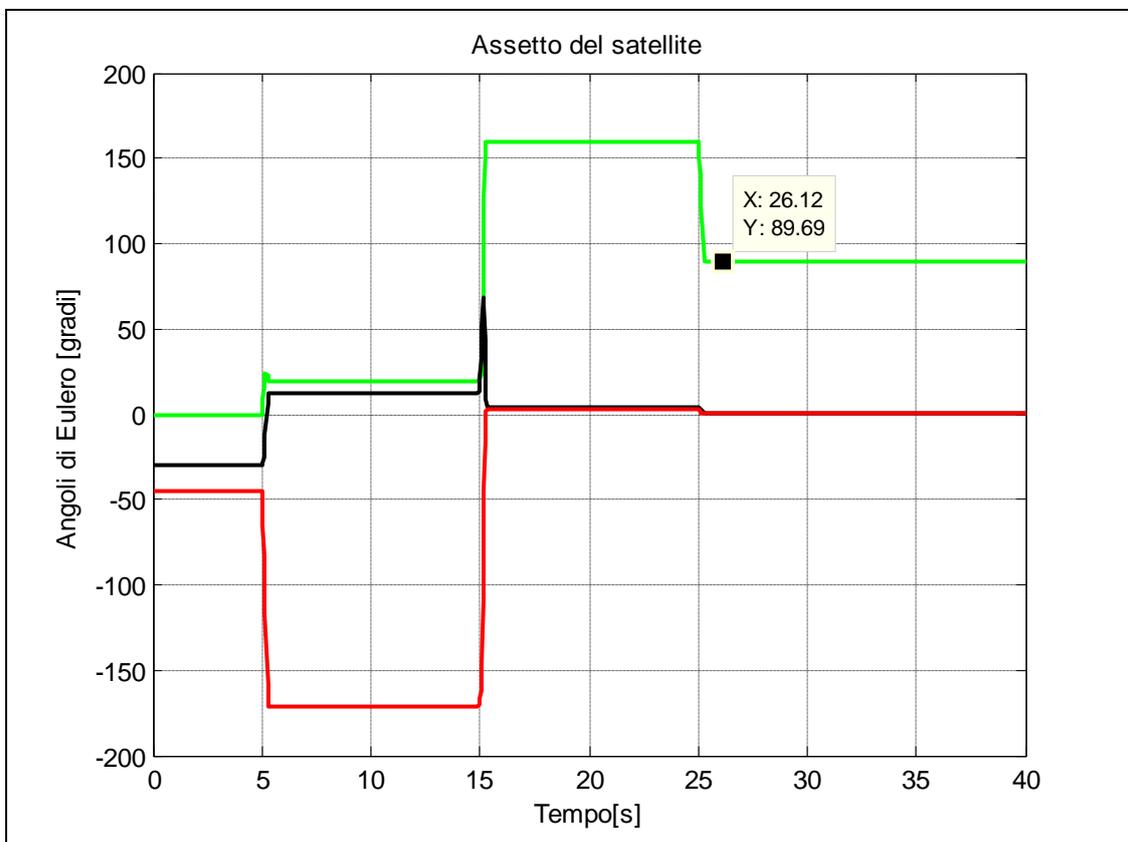
**Vettore L3:**

$$L_3 = [-0.003979560453444719 \ 0.000673703441862017 \ -0.053884632885456085] \text{ [Kg} \cdot \text{m}^2 \cdot \text{s}^{-1}\text{]}$$

**Risultati simulazioni con Simulink:**

Si riporta l'assetto del satellite in IJK in due casi:

- Assetto iniziale imposto nel blocco 6DoF uguale all'assetto fissato in principio:



**Figura 12.4 Assetto del satellite in IJK dopo la manovra.**

Assetto ottenuto dopo la manovra:

$$\begin{cases} \Phi = 89.694277527378^\circ \\ \theta = 0.19889619502584^\circ \\ \Psi = 0.40942575576958^\circ \end{cases}$$

Errore sull'assetto ottenuto:

$$\begin{cases} \Phi = 0.305722472621994^\circ \\ \theta = -0.19889619502584^\circ \\ \Psi = -0.40942575576958^\circ \end{cases}$$

- Assetto iniziale imposto nel blocco 6DoF uguale all'assetto ottenuto con il metodo apposito del sistema ACS:

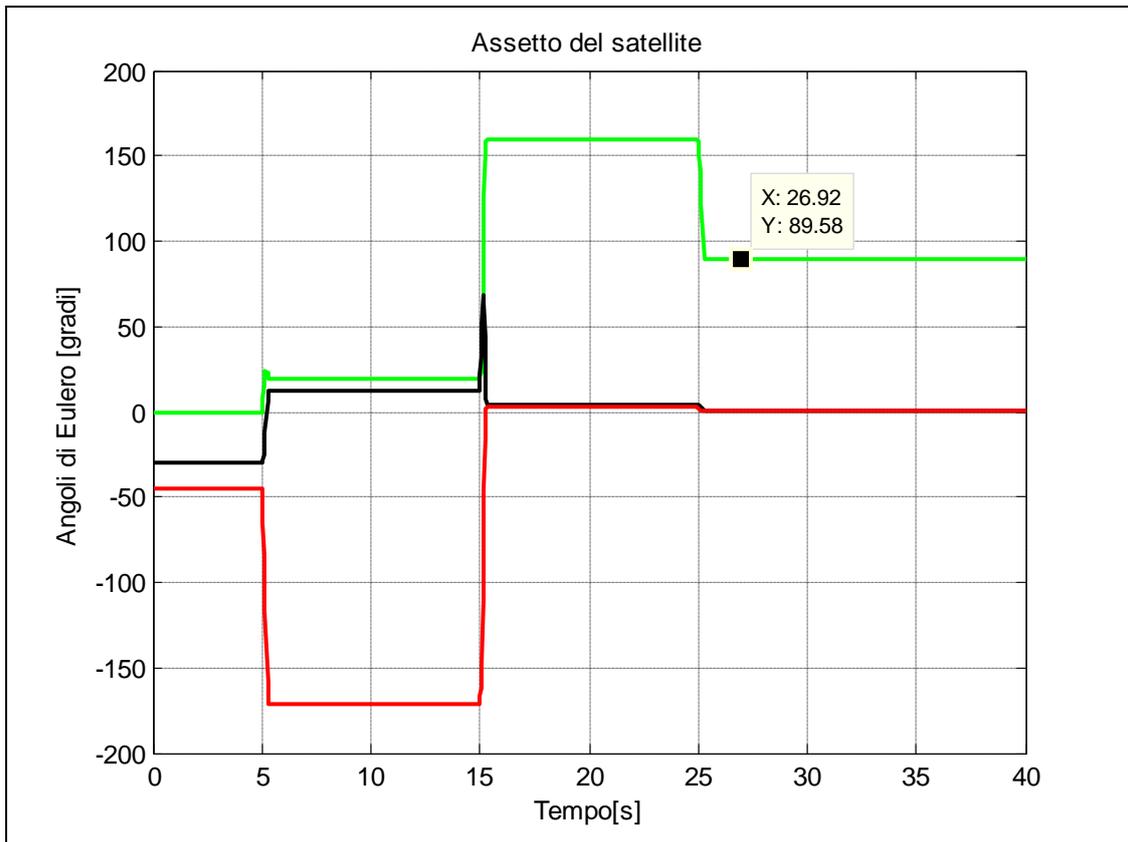


Figura 12.5 Assetto del satellite in IJK dopo la manovra.

Assetto ottenuto dopo la manovra:

$$\begin{cases} \Phi = 89.580786338592^\circ \\ \theta = 0.31433386286868^\circ \\ \Psi = 0.023403010539425^\circ \end{cases}$$

Errore sull'assetto ottenuto:

$$\begin{cases} \Phi = 0.419213661407994^\circ \\ \theta = -0.31433386286868^\circ \\ \Psi = -0.023403010539425^\circ \end{cases}$$

## 12.4 Test 3: rumore sui valori forniti dai sensori, 3 tiles illuminate con sensori di Sole funzionanti e 2 magnetometri funzionanti

In questo test è stato osservato il peggioramento sull'accuratezza della determinazione dell'assetto, nel caso in cui siano presenti un numero di magnetometri inferiore (soltanto le tiles 1 e 2 possiedono il magnetometro nel test in analisi). I valori utilizzati sono identici a quelli del test precedente.

**Vettore  $\vec{\Phi}^T$ :**

$$\vec{\Phi}^T = \begin{bmatrix} -11732.49727130680 \\ -16112.33653597182 \\ 15871.85390110657 \\ 4740.86328536340 \end{bmatrix} \text{ [unità di 10 nT]}$$

**Matrice d'assetto di errore ( $\underline{\underline{A}}_{\text{ottenuta}} \cdot \underline{\underline{A}}^T$ ):**

$$\underline{\underline{A}}_{eo} = \begin{bmatrix} 1.040219046506044 & 0.006296865608074 & 0.026303472892048 \\ -0.017243875219335 & 0.994093186360624 & -0.021997945620099 \\ -0.046321223106564 & -0.018851779576741 & 0.955726770509800 \end{bmatrix}$$

Come risulta semplice osservare gli errori sono maggiori.

**Matrice d'assetto di errore ( $\underline{\underline{A}} - \underline{\underline{A}}_{\text{ottenuta}}$ ):**

$$\underline{\underline{A}}_{e2} = \begin{bmatrix} -0.019781909816596 & 0.010876797073217 & -0.042888998985291 \\ 0.006958973588434 & 0.001394522370782 & 0.027672717347741 \\ 0.026043110995547 & 0.000617331356748 & 0.061502352999374 \end{bmatrix}$$

## 12.5 Test 4: rumore sui valori forniti dai sensori, 1 tile illuminata con sensore di Sole funzionante e 2 magnetometri funzionanti

In questo test sono stati utilizzati gli stessi valori del test precedente, ma ci si è posti nella situazione in cui sia presente un solo sensore di posizione del Sole.

**Matrice d'assetto di errore ( $\underline{\underline{A}}_{\text{ottenuta}} \cdot \underline{\underline{A}}^T$ ):**

$$\underline{\underline{A}}_{eo} = \begin{bmatrix} 0.993514126189986 & -0.005324074197593 & 0.183030339677962 \\ -0.043969238374707 & 1.000296738846094 & -0.034078974274671 \\ -0.171811509319546 & -0.000916672881688 & 0.987626627354770 \end{bmatrix}$$

**Matrice d'assetto di errore ( $\underline{\underline{A}} - \underline{\underline{A}}_{\text{ottenuta}}$ ):**

$$\underline{\underline{A}}_{e2} = \begin{bmatrix} 0.072447456485894 & -0.064918078548578 & -0.155265986919403 \\ 0.014666986646143 & -0.015086638746772 & 0.051497876644135 \\ 0.101486170201556 & -0.100189798980014 & 0.096621409701035 \end{bmatrix}$$

In questo caso le prestazioni peggiorano ancora.

## 12.6 Test 5: determinazione dello spin del satellite

In questa prova è stato testato l'algoritmo di determinazione dello spin del satellite. E' stato fissato il vettore di spin nel seguente modo:

$$\vec{\omega} = (1.0 \quad 0.0 \quad 0.0) \text{ [rad / s]}$$

A questo punto, prendendo in considerazione l'orientamento delle tiles fissato, è stato ricavato il vettore  $\vec{\Omega}^T$  atteso. Si ricorda che tale vettore contiene i valori di spin letti dai vari sensori giroscopici presenti sul satellite. I valori ottenuti sono (nella situazione in esame si presume che tutte le Power Management tiles del satellite cubico in esame possiedano il sensore giroscopico):

$$\vec{\Omega}^T = [0.0 \ 0.0 \ -1.0 \ 0.0 \ 0.0]^T \text{ [rad / s]}$$

Il valore di velocità angolare calcolato dall'algoritmo è dato da:

$$\vec{\omega} = (1.0 \quad 0.0 \quad 0.0) \text{ [rad / s]}$$

Questo risultato dimostra la correttezza dell'algoritmo di determinazione dello spin del satellite sviluppato.

# Capitolo 13

## 13 Conclusioni

Il progetto del sistema software ACS, indirizzato alla determinazione ed al controllo dell'assetto del satellite AraMiS, ha permesso al candidato di apprendere molte nozioni non trattate nel corso degli studi, soprattutto per quanto concerne tutti gli aspetti legati all'ambito aerospaziale e quelli legati allo sviluppo di un sistema software indirizzato ad un'applicazione significativa e molto interessante.

La fase preliminare dell'attività è stata dedicata allo studio delle principali definizioni impiegate, in ambito spaziale, nella descrizione delle orbite dei satelliti e dei corpi celesti, in generale, dei principali parametri adottati nella descrizione dell'assetto del satellite e dei principali sensori ed attuatori sfruttati al fine di governare la dinamica di un oggetto nello spazio. In questa fase iniziale sono stati opportunamente definiti i vari sistemi di riferimento propedeutici allo svolgimento del lavoro. In particolare sono stati definiti il sistema di riferimento inerziale, il sistema di riferimento del satellite ed il sistema di riferimento delle tiles componenti la struttura modulare del satellite.

Questo studio iniziale ha permesso di fissare le basi su cui costruire tutti gli algoritmi implementati durante l'attività.

Tutti gli algoritmi sviluppati sono stati preliminarmente scritti e collaudati utilizzando il linguaggio di programmazione C. Le simulazioni effettuate confermano l'attendibilità dei risultati ottenuti.

L'intero sistema ACS è stato descritto in UML utilizzando l'ambiente Visual Paradigm ad esso associato, mediante il diagramma dei casi d'uso ed il diagramma delle classi (si veda capitolo 11). Utilizzando il tool Instant Generator di Visual Paradigm tutto il sistema ACS è stato tradotto in linguaggio di programmazione C++ ed è stato opportunamente collaudato. Le simulazioni effettuate confermano, anche in questo caso, l'attendibilità dei risultati ottenuti.

Il sistema software ACS è stato progettato con l'ottica di ottenere un sistema il più indipendente possibile dall'architettura modulare del satellite AraMiS. Questo permette di utilizzare il software sviluppato su satelliti AraMiS in qualsiasi configurazione senza, o al limite con poche, modifiche significative.

Il software orientato alla determinazione dell'assetto del satellite prevede, al momento, l'interrogazione dei sensori di posizione del Sole e dei sensori di campo magnetico. Di fatto, è stato costruito generalizzando del tutto il problema ed è funzionante in presenza di un numero variabile di vettori di misura e di riferimento. Qualora, in futuro, siano considerati ulteriori sensori d'assetto, la loro integrazione, ai fini dell'algoritmo sviluppato, risulterà estremamente semplice. Sarà, infatti, sufficiente integrare le interrogazioni agli eventuali sensori d'assetto interessati. Si tenga anche presente che il modello di calcolo della posizione del Sole, come annunciato nel capitolo 7, risulta essere versatile e può essere impiegato anche come modello della posizione dei pianeti del sistema solare.

Il software orientato al controllo dell'assetto del satellite prevede, temporaneamente, uno spostamento di tipo punto-punto. In altre parole l'algoritmo di controllo dell'assetto attuale è in grado di calcolare il comando necessario a spostare il satellite nell'assetto desiderato, ma non contempla la possibilità che tale assetto sia mantenuto nel tempo, ovvero non è prevista la possibilità di inseguimento di una traiettoria (problema del *tracking*). In realtà una possibile soluzione al problema è stata pensata, ma resta ancora da implementare e testare opportunamente. L'idea di base è di sfruttare gli stessi algoritmi di calcolo sviluppati nel caso di spostamento punto-

punto e di effettuare il conto del comando molte volte, correggendo opportunamente l'assetto del satellite ad ogni passo.

L'intero software non è stato implementato sul processore di bordo del satellite AraMiS, dal momento che non risulta ancora disponibile. Tuttavia la parte di software indirizzata al calcolo della posizione del satellite (mediante l'algoritmo SGP4) è stata implementata su di un MSP430FG439, il quale rappresenta un possibile processore candidato a divenire il processore di bordo del satellite AraMiS (si veda capitolo 4).

In futuro il software sviluppato durante l'attività di tesi dovrà essere opportunamente testato sull'appropriato processore di bordo del satellite AraMiS, al fine di valutare correttamente le tempistiche necessarie ad eseguire i vari computi e stabilire, quindi, le prestazioni del sistema. Inoltre, si dovranno effettuare delle adeguate simulazioni sul sistema completo, al fine di verificare convenientemente l'integrazione del software con l'hardware utilizzato. Infatti, tutte le prove eseguite sul sistema ACS sono state, al momento, effettuate rimanendo esclusivamente nell'ambito del software.

# Bibliografia

- [1] David A. Vallado, *Fundamentals of Astrodynamics and Applications*, California: Microcosm Press, 2001
- [2] James R. Wertz, *Spacecraft Attitude Determination and Control*, Holland: D.Reidel Publishing Company, 1980
- [3] Roy, Archie Edminston, *Orbital Motion*, Bristol : Hilger, 1982
- [4] Darrel Ince, *Programmazione a Oggetti in C++*, Italia: Arti Grafiche Battaia snc, 1992
- [5] Wiley J. Larsen e James R. Wertz, *Space Mission Analysis and Design*, Torrance : Microcosm; Dordrecht: Kluwer, 1992
- [6] H. M. Deitel e P. J. Deitel, *C Corso completo di programmazione*, Milano: Apogeo, 2004
- [7] Brian W. Kernighan e Dennis M. Ritchie, *Linguaggio C*, Milano : Pearson Education, 2004
- [8] Calcolo della posizione dei pianeti.  
URL: <http://stjarnhimlen.se/comp/ppcomp.html>
- [9] Documentazione algoritmo SGP4.  
URL: <http://www.celestrak.com/NORAD/documentation/spacetrk.pdf>
- [10] Documentazione algoritmo SGP4.  
URL: <http://www.centerforspace.com/downloads/files/pubs/AIAA-2006-6753.pdf>
- [11] Campo magnetico terrestre.  
URL: <http://ngdc.noaa.gov/geomag/servers.shtml>