POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Aerospaziale

Tesi di Laurea Magistrale Rimozione di debris multipli in LEO: applicazione della teoria dei grafi ad un problema tempo - dipendente



Relatore

Prof. Lorenzo Casalino

Candidata

Selena Mastrodonato

Anno Accademico 2020/2021

Abstract

A seguito dell'esplosione di un satellite in orbita eliosincrona, l'effetto Kessler ha innescato una serie di collisioni che hanno compromesso gravemente l'ambiente spaziale in orbita LEO. Agenzie spaziali e compagnie private cercano un modo per rimuovere 123 detriti per poter tornare ad operare in sicurezza in questa regione di spazio.

È questo lo scenario ipotizzato dalla *Global Trajectory Optimisation Competition 9*, su cui il seguente lavoro di tesi si costruisce.

L'obiettivo è quello di trovare una sequenza per rimuovere i 123 detriti in orbita LEO, minimizzando il costo della missione, in modo che la soluzione sia confrontabile con quella proposta dal team del JPL, vincitore della competizione. Il metodo usato è un algoritmo di ricerca nel grafo e si propone come alternativa all'algoritmo dell'*Ant Colony Optimization* (ACO) e agli algoritmi evolutivi principalmente usati per questo tipo di problema.

Una volta creata la struttura del grafo, in cui ogni nodo rappresenta un detrito e gli archi tutti i possibili trasferimenti tra un detrito e l'altro, si procede con una ricerca in profondità nel grafo per ottenere delle sequenze preliminari di missioni, tenendo in considerazione il costo dei trasferimenti orbitali. Le sequenze preliminari vengono poi completate con i detriti mancanti e si creano delle vere e proprie sotto-missioni cercando di rispettare il limite di massa iniziale del chaser che lascerà ai detriti i kit di de – orbit necessari a farli rientrare in atmosfera.

Raccolti tutti i detriti, si procede con l'ottimizzazione temporale della missione, rispettando i vincoli temporali imposti dalla competizione, cercando di ridurre il più possibile i costi monetari. Infine, un'ulteriore ottimizzazione viene eseguita sulla lista dei detriti, effettuando permutazioni sul loro ordine fino a convalidare la scoperta di un minimo locale della funzione di costo.

Il metodo utilizzato è stato in grado di trovare una soluzione, una missione in grado di raccogliere tutti i 123 detriti, con un costo sia in termini di ΔV che in milioni di euro non troppo distanti dalla soluzione del JPL.

Following the explosion of a Sun – synchronous satellite, the Kessler effect caused further impacts and LEO environment was severely compromised. Space agencies and private space companies are searching a way to remove 123 debris to restore the possibility to operate safely in this environment.

This is the scenario supposed by *Global Trajectory Optimisation Competition 9*, on which the following thesis work is based.

The main aim is to find a sequence to remove the 123 in LEO orbit, minimising the cost of the mission, so that the solution will be comparable to the solution proposed by the JPL team, winner of the competition. The approach used is a graph search algorithm and it is proposed as an alternative to the *Ant Colony Optimization* algorithm (ACO) and the evolutionary algorithms, mainly used for this type of problems.

First, the structure of the graph is created. In the graph each node represents a debris and the arcs all the possible orbital transfers between one debris and the other. We proceed with a depth-first search in the graph to obtain preliminary mission sequences, taking into consideration the cost of orbital transfers. The preliminary sequences are completed with the missing debris and real sub - mission are created trying to respect the limit imposed on the initial mass of the chaser.

Once all the debris are collected, we proceed with the temporal optimisation of the mission, respecting the time constraints imposed by the competition and trying to reduce as far as possible the monetary costs. Finally, an additional optimisation is done on the debris list, by performing permutations on its order until validating the discovery of a local minimum of the cost function.

The method used was able to find a solution, a mission capable of collecting all 123 debris, with a cost both in terms of ΔV and in millions of euro not too far from the JPL solution.

Sommario

1	Int	roduzione	1
2	Sto	oria degli space debris	2
	2.1	Definizione, dimensioni e origini	2
	2.2	Sindrome di Kessler	5
	2.3	Metodi per far fronte al problema dei detriti spaziali	6
	2.3	.1 Linee guida per la mitigazione dei detriti spaziali	6
	2.3	.2 Rimozione dei detriti spaziali	6
	(ClearSpace-1	7
3	Ele	ementi di meccanica orbitale	9
	3.1	Il problema dei due corpi	9
	3.2	Costanti del moto	10
	3.3	Equazione della traiettoria	11
	3.4	Parametri orbitali	12
	3.5	Perturbazioni orbitali	13
	3.6	Modello dinamico semplificato	15
4	De	finizione del problema e strumenti per la risoluzione	17
	4.1	Global Trajectory Optimisation Competition	17
	4.2	Valutazione del costo della trasferta	20
	4.2	.1 Tempo di trasferimento ottimale	20
	4.2	.2 Tempo di trasferimento limitato	20
	4.3	Calcolo della massa	24
	4.4	Teoria dei grafi	25
	4.4	.1 Applicazioni	29
]	sette ponti di Königsberg	29
]	Il gioco dell'icosaedro	30
]	Il problema del commesso viaggiatore (TSP)	31

5	R	icerca	della sequenza e ottimizzazione	34
		Meto	do con un singolo albero	36
		Parti	zione dei grafi	37
		Ricer	c ca dei Δv minimi	37
		Ricer	rca euristica nel piano temporale	38
	5.1	Ric	erca delle sequenze	39
	5.2	Cos	struzione della soluzione e ottimizzazione	44
	5.	.2.1	Prima modalità di completamento della sequenza	47
	5.	.2.2	Seconda modalità di completamento della sequenza	48
	5.	.2.3	Ottimizzazione dei costi	49
		fmin	con Interior Point Algorithm	49
		Funz	ione da ottimizzare e creazione dei vincoli	50
		Ottin	nizzazione della lista dei nodi	52
6	R	isultat	ti e confronti	53
	6.1	Ris	ultati preliminari dopo l'ottimizzazione temporale	53
	6.2	Ris	ultati finali	55
	6.3	Coı	nfronto con la soluzione del JPL	60
		Test	1: rimozione del primo detrito da ogni sotto-missione	62
		Test	2: rimozione di più detriti da più sotto-missioni	64
		Test	3: rimozione di un'intera sotto-missione	64
		Test	4: rimozione di più sotto-missioni	65
		Clust	ering spettrale	66
7	C	onclu	sioni	70
8	В	ibliog	rafia	72

Lista delle figure

Figura 2.1 Distribuzione di detriti in LEO	2
Figura 2.2 Nuovi oggetti catalogati negli anni in base all'orbita	3
Figura 2.3 Oggetti in LEO con diametro maggiore di 10 cm	3
Figura 2.4 Evoluzione del numero di oggetti	4
Figura 2.5 Evoluzione della massa	4
Figura 2.6 Effetto della collisione di un pannello solare con un detrito	5
Figura 3.1 Moto relativo tra i due corpi	10
Figura 3.2 Equazione della conica in forma polare	11
Figura 3.3 Sezioni coniche	12
Figura 3.4 Parametri orbitali kepleriani	12
Figura 3.5 Perturbazioni orbitali	14
Figura 4.1 Soluzione JPL	19
Figura 4.2 Durata rendez-vous della soluzione JPL	19
Figura 4.3 Durata trasferte della soluzione JPL	19
Figura 4.4 Costo delle trasferte in m/s della soluzione JPL	19
Figura 4.5 Esempio di grafo non orientato	25
Figura 4.6 Esempio di grafo orientato	26
Figura 4.7 Esempio di grafo pesato	26
Figura 4.8 Esempio di percorso in un grafo	27
Figura 4.9 Esempio di grafo connesso (destra) e di grafo non connesso (sinistra)	27
Figura 4.10 Grafo di riferimento con possibile albero	28
Figura 4.11 Rappresentazione di un grafo	29
Figura 4.12 Mappa di Königsberg che evidenzia il fiume Pregel e i suoi ponti	30
Figura 4.13 Dalla mappa al grafo	30
Figura 4.14 Suddivisione complessità dei problemi	32
Figura 5.1 Costo medio	34
Figura 5.2 Deviazione standard dei costi nel tempo	35
Figura 5.3 Costo minimo nel tempo	35
Figura 5.4 Esempio con 8 sotto-missioni dall'albero iniziale a quello finale	36
Figura 5.5 Costi totali delle sotto-missioni e detriti rimasti	36
Figura 5.6 Esempio di partizione dei grafi	37
Figura 5.7 Sequenza con la ricerca dei costi minimi	38

Figura 5.8 Flow chart della costruzione della soluzione	39
Figura 5.9 Grafo 'g'	40
Figura 5.10 Dettaglio grafo 'g'	41
Figura 5.11 Generazione delle sequenze	43
Figura 5.12 Flow chart ricerca delle sequenze preliminari	44
Figura 5.13 Esempio di struct che contiene le informazioni della soluzione	46
Figura 5.14 Ampliamento della sequenza con la prima modalità	48
Figura 5.15 Seconda modalità di completamento della sequenza	49
Figura 5.16 Processo di convergenza	51
Figura 6.1 Timeline della missione	58
Figura 6.2 Masse iniziali delle sotto-missioni	59
Figura 6.3 Confronto sui costi dei trasferimenti	60
Figura 6.4 Somiglianze con le sotto-missioni del JPL	61
Figura 6.5 Somiglianza tra le sotto-missioni	62
Figura 6.6 Clustering con diversi valori di share	67
Figura 6.7 Nodi in comune tra il cluster PoliTO e il cluster JPL	68
Figura 6.8 Somiglianze di percorso	68
Figura 6.9 Esempio di un momento in cui entrambe le soluzioni prevedono una p	oausa69
Lista delle tabelle	
Tabella 3.1 Tipologia dell'orbita in base all'eccentricità	13
Tabella 6.1 Lista preliminare dei detriti nelle sotto-missioni	54
Tabella 6.2 Costi dei trasferimenti orbitali preliminari	54
Tabella 6.3 Masse iniziali e costo totale della soluzione preliminare	55
Tabella 6.4 Lista debris raccolti nelle 14 sotto-missioni	55
Tabella 6.5 Costi dei trasferimenti orbitali	56
Tabella 6.6 Durata dei trasferimenti orbitali	57
Tabella 6.7 Giorni [MJD2000] di inizio e fine delle sotto-missioni	58
Tabella 6.8 Risultati test 2 (notazione zero-based)	64
Tabella 6.9 Risultati test 3 (notazione zero-based)	65
Tabella 6.10 Risultati test 4 (notazione zero-based)	65

1 Introduzione

Da più di sessanta anni nello spazio vengono lanciati razzi e satelliti e l'avvento sul mercato di lanciatori sempre più economici, capaci di mettere in orbita carichi resi più leggeri ed efficienti grazie alla miniaturizzazione dei componenti, ha contribuito all'aumento di materiale in orbita intorno alla Terra. I satelliti alla fine della loro vita operativa continuano a rimanere in orbita e insieme a primi stadi di razzi, frammenti, vernici, utensili persi durante le missioni, pannelli solari, materiale espulso dai razzi costituiscono gli *space debris*, i detriti spaziali.

L'orbita bassa terrestre (LEO – Low Earth Orbit) e l'orbita geostazionaria (GEO – Geostationary Earth Orbit) sono le regioni di maggiore interesse dal punto di vista commerciale e scientifico e per questo motivo sono le zone più popolate dai detriti.

Negli ultimi anni è stata raggiunta una situazione critica per quanto riguarda i detriti spaziali: alcuni di essi hanno dimensioni tali da essere individuati ed evitati; altri, invece, hanno dimensioni così ridotte da rendere impossibile il rilevamento e, viaggiando a velocità molto elevate, mettono a rischio la sicurezza dei satelliti operativi e delle missioni con equipaggio umano. Un'eventuale collisione potrebbe causare enormi danni, un conseguente incremento dei detriti in seguito all'impatto e un maggiore rischio di collisione - uno scenario previsto dalla cosiddetta sindrome di Kessler.

Per mitigare questo rischio e per poter continuare a lanciare satelliti in orbita in completa sicurezza è necessario cominciare a rimuovere i detriti che orbitano intorno alla Terra, con particolare interesse verso quelli in orbita LEO.

Esistono da tempo normative e tecnologie volte a limitare il problema dei detriti spaziali. In particolare, le agenzie spaziali si stanno muovendo per la rimozione attiva dei detriti (Active Debris Removal – ADR), che comprende diversi metodi, dall'utilizzo di un braccio robotico che agganci il satellite per riportarlo a Terra, ad un sistema a rete.

2 Storia degli space debris

2.1 Definizione, dimensioni e origini

Per detrito spaziale si intende qualsiasi oggetto creato artificialmente in orbita intorno alla Terra e non più utilizzato, come satelliti ormai in disuso, primi stadi di razzi, scaglie di vernici, polveri, frammenti derivati da collisioni.

È il 4 ottobre 1957 quando l'Unione Sovietica lancia in orbita il primo oggetto creato artificialmente, lo Sputnik 1, che diede il via all'era dell'esplorazione spaziale. Con l'evoluzione delle tecnologie spaziali negli anni sono stati lanciati sempre più satelliti, principalmente in orbita LEO e GEO, le più importanti per i satelliti con fine scientifico o commerciale (satelliti per immagini ad alta risoluzione, comunicazioni, etc.).

La maggior parte dei detriti risiede infatti entro i 2000 km dalla superficie terrestre e la più alta concentrazione si trova in una fascia compresa tra i 750 e i 1000 km.



Figura 2.1 Distribuzione di detriti in LEO

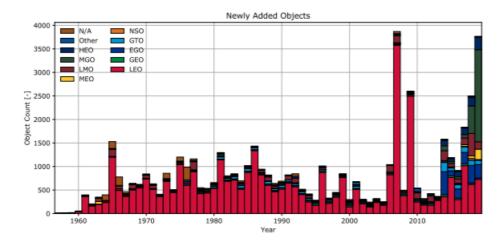


Figura 2.2 Nuovi oggetti catalogati negli anni in base all'orbita

Le ultime stime, risalenti a Gennaio 2021, indicano 28 210 detriti regolarmente tracciati e catalogati dallo *Space Surveillance Network*, 34 000 detriti di dimensioni maggiori ai 10 cm, 900 000 con dimensioni comprese tra 1 cm e 10 cm e 128 milioni di detriti con dimensioni tra 1 mm e 1 cm, per un totale di 9200 tonnellate di oggetti in orbita intorno alla Terra.

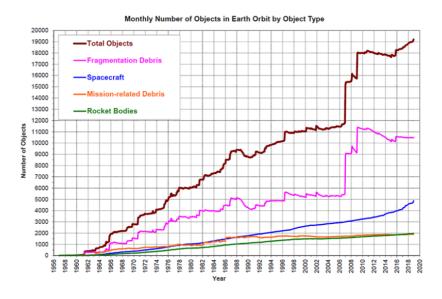


Figura 2.3 Oggetti in LEO con diametro maggiore di 10 cm

I detriti più grandi (>10 cm) possono essere tracciati, mentre quelli al di sotto dei 3 mm possono essere individuati da radar di terra. Una stima del numero di detriti di piccola dimensione, inferiore a 1 mm, può essere fatta esaminando gli impatti sulle superfici dei satelliti che rientrano.

Due sono gli eventi principali che hanno causato la maggior parte dei detriti oggi in orbita intorno alla Terra:

- nel 2007, la distruzione del satellite cinese Fengyun-1C come conseguenza di un test di un missile antisatellite (ASAT);
- nel 2009, un impatto accidentale tra il satellite Iridium-33 e un satellite russo, Cosmos-2251.

Prima di questi due incidenti la causa principale della formazione dei detriti spaziali derivava dall'esplosione dei primi stadi dei lanciatori e dei satelliti.

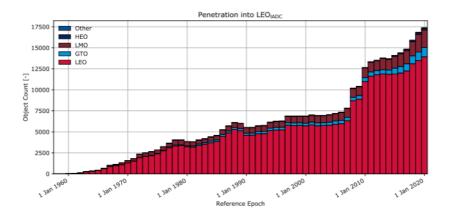


Figura 2.4 Evoluzione del numero di oggetti

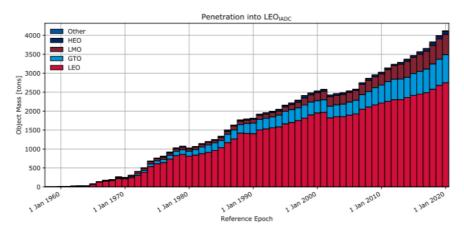


Figura 2.5 Evoluzione della massa

Il problema dei detriti spaziali è che le loro traiettorie non sempre sono calcolabili con precisione e dato che orbitano a velocità molto elevate anche il più piccolo dei detriti potrebbe avere conseguenze disastrose in caso di collisione con un altro oggetto. I detriti in LEO orbitano, infatti, ad una velocità di circa 7-8 km/s, ma la velocità media di impatto tra un detrito orbitante e un altro oggetto è approssimativamente di 10 km/s e può arrivare fino a 15 km/s.



Figura 2.6 Effetto della collisione di un pannello solare con un detrito.

2.2 Sindrome di Kessler

Donald J. Kessler, scienziato della NASA, nel 1978 propose uno scenario teorico in cui la densità degli oggetti in orbita LEO sarebbe stata abbastanza alta da causare collisioni a cascata; ogni collisione causerebbe altri detriti spaziali, aumentando così la probabilità di ulteriori collisioni. In *Collision Frequency of Artificial Satellites: The Creation of a Debris Belt [3]*, Kessler conclude che nell'anno 2000 il principale rischio per gli oggetti spaziali in orbita sarebbero stati i detriti spaziali.

Il concetto non nasce con gli studi sui detriti orbitali, ma dagli studi sul sistema solare e in particolare sulla formazione di meteoriti dagli asteroidi. Kessler dimostrò che un processo di collisione simile a quello degli asteroidi si sarebbe verificato tra i detriti in LEO in tempi molto più brevi di quanto si potesse pensare, decenni anziché miliardi di anni. Fu poi John Gabbard a coniare il termine *sindrome di Kessler* per riferirsi ad un accumulo incontrollato di detriti.

Si possono identificare tre diversi tipi di collisioni:

- non catastrofiche e trascurabili: questo tipo di collisioni produce una quantità trascurabile di detriti, non impattando sull'ambiente né nel lungo termine né nel breve termine.
- non catastrofiche: queste collisioni impattano sull'ambiente spaziale solo nel breve periodo. Si verificano, in generale, tra un frammento e un oggetto intatto, causando una quantità di detriti circa cento volte la massa del frammento.
- catastrofiche: impattano sull'ambiente spaziale sia nel breve che nel lungo periodo. Producono una popolazione di piccoli frammenti insieme ad una di frammenti più grandi che contribuisce significativamente al processo di collisione a cascata.

A seconda della massa e della velocità orbitale dei detriti, dunque, la collisione può essere

più o meno pericolosa per gli oggetti attivi e orbitanti intorno alla Terra.

2.3 Metodi per far fronte al problema dei detriti spaziali

Anche se tutti i lanci fossero sospesi, la popolazione dei detriti continuerebbe a crescere a causa delle continue collisioni, che genererebbero ulteriori detriti con un effetto a cascata. Dato il rischio per le future missioni e in particolare per le attività umane, dunque, le azioni più importanti al momento sono quelle di non creare ulteriori detriti - e questo può essere fatto attraverso un'attenta progettazione del veicolo e delle operazioni, oltre che alla catalogazione degli oggetti in orbita - e di rimuovere i detriti presenti. Al momento si contano circa 3000 satelliti dismessi o fuori uso.

2.3.1 Linee guida per la mitigazione dei detriti spaziali

La NASA e l'European Space Agency (ESA), come le nazioni che hanno la possibilità di lanciare nello spazio (Russia, Cina, Giappone, Francia) hanno delle linee guida per la mitigazione dei detriti orbitali. Gli aspetti più importanti e comuni a tutti riguardano:

- l'immissione in orbita: gli stadi di un lanciatore devono essere progettati in modo da eliminare o ridurre al minimo i detriti lasciati durante il lancio.
- la progettazione: durante questa fase è necessario dimostrare tramite metodi ingegneristici che il veicolo spaziale progettato abbia una probabilità di esplosione inferiore a 1/1000 durante il dispiegamento in orbita. Inoltre, è necessario limitare il rischio per gli altri sistemi spaziali in caso di esplosioni accidentali.
- la limitazione della probabilità di collisione con oggetti di piccole e grandi dimensioni. Inoltre, si dibatte anche sull'implementazione di un sistema di *Collision Avoidance* per tutti i mezzi immessi oltre i 400 km.
- lo smaltimento dei satelliti non più operativi: in orbita LEO, per esempio, si prevede il rientro atmosferico entro i 25 anni dal termine delle operazioni, tramite l'uso della propulsione chimica oppure per decadimento naturale verso l'atmosfera terrestre.

2.3.2 Rimozione dei detriti spaziali

La pulizia dell'ambiente spaziale rimane ancora una sfida tecnica ed economica, ma sarà anch'essa un'azione importante per rendere più sicure le future missioni in orbita LEO. Le principali strategie sono:

• il decadimento naturale, ad opera della resistenza aerodinamica, che farà in modo

che il detrito raggiunga l'atmosfera terrestre e quindi bruci. Questa opzione è valida quando è possibile mantenere il tempo di de-orbiting al di sotto dei 25 anni (secondo le politiche sulla mitigazione dei detriti spaziali) e questo non può essere garantito per gli oggetti con un'orbita iniziale al di sopra dei 700 km.

- la propulsione chimica, che permette un rientro in tempi brevi e rappresenta una buona opzione per le basse orbite LEO e per satelliti con massa piccola.
- la propulsione elettrica, che richiede tempi maggiori per il rientro, a causa del minore livello di spinta rispetto alla propulsione chimica.
- l'uso di dispositivi di "Drag Augmentation", in genere dispositivi gonfiabili (drag balloons) oppure strutture sottili supportate da bracci rigidi ad uso di vela (drag sails).
- sistema EDT (Bare Electro-Dynamic Tether), costituito da un filo conduttore collegato al satellite. Il sistema sfrutta la forza di Lorentz generata dal processo elettrodinamico dovuto al moto relativo del filo rispetto al campo magnetico terrestre, con conseguente generazione di corrente nel filo. La forza viene trasmessa al sistema opponendosi al moto del satellite, riducendo la sua velocità ed abbassando la sua orbita.
- la più importante è la rimozione attiva, *Active Debris Removal* (ADR), che avviene mediante l'uso di reti, arpioni, bracci robotici, dispositivi che servono a catturare il detrito o a far sì che la sua orbita decada. In particolare, nel presente lavoro di tesi la rimozione attiva avviene mediante un satellite chaser che deve effettuare una manovra di rendez-vous con i detriti target, in modo da agganciare loro il kit di deorbit, che permetterà di avere la spinta necessaria a far decadere la loro orbita.

L'ADR è importante, perché a differenza di altri metodi, non dipende da come è stato progettato il veicolo o i dispositivi che provvedono alla rimozione e pertanto può essere applicato ad ogni tipo di detrito.

ClearSpace-1

Ad oggi non sono ancora state svolte missioni di rimozione attiva dei detriti, ma l'European Space Agency si prepara a lanciare nel 2025 *ClearSpace-1*, la prima missione di raccolta dei detriti spaziali. Obiettivo della missione è, infatti, non solo quello di contribuire alla pulizia dell'ambiente spaziale, ma anche quello di dimostrare la fattibilità di questo tipo di operazioni e la maturità delle tecnologie necessarie.

La missione avrà come obiettivo il modulo VESPA (VEGA Secondary Payload Adapter),

utilizzato per il secondo lancio del 2013 dei satelliti Proba-V, ESTCube-1 e VNREDSAT, e lasciato in orbita a circa 800 km. Il modulo ha una massa di 100 kg, quindi paragonabile ad un piccolo satellite, e proprio per la sua struttura risulta essere un primo obiettivo ideale. *ClearSpace-1* verrà lanciato in un'orbita di 500 km, che verrà poi innalzata per raggiungere il target e catturarlo per mezzo di quattro bracci robotici. Target e chaser verranno poi fatti deorbitare insieme con un rientro distruttivo in atmosfera.

3 Elementi di meccanica orbitale

In questo capitolo vengono richiamati alcuni concetti di base della meccanica del volo spaziale, con particolare attenzione ai fenomeni perturbativi. Viene, infine, descritto il modello dinamico che sta alla base del moto dei detriti per il seguente lavoro.

3.1 Il problema dei due corpi

Il moto di un satellite intorno alla Terra può essere approssimato con il problema dei due corpi, che è l'unico problema gravitazionale ad avere una soluzione generale e completa. Due sono le assunzioni che riguardano questo modello:

- 1. i corpi sono sfericamente simmetrici: questo consente di trattarli come se la loro massa fosse concentrata nel loro centro;
- 2. non ci sono forze esterne o interne che agiscono sul sistema, oltre all'azione di reciproca attrazione gravitazionale descritta dalla legge di gravitazione universale di Newton:

$$F_g = -\frac{GMm \, \vec{r}}{r^2} \tag{3.1}$$

Dove $G = 6.67 * 10^{-11} N \cdot m^2/kg^2$ è la costante di gravitazione universale, $Me\ m$ sono le masse tra cui agisce la forza gravitazionale e r è la distanza tra i centri delle masse.

Per determinare l'equazione del moto relativo tra i due corpi con la seconda legge di Newton, bisogna considerare innanzitutto un sistema di riferimento inerziale.

Si consideri, quindi, un sistema inerziale XYZ. Applicando la seconda legge di Newton ai due corpi si ottiene:

$$\begin{cases} m\ddot{r}_{m} = -\frac{GMm\vec{r}}{r^{2}}\frac{\vec{r}}{r} \\ M\ddot{r}_{M} = \frac{GMm\vec{r}}{r^{2}}\frac{\vec{r}}{r} \end{cases}$$
(3.2)

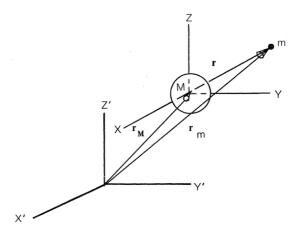


Figura 3.1 Moto relativo tra i due corpi

Sottraendo la (3.3) alla (3.2) si ottiene

$$\ddot{r} = -\frac{G(M+m)}{r^3}\vec{r} \tag{3.4}$$

Il problema dei due corpi ristretto è un caso particolare del problema dei due corpi: nel caso di satelliti artificiali, missili, sonde la massa del corpo orbitante m può essere trascurata in quanto molto più piccola della massa del corpo principale M. Si ottiene, quindi:

$$\ddot{r} = -\frac{GM}{r^3}\vec{r} \tag{3.5}$$

Per convenzione si definisce il parametro gravitazionale $\mu \equiv GM$, per cui si può riscrivere l'equazione (3.5) come

$$\ddot{r} + \frac{\mu}{r^3} \vec{r} = 0 \tag{3.6}$$

3.2 Costanti del moto

Dalla fisica e dalla meccanica si sa che un campo gravitazionale è conservativo, per cui un corpo che si muove sotto l'azione della gravità non perde né guadagna energia meccanica, ma semplicemente cambia l'energia cinetica in energia potenziale. Pertanto, una delle costanti del moto è l'energia meccanica:

$$E = \frac{v^2}{2} - \frac{\mu}{r}$$
 (3.7)

Con v velocità orbitale del corpo di massa m.

Inoltre, dato che la forza gravitazionale agisce sempre in direzione radiale rispetto al centro della massa ci si aspetta che anche il momento angolare del corpo m rispetto all'origine del sistema di riferimento non cambi:

$$\vec{h} = \vec{r} \times \vec{v} \tag{3.8}$$

Il momento angolare rimane costante in modulo, direzione e verso. Ne consegue che il corpo di massa m ha una traiettoria che segue un moto piano intorno al corpo di massa M.

3.3 Equazione della traiettoria

Dell'equazione (3.6) si può ottenere una soluzione, che rappresenta l'equazione della traiettoria del corpo secondario orbitante intorno al corpo principale.

$$r = \frac{h^2/\mu}{1 + \left(\frac{B}{\mu}\right)\cos\nu} \tag{3.9}$$

Dove B è il modulo di \vec{B} complanare all'orbita e con la stessa direzione di \vec{e} , vettore eccentricità.

L'equazione (3.9) si può riscrivere in coordinate polari:

$$r = \frac{p}{1 + e \cos \nu} \tag{3.10}$$

Dove p è il semilatus rectum, una costante geometrica della conica; e è il modulo dell'eccentricità e ν è l'anomalia vera, l'angolo tra \vec{r} e il punto della conica più vicino al fuoco (periastro).

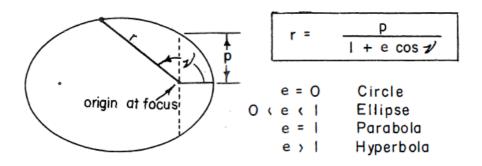


Figura 3.2 Equazione della conica in forma polare

1 1

 $[\]vec{B} = -\hat{r}\mu - \vec{h} \times \frac{d\vec{r}}{dt}$. Fundamentals of Astrodynamics, Roger R. Bate, Chap. 1.5

La somiglianza tra l'equazione della traiettoria (3.9) e l'equazione della conica (3.10) verifica la prima legge di Keplero e permette di considerare come traiettoria non l'ellisse, ma anche le altre coniche: cerchio, parabola, iperbole.

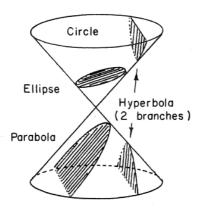


Figura 3.3 Sezioni coniche

3.4 Parametri orbitali

Per descrivere in modo completo la grandezza, la forma e l'orientamento di un'orbita sono sufficienti sei parametri.

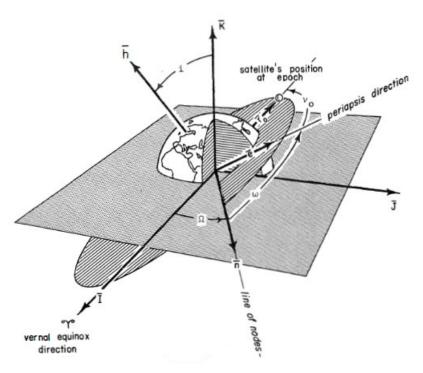


Figura 3.4 Parametri orbitali kepleriani

1. Inclinazione i: l'angolo tra il piano orbitale e il piano di riferimento, di solito il

- piano equatoriale. Assume valori compresi tra 0° e 180°.
- 2. Ascensione retta del nodo ascendente (RAAN) Ω: l'angolo tra il punto vernale e il nodo ascendente. Assume valori compresi tra 0° e 360°.
- 3. Argomento del periastro ω : l'angolo tra la linea dei nodi e il vettore eccentricità \vec{e} . Assume valori compresi tra 0° e 360° .
- 4. Semiasse maggiore a: una costante che definisce la grandezza dell'orbita.
- 5. Eccentricità e: una costante che definisce la forma dell'orbita

e = 0	cerchio
e > 1	parabola
e < 1	iperbole
0 < e < 1	ellisse

Tabella 3.1 Tipologia dell'orbita in base all'eccentricità

6. Anomalia vera ν : l'angolo compreso tra il periastro e la posizione del corpo orbitante nel tempo di riferimento. A volte, invece dell'anomali vera, si utilizza l'anomalia media M, che si può calcolare come $M=E-e\sin E$, dove E è l'anomalia eccentrica ed è legata a ν con la seguente formula: $\tan \frac{\nu}{2} = \sqrt{\frac{1+e}{1-e}} \tan \frac{E}{2}$.

3.5 Perturbazioni orbitali

Nel moto reale di un corpo orbitante non valgono rigorosamente le ipotesi del problema dei due corpi, per cui l'orbita effettiva del corpo non è una conica kepleriana. Questo è dovuto all'esistenza di fenomeni perturbativi che fanno discostare i parametri dell'orbita dai suoi valori nominali.

Questi fenomeni possono essere suddivisi in quattro categorie:

- *perturbazioni secolari:* sono proporzionali al tempo e comportano una continua riduzione o aumento del parametro di riferimento.
- perturbazioni periodiche di lungo periodo: causano variazioni armoniche dei parametri in un tempo che corrisponde al periodo di rotazione del periastro.
- *perturbazioni periodiche di corto periodo*: causano variazioni armoniche dei parametri con un periodo che è dell'ordine del periodo di rivoluzione del corpo intorno alla Terra.
- perturbazioni risonanti: causano variazioni dei parametri mediante l'incremento di

energia in un sistema altrimenti conservativo.

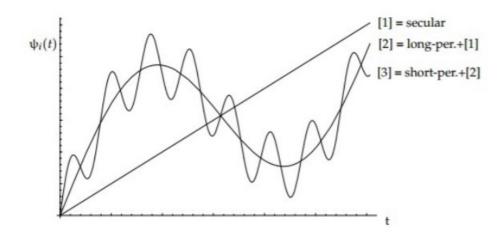


Figura 3.5 Perturbazioni orbitali

I principali fenomeni perturbativi sono la non sfericità e non omogeneità della Terra, le forze gravitazionali perturbatrici generate dagli altri corpi, la resistenza atmosferica e la pressione della radiazione solare.

Al fine della propagazione delle orbite in LEO le perturbazioni più importanti sono la resistenza aerodinamica, che è tanto maggiore quanto più l'orbita è bassa, e la non sfericità e non omogeneità della Terra.

Quest'ultima è una perturbazione di natura periodica e conservativa che determina distorsioni del campo gravitazionale terrestre.

Si definisce il potenziale gravitazionale in forma polare

$$V(r) = -\frac{\mu}{r} \tag{3.11}$$

e si aggiunge un contributo perturbativo

$$V(r,\phi) = -\frac{\mu}{r} \left[1 + \sum_{k=2}^{\infty} J_k \left(\frac{r_E}{r} \right)^k P_k(\sin \phi) \right]$$
 (3.12)

I coefficienti J_k rappresentano le armoniche dell'espansione, mentre $P^k \sin(\phi)$ è il k-esimo polinomio di Legendre

$$P_k(x) = (2^k k!)^{-1} \frac{d^k}{dx^k} [(x^2 - 1)^k]$$
(3.13)

Di tutte le armoniche il termine dominante è $J_2 = 1.082629 \cdot 10^{-3}$. Esplicitando il termine per k = 2 si ottiene il contributo di J_2 al potenziale gravitazionale:

$$P_2(x) = \frac{1}{2}(3x^2 - 1) \tag{3.14}$$

$$V(r,\phi)_{J_2} = -\frac{\mu J_2 r_E^2}{2r^3} (3\sin^2 \phi - 1)$$
 (3.15)

Gli effetti principali di J_2 sono la precessione della linea dei nodi e la regressione della linea degli apsidi. Lo schiacciamento della Terra modifica l'orientamento dell'orbita a seconda del semiasse maggiore, eccentricità e inclinazione. In base all'altezza dell'orbita, J_2 agisce in modo favorevole e riduce il Δv richiesto per il passaggio orbitale da un corpo all'altro quando fa sì che i nodi ascendenti delle due orbite si avvicinino.

3.6 Modello dinamico semplificato

È molto complicato predire in modo esatto il moto degli oggetti in orbita, proprio a causa dei fenomeni perturbativi che agiscono modificando i parametri orbitali. I modelli utilizzati generalmente per la propagazione delle orbite sono i *Simplified Perturbations models* (SGP, SGP4, SDP4, SGP8, SDP8), modelli matematici che permettono il calcolo dei vettori orbitali di stato (vettore posizione e vettore velocità) degli oggetti in orbita in un sistema di riferimento inerziale che ha come origine il centro della Terra. Questi modelli si basano sul *two-line element set* (TLE), un formato dati sviluppato da NORAD (North American Aerospace Defense Command) e NASA che fornisce informazioni utili riguardo ai satelliti, compresi i parametri orbitali. Le soluzioni tengono conto delle varie perturbazioni. Tuttavia, possiedono già all'epoca iniziale un errore di circa 1 km, che aumenta di 1-3 km ogni giorno. Per questo motivo, questi modelli non sono molto accurati sul lungo periodo e necessitano di correzioni dopo pochi giorni.

Pertanto, nel seguente lavoro di tesi è stato adottato un modello dinamico semplificato per descrivere la propagazione delle orbite dei detriti spaziali nel tempo. In particolare, è stato tenuta in considerazione solamente la perturbazione secolare dovuta alla non sfericità della Terra. Ne consegue che le variazioni di semiasse maggiore a, eccentricità e e inclinazione i sono trascurabili, mentre l'ascensione retta del nodo ascendente Ω , l'argomento del periastro ω e l'anomalia media M variano secondo le seguenti equazioni:

$$\frac{d\Omega}{dt} = -\frac{3}{2} \left(\frac{r_E}{p}\right)^2 n J_2 cosi \tag{3.16}$$

$$\frac{d\omega}{dt} = \frac{3}{4} \left(\frac{r_E}{p}\right)^2 n J_2(5\cos^2 i - 1)$$
 (3.17)

$$\frac{dM}{dt} = n + \frac{3}{4} \left(\frac{r_E}{p}\right)^2 n J_2 \sqrt{1 - e^2} (3\cos^2 i - 1)$$
 (3.18)

Dove $n=\sqrt{\frac{\mu}{a^3}}$ è la velocità orbitale media, μ è il parametro di gravitazione terrestre, r_E è il raggio equatoriale terrestre e $p=a(1-e^2)$ è il semilatus rectum.

Dato un determinato valore iniziale dei parametri orbitali, attraverso queste equazioni, è possibile calcolare per ogni tempo t i valori perturbati.

4 Definizione del problema e strumenti per la

risoluzione

In questo capitolo viene presentato il problema di cui si occupa il presente lavoro di tesi,

ossia la rimozione di 123 detriti in orbita LEO, la valutazione del costo dei trasferimenti

orbitali e delle masse iniziali. Viene, inoltre, fornita una panoramica sulla teoria dei grafi,

che viene applicata per la risoluzione del problema.

4.1 Global Trajectory Optimisation Competition

La Global Trajectory Optimisation Competition 9, indetta nel 2017 dall'European Space

Agency, ipotizza uno scenario futuro in cui, a seguito di un'esplosione di un satellite in

orbita eliosincrona, a causa dell'effetto Kessler, le orbite LEO eliosincrone risultano

gravemente compromesse. L'unica soluzione per poter tornare ad operare in sicurezza in

questa regione di spazio è quella di rimuovere 123 detriti, cercando di minimizzare il costo

della missione.

Ogni detrito è identificato da un numero, da 0 a 122, e per ognuno di essi vengono forniti i

parametri orbitali, riferiti all'istante in cui sono stati individuati:

• semiasse maggiore a

• eccentricità e

• inclinazione i

• Right Ascension of Ascending Node (RAAN) Ω

argomento del periastro ω

• anomalia media *M*

Viene fornita, inoltre, la data a cui sono riferiti i parametri dei detriti in MJD2000. Tutti i

detriti hanno orbite quasi circolari e l'inclinazione è compresa tra i 96° e i 101°.

Questo si può notare analizzando i dati forniti: sono stati creati tre video² dell'inclinazione,

dell'eccentricità e del semiasse maggiore in funzione del RAAN.

L'obiettivo della missione, come detto sopra, è raccogliere tutti i 123 detriti minimizzando

la seguente funzione di costo:

² Semiasse maggiore: https://www.youtube.com/watch?v=5Jb0lzLSKTY

$$J = \sum_{i=1}^{n} C_i = \sum_{i=1}^{n} [c_i + \alpha (m_{0_i} - m_{dry})^2]$$
 (4.1)

Con n numero di lanci necessario al completamento della missione. C_i rappresenta il costo del lancio per la missione i-esima ed è composto da un costo fisso c_i e da un termine $\alpha (m_{0_i} - m_{dry})^2$ che favorisce spacecraft più leggeri. All'inizio della i-esima missione, m_{0_i} è la massa iniziale dello spacecraft e m_{dry} è la massa a vuoto. Il parametro α è settato a $2.0 \cdot 10^{-6}$ [MEUR/Kg²].

Le sole manovre permesse per controllare la traiettoria dello spacecraft sono cambi istantanei di velocità, per cui dopo ogni manovra la massa dello spacecraft si aggiorna secondo l'equazione di Tsiolkovsky:

$$m_f = m_i \exp\left(-\frac{\Delta v}{v_e}\right) \tag{4.2}$$

Dove $v_e = I_{sp}g_0$ è la velocità efficace di scarico, I_{sp} è l'impulso specifico (fissato a 340 sec) e g_0 l'accelerazione di gravità a livello del mare, pari a 9.80665 m/s^2 .

Ogni spacecraft ha una massa iniziale pari a $m_0 = m_{dry} + N m_{de} + m_p$, con m_{dry} che è fissata per tutti a 2000 kg e la massa iniziale di propellente m_p che può essere al massimo 5000 kg. La m_{de} rappresenta la massa del kit di de-orbit che ogni detrito target deve ricevere dallo spacecraft ed è pari a 30 kg.

Inoltre, la missione è soggetta ai seguenti vincoli operativi:

- il tempo tra un rendez-vous tra due detriti e un altro non può superare i 30 giorni;
- devono passare almeno 30 giorni tra la fine di una missione e l'inizio della successiva;
- tutti gli eventi della missione (arrivi, partenze, etc.) devono essere all'interno di una finestra temporale di 8 anni, che va dal 23467 MJD2000 al 26419 MJD2000.
- Il periastro dell'orbita non deve essere inferiore ai 6600000 m.

La competizione è stata vinta dal *Jet Propulsion Laboratory* (JPL) della NASA. La loro soluzione prevede la rimozione dei 123 detriti in 10 lanci. Di seguito il dettaglio della

soluzione, che verrà presa come riferimento per questo lavoro.

	Start	End	Lounah	Number		
			Launch			
Mission	MJD2000	MJD2000	Mass, kg	of objects	Debris ID	UTC
1	23557.18	23821.03	5665.38	14	23,55,79,113,25,20,27,117,121,50,95,102,38,97	20:17
2	23851.08	24024.53	4666.15	12	19,115,41,26,45,82,47,85,7,2,11,77	20:17
3	24057.47	24561.49	6589.58	21	72,107,61,10,28,3,64,66,31,90,73,87,57,35,69,65,8,43,71,4,29	21:42
4	24637.26	24916.44	5679.10	11	108,24,104,119,22,75,63,112,37,32,114	20:18
5	24946.47	25232.94	4906.59	14	84,59,98,1,40,51,36,67,62,99,54,122,76,15	20:18
6	25262.95	25455.15	5062.74	10	101,48,53,5,12,39,58,13,60,74	20:18
7	25485.20	25682.33	4082.33	10	49,9,70,93,105,46,88,118,18,91	20:18
8	25712.38	25915.53	3725.73	9	86,34,100,30,92,6,110,96,81	20:19
9	25946.06	26237.29	4897.35	12	33,68,116,106,14,52,120,80,16,94,83,89	20:19
10	26267.80	26416.00	3438.62	10	44,111,56,78,0,17,109,103,42,21	20:19

Figura 4.1 Soluzione JPL

Mission	Rendezvous Duration, days
1	5.00,5.00,5.04,5.01,5.01,5.03,5.00,5.00,5.00,5.03,5.04,5.04,5.00
2	5.00,5.02,5.02,5.00,5.04,5.00,5.05,5.02,5.07,5.03,5.02,5.00
3	5.00,5.06,5.01,5.02,5.07,5.02,5.04,5.02,5.01,5.02,5.01,5.07,5.06,5.02,5.01,5.01,5.06,5.01,5.02,5.04,5.00
4	5.00,6.01,6.01,6.03,6.05,6.05,6.04,6.01,6.06,6.04,5.00
5	5.00,5.02,5.07,5.04,5.01,5.01,5.02,5.06,5.06,5.02,5.06,5.01,5.07,5.00
6	5.00,5.02,5.01,5.04,5.07,5.02,5.01,5.02,5.00
7	5.00,5.00,5.06,5.06,5.04,5.06,5.04,5.06,5.03,5.00
8	5.00,5.01,5.03,5.00,5.01,5.04,5.07,5.02,5.00
9	5.00,5.51,5.53,5.53,5.53,5.55,5.54,5.53,5.54,5.55,5.52,5.00
10	5.00,5.54,5.50,5.50,5.52,5.54,5.53,5.52,5.00

Figura 4.2 Durata rendez-vous della soluzione JPL

Mission	Transfer Duration, days
1	24.86,24.98,22.42,24.99,0.29,10.63,25.00,2.70,1.51,1.41,24.67,24.31,5.86
2	24.93,0.28,0.73,0.39,17.07,1.61,22.42,2.39,15.88,24.97,2.49
3	14.16,24.94,2.87,8.10,9.00,23.13,23.09,23.09,22.83,24.98,24.98,24.93,24.94,9.10,13.44,24.99,24.94,24.99,24.98,24.96
4	23.96,6.48,16.72,23.97,23.95,23.95,23.96,23.99,23.94,23.96
5	0.45,3.17,24.93,10.34,12.53,7.11,13.44,24.94,24.94,24.98,22.19,24.99,22.01
6	24.91,0.30,18.39,3.08,20.24,24.96,24.85,24.97,0.28
7	15.69,0.50,9.83,24.94,24.90,24.48,20.87,24.91,0.66
8	10.03,24.00,2.83,24.99,24.99,24.96,21.19,24.98
9	22.69,4.24,24.47,24.46,24.47,24.44,24.46,24.46,18.54,9.22
10	0.81,11.59,7.66,1.11,17.46,6.47,20.47,24.47,3.99

Figura 4.3 Durata trasferte della soluzione JPL

Mission	ΔV , m/s
1	161.8,139.2,65.8,208.2,115.2,300.1,564.9,78.3,105.0,233.3,453.5,340.4,300.8
2	659.0,301.1,252.1,143.8,146.8,68.6,40.6,84.2,105.3,448.5,148.0
3	219.1,80.8,105.2,55.2,140.2,85.5,95.0,237.6,205.9,149.9,245.2,71.6,197.3,160.4,132.2,240.0,161.2,364.3,230.4,232.5
4	86.1,103.1,62.6,222.9,709.1,553.9,219.9,233.9,739.0,232.6
5	129.6,45.2,172.9,52.6,160.7,280.8,221.1,163.5,98.2,115.7,164.8,674.8,291.1
6	156.0,198.0,305.8,71.2,194.4,920.5,314.1,353.0,272.8
7	400.6,173.6,211.3,374.4,109.6,171.2,145.1,194.3,233.0
8	287.9,111.9,112.2,144.5,540.0,260.1,198.8,82.7
9	83.3,148.1,495.9,464.9,405.2,285.9,254.8,62.3,156.6,36.5,174.9
10	189.4,112.9,110.0,121.3,117.9,280.1,300.4,120.6,70.2

Figura 4.4 Costo delle trasferte in m/s della soluzione JPL

4.2 Valutazione del costo della trasferta

Per trovare la sequenza ottimale di detriti da rimuovere è necessario innanzitutto stimare il costo della trasferta tra un detrito e l'altro. In questo caso viene utilizzato un metodo semplificato che approssima il costo delle trasferte in modo sufficientemente accurato.

Il modello è basato sull'ipotesi che le trasferte più favorevoli vi siano in caso il cambio di piano orbitale tra chaser e target sia di piccola entità. Per questo viene sfruttato l'effetto perturbativo di J_2 , il quale è in grado di modificare il RAAN dei corpi orbitanti con un rateo in funzione del semiasse maggiore e dell'inclinazione. Tuttavia, questo tipo di soluzione non può essere applicata in caso di stretti vincoli temporali, per cui si rende necessario trovare un altro metodo di approssimazione, descritto nella sezione 4.2.2 anziché quello riportato nella sezione 4.2.1.

4.2.1 Tempo di trasferimento ottimale

In questo caso il costo della trasferta di Hohmann per variazioni di raggio trascurabili è:

$$\frac{\Delta v}{v} = 0.5 \, \frac{\Delta r}{r} \tag{4.3}$$

In questo caso si tiene in considerazione la variazione del semiasse maggiore Δa e si ottiene una formula empirica che aggiunge il contributo dovuto alla variazione dell'eccentricità Δe :

$$\frac{\Delta v}{v} = 0.5 \sqrt{\left(\frac{\Delta a}{a}\right)^2 + \Delta i^2 + \Delta e^2}$$
 (4.4)

4.2.2 Tempo di trasferimento limitato

Il tempo di trasferimento dal detrito k al detrito k+1 può essere posto pari a t. Si assumono inoltre le seguenti condizioni:

- $\Omega_{k+1}(t) \Omega_k(t) \neq 0$,
- $\bullet \quad a_{k+1} a_k \neq 0,$
- $i_{k+1} i_k \neq 0$.

Le corrispondenti manovre per il cambio di velocità sono denotate da x, y, z:

$$x = (\Omega_{k+1}(t) - \Omega_k(t)) \sin i_0 v_0$$
(4.5)

$$y = \frac{a_{k+1} - a_k}{2a_0} v_0 \tag{4.6}$$

$$z = (i_{k+1} - i_k)v_0 (4.7)$$

Dove
$$a_0 = \frac{a_{k+1} - a_k}{2}$$
, $i_0 = \frac{(i_{k+1} - i_k)}{2}$ e $v_0 = \sqrt{\frac{\mu}{a_0}}$, tutti e tre parametri costanti.

Per un'analisi approssimata si considerano trasferte a due impulsi: un primo impulso comporta un cambiamento parziale di x,y,z e il secondo impulso completa la manovra. Per risparmiare propellente le manovre sono combinate, e il primo impulso Δv_a si può scrivere come:

$$\Delta v_a = \sqrt{(s_x x)^2 + (s_y y)^2 + (s_z z)^2}$$
 (4.8)

I coefficienti s_x , s_y ed s_z non hanno restrizioni sul valore che possono assumere. Infatti, i cambiamenti di semiasse maggiore ed inclinazione possono essere maggiori di quelli previsti in origine, in modo da sfruttare l'effetto J_2 per ridurre la differenza di RAAN. Questo caso si verifica quando la differenza di RAAN è troppo grande e la durata della manovra non è sufficientemente lunga.

La variazione del RAAN nel tempo (RAAN-rate) è data dalla seguente formula:

$$\frac{\Delta\Omega}{dt} = -\frac{3}{2} \sqrt{\frac{\mu}{a^3}} \frac{J_2 cosi}{(1 - e^2)^2} \left(\frac{r_E}{a}\right)^2$$
 (4.9)

Per valori di eccentricità piccoli, la variazione di RAAN-rate per piccole variazioni di a (δa) e i (δi) si ottiene derivando la (4.9) rispetto ad a ed i:

$$\frac{\delta \dot{\Omega}}{\dot{\Omega}} = -\frac{7}{2} \frac{\delta a}{a} - \tan i \, \delta i \tag{4.10}$$

Sapendo che $\delta a = a_{k+1} - a_k$ e che $\delta i = i_{k+1} - i_k$, si può riscrivere $\delta \dot{\Omega}$ come:

$$\delta\dot{\Omega} = -7\dot{\Omega}_0 \frac{y s_y}{v_0} - \dot{\Omega}_0 \tan i_0 \frac{z s_z}{v_0}$$
(4.11)

Per comprendere meglio l'effetto della variazione del RAAN e del RAAN-rate sulle orbite dei detriti è stata creata un'animazione³ che prende in considerazione i primi tre detriti con i relativi parametri orbitali forniti dall'ESA per la competizione e mostra l'evolversi nel tempo delle loro orbite. Contestualizzando il modello orbitale semplificato con le informazioni fornite dal GTOC9 è interessante notare come, seppur tutti i detriti abbiano parametri molto simili di inclinazione e RAAN, i loro ratei sono distinti e provocano l'addensamento e la dilatazione di più orbite nel tempo. Sulla base di questo concetto si svilupperà la ricerca di una soluzione ottimale, poiché il Δv richiesto per il passaggio da un'orbita all'altra si riduce fortemente se le orbite sono quasi complanari.

Moltiplicando la (4.11) per la durata della manovra t si ottiene la differenza di RAAN dovuta alla prima manovra, ed inserendola all'interno della (4.5) si ottiene la differenza di velocità residua Δx :

$$\Delta x = 7\dot{\Omega}_0 y s_y \sin i_0 + \dot{\Omega}_0 \tan i_0 z s_z \sin i_0 \tag{4.12}$$

Ponendo $m = 7\dot{\Omega}_0 \sin i_0 t$ ed $n = \dot{\Omega}_0 \tan i_0 \sin i_0 t$, si può riscrivere Δx come:

$$\Delta x = mys_y + nzs_z \tag{4.13}$$

Il secondo impulso Δv_b può essere allora espresso come:

$$\Delta v_b = \sqrt{(x - s_x x - \Delta x)^2 + (y + s_y y)^2 + (z + s_z z)^2}$$
 (4.14)

E l'impulso totale:

 $\Delta v_a + \Delta v_b = \sqrt{(s_x x)^2 + (s_y y)^2 + (s_z z)^2} + \sqrt{(x - s_x x - \Delta x)^2 + (y + s_y y)^2 + (z + s_z z)^2}$ (4.15)

È complicato trovare il minimo di $\Delta v = \Delta v_a + \Delta v_b$ in forma chiusa. È possibile, però,

-

³ https://www.youtube.com/watch?v=CFRYYpXmVmE

derivare l'espressione (4.15) rispetto ai coefficienti s_x , s_y e s_z per trovare il minimo costo della manovra. Una possibile approssimazione analitica consiste nell'elevare al quadrato i termini delle due velocità:

$$\Delta v_a^2 + \Delta v_b^2 = (s_x x)^2 + (s_y y)^2 + (s_z z)^2 + (4.16)$$

$$+ (x - s_x x - \Delta x)^2 + (y + s_y y)^2 + (z + s_z z)^2$$

Il doppio prodotto $2\Delta v_a \Delta v_b$ viene trascurato. Si deriva poi la (4.16) rispetto ai coefficienti s_x , s_y , s_z e se ne annullano le derivate:

$$\frac{\partial(\Delta v_a^2 + \Delta v_b^2)}{\partial s_x} = 4x^2 s_x + 2mxy s_y + 2nxz s_z - 2x^2 = 0 \tag{4.17}$$

$$\frac{\partial(\Delta v_a^2 + \Delta v_b^2)}{\partial s_v} = 2mxys_x + (4y^2 + 2m^2y^2)s_y + 2mnyzs_z + 2y^2 - 2mxy = 0$$
 (4.18)

$$\frac{\partial(\Delta v_a^2 + \Delta v_b^2)}{\partial s_z} = 2nxzs_x + 2mnyzs_y + (4z^2 + 2n^2z^2)s_z + 2z^2 - 2nxz = 0$$
 (4.19)

I coefficienti s_x , s_y , s_z possono essere ricavati dal sistema:

$$s_x = \frac{2x + my + nz}{(4 + m^2 + n^2)x} \tag{4.20}$$

$$s_y = \frac{2mx - (4 + n^2)y + mnz}{(8 + 2m^2 + 2n^2)y}$$
(4.21)

$$s_{z} = \frac{2nx + mny - (4 + m^{2})z}{(8 + 2m^{2} + 2n^{2})z}$$
(4.22)

Sostituendoli nelle espressioni di Δv_a e Δv_b si può ottenere un costo approssimato della trasferta. Si osserva che quando s_x è zero, sia s_y che s_z sono pari a $-\frac{1}{2}$. Questo significa che la variazione di semiasse maggiore e inclinazione richiede due uguali valori di Δv . Si può anche tenere in considerazione il contributo dato dall'eccentricità. Per piccole variazioni di e si ottiene:

$$\Delta v_e = \frac{1}{2} v_0 \sqrt{(\Delta e_x)^2 + (\Delta e_y)^2}$$
 (4.23)

Dove $e_y = e \sin \omega$ e $e_x = e \cos \omega$.

Si può ottenere una formula empirica per il cambio di velocità richiesto, supponendo che la variazione di velocità dovuta al cambiamento di eccentricità sia distribuita equamente sulle due manovre:

$$\Delta v = \sqrt{(0.5\Delta v_e)^2 + \Delta v_a^2} + \sqrt{(0.5\Delta v_e)^2 + \Delta v_b^2}$$
 (4.24)

4.3 Calcolo della massa

Il chaser è il satellite che viene lanciato in ogni missione con lo scopo di portare ad ogni detrito un kit di de – orbit che gli permetta di deorbitare ed essere così rimosso. Pertanto, ad ogni rendez–vous la massa del chaser diminuisce e non è possibile calcolare la sua massa iniziale a partire dal Δv totale. È necessario, dunque, partire dalla fine della missione. È semplice ricavare un modello per il calcolo della massa iniziale di missione che prende in input i valori della dry mass, $m_{dry} = 2000kg$, della massa del kit di de – orbit, $m_{de} = 30 kg$, una sequenza di costi in termini di Δv , il valore di g_0 , l'accelerazione di gravità a livello del mare e l'impulso specifico del motore I_{sp} , fissato a 340 secondi dalla competizione.

Supponendo di avere una missione che raccoglie n debris, ci saranno allora n-1 trasferte con i corrispondenti costi $\Delta v_1, \Delta v_2, ..., \Delta v_{n-1}$. Per il calcolo della massa iniziale si utilizza la legge di Tsiolkovsky, per cui:

$$m_i = m_f e^{\Delta v/c} \tag{4.25}$$

Partendo dalla fine della missione, dopo che il chaser avrà rilasciato 1'n - esimo kit, la massa sarà:

$$m_{f,n} = m_{dry} (4.26)$$

Mentre la massa prima di rilasciare l'ultimo kit sarà:

$$m_{i,n} = m_{dry} + m_{de} \tag{4.27}$$

Applicando l'equazione di Tsiolkovsky per la trasferta dal detrito n-1 al detrito n, caratterizzata dal costo Δv_{n-1} , si ottiene la differenza di massa dovuta al consumo di propellente:

$$m_{f,n-1} = m_{i,n} e^{\Delta v_{n-1}/c} \tag{4.28}$$

Con c che rappresenta la velocità efficace di scarico ed è pari a $c = g_0 I_{sp}$.

Iterando in questo modo fino al primo detrito della missione, si ottiene la massa iniziale del chaser.

4.4 Teoria dei grafi

Per risolvere il problema e ricercare delle sequenze che permettessero di rimuovere tutti e 123 i detriti è stata applicata la teoria dei grafi. In questa sezione vengono presentati solamente i concetti fondamentali legati alle definizioni di grafi e alberi.

La teoria dei grafi si colloca nell'ambito della Ricerca Operativa, in cui si risolvono problemi di minimo o di massimo con opportune restrizioni. Essa possiede degli algoritmi che permettono di capire quale sia il percorso minimo su un grafo.

Un grafo è una struttura costituita da nodi collegati tra loro.

I collegamenti tra i nodi possono essere:

• non orientati: hanno una direzione, ma non un verso; in questo caso vengono definiti spigoli e il grafo viene definito "non orientato";

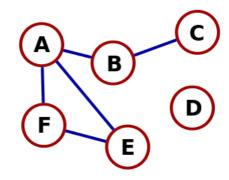


Figura 4.5 Esempio di grafo non orientato

• orientati: hanno una direzione e un verso; in questo caso vengono definiti archi e il grafo è detto "orientato" o "digrafo";

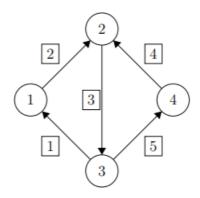


Figura 4.6 Esempio di grafo orientato

• il grafo si dice "pesato" se ad ogni collegamento è associato un valore numerico, detto appunto peso.

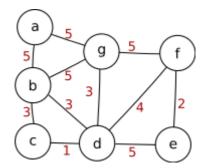


Figura 4.7 Esempio di grafo pesato

Quello che conta nel grafo sono solamente i nodi e le relazioni tra di essi, per cui un grafo può essere rappresentato in molti modi diversi.

Formalmente si dice grafo una coppia G = (V, E) di insieme, con V insieme dei nodi ed E insieme degli archi, tali che gli elementi di E siano coppie di elementi di E è un multiinsieme, allora si parla di multigrafo. I vertici (u, v) connessi da un arco e vengono detti estremi dell'arco.

Scheletro

Lo scheletro sk(G) di G è il grafo che si ottiene eliminando da G tutti gli archi che hanno estremi coincidenti e sostituendo ogni multiarco con un solo arco avente gli stessi estremi. $Grafo \ completo$

Un grafo è definito completo se due qualsiasi dei suoi vertici sono adiacenti, cioè esiste un arco che li connette.

Cammino, circuito, ciclo

Un percorso di lunghezza n in G è dato da una sequenza di vertici $v_0, v_1, ..., v_n$, non necessariamente tutti distinti e da una sequenza di archi che li collegano. I vertici v_0 e v_n sono gli estremi del percorso. Se il percorso ha i lati a due a due distinti tra loro si chiama cammino.

Se il cammino è chiuso $(v_0 = v_n)$ senza archi ripetuti si parla di circuito; se il cammino è chiuso senza archi né nodi ripetuti si parla di ciclo.

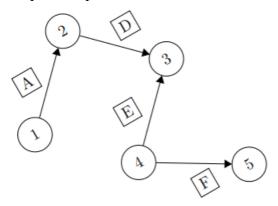


Figura 4.8 Esempio di percorso in un grafo

Dato un generico grafo G = (V, E) due vertici $u, v \in V$ si dicono connessi se esiste un cammino con estremi v ed u. Se il cammino non esiste u e v sono sconnessi.

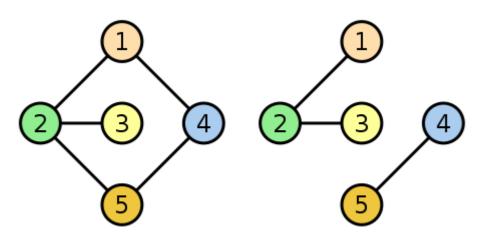


Figura 4.9 Esempio di grafo connesso (destra) e di grafo non connesso (sinistra)

Sotto-grafi e alberi

I sotto-grafi rappresentano una qualsiasi partizione del grafo originario. Se i nodi nel sotto-

grafo preservano gli archi originari allora si parla di sotto-grafo completo.

Un albero è un sotto-grafo che contiene tutti i nodi collegati. Poiché tutti i nodi sono collegati, l'albero contiene N-1 lati del grafo di origine. Inoltre, la scelta dell'albero non è univoca. Se l'albero è un sotto-grafo di un grafo G connesso, si dice che l'albero è ricoprente (spanning tree).

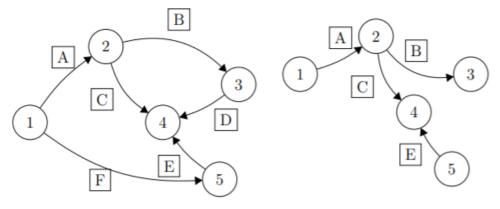


Figura 4.10 Grafo di riferimento con possibile albero

Circuito euleriano ed hamiltoniano

Se il grafo G è connesso e tutti i vertici hanno grado pari, dove per grado si intende il numero di archi che si connette al vertice, allora esiste un circuito euleriano E, ovvero un circuito che tocca tutti i suoi archi una e una volta sola.

Se, invece, dato un grafo, esiste un cammino attraverso alcuni archi che torni al punto iniziale percorrendo tutti i vertici una volta sola si parla di circuito hamiltoniano. Quindi, un circuito hamiltoniano *H* tocca tutti i vertici una ed una volta sola.

Rappresentazione di grafi e alberi

Un grafo può essere rappresentato mediante:

- liste di adiacenza: per ogni vertice $u \in V$ viene costruita una lista di vertici $v_1 \dots v_k$ adiacenti ad u, tali che $(u, v_i) \in E$ per ogni $i = 1, \dots, k$.
- Matrice di adiacenza: il grafo viene rappresentato con una matrice quadrata di ordine n = |V|, in cui M_{i,j} = 1 se (v_i, v_j) ∈ E e M_{i,j} = 0 altrimenti. Da un punto di vista pratico, all'aumentare dei nodi le informazioni vengono fornite con cataloghi di tipo sparse anziché full, per non dover memorizzare la presenza degli zeri che solitamente costituiscono la maggioranza nelle matrici di adiacenza.

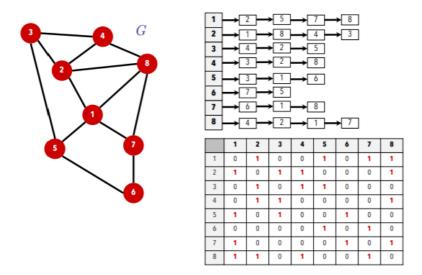


Figura 4.11 Rappresentazione di un grafo

Esempi di algoritmi risolutori per problemi su grafi

- Visita in ampiezza e in profondità di un grafo
- Ricerca del minimo albero ricoprente (minimum spanning tree algoritmo di Kruskal⁴)
- Ricerca del cammino minimo per raggiungere ogni altro vertice a partire da una sorgente (algoritmo di Dijkstra⁵).

4.4.1 Applicazioni

I sette ponti di Königsberg

La nascita della teoria dei grafi si fa risalire al 1736, anno in cui il matematico Leonhard Euler risolse il problema noto come i sette ponti di Königsberg, un problema ispirato da una città reale e da una situazione concreta.

http://bioinfo.ict.ac.cn/~dbu/AlgorithmCourses/Lectures/Prim1957.pdf

⁴ Per approfondimenti sull'algoritmo:

⁵ Per approfondimenti: https://en.wikipedia.org/wiki/Dijkstra%27s algorithm

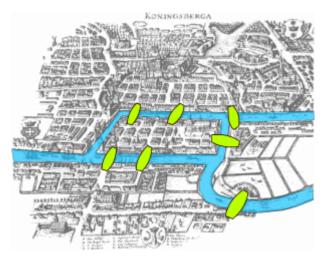


Figura 4.12 Mappa di Königsberg che evidenzia il fiume Pregel e i suoi ponti

Ci si chiedeva se fosse possibile con una passeggiata attraversare ogni ponte della città una volta soltanto. Eulero dimostrò che non era possibile, formulando il problema in termini della teoria dei grafi.



Figura 4.13 Dalla mappa al grafo

Sulla base delle osservazioni fatte da Eulero, un grafo qualsiasi è percorribile passando una sola volta sui suoi vertici se e solo se ha tutti i nodi di grado pari, o due di essi sono di grado dispari. È impossibile percorrere Königsberg passando una sola volta sui suoi ponti, in quanto tutti i nodi hanno grado dispari.

Il gioco dell'icosaedro

Nel 1859 Sir William Rowan Hamilton propose il gioco dell'icosaedro: ad ogni vertice era stato assegnato il nome di una città e si richiedeva di trovare un percorso che visitasse tutte le citta una sola volta e tornare poi al punto di partenza. Determinare se questo percorso esista è un problema NP-completo. Per determinare un cammino hamiltoniano si deve ricercare una permutazione $(v_0, v_1, ..., v_{n-1})$ dei nodi tali che $(v_i, v_{i+1}) \in E$ per ogni $0 \le i \le n-2$, dove E è l'insieme degli archi del grafo. Si

conclude che si ha un ciclo hamiltoniano se in un cammino hamiltoniano esiste un arco che collega l'ultimo vertice con il primo.

Il problema del commesso viaggiatore (TSP)

Il problema consiste, dato un insieme di città e note le distanze tra ciascuna coppia di esse, nel trovare il tragitto di minima percorrenza che il commesso viaggiatore deve seguire per visitare tutte le città una sola volta e ritornare nella città di partenza. La complessità del problema è racchiusa nel fatto di trovare il migliore circuito chiuso. Pertanto, questo problema è contenuto nella classe di problemi NP-completi. Trovare un algoritmo che risolva ogni esempio di TSP sarebbe un traguardo molto importante nel campo matematico, perché si sarebbe così in grado di risolvere efficientemente ogni problema computazionale per cui la risposta sia facilmente verificabile. Ad oggi, infatti, il problema è ancora aperto. Le strategie per risolvere i problemi NP-completi sono:

- trovare un caso specifico del problema per il quale sia possibile una soluzione esatta o un'euristica migliore;
- progettare algoritmi euristici con soluzioni probabilmente buone, ma impossibili da verificare che siano quelle ottimali;
- progettare algoritmi per trovare la soluzione esatta, con un numero di città relativamente bassi.

Tra gli algoritmi esatti utilizzati per il TSP si trovano il branch and bound⁶, il branch and cut⁷o algoritmi che utilizzano tecniche derivate dalla programmazione lineare. Tra gli algoritmi euristici, invece, uno dei più usati è il nearest neighbor⁸, che normalmente si avvicina alla soluzione ottima.

Classe di complessità NP

Per comprendere cosa significhi definire i problemi sopra descritti come appartenenti alla classe di problemi NP-completi, è meglio riportare brevemente che cosa si intende con questa denominazione.

Il problema delle classi P e NP è un problema ancora aperto nella teoria della complessità computazionale. La classe P comprende i problemi che possono essere

⁶Per approfondire l'algoritmo di branch and bound: https://it.wikipedia.org/wiki/Branch_and_bound

⁷ Per approfondire l'algoritmo di branch and cut:https://en.wikipedia.org/wiki/Branch and cut

⁸ Per approfondire l'algoritmo di nearest neighbor<u>https://www.mathworks.com/help/stats/classificationusing-nearest-neighbors.html#bsehylk</u>

risolti con un algoritmo dotato di complessità polinomiale, ossia se il tempo di risoluzione è una funzione polinomiale delle dimensioni dei valori in input. Questo si traduce in una complessità di calcolo dell'algoritmo $O(n^{\alpha})$ con $\alpha > 0$; generalmente se α è piccolo è probabile che il problema sia facilmente risolvibile.

Un problema NP, invece, è definito tale se, posto in una qualsiasi formulazione matematica, è verificabile, ma non risolvibile, in tempo polinomiale: si tratta, dunque, di verificare il problema in tempo polinomiale, data un'istanza affermativa del problema, cioè una soluzione ad esso.

La maggior parte dei problemi di riconoscimento, decisione ed ottimizzazione appartiene ai problemi NP.

Un problema NP-hard è un problema tale che un algoritmo per risolvere uno di questi problemi può essere convertito in un algoritmo per risolvere un qualunque problema NP. I problemi NP-hard sono, dunque, almeno complicati quanto i problemi NP. Se un problema è sia NP che NP-hard si definisce NP-completo ed è la classe più difficile di problemi da risolvere.

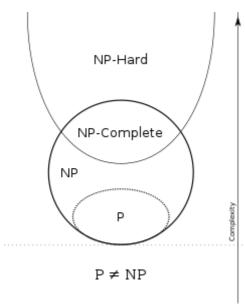


Figura 4.14 Suddivisione complessità dei problemi

Per molti aspetti il problema presentato nel seguente lavoro di tesi si avvicina al problema del commesso viaggiatore: l'obiettivo è, infatti, quello di trovare il percorso migliore che permetta di rimuovere tutti i detriti, senza la possibilità di ritornare su un detrito già rimosso, con il fine di minimizzare il costo della missione. A complicare il problema, però, è il fatto che l'ambiente spaziale cui fa riferimento sia dipendente dal tempo, per cui la posizione dei detriti evolve così come le interazioni tra di essi,

sottoponendo il grafo ad un'evoluzione tempo dipendente. Quindi, dato che il TSP nella sua definizione classica, cioè tempo indipendente, è attribuito alla classe dei problemi NP-completi, segue che il problema che si tenterà di risolvere ha almeno una complessità NP-completa.

5 Ricerca della sequenza e ottimizzazione

In questo capitolo verranno descritti il metodo e gli algoritmi, implementati in ambiente MATLAB, utilizzati per ricercare la sequenza dei detriti da rimuovere che sarà soluzione del problema e la relativa ottimizzazione.

Prima di trovare l'approccio che ha portato alla soluzione sono stati considerati altri tre metodi differenti, che verranno descritti brevemente di seguito.

Quello che accomuna tutti gli approcci e facilita il lavoro è la creazione di un database che contiene tutti i costi in Δv di ogni detrito con ogni altro detrito nel tempo. Le figure che seguono mostrano la situazione dei costi tra i detriti.

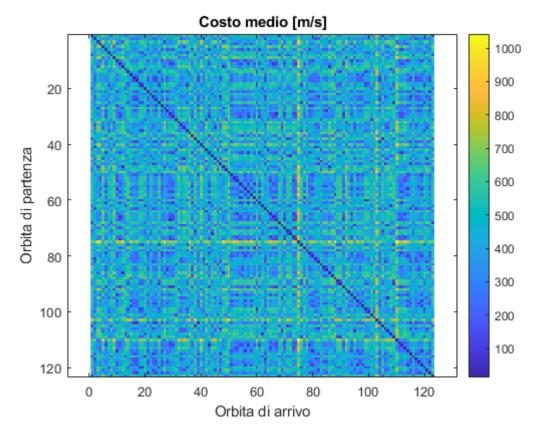


Figura 5.1 Costo medio

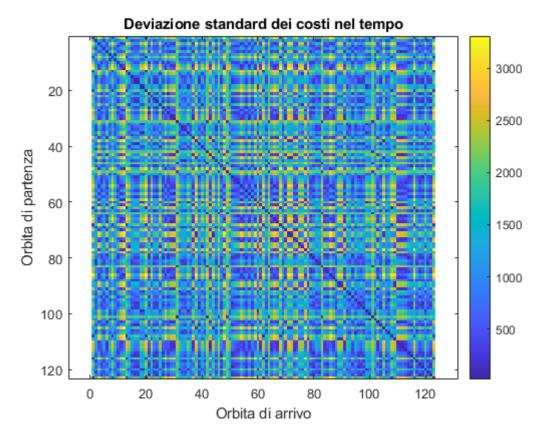


Figura 5.2 Deviazione standard dei costi nel tempo

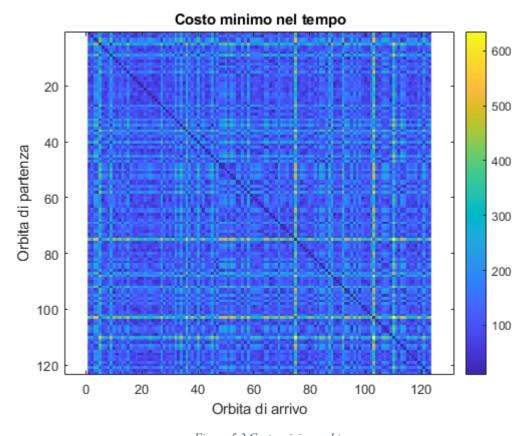


Figura 5.3 Costo minimo nel tempo

Metodo con un singolo albero

In una fase iniziale si era pensato di raggruppare i nodi in maniera iterativa con un albero singolo non orientato. L'albero, in questo modo, evolveva nel tempo e veniva ricalcolato nel tempo sia per le variazioni dei parametri orbitali sia perché dall'albero, nell'iterazione precedente, era stato estratto un numero minimo di nodi k. La creazione dell'albero a partire dal grafo avveniva grazie alla funzione built-in di MATLAB *graphminspantree*, che evidenzia lo scheletro con la minima estensione. Variando il numero minimo di detriti da raccogliere e il numero di sotto-missioni (da 8 a 11) si riuscivano a trovare delle missioni in grado di raccogliere un centinaio di detriti, ma in alcune iterazioni l'albero non forniva nessuna sequenza praticabile, risultando così un tempo morto nella missione. Questo approccio, tuttavia, si è rivelato non essere sufficientemente adeguato alla risoluzione del problema, a causa della forte dipendenza dal tempo di quest'ultimo.

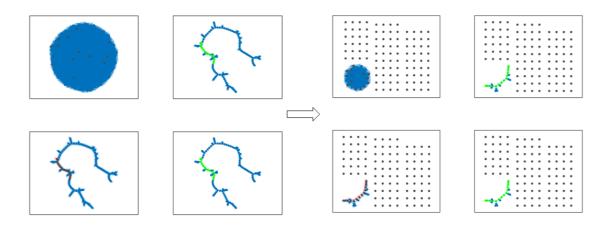


Figura 5.4 Esempio con 8 sotto-missioni dall'albero iniziale a quello finale

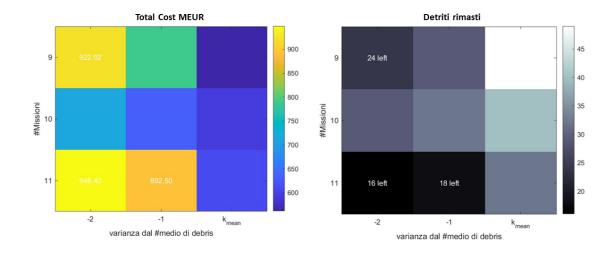


Figura 5.5 Costi totali delle sotto-missioni e detriti rimasti

Partizione dei grafi

Un altro metodo per suddividere i nodi avrebbe potuto essere la suddivisione del grafo in più sotto-grafi. La difficoltà risiede nel criterio scelto per la suddivisione in quanto i nodi sono fortemente connessi.

Il processo risulterebbe molto simile a quello della costruzione della missione adottando un solo albero. Partizionando il grafo in più sotto-grafi e poi traducendoli nei loro alberi ad estensione minima si otterrebbe una semplificazione più efficiente, poiché in un singolo albero la struttura dei nodi connessi contiene anche legami dal costo elevato: rimuovendo questi legami inefficienti sarebbe più facile identificare una sequenza di detriti scegliendo arbitrariamente tra i sotto-grafi. Bisogna ricordare che il grafo è tempo dipendente, per cui la clusterizzazione dovrà essere reiterata aggiornando le connessioni dei nodi allo scorrere del tempo.

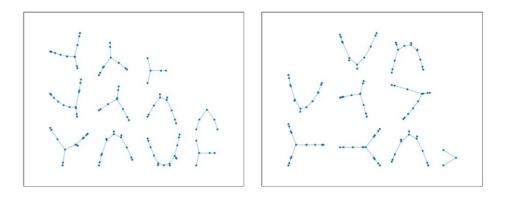


Figura 5.6 Esempio di partizione dei grafi

Ricerca dei **Av** minimi

L'obiettivo in questo approccio è identificare il valore minimo di Δv tra tutte le griglie temporali e, da quei detriti caratterizzati dal costo minimo, far partire la sequenza, incrementandola sempre ricercando il Δv minimo. Per esempio, se all'istante i avviene il passaggio ottimale da A a B e all'istante i+2 il passaggio ottimale da C a D, allora si inserisce automaticamente il passaggio intermedio tra B e C. Il problema è che i primi due passaggi erano altamente convenienti, il terzo prevede quasi sempre un costo elevato. La sequenza così costruita vedeva un alternarsi di costi bassissimi e costi altissimi, portando all'abbandono di questo metodo.

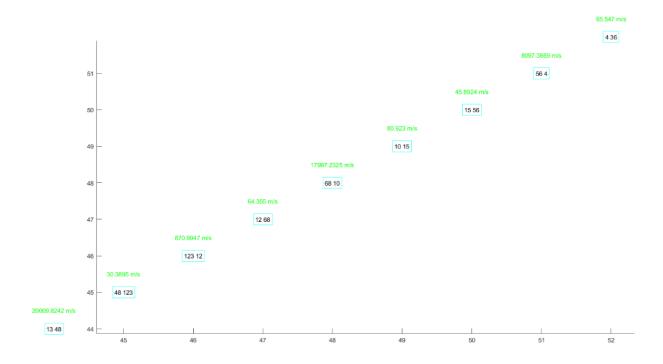


Figura 5.7 Sequenza con la ricerca dei costi minimi

Ricerca euristica nel piano temporale

L'approccio che invece ha permesso di giungere ad una soluzione prevede una ricerca euristica delle sequenze nel piano temporale. Il flow chart che segue spiega il procedimento di costruzione della soluzione.

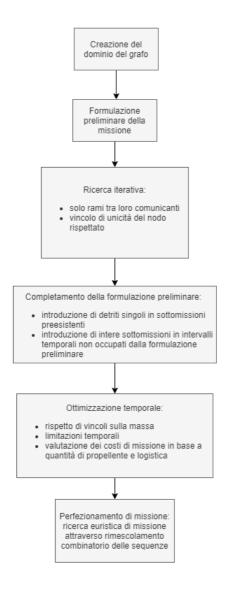


Figura 5.8 Flow chart della costruzione della soluzione

5.1 Ricerca delle sequenze

Nell'algoritmo di ricerca delle sequenze la linea temporale degli 8 anni, consentiti dalla competizione per lo svolgimento della missione, è stata suddivisa in numero sufficientemente elevato di frames, in questo caso 300. Tramite la creazione di un grafo orientato si possono memorizzare solo i passaggi tra un detrito e un altro con un costo in termini di Δv inferiore ad una determinata soglia, scelta arbitrariamente. In questo caso, la soglia è stata imposta a 370 m/s: una soglia troppo bassa avrebbe reso il grafo poveramente connesso, ma una soglia troppo alta non sarebbe stata un filtro adeguato e il risparmio in termini economici sarebbe stato compromesso.

In una matrice si tiene traccia di tutti i detriti coinvolti nella ricerca. Ogni detrito è definito in modo univoco da due parametri: il suo identificativo (da 1 a 123) e il frame temporale a

cui fa riferimento il passaggio orbitale (da 1 a 300).

La valutazione del costo dei trasferimenti orbitali viene ripetuta con tre diversi valori di durata del trasferimento, considerando fissa a 5 giorni la durata del rendez-vous⁹: i calcoli, dunque, sono stati svolti rispettivamente con 5, 15 e 25 giorni di trasferimento. Solo i trasferimenti con un costo inferiore alla soglia specificata vengono poi presi in considerazione. In un grafo, g, vengono memorizzati i legami assegnando come peso un valore tra 1, 2 e 3 in base alla durata del trasferimento orbitale (1 per 5 giorni, 2 per 15 giorni, 3 per 25 giorni), mentre in un altro grafo, DV, il peso del legame coincide con il costo in Δv (m/s).

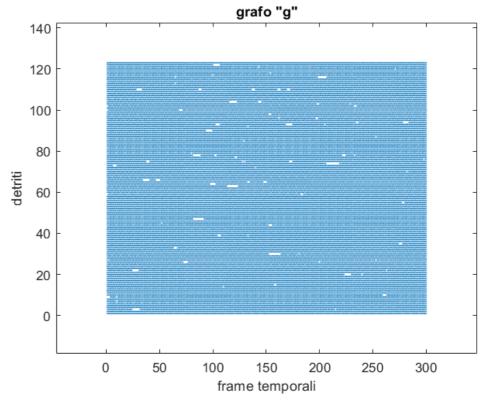


Figura 5.9 Grafo 'g'

.

⁹ 5 giorni sono sufficienti per il rendez-vous, dal momento che le orbite LEO eliosincrone hanno un periodo medio di circa 90-100 minuti, senza contare che la quantità di propellente per le manovre di rendez-vous è molto più bassa di quella necessaria per un trasferimento tra orbite. Pertanto, la difficoltà del problema si sposta da un punto di vista logistico (come effettuare concretamente la rimozione del detrito) a uno di tipo decisionale (verso quale orbita orientarsi).

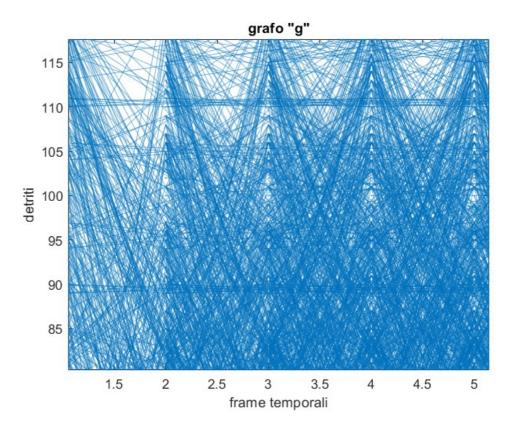


Figura 5.10 Dettaglio grafo 'g'

Per la comprensione dell'algoritmo si può introdurre il concetto di filone: una sequenza in cui sono presenti informazioni sui nodi e sugli istanti temporali in cui vengono raccolti. Ogni singolo filone costituisce la base di partenza con cui giungere ad una soluzione definitiva.

La ricerca viene svolta prendendo più filoni temporali, in questo caso 20, nei quali viene memorizzato un possibile percorso. La struttura del grafo, avendo una bassa copertura (3%), impedisce la creazione di un filone senza interruzioni. Ogni filone viene ampliato a più riprese, svolgendo ricerche nel grafo indagando i nodi cronologicamente successivi a partire da un determinato nodo, scelto arbitrariamente. Quando il filone va incontro ad una interruzione, viene ampliato con un nuovo blocco di detriti ottenuto sempre da una ricerca nel grafo, a partire dal punto in cui ci si era interrotti. Poiché la ricerca nel grafo risulta dispersiva (il grafo possiede 123x300 nodi), è necessario porvi un limite, scegliendo un numero opportuno di cicli e al termine di questi arricchire il filone con il blocco migliore trovato.

Ci sono più cause per le quali un filone può subire un'interruzione: il nodo potrebbe non avere connessioni con detriti per cui il trasferimento orbitale sia sotto la soglia imposta oppure i nodi successori si riferiscono a detriti già raccolti. Tra un'interruzione e l'altra si

dovrà identificare e far coincidere l'inizio e la fine di due sotto-missioni. Al termine della ricerca ogni filone può essere elaborato e modificato per giungere ad una soluzione del problema, poiché opera indipendentemente dagli altri filoni.

La ricerca termina quando almeno uno dei filoni colleziona quasi la totalità dei detriti.

Osservando più nel dettaglio il processo di ampliamento dei filoni a seguito di un'interruzione, si presenta in via del tutto schematica una sequenza di istruzioni che l'algoritmo deve eseguire.

Innanzitutto, viene inizializzata una variabile temporanea con diverse caratteristiche (chiamata nel codice *traveller*, adottato per comodità anche nel testo), a cui viene assegnato un nodo di partenza in modo che:

- il detrito rappresentato da quel nodo non sia già presente nel filone
- il nodo sia distante almeno 30 giorni dall'ultimo nodo nella sequenza del filone
- il nodo possieda legami con altri nodi.

Se esistono più nodi con queste caratteristiche l'assegnazione è casuale; se nessun nodo le rispetta, invece, l'assegnazione si concentra sul gruppo di nodi appartenenti a frames temporali successivi.

Ogni variabile *traveller* tiene traccia della sequenza che verrà costruita, dell'insieme dei nodi alternativi per costruire la lista e del numero più alto di detriti raccolti senza interruzioni. Alla fine delle 500 iterazioni, si ritiene di aver svolto una ricerca piuttosto approfondita e ogni filone viene incrementato concatenando la sequenza più lunga mai registrata dalla variabile *traveller*.

Riassumendo, lo schema con cui la sequenza viene costruita è il seguente:

- si cercano i nodi consecutivi all'ultimo nodo del *traveller*. Questo avviene tramite la funzione *successors*, una funzione built-in di MATLAB negli algoritmi della teoria dei grafi, che permette di trovare un nodo successore in un grafo diretto a partire da un determinato nodo.
- se ci sono successori si rimuovono i detriti che sono già stati presi dal filone o dal *traveller* stesso.
- se l'insieme che si ottiene dallo step precedente non è vuoto, si memorizza nel *traveller* tale insieme e da questo viene scelto casualmente un nodo per ampliare la lista.
- si ripete la procedura sopra descritta.

Nella figura 5.11 si può osservare l'algoritmo nella sua esecuzione, poiché è stato dotato di un'interfaccia per sapere come procede qualitativamente la ricerca in tempo reale. L'interfaccia mostra il numero di detriti già raccolti e non più modificabili di ogni filone (numeri in verde) e accanto ad essi viene rappresentato il contenuto del rispettivo *traveller* che arricchirà la sequenza. La sequenza del *traveller* è ancora modificabile e l'algoritmo la sta formulando orientandosi all'interno del grafo. Ogni singola modifica al *traveller* (aggiunta, rimozione o sostituzione) viene conteggiata come un'iterazione. Al termine del limite prestabilito di iterazioni, la lista più lunga che è stata formulata viene inglobata nel filone ed incrementa il quantitativo di detriti raccolti.

Nell'immagine di sinistra i filoni hanno subìto una sola interruzione e hanno collezionato dai 30 ai 50 detriti circa; l'algoritmo ha successivamente aperto una nuova sessione di ricerca a partire dal risultato ottenuto fino ad arrivare al risultato che si vede nell'immagine di destra. Si può notare che nelle fasi iniziali quasi tutti i filoni trovano con facilità lunghe sequenze di detriti, mentre nelle fasi successive, quando il numero di detriti collezionati comincia ad essere consistente, le sequenze di incremento sono generalmente più corte.

Questo aspetto è comprensibile: un filone ricco di detriti esercita limitazioni più stringenti nella ricerca, dal momento che nessun detrito può essere ripetuto più di una volta nella sequenza.

Figura 5.11 Generazione delle sequenze

Prima di procedere nella descrizione delle fasi successive, viene riportato di seguito uno schema che riassume le operazioni svolte finora per la costruzione della soluzione definitiva del problema.

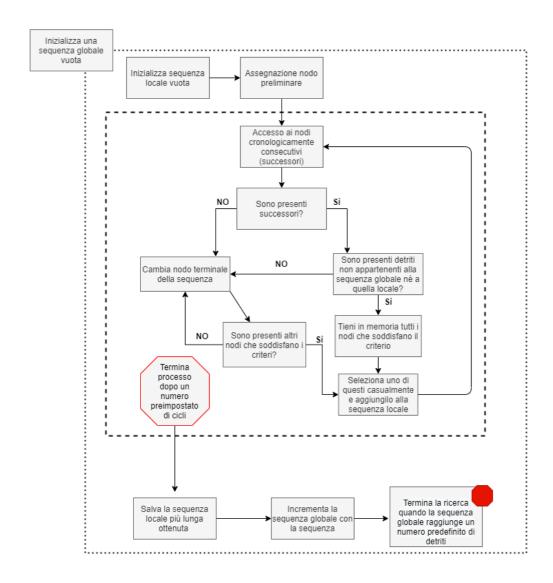


Figura 5.12 Flow chart ricerca delle sequenze preliminari

5.2 Costruzione della soluzione e ottimizzazione

Dopo aver lanciato più volte il codice per la costruzione delle possibili sequenze preliminari (i filoni della sezione precedente), le migliori sono state memorizzate. Una soluzione unica e nettamente migliore di altre non esiste per un problema di questo tipo. È importante sottolineare che nessun filone restituisce una soluzione completa: alcuni detriti non sono stati raccolti e devono essere introdotti in un secondo momento. Inoltre, la ricerca nel grafo non ha tenuto in considerazione le limitazioni sulla massa iniziale dello spacecraft, ma solo i vincoli sulla non ripetizione dei detriti. Pertanto, nelle prossime sezioni verranno esposte tutte le tecniche con cui le sequenze preliminari sono state migliorate.

Quello che è importante controllare ora, e che costituisce anche uno dei vincoli della competizione, è la massa iniziale di ciascuna sotto-missione. Focalizzandosi sul valore di

soglia con cui partizionare opportunamente una lista di detriti, è risultato conveniente imporre una soglia lievemente superiore a quella indicata nella competizione. In particolare, la massa iniziale non potrà superare gli 8000 kg - 7000 kg sarebbe il valore ufficiale non tenendo in considerazione la massa dei kit di de-orbit, ma si è notato come nel processo di ottimizzazione anche le masse diminuiscano, per cui la soglia imposta è leggermente superiore. In questo modo, l'algoritmo può operare con una libertà maggiore in uno scenario in cui le informazioni non sono definitive.

Se la massa della sotto-missione supera la soglia, la sotto-missione viene suddivisa in due sotto-missioni. Questo taglio avviene in un istante preciso, cioè dove i trasferimenti orbitali sono più lunghi (30 giorni). Una sequenza è costituita da un susseguirsi di trasferimenti orbitali di durata variabile, alcuni molto brevi, alcuni di durata intermedia e alcuni molto lunghi. Solo tra i più lunghi viene scelto quello posizionato più vicino al centro della sequenza, in modo da suddividere più equamente le due sotto-missioni. Il taglio avviene solo sui trasferimenti da 30 giorni, poiché, se avvenisse dove il trasferimento orbitale è più breve, si introdurrebbe con la pausa forzata dei 30 giorni tra un lancio e il successivo uno shift nella cronologia che danneggerebbe l'intera sotto-missione.

Per poter rispettare i numerosi vincoli è necessario poter avere facile accesso alle informazioni sulle missioni che si stanno delineando.

Sostanzialmente è necessario monitorare una missione in base a queste caratteristiche:

- la suddivisione delle sotto-missioni
- il numero di detriti raccolti
- l'identificativo di ogni detrito
- il tempo in cui ogni detrito viene raccolto
- i costi in termini di Δv dei trasferimenti orbitali
- il costo in MEUR della missione
- il rispetto delle condizioni di massa.

```
n: [12 14 13 10 8 9 8 13 8 4 7 5 6 6]

list: [1×123 uint8]

pat: {1×14 cell}

feas: [14×1 double]

estdV: {1×14 cell}

total_cost: [1×14 double]

costo_medio_min: [1×14 double]

estM: [1×14 double]

massMEUR: [1×14 double]

timeMEUR: [1×14 double]

timeMEUR: [1×14 double]

timeMEUR: [1×14 double]

interval: [1×123 double]
```

Figura 5.13 Esempio di struct che contiene le informazioni della soluzione

Due sono le esigenze da soddisfare per la creazione della soluzione: in un primo step l'obiettivo è quello di incrementare la sequenza, cercando di raccogliere tutti i 123 detriti; successivamente si pensa all'ottimizzazione dei costi giocando con la cronologia dei trasferimenti senza alterare la partizione o la sequenza globale.

Per l'incremento della sequenza è stato implementato un algoritmo complesso data la difficoltà del problema da affrontare: l'aggiunta dei detriti procede con miglioramenti localizzati anziché ottimizzando globalmente la sequenza, poiché un'ottimizzazione globale è complicata da codificare.

I punti che seguono rappresentano un riassunto dei principali step dell'algoritmo. Nei paragrafi successivi saranno descritti nel dettaglio i processi più importanti.

- Viene presa in considerazione la lista dei detriti rimasti, che non sono ancora stati raccolti;
- si calcola il tempo che intercorre tra una sotto-missione e la seguente. Si isolano solo le pause maggiori di una certa quantità di tempo, poiché non possono essere inferiori ai 30 giorni secondo i vincoli imposti dalla competizione, ma al tempo stesso bisogna garantire un trasferimento orbitale con un costo sufficientemente ridotto con quantità di tempo ragionevoli necessarie alla trasferta;
- a questo punto vengono messe in atto due diverse modalità di completamento, una che prevede l'ampliamento di sotto-missioni già esistenti e una che tenta di creare nuove sotto-missioni.

5.2.1 Prima modalità di completamento della sequenza

La prima modalità è la più semplice e prevede che un detrito venga incluso all'inizio di una sotto-missione, a patto che la sua introduzione permetta di soddisfare ancora il vincolo sulla massa. Il motivo per cui il detrito viene introdotto all'inizio e non alla fine è dovuto al fatto che il calcolo della massa iniziale viene svolto a ritroso, quindi la scelta migliore è non cambiare il termine iniziale della successione dei Δv . Se invece non ci fosse la massa del kit di de-orbit da scaricare dopo il rendez-vous con ogni detrito, l'ordine dei Δv non influirebbe sulla massa risultante e si potrebbe pensare di collocare nuovi detriti nella sotto-missione sia all'inizio sia alla fine.

L'introduzione del nuovo Δv non comprime sull'asse temporale i detriti già presenti nella sotto-missione, quindi è necessario che la pausa tra la sotto-missione in questione e quella che la precede sia sufficientemente larga, in modo che siano ancora disponibili 30 giorni di separazione dopo l'aggiunta del nuovo detrito. Questo processo può essere effettuato più e più volte su una stessa sotto-missione. Poiché questa modalità di ampliamento non genera nuove sotto-missioni, risulta solitamente molto conveniente in termini economici perché tutte le sotto-missioni contribuiscono con un costo fisso al costo complessivo. Proprio per questo, l'algoritmo tenta prima di introdurre detriti singoli invece di inserirli in blocco come nuova sotto-missione.

L'algoritmo che sviluppa questo processo è il seguente e sfrutta le informazioni temporali della soluzione preesistente:

- individua i detriti iniziali di ogni sotto-missione anticipata da una pausa sufficientemente estesa
- svolge il calcolo del Δv tra tutti i detriti avanzati e con i detriti del punto precedente
- elegge la combinazione "detrito avanzato-detrito di aggancio" con il Δv più basso
- ricalcola la massa della specifica sotto-missione a cui appartiene il detrito di aggancio e convalida l'introduzione solo se la massa non supera il valore di soglia imposta. Se la convalida non va a buon fine, allora nessun Δv può essere introdotto in quel punto.

```
sto cercando di ampliare la missione prima del nodo #73
metto nella 11° submission il detrito #50
prevedo 3171.9 [kg]
detriti attuali: 103
sto cercando di ampliare la missione prima del nodo #50
metto nella 11° submission il detrito #16
prevedo 3459.7 [kg]
detriti attuali: 104
sto cercando di ampliare la missione prima del nodo #74
metto nella 10° submission il detrito #111
prevedo 3487.9 [kg]
detriti attuali: 105
sto cercando di ampliare la missione prima del nodo #111
metto nella 10° submission il detrito #116
prevedo 4412.0 [kg]
detriti attuali: 106
sto cercando di ampliare la missione prima del nodo #116
metto nella 10° submission il detrito #98
prevedo 4739.7 [kg]
detriti attuali: 107
sto cercando di ampliare la missione prima del nodo #9
metto nella 9° submission il detrito #77
prevedo 5026.9 [kg]
detriti attuali: 108
```

Figura 5.14 Ampliamento della sequenza con la prima modalità

5.2.2 Seconda modalità di completamento della sequenza

Se non si può applicare il metodo appena descritto, si procede con la seconda modalità di completamento: se esiste una pausa sufficientemente lunga da poter fare più trasferimenti orbitali, si cerca di inserire un'intera nuova sotto-missione con una porzione dei detriti non ancora raccolti. Se la pausa si trova tra due sotto-missioni già esistenti è necessario rispettare il vincolo dei 30 giorni di pausa tra un lancio e il successivo. Se la pausa, invece, si trova dopo l'ultima sotto-missione esistente è necessario garantire il completamento della missione prima della scadenza degli 8 anni.

L'algoritmo si occupa dell'aggiunta di nuove sotto-missioni. Solitamente i detriti rimasti sono quelli che richiedono un trasferimento orbitale costoso, per cui la via più semplice per ridurre il loro costo ed introdurli nella missione è di dilatare i tempi del trasferimento. Ipotizzando, quindi, la massima durata del trasferimento orbitale, 30 giorni, viene calcolato il numero di detriti che possono essere raccolti in una determinata pausa. I detriti vengono collocati in un vettore temporale equi-spaziato in base al numero di detriti che è possibile raccogliere.

Per determinare la sequenza ottimale di detriti da aggiungere in blocco si procede per via combinatoria, introducendo 7 detriti alla volta. Il numero 7 non è scelto casualmente, ma aumentandolo si avrebbe un incremento nei tempi di calcolo o la RAM necessaria sarebbe così elevata da rendere impossibile l'operazione.

Se il numero degli slot temporali disponibili è maggiore di 7, ci si concentra su una porzione di essi (precisamente 7) e la sequenza viene incrementata a partire dal fondo. Calcolati i

costi in termini di Δv di tutti i detriti con ogni altro detrito in tutti gli slot temporali, si ottiene la lista delle combinazioni scegliendo un numero di detriti dall'insieme di quelli rimasti che equivale al numero di detriti che è possibile raccogliere, calcolato in precedenza.

Per ogni possibile combinazione, poi, si ottiene la lista delle permutazioni e si identifica in ogni permutazione dopo quanti detriti si supera la soglia di massa imposta a 7000 kg. Ogni permutazione offre un numero differente di massimi detriti raccoglibili, per cui la soluzione che verrà scelta, ovvero il set di detriti che andrà a formare una nuova sotto-missione, sarà quella con il maggior numero di detriti e con la massa inferiore. Quando una sotto-missione viene innestata, l'algoritmo tenta di innestarne una nuova concentrandosi su una nuova pausa sufficientemente lunga in cui agire con i detriti avanzati.

Quando tutti i detriti sono stati raccolti oppure quando l'algoritmo non trova il modo di inserirne altri, il processo termina.

```
sto cercando di ampliare la missione prima del nodo #77 non riesco ad introdurre detriti: dV troppo alti introduco una submission dopo la 10° prevedo 5595 kg per 5 detriti detriti attuali: 113 sto cercando di ampliare la missione prima del nodo #98 non riesco ad introdurre detriti: dV troppo alti introduco una submission alla fine prevedo 5552 kg per 6 detriti detriti attuali: 119 sto cercando di ampliare la missione prima del nodo #49 non riesco ad introdurre detriti: dV troppo alti introduco una submission dopo la 9° prevedo 4346 kg per 4 detriti detriti attuali: 123
```

Figura 5.15 Seconda modalità di completamento della sequenza

5.2.3 Ottimizzazione dei costi

Raccolti tutti i detriti, si procede con l'ottimizzazione dei costi, facendo variare gli istanti temporali e le durate dei trasferimenti orbitali, senza alterare la sequenza di rimozione dei detriti. Per questo processo è stata utilizzata la funzione di MATLAB *fmincon*, ideata per l'ottimizzazione di problemi non lineari soggetti a vincoli. L'algoritmo che *fmincon* utilizza di default è l'Interior Point Algorithm.

fmincon Interior Point Algorithm

Questo tipo di approccio prevede la risoluzione di una sequenza di problemi di minimo approssimati. Il problema originale è: $\min f(x)$ (target) tale che h(x) = 0 e $g(x) \le 0$

(constraints).

Per ogni $\mu > 0$ il problema approssimato si presenta come:

$$\min f_{\mu}(x,s) = \min f(x) - \mu \sum_{i} \ln (s_{i})$$

Tale che h(x) = 0 e g(x) + s = 0. Ci sono tante variabili slack s_i quanti sono i vincoli di disuguaglianza g. Quando μ tende a 0 il minimo di f_{μ} dovrebbe avvicinarsi al minimo di f.

L'equazione del problema approssimato è una sequenza di problemi vincolati dall'uguaglianza e questi sono più facili da risolvere rispetto al problema originale vincolato dalla disuguaglianza.

Partendo da un punto interno alla regione ammissibile (insieme delle soluzioni ammissibili), l'algoritmo si sposta iterativamente lungo la direzione di massima diminuzione di f_{μ} raggiungendo il confine della regione ammissibile solo alla fine.

Funzione da ottimizzare e creazione dei vincoli

Contestualizzando il processo di ottimizzazione al problema da affrontare, è stata necessaria una formulazione matematicamente compatta. Se *fmincon* risolve problemi di minimo con vincoli, allora la funzione f(x) coincide con la funzione costo di missione. Il vettore x contiene le informazioni temporali, mentre g(x) rappresenta tutte le limitazioni dal punto di vista temporale. La sequenza dei nodi e la loro partizione non sono dei veri input della funzione, poiché non possono essere modificati.

La funzione da ottimizzare è una funzione del vettore temporale, che viene preso come input da una function, la quale restituisce sia il costo totale della missione sia l'oggetto in cui vengono raccolti tutti i parametri di ogni sotto-missione, tra cui la lista dei detriti, la lista dei detriti avanzati, il costo in Δv , il costo totale e le masse.

La creazione dei limiti entro cui ogni variabile x_i può variare è stata automatizzata tramite un'altra funzione, poiché dipende dalla partizione dei detriti. In presenza di un trasferimento orbitale i giorni a disposizione vanno da 5 a 30, mentre gli intervalli temporali tra una sotto-missione e la successiva partono da un minimo di 30 giorni fino ad un valore arbitrario. L'unica eccezione è il valore della prima incognita che sancisce l'inizio di tutta la missione. A seconda della partizione con cui i detriti sono stati raccolti si otterrà:

• un limite inferiore: 0 [5 5 ... 5] 30 [5 5 ... 5] 30 [5 5 ... 5]. Il lower bound è, dunque, un vettore di (1+122) elementi in cui il primo valore sancisce il momento,

- a partire dall'inizio della competizione, in cui partirà la prima sotto-missione. I restanti elementi rappresentano gli intervalli minimi tra un detrito e il successivo.
- un limite superiore: un vettore in cui il primo elemento rappresenta il massimo ritardo della prima sotto-missione rispetto al momento di inizio della competizione (impostato a 10 giorni nel codice). Il trasferimento orbitale tra i detriti è imposto al valore massimo di 30 giorni, mentre per le pause tra le sotto-missioni si può fare riferimento ad un arbitrario ritardo Δd, che nel caso specifico è stato impostato pari alla durata più lunga presente nella missione preesistente più un supplemento di 10 giorni. Il limite superiore è nella forma: ΔD [30 30 ... 30] Δd [30 30 ... 30] Δd [30 30 ... 30].
- i termini della disequazione del tipo $Ax \le B$ (vincoli aggiuntivi di g(x)). È necessario, dunque, che la somma di tutti gli intervalli temporali non superi gli 8 anni. Pertanto, A si trasforma in un vettore di coefficienti unitari per ottenere la durata complessiva di missione e B si identifica con la durata di 8 anni espressa in giorni.
- un vettore di condizioni iniziali \vec{x}_0 ottenibile dalle informazioni della missione preesistente all'uscita dal processo di completamento.

Con questa prima ottimizzazione temporale si ottengono i primi risultati. Lo step successivo prevede l'ottimizzazione della lista dei nodi.

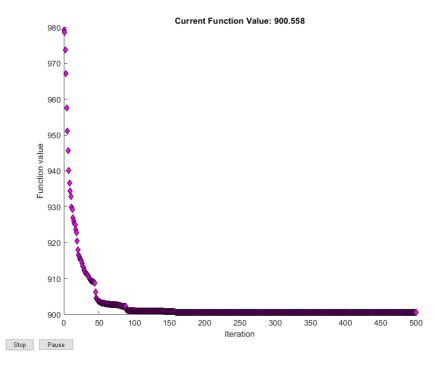


Figura 5.16 Processo di convergenza

Ottimizzazione della lista dei nodi

Si intende verificare che non ci siano combinazioni di detriti più valide di quelle già ottenute. In questo processo la timeline non viene modificata. L'algoritmo prevede lo scambio di posizione di alcuni detriti al fine di ridurre il costo complessivo. Viene determinata una sequenza casuale di permutazione con un numero casuale di detriti compreso tra 2 e 6. Calcolata la nuova missione, se questa ha un costo complessivamente minore di quella già ottenuta allora sarà questa nuova missione con i detriti scambiati a costituire la soluzione.

Per un'ulteriore verifica si tenta un'altra ottimizzazione tramite la modifica della timeline utilizzando la function *fmincon*, ma si nota che si è già in partenza su un minimo locale della funzione e quella che si ottiene è la soluzione complessiva. Per comprendere meglio l'obiettivo di questa fase finale si fa riferimento alla sezione dei risultati, in cui, con lo scambio di pochi detriti, si ottengono notevoli miglioramenti di costo.

6 Risultati e confronti

In questo capitolo viene presentata la soluzione al problema in esame, trovata tramite l'applicazione della teoria dei grafi. Questa soluzione verrà poi confrontata con la soluzione proposta dal team del JPL, vincitore della competizione GTOC9.

Tramite l'algoritmo di ricerca e ottimizzazione sopra descritto, si è riusciti a trovare una sequenza per rimuovere tutti i 123 detriti in orbita eliosincrona. Quella che viene presentata come soluzione definitiva del problema deriva da un filone che nella ricerca nel grafo aveva collezionato 102 detriti e che, grazie alle tecniche di completamento, ha incluso i 21 detriti mancanti e che poi ha subito ulteriori miglioramenti. È interessante notare che erano presenti sequenze preliminari con più detriti rispetto al filone da 102, ma al termine del processo si sono rivelate più costose.

La soluzione prevede la suddivisione in 14 sotto – missioni, ognuna che raccoglie un numero di detriti compreso tra 4 e 14 per un costo totale di 865.38 MEUR.

6.1 Risultati preliminari dopo l'ottimizzazione temporale

Dopo l'ottimizzazione temporale con *fmincon* sono stati ottenuti dei risultati preliminari, che vengono poi ulteriormente ottimizzati. La soluzione preliminare prevede sempre 14 sotto-missioni, ma un costo in MEUR complessivamente superiore. Quello che cambierà dopo l'ottimizzazione combinatoria sarà la lista dei detriti raccolti, che presenterà degli scambi al suo interno.

Lancio	# debris	Lista (soluzione preliminare)
1	12	34 39 82 63 104 115 23 96 17 119 58 97
2	14	122 51 24 118 88 21 85 26 80 114 79 37 56 6
3	13	40 46 83 8 71 86 48 27 105 3 69 123 1
4	10	109 68 7 19 12 25 45 121 76 38
5	8	53 22 65 31 101 67 20 70
6	9	94 10 66 91 15 107 29 78 95
7	8	93 112 13 57 43 72 110 44
8	13	52 99 60 92 103 55 100 2 36 98 5 41 113
9	8	77 9 28 117 108 62 18 81
10	4	120 33 32 89

11	7	98 116 111 74 87 35 11
12	5	61 54 4 84 14
13	6	16 50 73 64 59 42
14	6	49 106 47 75 102 30

Tabella 6.1 Lista preliminare dei detriti nelle sotto-missioni

Lancio	Δv [m/s] (soluzione preliminare)		
1	355.4 242.9 177.3 354.2 84.9 337.7 207.6 128.7 275 105.8 285.2		
2	154.3 326.7 363.9 306.6 210.5 232.4 395.5 312 359.8 112.2 125.9 322.5 233.2		
3	228.3 149.8 181.5 395.4 122.9 118.8 271.5 393.7 118.6 334.2 246.3 224.5		
4	129.9 132.3 259.5 145.1 397.6 230 106.4 203.9 165.3		
5	279.4 192 208.4 200.2 145.9 154.3 382.9		
6	296.5 138.1 285.5 168.9 284.3 258.1 332.6 241.4		
7	152.3 298.4 241.2 142.7 198.1 341 344.5		
8	163 28.8 213.6 128.8 286.6 139.6 234.8 254.6 383 264.8 241.9 163.8		
9	1154.3 224.9 57.8 172.8 200 432.7 155		
10	213.8 1496.5 682.2		
11	221.4 746.5 864.8 256.5 199.2 302.4		
12	474.4 554.9 1713.4 221.3		
13	236.6 508.6 135.2 369.7 180.8		
14	233.7 428.5 332.3 1252.3 795.1		

Tabella 6.2 Costi dei trasferimenti orbitali preliminari

La timeline della missione e le durate dei trasferimenti orbitali non variano dopo l'ottimizzazione combinatoria, a seguito della quale si ottiene la soluzione finale. Pertanto, i risultati riguardanti i trasferimenti e la timeline verranno presentati nella sezione che segue.

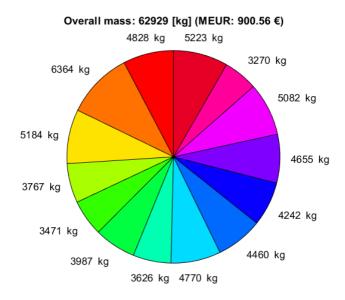


Tabella 6.3 Masse iniziali e costo totale della soluzione preliminare

A seguito dell'ottimizzazione temporale si ottengono determinate masse iniziali e un costo in termini monetari pari a 900.56 MEUR. Con l'ulteriore ottimizzazione sulla lista dei detriti si noterà come sia le masse iniziali sia il costo complessivo verranno diminuiti.

6.2 Risultati finali

Lancio	# debris	Lista (soluzione definitiva)
1	12	34 5 82 39 104 115 23 96 17 58 24 97
2	14	118 122 51 119 88 21 85 26 80 114 79 37 56 6
3	13	40 46 83 8 71 86 48 27 3 105 12 69 1
4	10	68 7 109 19 123 25 45 121 76 38
5	8	53 22 65 31 101 67 20 70
6	9	15 10 66 91 94 107 29 78 95
7	8	93 112 13 57 43 72 110 44
8	13	52 99 60 92 103 55 100 63 36 2 90 41 113
9	8	77 9 28 117 108 62 18 81
10	4	120 33 32 89
11	7	98 116 111 87 74 35 11
12	5	61 54 4 14 84
13	6	16 50 73 64 59 42
14	6	49 106 47 75 102 30
		a 6.4 Lista dobnis na coolti nello 14 sotto missioni

Tabella 6.4 Lista debris raccolti nelle 14 sotto-missioni

In dettaglio i costi in Δv dei trasferimenti orbitali tra un debris e l'altro e la durata del trasferimento, tenendo in considerazione che il tempo necessario per il rendez-vous, in questo caso, viene tenuto fisso a 5 giorni.

Lancio	Δv [m/s] (soluzione defnitiva)		
1	161.8 289.3 539.9 53.8 84.9 337.7 207.6 128.7 242.7 266.6 231.5		
2	63.9 127.7 81.5 273.7 210.5 232.4 395.5 312 359.8 112.2 125.9 322.5 233.2		
3	228.3 149.8 181.5 395.4 122.9 118.8 271.5 164.5 118.6 80.3 370.9 268.9		
4	124.3 96.4 239 161.1 424.5 230 106.4 203.9 165.3		
5	279.4 192 208.4 200.2 145.9 154.3 382.9		
6	85.7 138.1 285.5 131.1 328.5 258.1 332.6 241.4		
7	152.3 298.4 241.2 142.7 198.1 341 344.5		
8	163 28.8 213.6 128.8 286.6 139.6 138.6 253 299.2 119.4 92.9 163.8		
9	1154.3 224.9 57.8 172.8 200 432.7 155		
10	213.8 1496.5 682.2		
11	221.4 746.5 463.8 256.5 241.9 302.4		
12	474.4 554.9 1642.2 221.3		
13	236.6 508.6 135.2 369.7 180.8		
14	233.7 428.5 332.3 1252.3 795.1		

Tabella 6.5 Costi dei trasferimenti orbitali

Come si può notare, la maggior parte dei Δv ha un valore piuttosto contenuto: questo è dovuto al fatto che nella ricerca della sequenza preliminare è stata impostata una soglia di 370 m/s. Pertanto, sono stati scelti solamente detriti tra i quali il trasferimento orbitale fosse in termini di Δv sotto tale soglia. Sono presenti alcuni trasferimenti, invece, piuttosto costosi, finanche a circa 1600 m/s: una volta raccolti tutti i detriti più convenienti, per quelli rimasti si è dovuto trovare una combinazione in determinati istanti temporali perché il trasferimento costasse il meno possibile, ma in ogni caso, fuori dalla sequenza preliminare, erano rimasti i detriti più scomodi da prendere e che hanno penalizzato il processo di ottimizzazione globale.

Lancio	Durata del trasferimento [giorni]		
1	27.6 30 24.47 30 5 5 30 5 30 28.36 19.44		
2	5 30 23.05 30 13.16 25.74 30 30 30 7.63 5 30 5		
3	12.39 5 5 30 27.59 30 30 30 12.19 5 5.34 22.71		
4	5 10.04 30 30 30 29.27 30 5 16.4		
5	30 5 30 29.99 5 29.96 5		
6	29.24 6.67 30 30 22.66 30 30 30		
7	5 26.31 30 14.15 5 25.64 24.05		
8	10.15 30 7.37 9.13 20.07 5 30 16.94 30 5 21.63 5		
9	30 30 30 5 19.65 30 30		
10	30 30 30		
11	30 30 30 5 24.38 30		
12	30 30 30 30		
13	25.34 30 30 30 5		
14	30 30 30 30 30		

Tabella 6.6 Durata dei trasferimenti orbitali

Le durate dei trasferimenti orbitali variano da un minimo di 5 giorni, ossia semplicemente il tempo necessario per il rendez-vous, ad un massimo di 30 giorni, che è il limite fissato dalla competizione. Alcuni trasferimenti risultano molto convenienti, sia in termini di tempo che di costo in Δv .

La missione globale può partire dal giorno MJD2000 dell'inizio della competizione, cioè <u>23467 MJD2000</u>, e deve concludersi in un arco temporale di 8 anni, che corrisponde al <u>26419 MJD2000</u>.

Si ricorda che nel processo di ottimizzazione temporale, descritto nella sezione 5.2.3, il ritardo massimo concesso per l'inizio della missione è stato di 10 giorni dall'inizio della competizione. In tutta risposta, l'ottimizzazione ha fornito un risultato in cui la missione inizia subito, quindi il 23467 MJD2000, e si conclude nel 26374.70 MJD2000, con 44.4 giorni di anticipo dalla scadenza. Ogni sotto-missione dura tra i 100 e i 250 giorni circa.

Lancio	# debris	Inizio [MJD2000]	Fine [MJD2000]
1	12	23467	23701.88
2	14	23731.88	23996.46
3	13	24026.46	24241.68
4	10	24271.68	24457.38
5	8	24487.38	24622.33
6	9	24652.33	24860.89
7	8	24899.14	25029.29
8	13	25059.29	25249.56
9	8	25301.27	25475.91
10	4	25505.92	25595.92
11	7	25630.06	25779.44
12	5	25848.97	25968.97
13	6	26045.65	26165.99
14	6	26224.70	26374.70

Tabella 6.7 Giorni [MJD2000] di inizio e fine delle sotto-missioni

Si offre anche una rappresentazione grafica delle informazioni temporali:

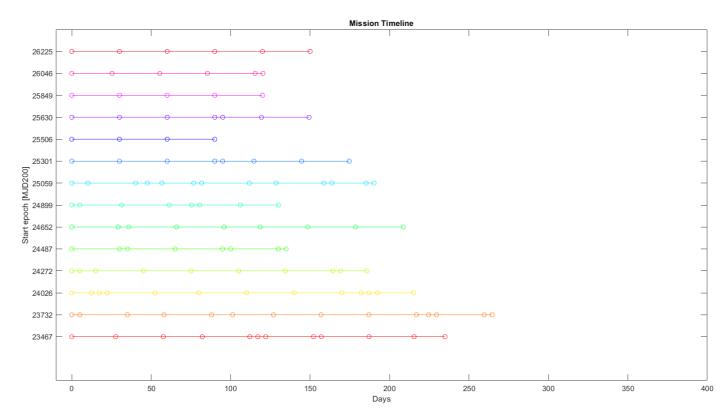


Figura 6.1 Timeline della missione

Altro punto importante della soluzione è il rispetto dei limiti sulla massa. Si ricorda che la massa a vuoto del satellite chaser è fissata dalla competizione a 2000 kg e che ogni kit di de – orbit ha una massa di 30 kg, che va dunque moltiplicata per il numero di detriti che il chaser deve raccogliere nella sotto-missione. La massa di propellente iniziale può essere al massimo 5000 kg.

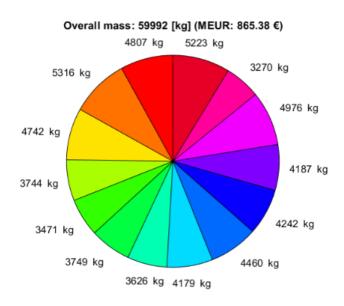
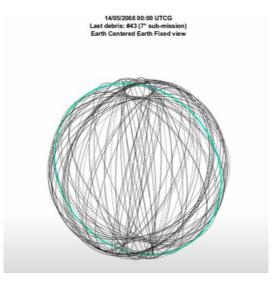


Figura 6.2 Masse iniziali delle sotto-missioni

A seguito dell'ottimizzazione combinatoria e dopo aver applicato *fmincon* alla nuova sequenza, il costo totale della missione risulta essere di 865.38 MEUR.

Per osservare una rappresentazione più concreta degli effetti benefici che si otterrebbero dalla rimozione sequenziale dei detriti in ambiente LEO viene presentata un'animazione ¹⁰ del processo.



https://www.youtube.com/watch?v=p4ed-tIwxCU (visuale co-rotante con la Terra) e https://www.youtube.com/watch?v=6s-11Pn0peg (visuale non rotante)

_

6.3 Confronto con la soluzione del JPL

Dovendo fare riferimento ai risultati del JPL, innanzitutto sono stati ricalcolati i Δv sostituendo il modello di stima del JPL con il modello Shen, che è quello utilizzato in questo lavoro di tesi e che è descritto nel capitolo 2. Dopodiché sono stati eseguiti diversi test per vedere quanto la sequenza ottenuta fosse simile alla sequenza ottenuta dal JPL.

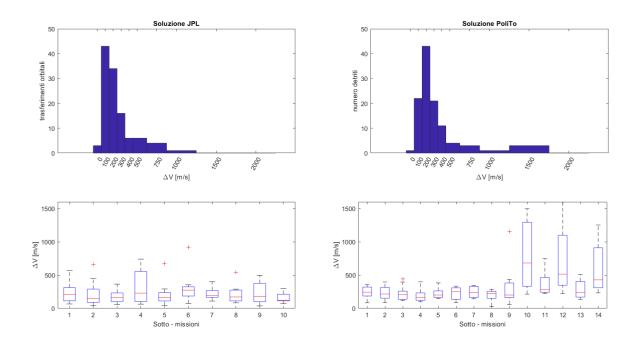


Figura 6.3 Confronto sui costi dei trasferimenti

Quello che si può notare da questi grafici è che le prime sotto-missioni della soluzione ottenuta con questo lavoro risultano essere altamente ottimizzate, a scapito poi delle ultime sotto-missioni in cui si raggiungono picchi di Δv superiori al migliaio di m/s. Come si può vedere dai boxplot, infatti, nella soluzione del JPL i primi box sono mediamente più grandi rispetto a quelli della soluzione PoliTO, ma in generale sono più omogenei tra loro, poiché il JPL ha saputo incastrare i detriti semplici da raggiungere con i più ostili.

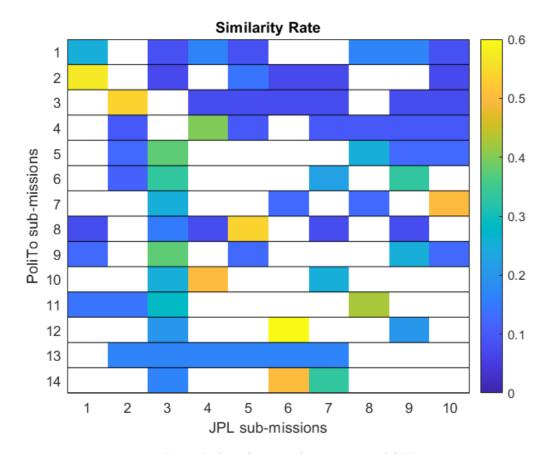


Figura 6.4 Somiglianze con le sotto-missioni del JPL

Il Similarity Rate rappresenta invece la frazione di detriti in comune tra una sotto-missione della soluzione PoliTO con una sotto-missione del JPL. Nella tabella possono esserci valori nulli, per cui rimane lo spazio bianco. Immaginando la griglia del Similarity Rate come una matrice, si nota una correlazione più forte lungo la diagonale principale. Si è deciso di verificare solamente l'appartenenza di detriti in entrambe le soluzioni, anziché ricercare una correlazione sequenziale, dal momento che il problema è fortemente tempo dipendente e i Δv , così come il grafo che li rappresenta, evolvono molto velocemente.

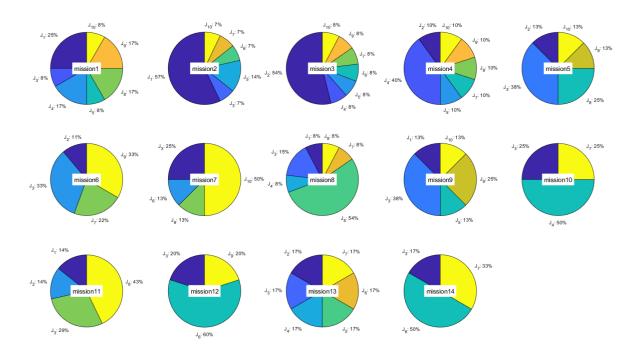


Figura 6.5 Somiglianza tra le sotto-missioni

I grafici a torta rappresentano il numero di detriti che in ognuna delle sotto-missioni sono presenti anche in una sotto-missione del JPL.

Per testare la bontà dell'algoritmo sono state sfruttate le informazioni fornite dal JPL per un ulteriore confronto tra le soluzioni. Nei paragrafi sottostanti vengono descritti i test che sono stati condotti, riguardanti la rimozione di detriti o intere sotto-missioni del JPL, per verificare se il codice sia in grado di ricostruire la soluzione che ha vinto la competizione GTOC9.

Test 1: rimozione del primo detrito da ogni sotto-missione

Il primo test per il confronto tra le due sequenze di rimozione dei detriti prevede la rimozione del primo detrito di ogni sotto-missione della soluzione presentata dal JPL, per un totale di 10 detriti. È risultato che in tre sotto-missioni (precisamente la seconda, la quarta e la sesta) la ricostruzione è avvenuta con successo, reinserendo i rispettivi detriti nelle sotto-missioni a cui appartenevano. Per quanto riguarda le altre sotto-missioni, che non sono state ricostruite, è doveroso dire che l'algoritmo ha identificato spesso il detrito che era stato rimosso, poiché il codice predilige il passaggio orbitale con minore costo, dimostrando quindi l'elevato grado di ottimizzazione globale della soluzione JPL nonché la capacità dell'algoritmo PoliTO di competere in efficienza. Il motivo per cui non è stato

possibile inserire i detriti in questione è che il calcolo della massa della sotto-missione superava la soglia per permetterne la reintroduzione. Se per la stima dei Δv fossero stati usati gli stessi criteri del JPL questo problema molto probabilmente non si sarebbe verificato.

Guardando globalmente i risultati, l'algoritmo ha cercato degli slot temporali in cui inserire sotto-missioni più piccole con i detriti rimasti – 2 sotto-missioni con 2 detriti. Solo 3 detriti non sono stati inseriti affatto.

Le cause di questo comportamento si possono riassumere nei seguenti punti:

- tra il modello di stima dei Δv adottato dal JPL e quello adottato in questo lavoro di tesi esistono divergenze che impattano sensibilmente sulla decisione di quali detriti prendere. Infatti, i detriti che non sono stati ricongiunti spesso sono caratterizzati da un Δv che è quasi il doppio rispetto a quello stimato dal JPL.
- la funzione che nel codice si occupa di inserire i detriti mancanti, se possibile, dà per scontato che i detriti avanzati siano solitamente quelli con un alto costo di trasferimento orbitale. Poiché con il modello di stima qui adottato, aumentando il tempo di trasferimento se ne riduce il costo, la funzione è stata programmata con una durata di trasferimento fissa e alta (circa 30 giorni comprensivi di rendezvous). Nella soluzione JPL, invece, le sotto-missioni sono già globalmente ottimizzate ed emerge quanto sia fondamentale integrare la variabile tempo in un problema in cui i costi di Δν sono fortemente contingentati.
- i detriti che sono stati reinseriti dall'algoritmo, non a caso, sono detriti connotati nella soluzione del JPL da un trasferimento di circa 25 giorni più 5 giorni di rendezvous. Non stupisce, quindi, che il codice abbia identificato con più facilità questa categoria piuttosto che i detriti con durata del trasferimento differente.

Rimane comunque alla base del ragionamento il fatto che la porzione troncata di ogni sotto-missione stabilisca a priori se un detrito sia inseribile o meno a causa dei requisiti di massa. Alcune sotto-missioni risultano già prossime al valore limite di massa e l'unico modo per ridurre il carico di propellente sarebbe quello di ridurre i Δv , alterandone la cronologia, ma non essendo ancora iniziata l'ottimizzazione temporale non c'è modo di rilassare o contrarre i Δv delle porzioni inalterate. Il motivo per cui l'ottimizzazione temporale avviene successivamente alla fase di completamento è che tendenzialmente le pause tra le sotto-missioni si riducono, dando più spazio alle durate dei trasferimenti e togliendo la possibilità ad alcuni detriti di essere introdotti per mancanza di slot temporali disponibili.

Test 2: rimozione di più detriti da più sotto-missioni

In questo caso sono stati rimossi alcuni detriti da due sotto-missioni distinte. Nell'esempio proposto sono stati tolti i primi 5 detriti dalla seconda sotto-missione e i primi 4 detriti dalla sesta sotto-missione. Le sotto-missioni non sono state scelte casualmente: la numero 2 e la numero 6 sono state quelle che nel primo test avevano risposto bene.

Dal momento che l'algoritmo di completamento ipotizza una durata fissa del trasferimento orbitale (30 giorni), per la riuscita di questo test è stata necessaria la modifica manuale della durata. Questo ha impedito uno shift temporale che alterasse tutta la missione del JPL, permettendo ai blocchi che dovevano essere reintrodotti di rimanere negli stessi intervalli temporali a cui appartenevano.

Ne è risultato che la seconda sotto-missione è stata ricostruita ad eccezione del primo detrito e scambiando di posizione altri due detriti. La sesta, invece, è stata ricostruita completamente ma con un ordine leggermente diverso.

Le cause di questo comportamento sono analoghe a quelle descritte per il primo test: la divergenza tra i due modelli di stima dei Δv e il fatto che l'algoritmo riesca a ricostruire meglio le sequenze nel caso in cui la durata dei trasferimenti sia di circa 30 giorni.

Mission JPL	Lista	Ricostruzione del codice
2	19 115 41 26 45 82 47 85 7 2 11 77	115 41 45 26 82 47 85 7 2 11 77
6	101 48 53 5 12 39 58 13 60 74	53 48 101 5 12 39 58 13 60 74

Tabella 6.8 Risultati test 2 (notazione zero-based)

Test 3: rimozione di un'intera sotto-missione

Il terzo test prevede la rimozione di un'intera sotto-missione dalla soluzione del JPL. In questo caso è stata rimossa la numero 7 costituita da 10 detriti.

Il codice non la ricostruisce completamente (7 detriti in un unico blocco) e non ha rispettato lo stesso ordine con cui era definita all'inizio. Inoltre, 2 detriti tra i 3 esclusi sono stati riposizionati in slot temporali differenti, mentre l'ultimo rimasto non è stato inserito affatto. Alle cause di questo comportamento già elencate prima se ne aggiunge un'altra che riguarda il fatto che il codice non è capace di innestare sotto-missioni con più di 7/8 detriti alla volta a causa di limitazioni di calcolo combinatorio, memoria e tempo.

Mission JPL	Lista	Ricostruzione del codice	Appartenenza alla sotto- missione
7	49 9 70 93 105 46 88 118 18 91	9 70 18 105 46 88 118	7
		91	1
		49	5

Tabella 6.9 Risultati test 3 (notazione zero-based)

Test 4: rimozione di più sotto-missioni

In questo ultimo test sono state rimosse due sotto-missioni consecutive, la numero 7 e la numero 8, per un totale di 19 detriti mancanti. Il codice non è riuscito a raccogliere tutti e 123, ne rimane fuori uno. La sotto-missione 8, però, è stata ricostruita completamente e con lo stesso ordine, prima innestando una sotto-missione con 7 detriti e poi ampliandola con altri 2. La sotto-missione 7, invece, ha recuperato 9 detriti su 10, con un ordine leggermente permutato, sempre prima creando una sotto-missione da 7 detriti e poi aggiungendone 2. Le cause sono sempre le stesse: la divergenza nella stima dei Δv , le durate dei trasferimenti fisse e alte e il fatto che il codice non sia in grado di inserire in una nuova sotto-missione più di 7/8 detriti per volta. Questo è un difetto nel processo che potrebbe peggiorare se il numero di detriti non presenti nel filone aumentasse ed è il motivo per cui non sono state rimosse 3 o più sotto-missioni. Non si esclude che, selezionando altre sotto-missioni da rimuovere, si otterrebbe più frequentemente una ricostruzione imperfetta dal punto di vista dell'ordine di rimozione dei detriti.

Mission JPL	Lista	Ricostruzione del codice
7	49 9 70 93 105 46 88 118 18 91	49 9 93 70 18 105 46 88 118
8	86 34 100 30 92 6 110 96 81	86 34 100 30 92 6 110 96 81

Tabella 6.10 Risultati test 4 (notazione zero-based)

Quello che risulta dai 4 test condotti è che il modello Shen di stima dei Δv a volte può essere fuorviante. La soluzione PoliTO non avrebbe mai potuto essere identica a quella proposta dal JPL, in quanto le missioni si svolgono in tempi differenti e sono soluzioni ad

un problema in cui la variabile tempo è fondamentale.

Forzando il codice a ricreare la soluzione del JPL, si nota come una delle cause principali che impediva la ricostruzione della sequenza fosse proprio la divergenza nei modelli di stima dei Δv , che di fatto impediva che un detrito venisse inserito in una sotto-missione a causa della violazione del limite sulla massa.

Clustering spettrale

Un ulteriore confronto è stato svolto con il metodo del clustering spettrale, un algoritmo basato sui grafi per il partizionamento di dati e osservazioni in un numero k di cluster. La funzione *spectralcluster* di MATLAB esegue il clustering su una matrice di dati. In questo caso la matrice contiene i Δv minimi in ogni step temporale. La clusterizzazione avviene per ogni frame temporale ed il vettore temporale, con un passo di 5 giorni, arriva fino a 2920 giorni (8 anni). Si può ottenere, quindi, una partizione del grafo che evolve nel tempo. *spectralcluster* restituisce tre diversi output: gli indici di appartenenza ai cluster, una matrice contenente k autovettori della matrice laplaciana e un vettore di autovalori corrispondenti agli autovettori.

Per il confronto tra le soluzioni PoliTO e JPL, il parametro di clusterizzazione k è stato considerato pari a 25, un buon numero di clusters per riuscire ad indentificare i nodi realmente vicini su una popolazione di 123 elementi mutuamente e periodicamente connessi.

I nodi del grafo non rappresentano la reale disposizione dei detriti nello spazio con l'evolversi del tempo, ma sono disposti in base ai Δv : più i nodi sono vicini, minore sarà il costo per il trasferimento. Rappresentando visivamente il grafo con un'istantanea per ogni frame temporale, è stato utile introdurre un piano bidimensionale astratto su cui posizionare i nodi in modo tale che tutte le proporzioni tra i Δv fossero soddisfatte. Questo tipo di rappresentazione simbolica è molto simile alle rappresentazioni dello spazio 3D delle molecole chimiche in cui però i legami chimici sono stati sostituiti con i pesi dei Δv .

Nella parte inferiore di questa istantanea è stata anche inserita una porzione della timeline, sia della soluzione JPL sia della soluzione PoliTO, con l'elenco dei detriti che mano a mano vengono raccolti.

La figura 6.6 mostra quattro fotogrammi di un video¹¹ che rappresenta l'evoluzione del grafo.

¹¹ https://www.youtube.com/watch?v=sIxnJdQVDgM

Quello che si può notare dall'analisi è un confronto di tipo qualitativo. In ogni istantanea sono presenti delle frecce in entrambi i clusters. Queste frecce indicano il prossimo detrito che verrà raccolto a partire dal detrito su cui la sotto-missione attualmente si concentra. Entrambe le frecce cadono sempre all'interno del cluster di appartenenza, sia per quanto riguarda la missione PoliTO che per la missione JPL, e questo è sintomo di un buon grado di ottimizzazione, soprattutto dal momento che un numero di clusters più alto renderebbe ancora minore la probabilità di un nodo di essere classificato nel cluster in cui si trova il nodo precedente.

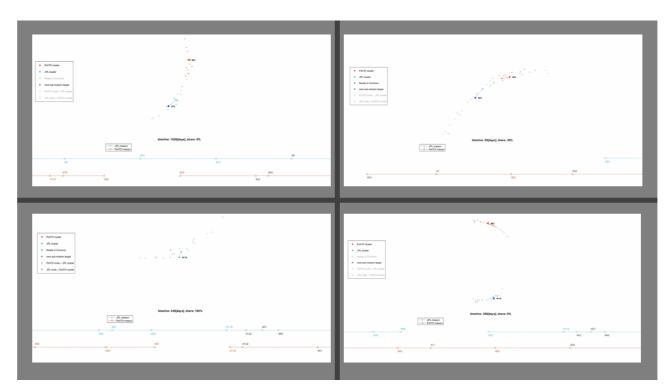


Figura 6.6 Clustering con diversi valori di share

Inoltre, si nota chiaramente che in alcuni istanti temporali i cluster (rossi e blu) che rappresentano le due soluzioni si avvicinano e in altri si allontanano. Questo fatto non può essere controllato poiché è figlio di un processo decisionale differente da quello adottato dal JPL.

I due subset si allontanano poiché la sequenza temporale influisce fortemente sulla regione del grafo verso cui ci si sposta. I punti in cui si avvicinano fanno invece capire come anche la soluzione PoliTO sia stata in grado di individuare delle regioni ad alta convenienza, proprio come il JPL. In un'analisi di questo tipo è utile monitorare il numero di nodi facilmente raggiungibili sia dal PoliTO sia dal JPL. Quando le due soluzioni coincidono naturalmente anche i cluster coincidono e la percentuale di share raggiunge il 100%. È

interessante notare, però, che anche nel momento in cui non ci sono nodi in comune (share: 0%) i due subset a volte sono comunque molto vicini. Riducendo il numero k di cluster, probabilmente, la percentuale dei nodi in comune sarebbe stata generalmente più alta.

Queste considerazioni si possono anche notare nelle figure che seguono, che mostrano la percentuale di nodi in comune tra i due subset e le somiglianze tra le sequenze di detriti. La somiglianza è vera proprio nei punti in cui i due subset presentano una percentuale di nodi in comune non nulla.

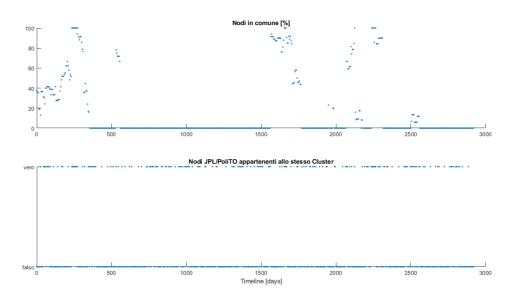


Figura 6.7 Nodi in comune tra il cluster PoliTO e il cluster JPL

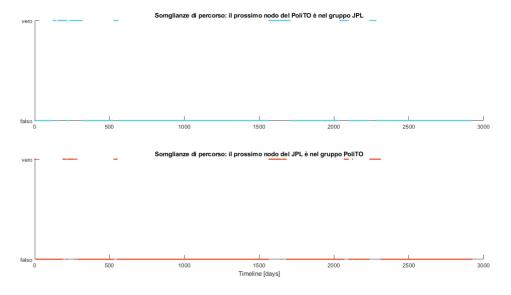


Figura 6.8 Somiglianze di percorso

Focalizzandosi, invece, sui periodi di inattività tra le sotto-missioni anziché sui

trasferimenti, risulta che il JPL fa meno pause: la partizione della soluzione PoliTO è, infatti, più grande e questo è anche uno dei motivi del costo in termini monetari più alto rispetto a quello del JPL. Tuttavia, i momenti in cui entrambe le soluzioni prevedono la raccolta di detriti e quelli di pausa sono correlati: entrambe hanno saputo individuare gli istanti che non sembravano essere convenienti per la raccolta. Lo si nota dal fatto che in specifici intervalli temporali l'interruzione delle attività tra JPL e PoliTO è quasi coincidente e ammette un'escursione non superiore ai 60 giorni.

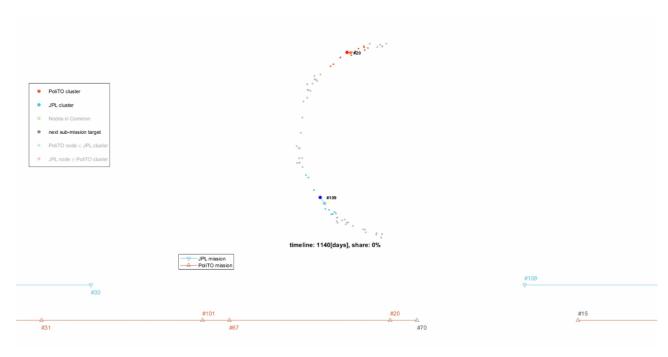


Figura 6.9 Esempio di un momento in cui entrambe le soluzioni prevedono una pausa

7 Conclusioni

Dato il rapido ed importante aumento negli ultimi dieci anni dei lanci di piccoli satelliti in orbita bassa terrestre, si rende necessaria la rimozione da questa regione di spazio di tutti quegli elementi classificati come detriti spaziali, al fine di evitare collisioni tra di essi e dei detriti con i satelliti operativi, rendendo più sicure le future missioni, soprattutto quelle con equipaggio umano.

Questo lavoro di tesi, basandosi su una competizione indetta dall'ESA, la *Global Trajectory Optimization Competition 9* (GTOC9), ha provato a trovare un metodo per la rimozione di un set piuttosto consistente di detriti in orbita LEO. L'approccio che è stato utilizzato si basa sulla teoria dei grafi e si propone come metodo alternativo ai più sfruttati algoritmi evolutivi utilizzati generalmente nei problemi di ottimizzazione.

La principale difficoltà nella ricerca della sequenza di rimozione dei detriti risiede nella forte dipendenza dal tempo del problema: i parametri che definiscono la posizione e l'orientamento dei detriti nello spazio cambiano nel tempo a causa delle perturbazioni orbitali. La variabile temporale è, dunque, fondamentale e insieme all'elevato numero di detriti rende la risoluzione del problema piuttosto complicata.

La ricerca nel grafo sul piano temporale ha prodotto comunque risultati soddisfacenti, permettendo la rimozione di tutti i 123 detriti previsti dalla competizione, con un costo sia in termini di Δv che in termini monetari non così lontani dalla soluzione del JPL, vincitore della competizione. Proprio con la soluzione vincitrice sono stati confrontati i risultati ottenuti da questo lavoro, evidenziando non solo come naturalmente la soluzione del JPL sia globalmente ottimizzata, ma anche come i diversi modelli di stima dei Δv abbiano impattato sulla determinazione della sequenza dei detriti da rimuovere.

Non esiste una soluzione nettamente migliore di altre data la complessità del problema, ma il motivo per cui si è scelto di adottare un approccio basato sui grafi e non direttamente un algoritmo genetico è che, nonostante quest'ultimo permetta di ottimizzare parametri di diversa natura contemporaneamente, si basa su una funzione di fitness complicata da formulare e che richiederebbe un elevato costo computazionale. L'approccio adottato nel presente lavoro permette di arrivare ad una soluzione che viene costruita e ottimizzata per fasi e anche di risparmiare tempo e memoria.

Il codice sviluppato presenta delle limitazioni che potranno essere risolte in futuro. Tra gli ulteriori sviluppi possibili si possono includere:

 \triangleright cambiare la soglia di Δv che consente la memorizzazione dei passaggi orbitali nel grafo, consci delle conseguenze che un eventuale aumento o riduzione

comporterebbe;

- ➤ provare altri metodi di ampliamento delle sequenze preliminari, per esempio inserire i detriti anche alla fine delle sotto-missioni e non solo all'inizio come è stato fatto nel presente lavoro;
- \triangleright capire il motivo della divergenza tra i modelli di stima dei Δv , quello Shen e quello adottato dal JPL;
- ➤ prevedere una formulazione del problema più generalizzata, in modo da manipolare contemporaneamente la partizione e lo storico dei trasferimenti orbitali a partire da una lista prestabilita di detriti. Nel lavoro svolto finora, infatti, la partizione non è mai stata messa in discussione in quanto generata *step by step*.
- > ottenere direttamente dalla funzione obiettivo anche la sequenza ottimale dei detriti, eliminando così l'ottimizzazione per fasi, dal momento che tutte le variabili fondamentali di missione concorrerebbero simultaneamente al processo di ottimizzazione.

8 Bibliografia

- [1] Shen H.X, Casalino L., Simple ΔV Approximation for Optimization of Debris-to-Debris Transfers.
- [2] Shen H.X., Zhang T.J., Casalino L., Pastrone D., Optimization of Active Debris Removal Missions with Multiple Targets.
- [3] Kessler D.J., Cour-Palais B.G., Collision Frequency of Artificial Satellites: The Creation of a Debris Belt.
- [4] Kessler D.J., Johnson N.L., Liou J.C., Matney M., The Kessler Syndrome: implications to future space operations.
- [5] Izzo D., Problem description for the 9th Global Trajectory Optimisation Competition, The Kessler Run.
- [6] Petropoulos A., Grebow D., Jones D.R., Lantoine G., Nicholas A., Roa J., Senent J., Stuart J., Arora N., Pavlak T., Lam T., McElrath T., Roncoli R., Garza D., Bradley N., Landau D., Tarzi Z., Laipert F., Bonfiglio E., Wallace M., Sims J.A., GTOC9: Methods and Results from the Jet Propulsion Laboratory Team.
- [7] Bate R.R., Mueller D.D., White J.E., Fundamentals of Astrodynamics, New York, Dover Publications, Inc., 1971.
- [8] Casolo C., Corso di Teoria dei Grafi, 2009.
- [9] Bonaccorsi S., Teoria dei grafi e applicazioni.
- [10] fmincon Interior Point Algorithm.

Available at: https://www.mathworks.com/help/optim/ug/constrained-nonlinear-optimization-algorithms.html#brnpd5f

[11] Partition Data Using Spectral Clustering.

Available at: https://www.mathworks.com/help/stats/partition-data-using-spectral-clustering.html