



SAPIENZA  
UNIVERSITÀ DI ROMA

## Satellite Edge Computing - Modellazione e Simulazione dell' Handover

Facoltà di Ingegneria dell'Informazione, Informatica e Statistica  
Corso di Laurea Triennale in Informatica

**Vincenzo Salvatore**

Matricola 1918339

Responsabile

Prof. Emiliano Casalicchio

Anno Accademico 2024/2025

---

**Satellite Edge Computing - Modellazione e Simulazione dell' Handover**  
Sapienza Università di Roma

© 2025 Vincenzo Salvatore. Tutti i diritti riservati

Questa tesi è stata composta con L<sup>A</sup>T<sub>E</sub>X e la classe Sapthesis.

Email dell'autore: salvatore.1918339@studenti.uniroma1.it

*Dedicato ai miei Nonni  
e ai miei Genitori*

# Indice

|   |           |
|---|-----------|
| <b>Acronomi</b>   | <b>vi</b> |
| <b>1 Introduzione</b>   | <b>1</b>  |
| 1.1 Contributo . . . . .  | 1         |
| 1.2 Struttura della tesi . . . . .  | 2         |
| <b>2 Stato dell'arte</b>  | <b>4</b>  |
| 2.1 Edge Computing: Una risposta ai limiti dell'elaborazione centralizzata  | 4         |
| 2.2 Tecnologie satellitari . . . . .  | 5         |
| 2.2.1 Tipi di costellazioni satellitari . . . . .   | 6         |
| 2.3 Simulazione Orbitale e Modelli di perturbazione . . . . .   | 6         |
| 2.4 Sfide nello scambio dati tra satelliti in movimento . . . . .   | 7         |
| 2.4.1 ISLs . . . . .  | 7         |
| 2.4.2 LISLs . . . . .   | 7         |
| <b>3 Studio del Problema : Copertura Satellitare, Comunicazione e Simulazione</b>   | <b>10</b> |
| 3.1 Descrizione delle dinamiche di movimento nei satelliti . . . . .  | 10        |
| 3.1.1 Classificazione e Caratteristiche dei Satelliti LEO, MEO e GEO:   | 10        |
| 3.1.2 Modelli di distribuzione dei satelliti in <i>Low Earth Orbit</i> (LEO)  | 12        |
| 3.1.3 Copertura dell'utente a Terra: Numero di satelliti con buona qualità del segnale e senza disturbi dovuti all'angolo di elevazione | 15        |
| 3.1.4 Strategie di Selezione del Satellite per la Connessione . . . . .   | 19        |
| 3.2 Implicazione del movimento dei satelliti per la copertura . . . . .   | 23        |
| 3.2.1 Densità Satellitare e Qualità del Segnale . . . . .   | 23        |
| 3.2.2 Impatto dell'Angolo di Elevazione sulla Qualità del Link . . .  | 24        |
| 3.2.3 Difficoltà nel prevedere i movimenti dei satelliti . . . . .  | 26        |
| 3.3 Simulazione satellitare - Satellite Edge Computing (SEC) . . . . .  | 27        |
| 3.3.1 Gestione del carico computazionale . . . . .  | 28        |
| 3.3.2 Latenza e Scheduling Decentralizzato dei Task . . . . .   | 28        |

|   |           |
|---|-----------|
| <b>4 Metodologia</b>  | <b>31</b> |
| 4.1 Oggetto della ricerca . . . . .                                 | 31        |
| 4.2 Dati e Strumenti di base . . . . .                              | 31        |
| 4.2.1 Two-Line Elements (TLE) e dati orbitali . . . . .             | 31        |
| 4.2.2 Observer e sistema di riferimento . . . . .                   | 35        |
| 4.3 Modellazione e simulazione della rete satellitare . . . . .     | 36        |
| 4.3.1 Identificazione dei satelliti visibili . . . . .              | 36        |
| 4.3.2 Definizione dell'insieme $S_i$ . . . . .                      | 36        |
| 4.3.3 Relazioni tra Satelliti e il concetto di Vicinato . . . . .   | 37        |
| 4.3.4 Topologia Statica . . . . .                                   | 37        |
| 4.4 Topologia Dinamica . . . . .                                    | 40        |
| 4.4.1 Neighborhood . . . . .  | 41        |
| 4.4.2 AP-SAT . . . . .  | 43        |
| 4.4.3 Buffer Zone . . . . .   | 46        |
| 4.5 Dalla Teoria alla Computazione: Elaborazione dei Dati . . . . . | 50        |
| 4.5.1 Pre-elaborazione dei Dati . . . . .                           | 50        |
| 4.5.2 Intersezioni delle Configurazioni . . . . .                   | 53        |
| <b>5 Conclusioni</b>  | <b>56</b> |
| 5.1 Innovatività della ricerca . . . . .                            | 56        |
| 5.2 Sviluppi futuri . . . . .                                       | 57        |
| <b>Glossario</b>  | <b>58</b> |
| <b>Bibliografia</b>   | <b>59</b> |

# Elenco delle figure

|      |  |    |
|------|--|----|
| 2.1  | Dispositivi e nodi <i>edge</i> in relazione al Cloud [18] . . . . .  | 4  |
| 3.1  | <i>Walker Star Pattern</i> [14] . . . . .  | 13 |
| 3.2  | Delta Pattern[17] . . . . .  | 14 |
| 3.3  | Rappresentazione del Piano dell’orizzonte rispetto l’ utente [9] . . . . .   | 16 |
| 3.4  | Analisi Velocità, Raggio, Periodo orbitale e numero passaggi giornalieri dei satelliti in base alla loro altitudine. [9] . . . . . | 18 |
| 3.5  | Cupola di ascolto dei satelliti. [8] . . . . .   | 19 |
| 3.6  | Rappresentazione grafica di come al variare di $\delta$ cambi il percorso del satellite nella cupola. [8] . . . . .                | 19 |
| 3.7  | Tabella rappresentativa della percentuale di copertura e della quantità di satelliti richiesta [12]. . . . .                       | 23 |
| 3.8  | Capacità totale media di <i>backhaul</i> di ciascuna utente rispetto alla densità delle utenti [12]. . . . .                       | 24 |
| 3.9  | Numero di satelliti LEO dispiegati rispetto al grado di copertura [12].  | 24 |
| 3.10 | Tabella rappresentativa della percentuale di Copertura e della quantità di satelliti richiesta [12]. . . . .                       | 25 |
| 3.11 | Comparazione fra la massima altitudine del satellite in gradi rispetto alla sua distanza [20] . . . . .                            | 25 |
| 3.12 | Differenti piani orbitali [20]   | 25 |
| 3.13 | Illustrazione della struttura adottata nel simulatore presentato nell’articolo [7]. . . . .  | 27 |
| 4.1  | Rappresentazione della copertura globale del 15% dei satelliti StarLink  | 34 |
| 4.2  | Rappresentazione della copertura globale del 50% dei satelliti StarLink  | 34 |
| 4.3  | Rappresentazione della copertura globale del 100% dei satelliti StarLink   | 34 |
| 4.4  | Zona di copertura della cupola con un angolo di ascolto di 40° . . . . .   | 36 |
| 4.5  | Prospettiva laterale della topologia . . . . .   | 38 |
| 4.6  | Prospettiva dallo zenith della topologia . . . . .   | 38 |
| 4.7  | Rappresentazione dell’insieme di tutte le configurazioni . . . . .   | 40 |

|  |    |
|--|----|
| 4.8 Rappresentazione delle relazioni intra-satellitari . . . . .                                 | 41 |
| 4.9 Illustrazione della componente $v_{rel}$ di un satellite rispetto ad un Osservatore. . . . . | 44 |
| 4.10 Illustrazione del caso particolare centrato in Oulu, Finlandia. . . . .                     | 46 |
| 4.11 Rappresentazione della zona di Buffer . . . . .   | 49 |
| 4.12 Topologia completa Starlink istante 1 . . . . .   | 49 |
| 4.13 Topologia completa Starlink istante 2 . . . . .   | 49 |
| 4.14 Topologia completa Starlink istante 3 . . . . .   | 49 |

# Acronimi

**AOS** *Acquisition of the Satellite.* 15, 16, 18, 23

**EC** *Edge Computing.* 4

**GEO** *Geostationary Earth Orbit.* 6, 10, 11

**GPS** *Global Positioning System.* 12

**GS** *Ground Station.* 17, 18, 23, 24, 33, 41, 47

**IoT** *Internet of Things.* 4, 6

**ISLs** *Inter-Satellite Links.* 7

**LEO** *Low Earth Orbit.* iii, 5–7, 11, 12, 17, 23, 27, 38

**LISLs** *Laser Inter-Satellite Links.* 7, 8, 37

**LOS** *Loss of Signal.* 15, 16, 23, 53

**MCNs** *Mega-constellation networks.* 7

**MEO** *Medium Earth Orbit.* 10, 11

**OBP** *On-Board Processing.* 7

**QoS** *Quality of Service.* 23

**RAAN** *right ascension of ascending node.* 14, 15

**SDP4** Simplified Deep space Perturbations 4. 26

**SEC** *Satellite Edge Computing.* 4, 28

**SEN** *Satellite Edge Nodes.* 27, 28

**SGP4** *Simplified General Perturbation 4.* 4, 6, 26, 33, 35, 56

# Capitolo 1

## Introduzione

Negli ultimi anni, la crescente diffusione dell'Internet of Things (IoT) e l'aumento esponenziale del volume di dati generati dai dispositivi connessi hanno reso necessaria una trasformazione dell'infrastruttura di elaborazione dati. In questo contesto, il paradigma dell'Edge Computing si è affermato come una soluzione efficace per ridurre la latenza, migliorare la sicurezza e ottimizzare l'utilizzo della larghezza di banda, spostando l'elaborazione dei dati vicino alla sorgente di generazione.

Parallelamente, l'avanzamento delle tecnologie satellitari e la proliferazione delle costellazioni di satelliti in orbita bassa (LEO) hanno aperto nuove opportunità per l'integrazione di piattaforme computazionali nello spazio. Il Satellite Edge Computing (SEC) rappresenta una nuova frontiera in cui l'elaborazione decentralizzata viene estesa all'ambiente satellitare, consentendo la gestione locale dei dati e la riduzione della dipendenza dai data center terrestri.

### 1.1 Contributo

La ricerca si propone di esplorare la modellazione e simulazione di una rete di satelliti nel contesto del Satellite Edge Computing, con particolare attenzione al fenomeno dell'*handover*, ovvero il trasferimento dinamico della connessione tra satelliti durante il loro moto orbitale. Il lavoro di ricerca si basa sull'utilizzo di modelli orbitali, come il Simplified General Perturbation 4 (SGP4), e su dati orbitali aggiornati in formato Two-Line Elements (TLE) per simulare il comportamento dinamico della rete satellitare.

L'obiettivo principale è la definizione di un modello capace di rappresentare la topologia della rete, identificare i satelliti visibili in un dato istante e ottimizzare il processo di *handover*, garantendo una comunicazione continua ed efficiente tra

utenti terrestri e satelliti. Questo approccio si inserisce in un più ampio contesto di ricerca volto a migliorare la connettività globale e supportare applicazioni emergenti nell'ambito dell'IoT e delle comunicazioni distribuite.

## 1.2 Struttura della tesi

La struttura dei capitoli si articola nel seguente modo:

- Il **Capitolo 2** fornisce una panoramica dello stato dell'arte, analizzando i concetti chiave di Edge Computing, tecnologie satellitari e i modelli orbitali utilizzati per prevedere il movimento dei satelliti.
- Il **Capitolo 3** descrive lo studio del problema, illustrando le dinamiche di movimento dei satelliti, la copertura dell'utente a terra e le sfide nella comunicazione tra satelliti in movimento.
- Il **Capitolo 4** presenta la metodologia di simulazione, includendo la raccolta di dati orbitali, la costruzione della topologia e le tecniche di elaborazione dei dati.
- Il **Capitolo 5** discute i risultati ottenuti e l'innovatività della ricerca, evidenziando i vantaggi e i limiti del modello proposto.
- Infine, le conclusioni riassumono il lavoro svolto, delineando le potenzialità future e gli sviluppi della ricerca.



## Capitolo 2

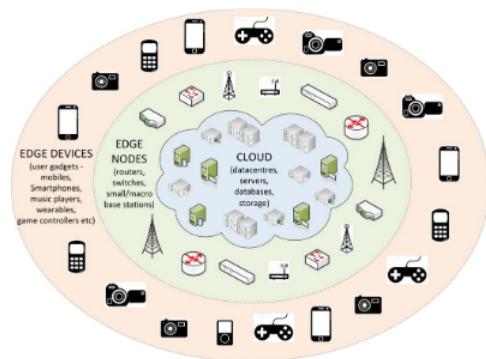
# Stato dell'arte

Il seguente capitolo è dedicato allo stato dell'arte, verranno analizzate le tecnologie necessarie per poter comprendere e contestualizzare *Satellite Edge Computing* (SEC); Saranno esaminati i concetti chiave come l'*Internet of Things* (IoT), l'*Edge Computing* (EC) e i modelli orbitali dei satelliti, tra cui i sistemi kepleriani e modelli avanzati come *Simplified General Perturbation 4* (SGP4). Questi rappresentano le fondamenta necessarie per affrontare le sfide e le opportunità offerte dal SEC.

### 2.1 Edge Computing: Una risposta ai limiti dell'elaborazione centralizzata

L'**Edge Computing** è un paradigma emergente che mira a spostare l'elaborazione dei dati vicino al luogo in cui vengono generati, piuttosto che trasferirli a un *data center* remoto o al *cloud* per l'elaborazione. Con la crescita esponenziale delle tecnologie dell'IoT e l'aumento del numero di dispositivi connessi, l'approccio centralizzato ha mostrato limiti significativi nell'elaborazione di grandi quantità di dati nei server al centro della rete.

In primo luogo, la quantità di dati generati è aumentata drasticamente, creando un'enorme pressione sulle infrastrutture di rete. La trasmissione continua di grandi volumi di dati verso server centrali può portare a colli di bottiglia nella rete, riducendo l'efficienza complessiva e aumentando i costi di gestione. Inoltre, il trasferimento di dati su lunghe distanze richiede un utilizzo intensivo della



**Figura 2.1.** Dispositivi e nodi *edge* in relazione al Cloud [18]

larghezza di banda, una risorsa spesso limitata in molti contesti applicativi.

Per secondo, la latenza, ovvero il tempo necessario per trasmettere i dati al server, elaborare le informazioni e ricevere una risposta, può diventare un fattore critico per molte applicazioni moderne. Scenari come il monitoraggio di sistemi critici, di veicoli autonomi o i dispositivi medici richiedono risposte in tempo reale o quasi, rendendo il ritardo introdotto dalla trasmissione a lungo raggio inaccettabile. Un esempio, le applicazioni di guida visiva basate su videocamere richiedono tempi di risposta compresi tra i 25 e i 50 ms, mentre l'attuale infrastruttura cloud può presentare ritardi anche superiori ai 175 ms su lunghe distanze [18].

Non meno importante sono le possibili implicazioni di sicurezza e privacy. La trasmissione continua di dati sensibili verso server remoti aumenta il rischio di intercettazioni, attacchi o violazioni della privacy, adottando pertanto misure aggiuntive per proteggere le informazioni.

La soluzione a queste problematiche è l'*edge computing*, questo consente di ridurre il volume di dati trasmessi sulla rete, soddisfare i requisiti di bassa latenza, e, migliorare la sicurezza attraverso l'elaborazione locale dei dati.

Per quanto riguarda la sostenibilità energetica, la gestione dei data center è una delle principali sfide del *cloud computing*; Si prevede che, nel prossimo decennio, il consumo energetico di questi centri triplicherà rispetto a oggi [18], rendendo necessarie strategie più efficienti. L'*edge Computing* ridurre il carico sui data center, spostando compiti analitici meno complessi sui nodi *edge*. Questi, essendo più vicini alla fonte dei dati, eseguono tali compiti senza impatti significativi sul consumo energetico globale, contribuendo così a una maggiore sostenibilità.

## 2.2 Tecnologie satellitari

Negli ultimi anni, i satelliti in orbita terrestre bassa LEO, hanno acquisito un ruolo centrale nello sviluppo di tecnologie innovative e nella realizzazione di infrastrutture avanzate per la comunicazione e l'elaborazione dei dati. Le *satellite communications (SatComs)* sono l'avanguardia delle comunicazioni per garantire una copertura globale, in particolar modo nelle aree con poca se non addirittura nessuna copertura, in quanto non servite da infrastrutture terrestri [13]. Il Punto di partenza che ha permesso lo sviluppo delle *SatComs* è rappresentato dall'integrazione con il sistema di comunicazione di quinta generazione (5G). Quest'ultima ha permesso infatti di unire le peculiarità che caratterizzano la tecnologia satellitare, come la copertura su

tutta la superficie terrestre e le trasmissioni *multicast*, per soddisfare le esigenze dei mercati in rapida espansione, tra cui IoT e Industria 4.0.

### 2.2.1 Tipi di costellazioni satellitari

Un satellite, dopo il suo lancio, rimane mediamente in orbita per lunghi periodi di tempo, la permanenza nello spazio è possibile grazie all'equilibrio tra la forza gravitazionale della Terra e la forza centrifuga generata dal suo movimento. Poiché i satelliti orbitano al di fuori dell'atmosfera terrestre non sono soggetti all'attrito dell'aria, mantenendo così la velocità costante e garantendo orbite stabili [3]. La forza gravitazionale diminuisce all'aumentare della distanza dalla Terra, mentre la forza centrifuga cresce all'aumento della velocità orbitale;

Ciò implica che un satellite posizionato a circa 800km dalla superficie terrestre, che prende il nome di LEO, si muoverà ad una velocità elevata per bilanciare la maggiore attrazione gravitazionale. Al contrario ad una distanza di 36.000 km il periodo orbitale di un satellite è pari a 24 h, lo stesso tempo impiegato dalla terra a completare una rivoluzione su se stessa. *Geostationary Earth Orbit* (GEO) posti sopra l'equatore appaiono come stazionari rispetto alla superficie terrestre.

## 2.3 Simulazione Orbitale e Modelli di perturbazione

Per comprendere e prevedere il movimento dei satelliti, è necessario utilizzare un modello matematico che descriva con precisione le dinamiche che riguardano le orbite che seguono i satelliti. Esso non viene usato solamente per prevedere l'orbita dei satelliti o di oggetti spaziali, ma anche per ottimizzare traiettorie e prevenire collisioni. Partendo dall'utilizzo di modelli semplificati che tengono conto solo delle forze gravitazionali principali, a modelli avanzati che includono le perturbazioni dovute alla resistenza atmosferica e alle irregolarità del campo gravitazionale terrestre.

**SGP4** È un modello matematico utilizzato per calcolare i vettori di stato orbitali di satelliti e detriti spaziali. Esso tiene conto del sistema di coordinate inerziali centrato sulla Terra. Considera perturbazioni come l'irregolarità del campo gravitazionale terrestre, l'attrazione della Luna e del Sole, e la resistenza atmosferica. È il modello di riferimento per la previsione della posizione di satelliti in tempo reale, usato in *tracking* satellitari e simulazioni di missioni. Ma non è applicabile a satelliti in orbite alte o per simulazioni a lungo termine, poiché la precisione diminuisce con l'altezza dell'orbita.

## 2.4 Sfide nello scambio dati tra satelliti in movimento

*Mega-constellation networks* (MCNs) basate su LEO stanno acquisendo sempre più un ruolo centrale nel panorama delle comunicazioni globali[11]. MCNs, composte da centinaia se non migliaia di satelliti sono in grado di fornire connettività a banda larga e a bassa latenza a livello mondiale; Adottando tecnologie avanzate come *On-Board Processing* (OBP) e *Inter-Satellite Links* (ISLs).

### 2.4.1 ISLs

ISLs ricoprono un ruolo importante nelle MCNs, permettono infatti la connessione fra i vari elementi della costellazione; La variabilità della distanza e della posizione tra satelliti in orbita è una caratteristica chiave che influenza significativamente le prestazioni delle comunicazioni.

Tra i parametri chiave da considerare vi è il numero di *hop* ISLs necessari per connettere due utenti a terra. La quantità di *hop* sono un indicatore che riflette la complessità del *routing* nella rete e incide sulle prestazioni come la latenza.

Anche la **configurazione della costellazione**, in termini di inclinazione orbitale e fattore di sfasamento, gioca un ruolo fondamentale nel determinare il numero medio di *hop* ISLs tra gli utenti. Inoltre, la possibilità per gli utenti a terra di essere serviti da satelliti sia ascendenti che discendenti aggiunge un ulteriore livello di variabilità ai percorsi di comunicazione.

### 2.4.2 LISLs

Un'evoluzione in questo ambito è rappresentata dall'adozione dei *Laser Inter-Satellite Links* (LISLs)[10]. Rispetto ai tradizionali collegamenti a radiofrequenza, i LISLs offrono diversi vantaggi, come maggiori velocità di trasmissione dati, dimensioni di antenna più compatte e minori requisiti di potenza. I LISLs possono essere classificati in base alla posizione:

- LISLs *intra-orbital plane*, tra satelliti sullo stesso piano orbitale. I collegamenti sono generalmente permanenti.
- LISLs *inter-orbital plane*, tra satelliti in piani orbitali diversi. Ulteriormente suddivisibili in:
  - *Nearby Orbital Plane LISLs* (NOPL): I satelliti sono situati in piani orbitali vicini ma non adiacenti, Anche in questo caso, sono generalmente permanenti.

- *Crossing Orbital Plane LISLs* (COPL): I satelliti sono situati in piani orbitali che si incrociano, ad esempio, in costellazioni come Starlink, metà dei satelliti orbita in direzione nord-est, mentre l'altra metà si muove in direzione sud-est, questo crea piani orbitali incrociati. I collegamenti sono generalmente temporanei o intermittenti.



## Capitolo 3

# Studio del Problema : Copertura Satellitare, Comunicazione e Simulazione

### 3.1 Descrizione delle dinamiche di movimento nei satelliti

Il movimento dei satelliti in orbita attorno alla Terra è governato da una complessa interazione di fattori fisici e geometrici. La traiettoria e la velocità sono determinati principalmente dall'altitudine dell'orbita e dall'inclinazione rispetto al piano equatoriale terrestre. Questi parametri non solo influenzano il percorso del satellite, ma anche la capacità di coprire porzioni della superficie terrestre, garantendo una copertura globale o regionale. Comprendere le differenze tra i satelliti in *Low Earth Orbit* (LEO), *Medium Earth Orbit* (MEO) e *Geostationary Earth Orbit* (GEO) permette di analizzare le dinamiche di copertura, comunicazione e servizio offerto da ciascuna categoria.

#### 3.1.1 Classificazione e Caratteristiche dei Satelliti LEO, MEO e GEO:

Al momento del lancio, un satellite o un veicolo spaziale può essere inserito in una delle orbite specifiche attorno alla Terra, oppure può essere destinato a un viaggio interplanetario. In quest'ultimo caso, l'oggetto, non orbiterà più attorno al nostro pianeta, ma, sarà trasferito in un'orbita intorno al Sole. Il satellite eventualmente potrebbe raggiungere una destinazione finale come un corpo planetario dotato di

sufficiente forza gravitazionale, che consenta al veicolo di stabilizzarsi in un'orbita intorno ad esso[4].

### **Geostationary Earth Orbit (GEO)**

Gli oggetti spaziali posizionati in GEO sorvolano l'equatore terrestre segnando una traiettoria da ovest verso est, sincronizzata esattamente con la rotazione della Terra. Essi impiegano 23 ore, 56 minuti e 4 secondi per completare un'orbita, ossia la durata di una giornata siderale. Di conseguenza appaiono "stazionari" rispetto a un punto fisso sulla superficie terrestre; Viaggiano a una velocità di circa 3 km al secondo, ad un'altitudine di 35.786 km, ed una distanza significativamente maggiore rispetto alle altre orbite.

Questo tipo di orbita è ideale per i satelliti che necessitano di rimanere costantemente sopra una specifica posizione sulla Terra, come quelli utilizzati per le telecomunicazioni e il monitoraggio meteorologico. Caratteristica, che, consente alle antenne terrestri di puntare stabilmente verso il satellite senza necessità di frequenti riposizionamenti. Inoltre, è possibile coprire vaste porzioni della superficie, richiedendo un numero significativamente inferiore di satelliti rispetto alle orbite più basse per garantire una copertura globale.

### **Medium Earth Orbit (MEO)**

La MEO, rispetto alla precedente, comprende altitudini intermedie che si estendono a partire dalla LEO ad arrivare alla GEO, posizionandosi al di sopra delle fasce di Van Allen. I satelliti situati nelle orbite MEO non seguono percorsi rigidamente definiti donando loro una caratteristica di estrema versatilità. Uno degli utilizzi principali dell'orbita MEO riguarda i satelliti di navigazione, come Galileo, il quale fornisce servizi di navigazione globali supportando attività che spaziano dal monitoraggio di aerei di linea, al fornire indicazioni di percorso.

### **Low Earth Orbit (LEO)**

L'orbita LEO, relativamente vicina alla superficie terrestre, si estende fino ad un'altitudine massima di 2000 km, limite determinato dalla presenza delle fasce di Van Allen, le quali creano un ambiente ostile per i satelliti. Il limite inferiore è intorno ai 180 km, al di sotto di questa altitudine la resistenza atmosferica diventa significativa, rendendo difficile per un satellite mantenere l'orbita. A differenza di GEO, i satelliti in LEO hanno i loro piani orbitali inclinati di vari angoli, ciò permette loro di coprire diverse aree del pianeta, adattandosi a molteplici applicazioni. Essi

viaggiano a una velocità di circa 7.8 km al secondo, completando un'orbita attorno alla Terra in circa 90 minuti.

### 3.1.2 Modelli di distribuzione dei satelliti in LEO

È fondamentale adottare la giusta costellazione satellitare per realizzare missioni spaziali di successo; La scelta dipende strettamente dal tipo di missione che si andrà ad affrontare e delle specifiche esigenze operative. Fattori quali la copertura geografica e la continuità del servizio determinano l'adozione di un modello di distribuzione rispetto ad un altro. Spesso, quando si progetta una costellazione questa viene creata con l'obiettivo che più satelliti abbiano orbite, eccentricità e inclinazione simili, in modo che le perturbazioni influenzino ogni satellite allo stesso modo. Preservando la geometria strutturale della costellazione si riducono al minimo gli interventi di riposizionamento dei satelliti, risparmiando carburante e aumentando l'aspettativa di vita di questi ultimi. Le costellazioni ampiamente usate sono la **Walker-Delta** e **Walker-Star**.

#### Walker Constellations

La Walker Constellations venne proposta da J.G. Walker, progettata per ottimizzare la copertura globale, attraverso l'uso di più satelliti disposti in orbite specifiche, con l'obiettivo di ridurre il numero di satelliti necessari a garantire una copertura continua della superficie terrestre. È ampiamente utilizzata da sistemi come *Starlink* e *Global Positioning System (GPS)*. I satelliti disposti nella Walker Constellation condividono le seguenti proprietà[16]:

- **Quota orbitale:** I satelliti si trovano tutti alla stessa altezza rispetto alla superficie terrestre.
- **Distribuzione Orbitale:** I satelliti sono equidistanti gli uni dagli altri nella loro orbita.
- **Orbita inclinata:** I satelliti hanno la stessa inclinazione rispetto al piano di riferimento.

#### Walker-Star

La Walker Star fa parte della famiglia Walker, ed è caratterizzata da orbite polari, le quali permettono una copertura globale della superficie terrestre. Esse hanno un'inclinazione prossima ai 90° rispetto all'equatore, consentendo ai satelliti di sorvolare le regioni polari ad ogni passaggio. La suddetta caratteristica le rende ideali

per il telerilevamento, il monitoraggio ambientale, la sorveglianza e le comunicazioni globali. I satelliti in piani orbitali adiacenti si muovono nella stessa direzione e sono sfalsati per evitare scontri nelle zone d'intersezione di tutti piani [23].

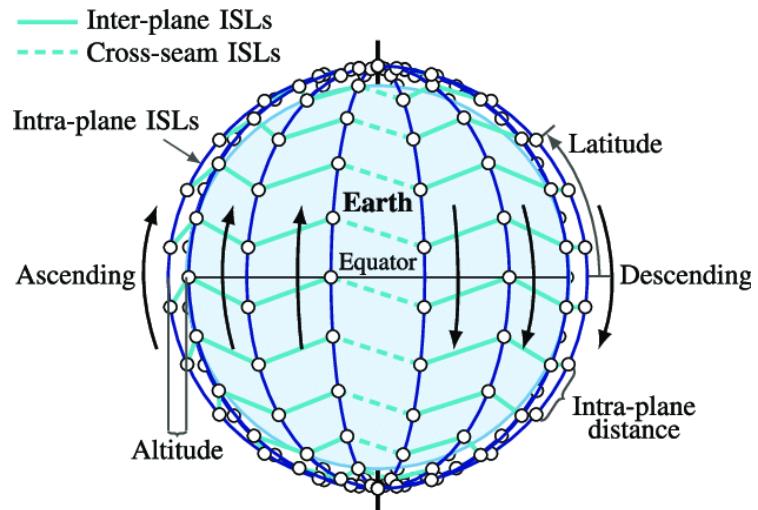


Figura 3.1. *Walker Star Pattern* [14]

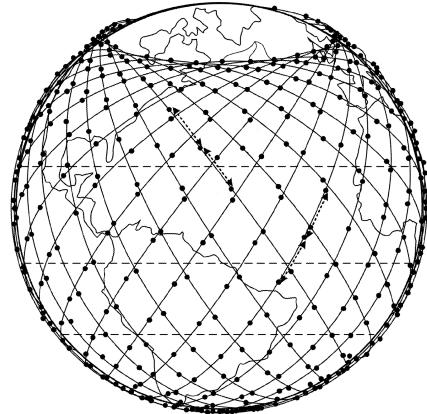
### Walker-Delta

La Walker-Delta è caratterizzata da una distribuzione uniforme dei satelliti su più piani orbitali inclinati, con la stessa inclinazione rispetto all'equatore [16]. Questo design consente un'elevata regolarità nella copertura riducendo la complessità nella gestione della rete, dove i piani orbitali non passano direttamente sopra i poli, creando zone senza copertura. Concentrando la disponibilità nelle fasce a latitudine medio-bassa, dove risiede la maggior parte della popolazione e delle necessità di comunicazione. Le relazioni tra i satelliti in diversi piani orbitali rimangono relativamente costanti nel tempo, semplificando il *routing* dei dati e le operazioni di *handover* tra i nodi. D'altra parte, durante il suo percorso, un satellite può interagire con nodi appartenenti a piani orbitali più distanti dal proprio, tale fenomeno si verifica a causa dell'intersezione tra più piani orbitali in punti specifici dell'orbita. Un esempio, durante l'ascesa del satellite da sud verso nord, incontrerà nodi discendenti di altri piani, stabilendo connessioni temporanee utili per il trasferimento delle informazioni. Per descrivere matematicamente la costellazione possiamo definire i seguenti parametri:

- **T:** Numero totale di satelliti.
- **P:** Numero di piani orbitali.
- **F:** *Phase factor*. Valore che determina la distribuzione dei satelliti tra i vari  $p \in P$  dove  $0 < F < P - 1$ .

Assumiamo che il *right ascension of ascending node* (RAAN) del primo piano orbitale è  $\Omega_0$ , dove RAAN rappresenta la posizione del piano che interseca la terra, e assumiamo anche che il primo satellite in  $\Omega_0$  è  $u_0$ .

$$\begin{cases} \Omega = \Omega_0 + (m - 1)\frac{2\pi}{P} \\ u = u_0 + (m - 1)F\frac{2\pi}{T} + (n - 1)P\frac{2\pi}{T} \end{cases} \quad (1)$$



**Figura 3.2.** Delta Pattern[17]

$\Omega$ 

$\Omega$  rappresenta la RAAN del piano orbitale  $m$  rispetto a  $\Omega_0$ . Poiché i piani devono essere distribuiti uniformemente lungo i  $360^\circ$  attorno al globo, la distanza angolare tra i piani, indicata con  $\Delta\Omega$ , è:

$$\Delta\Omega = (m - 1) \frac{360^\circ}{P} = (m - 1) \frac{2\pi}{P} \quad (\text{in radianti})$$

Pertanto, la posizione del piano  $m$  rispetto a  $\Omega_0$  è:

$$\Omega = \Omega_0 + (m - 1) \frac{2\pi}{P}$$

 $u$ 

$u$  permette di determinare la posizione del satellite lungo la sua orbita. Pertanto la formula ci assicura che i satelliti siano distribuiti uniformemente lungo l'orbita e tra i diversi piani. Dove:

- $u_0$  è la fase iniziale del primo satellite di  $\Omega_0$ .
- $(m - 1)F \frac{2\pi}{T}$  : Termine che controlla lo sfasamento tra i piani orbitali.
- $(n - 1)P \frac{2\pi}{T}$  : Termine che controlla la posizione del satellite nel suo piano orbitale.

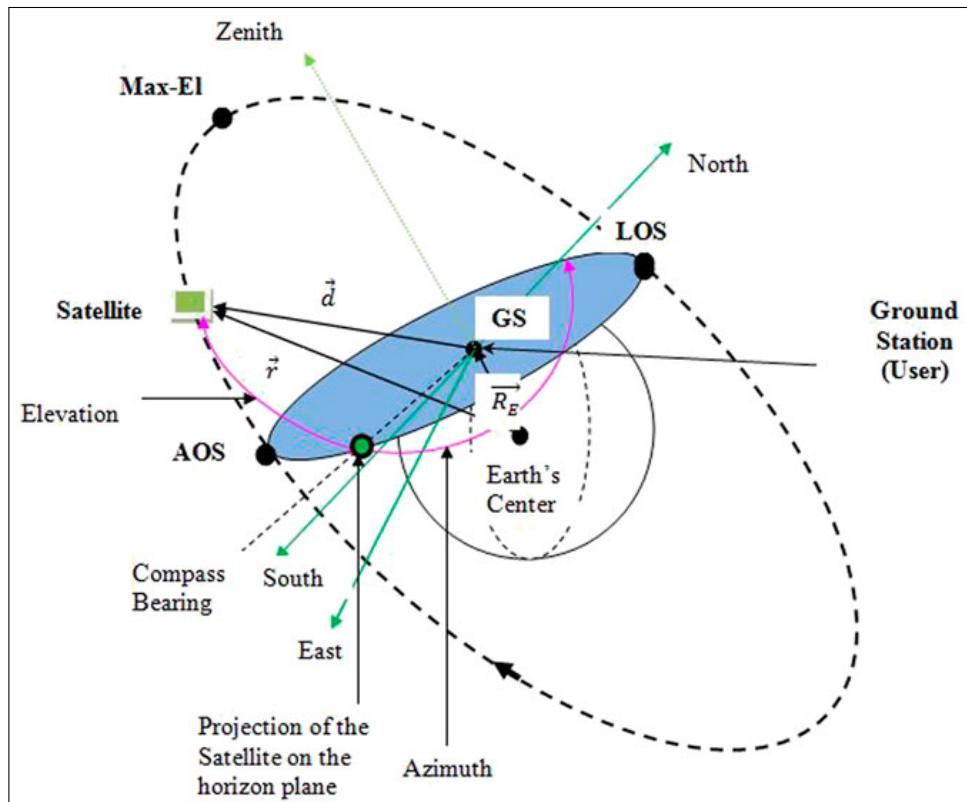
### 3.1.3 Copertura dell'utente a Terra: Numero di satelliti con buona qualità del segnale e senza disturbi dovuti all'angolo di elevazione

Lo studio presentato in [9], evidenzia l' importanza di considerare specifici angoli di elevazione, per garantire l' accesso ai servizi satellitari. Definiamo i punti lungo l' orbita del satellite rispetto la posizione dell'utente sulla Terra, Figura 3.3:

- **Acquisition of the Satellite (AOS)** : Il satellite sorge sull'orizzonte rispetto al campo visivo dell'utente che vuole effettuare la connessione.
- **Loss of Signal (LOS)** : Il Satellite tramonta oltre l'orizzonte rispetto al campo visivo dell'utente.
- **Maximal Elevation (Max-El)** : Punto di massima elevazione del satellite, nel piano orbitale rispetto al campo visivo dell'utente.

Fra AOS e LOS è possibile stabilire una connessione ed in seguito iniziare una comunicazione diretta con il satellite. Si fa altresì riferimento alle seguenti coordinate angolari:

- **Azimuth** : Angolo misurato in senso orario a partire dal Nord geografico fino alla proiezione dell'oggetto sull'orizzonte. Tale valore ci permette di capire da quale direzione il satellite raggiunge l'utente. Esso viene espresso in gradi e può assumere un valore compreso tra  $0^\circ$  e  $360^\circ$ . Dove:
  - $0^\circ$  : Nord.
  - $90^\circ$  : Est.
  - $180^\circ$  : Sud.
  - $270^\circ$  : Ovest
- **Zenith** : Punto sulla sfera celeste che si trova esattamente sopra l'osservatore, a  $90^\circ$  rispetto all'orizzonte; Nel seguito useremo questo punto come riferimento per confrontare l'elevazione del satellite.



**Figura 3.3.** Rappresentazione del Piano dell'orizzonte rispetto l' utente [9]

Ulteriori vettori utili al fine di capire il contesto nel quale analizziamo i movimenti dei satelliti sono:

- $\vec{R}_E$  : Vettore che parte dal centro della terra all'utente, il piano dell'orizzonte è considerato come tangente a questo vettore.
- $\vec{d}$  : Vettore che collega il satellite e la *Ground Station* (GS), la quale rappresenta la distanza effettiva tra il satellite e l'utente. Esso dipende da *Max-El* e dal percorso che il satellite percorre sopra l'utente.
- $\vec{r}$  : Raggio orbitale, vettore che collega il centro della terra al satellite, ogni orbita in LEO ha una sua altitudine  $H$  rispetto alla superficie.

$$\vec{r} = H + \vec{R}_E \quad (3.1)$$

Dove  $\vec{R}_E = 6,371 \text{ km}$  raggio della terra.

La velocità del satellite in Orbita è indicata dalla seguente formula:

$$v = \sqrt{\frac{\mu}{\vec{r}}} \quad (3.2)$$

Dove  $\mu = M \cdot G = 3.986 \times 10^5 \text{ km}^3/\text{s}^2$  parametro gravitazionale standard, prodotto tra la Massa della terra  $M$  e la costante gravitazionale universale  $G$ . Il periodo orbitale viene espresso con:

$$T = 2\pi \sqrt{\frac{r^3}{\mu}} \quad (3.3)$$

In figura 3.4  $n$  è il numero di volte al giorno in cui un satellite in LEO interseca la linea equatoriale, considerando il rapporto con il giorno di durata ( $23 \text{ h } 56 \text{ min } 4,1 \text{ s}$ ), rispetto all' altitudine  $H$ . Nello studio [9] viene fatta un analisi su come cambino i parametri cambiando solamente l' altitudine.

Collegarsi ad un satellite LEO che si trova in fascia più alta permette alla GS di guadagnare tempo in termini di trasmissione e connessione;

Tuttavia, questo modello ideale, non tiene conto di ostacoli naturali o artificiali quali montagne, alberi o edifici possono interferire con la linea di vista e quindi impedire o ritardare l' inizio della comunicazione. La soluzione adottata per tali interferenze, è aumentare l' angolo di elevazione minimo, indicato con  $\varphi_{\max}$ , dal

| Parameters | The first shell<br>$H = 550 \text{ km}$ | The second shell<br>$H = 1110 \text{ km}$ | The third shell<br>$H = 340 \text{ km}$ |
|------------|---|---|---|
| $r$ (km)   | 6,921                                   | 7,481                                     | 6,711                                   |
| $v$ (km/s) | 7.589                                   | 7.299                                     | 7.706                                   |
| $T$ (min)  | 95.5                                    | 107.3                                     | 91.2                                    |
| $n$        | 15.03                                   | 13.38                                     | 15.74                                   |

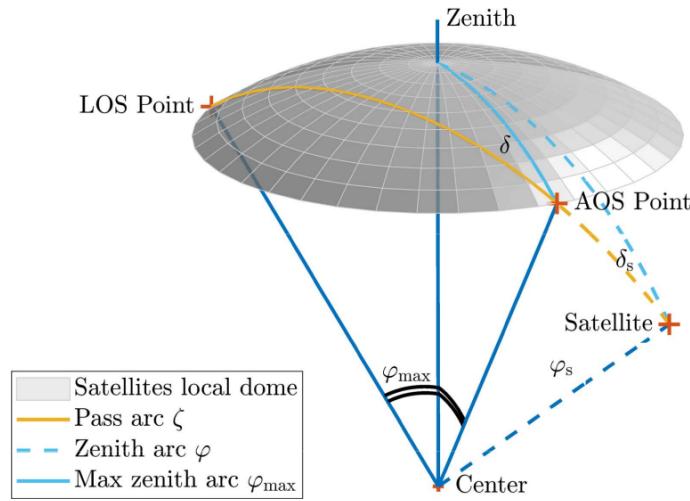
**Figura 3.4.** Analisi Velocità, Raggio, Periodo orbitale e numero passaggi giornalieri dei satelliti in base alla loro altitudine. [9]

quale la comunicazione può iniziare. Gli angoli fondamentali per predire la durata del passaggio dei satelliti nelle reti satellitari ultra dense, vengono rappresentati in figura 3.5:

- $\varphi_{\max}$  : Angolo fondamentale, indica da quando è possibile instaurare una connessione con il satellite. Questo angolo, ci permette di definire una cupola di ascolto del satellite, nella quale andremo a creare una prima **topologia**. Nei test condotti, il valore dell'angolo  $\varphi_{max}$  è stato fissato a  $40^\circ$ .
- $\varphi_s$  : Angolo di elevazione generale del satellite rispetto al sistema adottato.
- $\delta$  : Angolo di ingresso del satellite nella cupola, calcolato rispetto alla "Zenith reference line". Essa rappresenta la linea ideale che collega il punto di ingresso del satellite nella cupola locale con lo *zenith* dell'osservatore. Serve come riferimento per misurare il  $\delta$  che, definisce la deviazione del percorso del satellite, rispetto alla direzione zenitale nel momento in cui entra nella cupola.
- $\delta_s$  : In figura 3.5, rappresenta la rotazione dell'orbita rispetto alla linea di riferimento zenitale, influenzando così l'angolo di ingresso  $\delta$  del satellite nella cupola locale.

Considerando tali angoli, il valore varia in base al sistema che si è deciso di adottare. Nel contesto che stiamo analizzando il sistema può essere il centro della terra oppure una GS.

Nello studio [8], viene mostrato come  $\delta$  influenzi la durata del passaggio dei satelliti dentro la cupola (Figura 3.6). Riconoscendo in AOS il punto in cui il satellite sorge ed entra nel nostro campo di ascolto. Utilizzando  $\delta$ , possiamo determinare per quanto tempo il satellite rimarrà accessibile. Stabilendo un angolo limite, è possibile prevedere quando il contatto con il satellite diventa inefficace, evitando tentativi di comunicazione non produttivi.

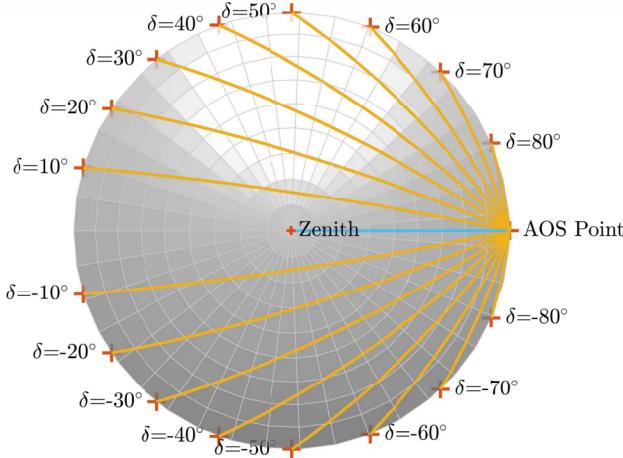


**Figura 3.5.** Cupola di ascolto dei satelliti. [8]

### 3.1.4 Strategie di Selezione del Satellite per la Connessione

L'angolo  $\delta$  influenza anche **Max-El**. Questa considerazione è doverosa poiché, come vedremo nelle sezioni successive, il punto nel quale si trova il satellite durante il suo percorso nella cupola influenza direttamente la quantità di dati trasmissibili.

Come mostrato in [8], diventa necessario trovare delle formule che permettano



**Figura 3.6.** Rappresentazione grafica di come al variare di  $\delta$  cambi il percorso del satellite nella cupola. [8]

di automatizzare il calcolo in fase di esecuzione. Introduciamo 3.4, 3.5, 3.6 e 3.7

<sup>1</sup> e comprendere come vengano applicate nel calcolo dei migliori satelliti ai quali collegarsi. Mostriamo degli esempi di applicazione delle formule riportate:

$$\delta = \arcsin \left( \frac{\sin \varphi_s}{\sin \varphi_{\max}} \sin \delta_s \right) \quad (3.4)$$

**Figure 3.4:** Angolo d'entrata dei satelliti nella cupola di ascolto.

$$\zeta = 2 \arctan(\cos \delta \tan \varphi_{\max}) \quad (3.6)$$

**Figure 3.6:** lunghezza dell'arco che attraversa la cupola.

$$T_p = \frac{\zeta}{2\pi} T_o = \frac{T_o}{\pi} \arctan(\cos \delta \tan \varphi_{\max}) \quad (3.5)$$

**Figure 3.5:** Tempo di transito del satellite sulla cupola osservabile come frazione del periodo orbitale.

$$T_o = 2\pi \sqrt{\frac{(\vec{R}_E + H)^3}{\mu}} \quad (3.7)$$

**Figure 3.7:** approssimazione del periodo orbitale.

---

<sup>1</sup>La formula 3.7 è più precisa rispetto a 3.3, poiché viene considerata anche la distanza tra la superficie terrestre e l'oggetto orbitante.

**Esempio Applicazione delle formula 3.4**

Consideriamo 3.4 per calcolare l'angolo di entrata nella cupola

$$\delta = \arcsin \left( \frac{\sin \varphi_s}{\sin \varphi_{\max}} \cdot \sin \delta_s \right)$$

Ad esempio, assumiamo che gli angoli abbiano i seguenti valori:

$$\varphi_s = 30^\circ, \quad \delta_s = 45^\circ, \quad \varphi_{\max} = 40^\circ$$

**1. Calcoliamo il seno degli angoli :**

$$\sin 30^\circ = 0.5, \quad \sin 40^\circ \approx 0.6428, \quad \sin 45^\circ \approx 0.7071$$

**2. Sostituire i valori nella formula :**

$$\delta = \arcsin \left( \frac{0.5}{0.6428} \cdot 0.7071 \right)$$

**3. Eseguiamo il calcolo della frazione :**

$$\frac{0.5}{0.6428} \approx 0.777$$

**4. Eseguiamo il prodotto :**

$$0.777 \cdot 0.7071 \approx 0.549$$

**5. Calcoliamo :**

$$\delta = \arcsin(0.549) \approx 33.3^\circ$$

Il risultato è l'angolo di entrata del satellite nella cupola.

Esempio Calcolo del tempo di transito del satellite nella cupola 3.5

Consideriamo 3.5 in funzione di 3.7:

$$T_p = \frac{\zeta}{2\pi} T_o = \frac{T_o}{\pi} \arctan(\cos \delta \tan \varphi_{\max})$$

con  $T_o$  approssimato come il periodo orbitale di un satellite intorno alla Terra:

$$T_o = 2\pi \sqrt{\frac{(R_\oplus + h)^3}{\mu}}$$

**Assumiamo i seguenti valori ai fini dell'esempio:**

- $\vec{R}_E : 6371 \text{ km}$
- $H : 500 \text{ km}$
- $\mu : 398\,600 \text{ km}^3/\text{s}^2$
- $\delta : 33.3^\circ$
- $\varphi_{\max} = 40^\circ$

1. **Calcoliamo**  $\vec{r} = (\vec{R}_E + H)$ : Distanza tra il satellite e il centro della terra.

$$\vec{R}_E + H = 6371 + 500 = 6871 \text{ km}$$

2. **Calcoliamo**  $T_o$ : Tempo che ci impiega un satellite per completare l'orbita del globo rispetto al piano orbitale sul quale è posizionato.

$$T_o = 2\pi \sqrt{\frac{(6871)^3}{398600}} = 5730 \text{ s} \approx 95.5 \text{ min}$$

3. **Valori dati per il calcolo di  $\zeta$**  : Sostituiamo i valori in 3.6.

$$\zeta = \arctan(\cos(33.3^\circ) \cdot \tan(40^\circ))$$

4. **Sostituiamo**  $\cos \delta$  e  $\tan \varphi_{\max}$  con le rispettive approssimazioni :

$$\zeta = \arctan(\cos 33.3^\circ \approx 0.834 \cdot \tan 40^\circ \approx 0.8391)$$

5. **Calcolo del prodotto e dell'arco tangente:**

$$\zeta = \arctan(0.834 \cdot 0.8391 \approx 0.700) \approx \arctan(0.700) \approx 34.9^\circ \approx 0.609 \text{ rad}$$

6. **Calcolo finale di  $T_p$**

$$T_p = \left( \frac{0.609}{\pi} \right) \cdot 95.5 \approx 18.5 \text{ min}$$

Il risultato finale indica quanto tempo il satellite impiegherà ad attraversare la cupola, con sistema riferimento rispetto al centro della terra.

Considerando l'angolo d'entrata dei satelliti nella cupola e del tempo che questi impiegano per andare da AOS a LOS, è possibile escludere o meno alcuni satelliti rispetto ad altri, dunque quantizzare i satelliti con i quali è possibile instaurare una connessione e avere miglioramenti in termini di efficienza.

## 3.2 Implicazione del movimento dei satelliti per la copertura

### 3.2.1 Densità Satellitare e Qualità del Segnale

Nel seguente studio [12] si effettua l'analisi di quale sia la formazione migliore per assicurare una connessione che abbia le capacità di soddisfare il maggior numero di utenti, con un livello di continuità elevata. A supporto di una copertura continua per la GS viene configurata una costellazione ultra-densa di satelliti LEO, in modo da garantire che almeno uno sorvoli l'area d'interesse in ogni intervallo di tempo. Si assume che, i satelliti vengano distribuiti su  $N$  piani orbitali ad altezza  $H$  rispetto alla superficie terrestre, dove  $n \in N$  piano orbitale, composto da  $M_n = \{1, 2, \dots, M_n\}$  satelliti. Ogni satellite è associato a tutte le GS presenti nella sua area di copertura, tutte condividono la larghezza di banda  $B$  che il satellite mette a disposizione per la comunicazione.

Viene evidenziata la correlazione tra l'aumento del numero di satelliti in orbita sullo stesso piano e i requisiti di *backhaul*<sup>2</sup> imposti dagli utenti. In particolare all'aumentare della quantità di dati che gli utenti desiderano trasmettere, cresce la necessità di incrementare il numero di satelliti in orbita, al fine di soddisfare tali richieste. Viene altresì analizzato il rapporto di copertura media, il quale deve essere garantito per una determinata percentuale di tempo. Si mira ad assicurare una qualità del servizio costante e affidabile per le comunicazioni satellitari.

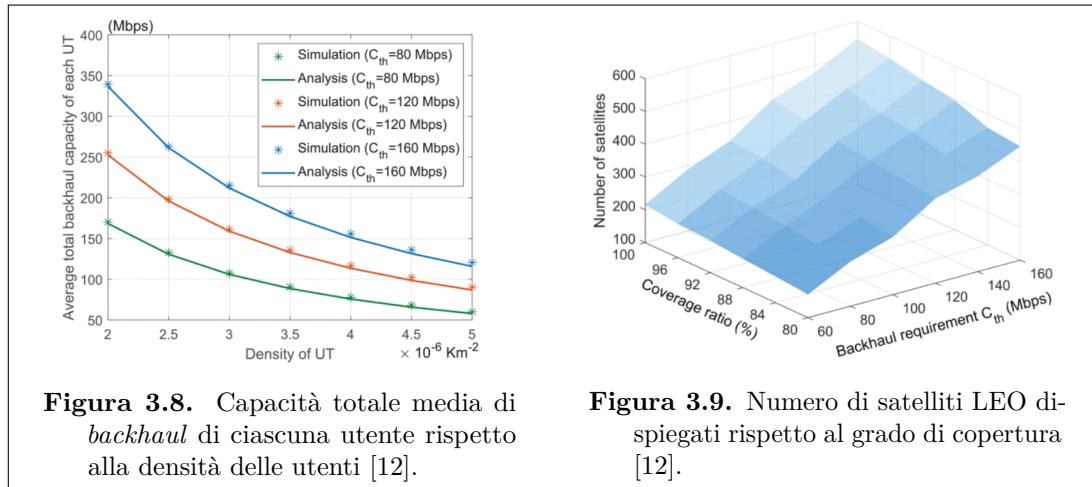
| Required coverage degree $k_{\min}$ | 3                       | 4   | 5   | 6   | 7   | 8   |
|-------------------------------------|-------------------------|-----|-----|-----|-----|-----|
| Number of LEO satellites            | Coverage ratio over 90% | 193 | 259 | 311 | 389 | 440 |
|                                     | Coverage ratio over 95% | 202 | 270 | 328 | 408 | 468 |

**Figura 3.7.** Tabella rappresentativa della percentuale di copertura e della quantità di satelliti richiesta [12].

Lo studio [12] dimostra come all'aumentare del grado minimo di copertura richiesto

<sup>2</sup>I Requisiti di backhaul comprendono: *bandwidth*, *latency*, *reliability*, *availability*, *Quality of Service* (QoS), *geographic coverage*, *security* e *handover* gestito correttamente

$(K_{min})$ , che rappresenta il numero di satelliti visibili contemporaneamente dalla GS, cresce proporzionalmente anche il numero totale di satelliti necessari garantendo un determinato livello di affidabilità; L'incremento dipende anche dal livello di copertura che si desidera offrire. Nello studio si dimostra che all'aumentare degli utenti nell'area di copertura diminuisce la capacità di *backhaul* per ognuno di essi (Figure 3.8), e come il numero di satelliti presenti nell'area influenzi anche il rapporto di copertura (Figure 3.9).



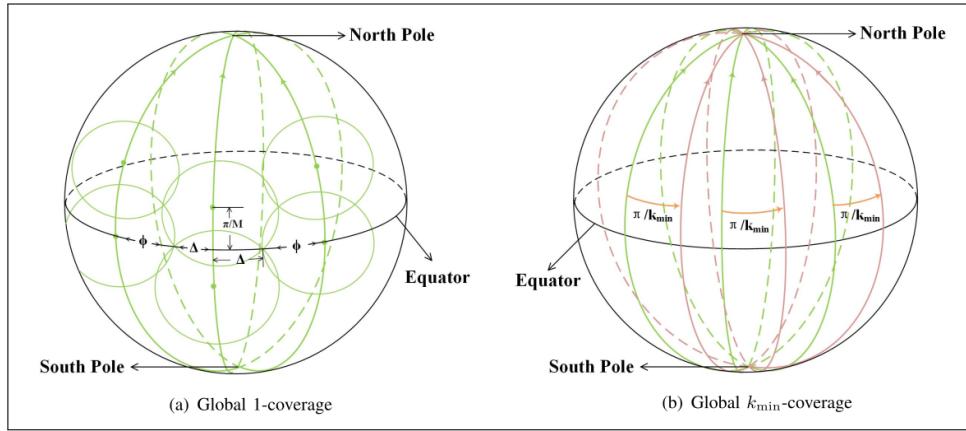
**Figura 3.8.** Capacità totale media di *backhaul* di ciascuna utente rispetto alla densità delle utenti [12].

**Figura 3.9.** Numero di satelliti LEO disposti rispetto al grado di copertura [12].

Lo studio fa emergere la necessità di trovare il miglior satellite con il quale stabilire una comunicazione, e come essa si presenti solo quando esso si trova nell'intersezione delle zone di copertura di questi ultimi. Nella Figura 3.10, viene mostrato il percorso che i satelliti seguono applicato alla Walker-Star Constellation sezione 3.1.2. Notiamo che le aree di copertura dei satelliti rispetto alla superficie terrestre, spesso si intersecano; Solo in quei casi, si ritene necessario trovare il migliore satellite con il quale effettuare una connessione.

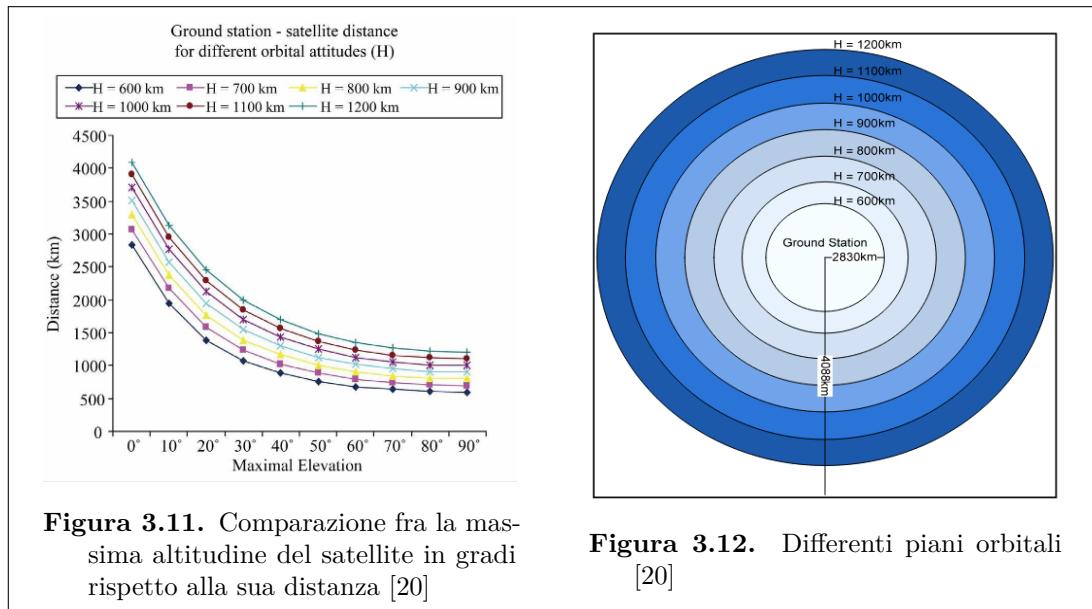
### 3.2.2 Impatto dell'Angolo di Elevazione sulla Qualità del Link

Di seguito si analizza il problema del ritardo nella trasmissione dei segnali satellitari, con particolare attenzione all'angolo di elevazione necessario per una comunicazione efficiente. Nello studio [20] si evidenzia come l'angolo di trasmissione teorico pari a  $0^\circ$ , pur rappresentando un piano ideale, non sia utilizzabile nella pratica. Trasmettere informazioni con un satellite posizionato sul limite del piano ideale comporterebbe ritardi significativi e perdite di segnale, dovuti alla presenza di ostacoli naturali e architettonici, come montagne o edifici. È necessario determinare un angolo di elevazione ottimale, che, garantisca un equilibrio tra la dimensione del piano e la distanza



**Figura 3.10.** Tabella rappresentativa della percentuale di Copertura e della quantità di satelliti richiesta [12].

tra l'utente e il satellite, trovare questo equilibrio è fondamentale per garantire una comunicazione efficiente. Un angolo troppo stretto, circoscriverebbe l'area del piano, diminuendo il numero di satelliti potenzialmente visibili e la probabilità di stabilire una connessione in tempi brevi. Tuttavia, l'angolo ridotto, migliorerebbe la qualità del segnale, poiché la distanza tra l'utente e il satellite verrebbe accorciata al minimo, diminuendo l'impatto degli ostacoli (Figure 3.11 3.12). D'altro canto, un angolo eccessivamente ampio aumenterebbe l'area visibile, ma potrebbe compromettere la qualità della comunicazione a causa della maggiore distanza e della maggiore interferenza con ostacoli locali.



### 3.2.3 Difficoltà nel prevedere i movimenti dei satelliti

Per la realizzazione di un modello funzionale, ai fini della simulazione, si è deciso di adottare il Semplified General Perturbation 4 (SGP4) che si basa su calcoli algebrici. Sviluppato nel 1960 è stato ampiamente utilizzato nel 1980 nel campo delle previsioni dei movimenti satellitari [2] [19] [21]. Grazie all'aumento del numero di siti di osservazione di alta qualità a basso costo, nonché all'impiego di dati radar (bi-statici) [15], è oggi possibile effettuare analisi statistiche sugli errori di posizione e velocità degli oggetti osservati, e dunque valutare la qualità del modello adottato. SGP4 è parte della famiglia degli SGP, che comprende altri modelli come SGP, SDP4, SGP4, SGP8 e SDP8. SGP4 e Simplified Deep space Perturbations 4 (SDP4) differivano per periodo orbitale:

- SGP4 : Se  $T \leq 225'$  (oggetti orbitali che rientrano nel raggio di  $R = 5877.5 \text{ km}$ ). Dove  $T$  è riportato in 3.7.
- SDP4 : Se  $T > 225'$ .

Tra la metà e la fine degli anni '70, il modello SGP4 è stato esteso per soddisfare anche i requisiti dello spazio profondo, consentendo l'inclusione di satelliti con un periodo orbitale superiore a 225 minuti. In particolare, il modello SDP4 introduce nel SGP4 le perturbazioni lunari e solari. SGP4 utilizza un algoritmo che tiene conto delle perturbazioni solari, della resistenza atmosferica, delle irregolarità del campo gravitazionale terrestre e delle variazioni dovute all'influenza lunare[22]. I due modelli si differenziano nella quantità di parametri dei quali tengono conto e della loro complessità. Prendono come input dati in formato TLE 4.2.1 producendo un errore trascurabile, che varia in base all'altitudine del satellite rispetto la superficie terrestre.

| Orbit    | Accuracy [km] |
|----------|---------------|
| LEO, NSO | $\sim 1$      |
| MEO      | $1 \div 2$    |
| GEO      | $2 \div 4$    |
| HEO, GTO | $6 \div 8$    |

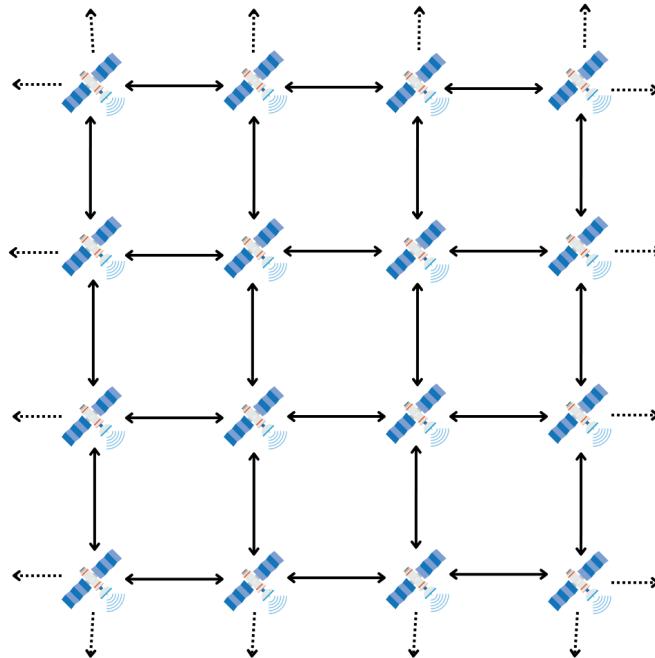
Tabella 3.1. SGP4 model accuracy for different kinds of orbits [19]

Diventa quindi fondamentale l'analisi e la previsione dei movimenti dei corpi orbitanti per poter evitare collisioni nei punti dove si ritrovano intersezione fra i vari piani.

### 3.3 Simulazione satellitare - Satellite Edge Computing (SEC)

*Satellite Edge Computing* (SEC) è un paradigma emergente che punta a trasformare la costellazione di satelliti in orbita bassa (LEO) in una piattaforma computazionale distribuita, con il potenziale di estendere le capacità dell'*edge computing* nello spazio. Tuttavia, la natura altamente dinamica delle costellazioni LEO introduce sfide significative nella gestione delle risorse, nella previsione dei movimenti dei nodi e nell'allocazione efficiente delle attività computazionali.

Uno degli aspetti fondamentali del SEC è la definizione della topologia della rete di satelliti. L'articolo [7] assume una costellazione fissa di 25 *Satellite Edge Nodes* (SEN), seguendo una struttura come mostrata in Figura 3.13, in cui i nodi ai margini possono comunicare con i loro opposti o meno. Questa configurazione rappresenta un'ottima base per confrontare le prestazioni dei diversi approcci di *scheduling*, sebbene introduca una semplificazione rispetto a un ambiente reale, dove i satelliti sono in costante movimento e la loro disponibilità varia nel tempo.



**Figura 3.13.** Illustrazione della struttura adottata nel simulatore presentato nell'articolo [7].

### 3.3.1 Gestione del carico computazionale

Lo studio [7] evidenzia chiaramente come un approccio decentralizzato al SEC, come *Decentralized Task Scheduling with Deadline (DTS-TMAX)*, garantisca prestazioni migliori rispetto a una soluzione centralizzata (CTS-TMAX). Il vantaggio principale di DTS-TMAX risiede nella distribuzione del carico computazionale direttamente tra i SEN, evitando la necessità di un nodo centrale come punto unico di decisione e mitigando il rischio di colli di bottiglia.

#### Access Point Satellitari

Un elemento cruciale nella gestione del SEC è la selezione degli Access Point Satellitari (AP-SAT), che fungono da punti d'ingresso per le richieste degli utenti. Questi ricevono i task da elaborare e li instradano all'interno della rete. La scelta dell'AP-SAT avviene in modo casuale e, una volta selezionato, i task possono essere eseguiti direttamente dal satellite di accesso o redistribuiti ad altri nodi della rete.

#### Caratteristiche dei Task

Ogni task elaborato dal sistema è caratterizzato dai seguenti attributi:

- *Require time*: durata del task, espressa in secondi, che indica per quanto tempo il processore del satellite dovrà eseguire il task.
- *Required CPU*: quantità di CPU necessaria per l'esecuzione del task, influenzando la capacità di elaborazione disponibile.
- *Required RAM*: quantità di memoria volatile (RAM) richiesta per il corretto funzionamento del task.
- *Required memory*: spazio di archiviazione occupato dal task nella memoria di massa del satellite.

### 3.3.2 Latenza e Scheduling Decentralizzato dei Task

Nell'algoritmo DTS-TMAX, la latenza gioca un ruolo cruciale nella fase di distribuzione dei task all'interno della rete di *Satellite Edge Nodes (SENs)*. L'algoritmo è progettato per minimizzare il ritardo nella trasmissione e nell'esecuzione dei task, assicurando che ogni attività venga completata entro la scadenza  $D_r$ .

Il *MONITOR* raccoglie continuamente dati sulle latenze dei collegamenti  $l_i^j$  e sulla banda disponibile  $b_i^j$  dove  $j \in E_i^t$  insieme dei nodi vicini del nodo SEN  $i$  al tempo  $t$ . Queste informazioni sono essenziali per determinare la disponibilità di ciascun

nodo e valutare l'impatto della latenza nella scelta della destinazione ottimale per un task. Inoltre, il *MONITOR* controlla i task in coda su ciascun server, tenendo conto delle priorità. Nei Nodi SEN i task ad alta priorità vengono infatti eseguiti prima di quelli a bassa. Ogni Server, dispone di un punteggio che varia in base al carico nelle code e le priorità dei task.

Durante la fase di ricerca di un nodo adatto, la funzione *SearchNode* applica un filtraggio basato sulla latenza, scartando i SEN che non possono garantire il completamento del task entro la deadline. La selezione dei nodi si basa sul valore  $R_{r,i,j}$ , che rappresenta il tempo di risposta totale per il task  $r$  inviato dal SEN  $i$  al SEN  $j$ . Questo valore è composto da:

$$R_{r,i,j}^t = X_{r,i,j} + c_{r,i} + T_{ij}^H + T_{ij}^L \quad (3.8)$$

dove  $X_{r,i,j}$  rappresenta il tempo di trasmissione del task,  $c_{r,i}$  il tempo di computazione richiesto dal task su  $SEN_i$ , e  $T_{ij}^H$  e  $T_{ij}^L$  i ritardi di comunicazione dovuti alle priorità dei task.

Se un nodo con risorse disponibili e latenza accettabile viene individuato, il task viene assegnato tramite la funzione *TaskAssignment*. Tuttavia, se nessun nodo soddisfa i vincoli, l'algoritmo attiva *PropagateSearch*, propagando la richiesta ad altri SEN vicini. In questo passaggio, la latenza influisce direttamente sulla decisione, poiché il valore  $D_r$  viene decrementato del tempo di trasmissione  $X_{r,i,j}$  per ogni hop, riducendo progressivamente il margine di esecuzione del task.

Questo garantisce un equilibrio tra efficienza e resilienza della rete, riducendo il rischio di task non eseguiti a causa di vincoli temporali troppo stringenti. Inoltre, l'algoritmo può adattarsi dinamicamente alle condizioni di rete variabili, aggiornando i parametri di selezione dei nodi in base ai dati raccolti dal *MONITOR*, migliorando così l'affidabilità e la reattività del sistema.



# Capitolo 4

## Metodologia

### 4.1 Oggetto della ricerca

L’oggetto della ricerca riguarda lo sviluppo di un modello integrabile nel simulatore Satellite Edge Computing (SEC), con particolare attenzione alla previsione e gestione della copertura satellitare. L’obiettivo principale, è elaborare un sistema capace di generare mappe satellitari dinamiche, basate sulle posizioni dei satelliti in tempo reale simulando lo scambio di dati.

Il modello proposto sfrutta dati orbitali aggiornati per rappresentare con accuratezza la posizione e il movimento dei satelliti, consentendo di simulare la distribuzione dei task IoT in un contesto realistico. Un aspetto cruciale della ricerca riguarda la gestione delle relazioni temporanee tra i satelliti, i quali variano continuamente a causa del loro moto orbitale. Questa dinamicità, introduce una complessità aggiuntiva nella pianificazione e ottimizzazione dell’allocazione delle risorse computazionali.

### 4.2 Dati e Strumenti di base

#### 4.2.1 Two-Line Elements (TLE) e dati orbitali

Il formato Two-Line Element (TLE) è uno standard per la distribuzione delle informazioni orbitali da parte del North American Aerospace Defense Command (NORAD). Per acquisire i TLE più recenti è stata utilizzata CelesTrak<sup>1</sup>, una delle principali fonti di dati orbitali per il monitoraggio dei satelliti artificiali. Questo sito fornisce le posizioni dei satelliti aggiornate, suddivisi in categorie specifiche, tra cui satelliti di navigazione, comunicazione e osservazione terrestre [1]; si è così scelto di utilizzare i satelliti della costellazione StarLink, in quanto l’azienda è attualmente

---

<sup>1</sup>I dati TLE sono reperibili sul sito [CelesTrack.com](http://CelesTrack.com)

tra le principali investitrici nel settore e dispone di una rete di oltre 6.900 satelliti attivi, dati ottenuti tramite l'analisi degli elementi TLE.

**Listing 4.1.** Esempio di dati TLE relativi a tre satelliti Starlink

STARLINK-1008

```
1 44714U 19074B    25043.51880833 -.00000405  00000+0
   -82922-5 0  9999
2 44714   53.0560  138.1627 0001487   82.4102  277.7055
   15.06377817290117
```

STARLINK-1009

```
1 44715U 19074C    25043.91668981 .00270735  37978-4
   88341-3 0  9993
2 44715   53.0462   97.0573 0008418   41.4042   44.5298
   15.90163923290710
```

STARLINK-1010

```
1 44716U 19074D    25043.52617303 .00002107  00000+0
   16036-3 0  9992
2 44716   53.0539  138.1307 0001879   94.8328  265.2875
   15.06389896290091
```

Vengono fornite tre righe per ogni satellite: la prima contiene il nome, mentre le ultime due riportano le informazioni orbitali necessarie per calcolare la sua posizione. Nell'esempio 4.1 vengono spiegati tutti i campi relativi al satellite *STARLINK-1008* 4.1 [5] [6]:

| Campo          | Descrizione   |
|----------------|---|
| 1              | Numero della linea  |
| 44714          | Numero del satellite nel catalogo NORAD   |
| U              | Classificazione (U = Unclassified)  |
| 19074B         | Designatore internazionale: Anno di lancio (2019), 74° lancio, parte B del carico   |
| 25043.51880833 | Epoca indica il 43° giorno dell'anno 2025, cioè 12 febbraio 2025, alle 12:27 UTC (perché $0.5188 * 24 \approx$ 12 ore e 27 minuti). |
| -0.00000405    | Prima derivata del moto medio (decadimento orbitale)  |
| 00000+0        | Seconda derivata del moto medio (di solito 0)   |
| -82922-5       | Termine di drag   |
| 0              | Tipo di efemeride   |
| 9999           | Numero del set di elementi  |
| 2              | Numero della linea  |
| 44714          | Numero del satellite (deve corrispondere alla prima riga)   |
| 53.0560        | Inclinazione orbitale (gradi)   |
| 138.1627       | Ascensione retta del nodo ascendente (gradi)  |
| 0001487        | Eccentricità  |
| 82.4102        | Argomento del perigeo (gradi)   |
| 277.7055       | Anomalia media (gradi)  |
| 15.06377817    | Moto medio (rivoluzioni al giorno)  |
| 290117         | Numero di rivoluzione all'epoca   |

**Tabella 4.1.** Descrizione dei campi del TLE per STARLINK-1008

Ottenuti i dati, vengono inseriti come input nel modello SGP4, permettendo di generare istanze di oggetti satellitari, determinando la posizione di ciascun satellite in base a un utente o a una GS. Il primo passo è stato determinare quanti satelliti sorvolano una specifica area in un dato istante, i TLE restituiscono l'elenco dei satelliti attivi. La loro posizione è rappresentabile da un vettore tridimensionale con coordinate x, y e z, che definisce la posizione nello spazio, essa ha permesso di visualizzare graficamente la disposizione dei satelliti. Inoltre, utilizzando la formula Euclidea della distanza tridimensionale 4.1, è possibile calcolare le distanze tra i satelliti.

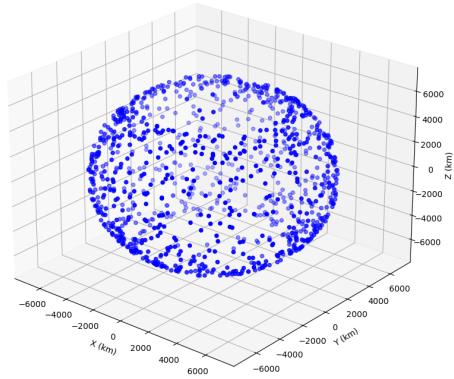
### Distanza Euclidea nello spazio Tridimensionale

Siano  $P_1$  e  $P_2$  i vettori che rappresentano la posizione di due satelliti nello spazio tridimensionale:

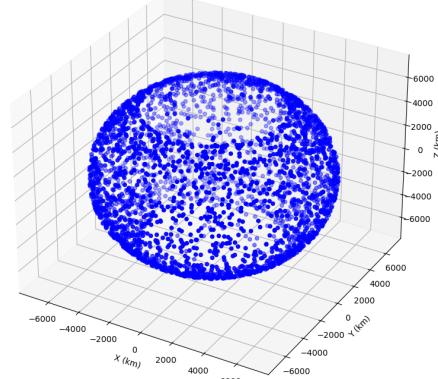
$$P_1(x_1, y_1, z_1) \quad \text{e} \quad P_2(x_2, y_2, z_2),$$

la distanza  $d$  tra i due punti è data da:

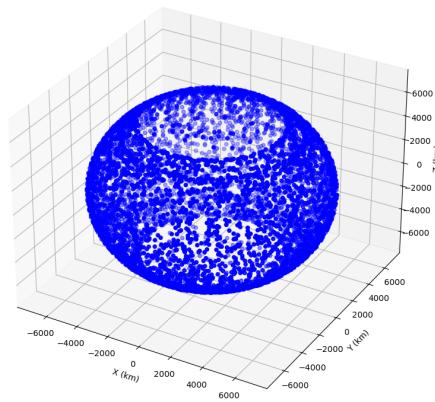
$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}. \quad (4.1)$$



**Figura 4.1.** Rappresentazione della copertura globale del 15% dei satelliti StarLink



**Figura 4.2.** Rappresentazione della copertura globale del 50% dei satelliti StarLink



**Figura 4.3.** Rappresentazione della copertura globale del 100% dei satelliti StarLink

A scopo dimostrativo, le figure 4.1, 4.2, 4.3 sono il risultato di stampa parziale della popolazione satellitare, incrementando la percentuale di satelliti caricati nel sistema, si osserva un aumento della loro copertura globale. Questo effetto consente di visualizzare in modo chiaro come, al crescere del numero di satelliti considerati, si arrivi a coprire l'intero globo.

#### 4.2.2 Observer e sistema di riferimento

Per definire la topologia, è stato necessario utilizzare l'istanza di osservazione del modello SGP4. L'**observer**, statico per tutta la durata della simulazione, richiede come input la latitudine e la longitudine in gradi. Ai fini della simulazione, si è scelto di adottare Roma come punto di riferimento, inserendo le coordinate  $41.8967^{\circ}$  N,  $12.4822^{\circ}$  E. Il sistema consente inoltre di specificare un valore per l'altitudine, utile nel caso in cui si vogliano considerare punti di osservazione situati a quote particolari.

## 4.3 Modellazione e simulazione della rete satellitare

### 4.3.1 Identificazione dei satelliti visibili

Per la costruzione della topologia, è stato fondamentale ridurre il numero di satelliti considerati. A tal fine, è stato calcolato l'angolo di elevazione  $\varphi_s$  di ciascun satellite rispetto all'osservatore, selezionando solo quelli con un angolo maggiore o uguale a  $\varphi_{max}$ <sup>3.1.3</sup>. Questo approccio ha permesso di ridurre drasticamente il numero di satelliti coinvolti, passando da oltre 6900 distribuiti globalmente a circa 25-30. La riduzione ha consentito un enorme risparmio di potenza computazionale, limitando il numero di oggetti caricati in memoria. Nella simulazione verranno quindi considerati solo i satelliti che, come mostrato in figura 4.4<sup>2</sup>, rientrano all'interno dell'area della cupola.



Figura 4.4. Zona di copertura della cupola con un angolo di ascolto di 40°

### 4.3.2 Definizione dell'insieme $S_i$

Per modellare e analizzare la rete satellitare, è necessario definire formalmente l'insieme dei satelliti disponibili in un dato istante di tempo. Indichiamo con  $S_i$  l'insieme di tutti i satelliti Starlink attivi osservabili all'istante  $i$ ; Formalmente possiamo esprimere questa definizione come segue:

$$S_i = \{s_k \mid \text{Starlink}(s_k) \wedge \text{Attivo}(s_k, i)\} \quad (4.2)$$

Dove :

- **Starlink( $s_k$ )** è una funzione che restituisce **vero** se  $s_k$  è un satellite Starlink.
- **Attivo( $s_k, i$ )** è una funzione che restituisce **vero** se il satellite  $s_k$  è attivo all'istante temporale  $i$ .

L'insieme  $S_i$  rappresenta il punto di partenza per la costruzione della topologia, in quanto contiene tutti i satelliti potenzialmente utilizzabili per stabilire connessioni

---

<sup>2</sup>Per la realizzazione dell'immagine è stato utilizzato il seguente sito che mostra il movimento dei satelliti in tempo reale [satellitemap.space](http://satellitemap.space)

all'interno della rete satellitare. Tuttavia, non tutti i satelliti in  $S_i$  contribuiscono effettivamente alla topologia in ogni istante, saranno selezionati solo quelli che soddisfano determinati criteri di visibilità e connessione.

### 4.3.3 Relazioni tra Satelliti e il concetto di Vicinato

Un aspetto cruciale nella modellazione della rete satellitare è la definizione delle relazioni di vicinanza tra i satelliti, che determinano quali di essi possono comunicare direttamente tra loro.

Assumiamo l'impiego della tecnologia LISLs già affrontata nelle sottosezione 2.4.2, per stabilire l'esistenza di una distanza massima  $r_{max}$  entro la quale due satelliti possono comunicare. Le relazioni tra satelliti all'interno della rete possono essere formalmente descritte come un insieme di coppie di satelliti connessi:

$$R_i = \{(s_h, s_k) \forall s_h, s_k \in S_i \mid \text{Distance}(s_h, s_k) \leq r_{max}\} \quad (4.3)$$

Dove:

- **Distance**( $s_h, s_k$ ) è una funzione che prende in input due satelliti e calcola la distanza utilizzando la formula Euclidea della distanza 4.1.

Questa formulazione consente di identificare la struttura della rete in ogni istante  $i$ , definendo come i satelliti formano effettivamente la topologia della rete e come possono essere coinvolti nelle operazioni di comunicazione e calcolo distribuito.

### 4.3.4 Topologia Statica

Per analizzare le interazioni tra i satelliti che, in un dato istante, sorvolano l'area dell'osservatore, definiamo un modello matematico che ne descrive la topologia.

Sia l'insieme  $S_i$  4.2 come l'insieme di tutti i satelliti disponibili, introduciamo  $T_i$  come il sottoinsieme di satelliti che costituiscono la **Topologia** nell'istante  $i$ , ovvero:

$$T_i = \{s_k \in S_i \mid \varphi_{s_k} \geq \varphi_{max}\} \quad (4.4)$$

In  $T_i$  sono inclusi tutti i satelliti Starlink attivi il cui angolo di elevazione  $\varphi_s$  è superiore alla soglia  $\varphi_{max}$ .

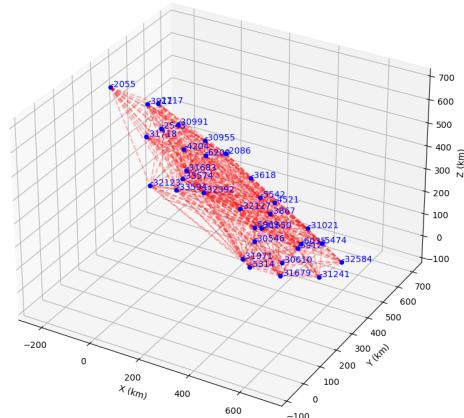
**Esempio di dati generati** I dati risultanti dal software sviluppato sono strutturati come segue:

Time: 2025-02-24T16:33:08Z

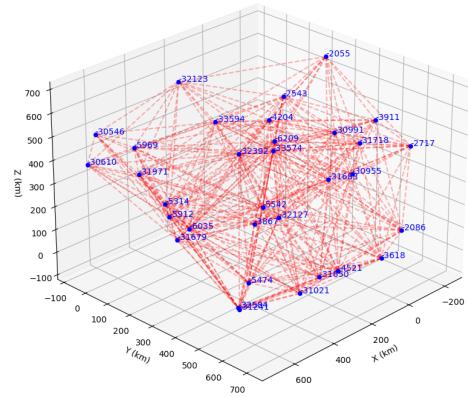
- |                               |                     |
|-------------------------------|---------------------|
| 0. Satellite: STARLINK-2055   | Elev: 45.20 degrees |
| 1. Satellite: STARLINK-2086   | Elev: 47.08 degrees |
| 2. Satellite: STARLINK-2543   | Elev: 60.86 degrees |
| 3. Satellite: STARLINK-2717   | Elev: 45.14 degrees |
| ...                           |                     |
| 29. Satellite: STARLINK-32392 | Elev: 78.52 degrees |
| 30. Satellite: STARLINK-32584 | Elev: 45.57 degrees |
| 31. Satellite: STARLINK-33594 | Elev: 61.74 degrees |
| 32. Satellite: STARLINK-33574 | Elev: 75.85 degrees |

**Listing 4.2.** Esempio di lista di satelliti visibili con relativa elevazione

**Visualizzazione della Topologia** Le figure 4.5 4.6 mostrano una rappresentazione della topologia da due prospettive differenti, generate a partire dai dati elencati nel Listing 4.2:



**Figura 4.5.** Prospettiva laterale della topologia



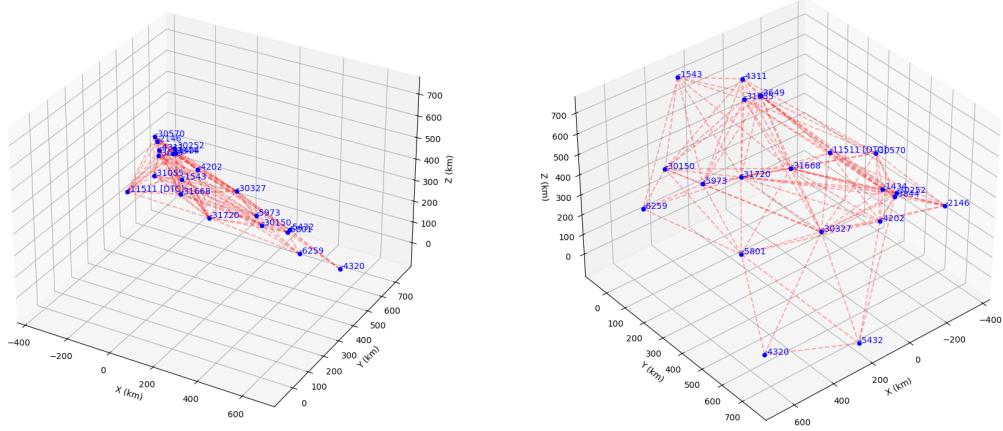
**Figura 4.6.** Prospettiva dallo zenith della topologia

Inoltre, nella Figura 4.5 è possibile osservare come i satelliti mostrati nel plot appartenenti alla fascia LEO, seguano la curvatura terrestre. Questo aspetto è una conseguenza della loro altitudine relativamente bassa rispetto alla superficie, che li mantiene in un'orbita compatta e vicina alla geometria del globo.

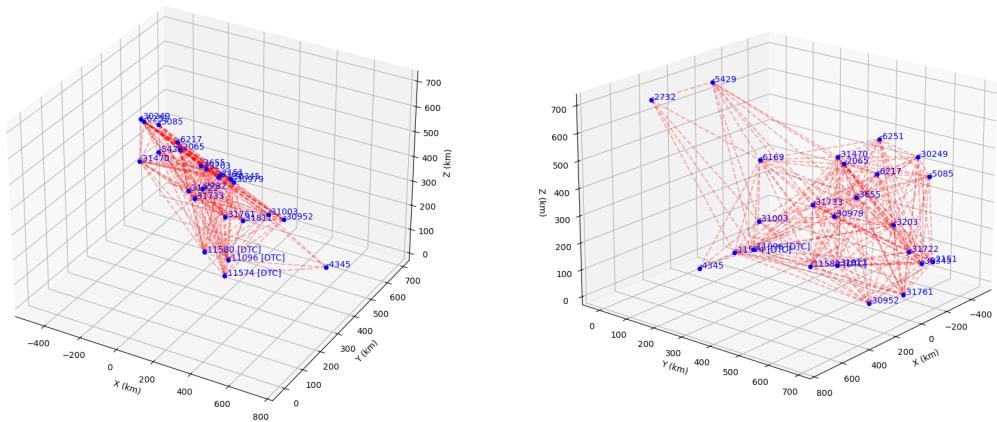
### Esempi di Topologia e Variazioni Temporali

A seconda dell'istante considerato, la topologia dei satelliti visibili può assumere forme differenti, con configurazioni che dipendono dalla distribuzione orbitale e dall'area osservata. Le seguenti figure mostrano esempi di topologie osservate in diversi momenti, evidenziando come la disposizione dei satelliti cambi nel tempo:

Istante di generazione: 2025-02-24T18:30:00Z

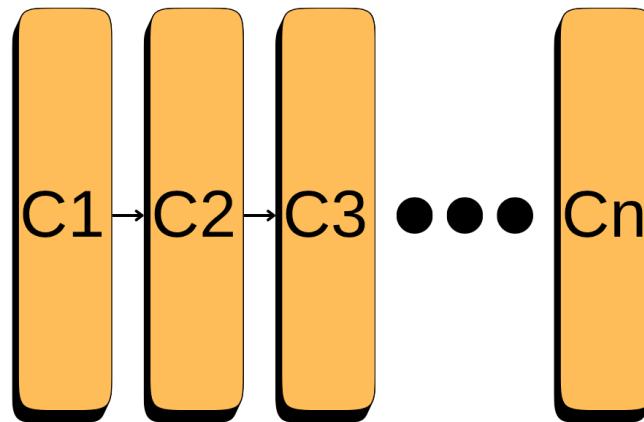


Istante di generazione: 2025-02-24T18:40:00Z



## 4.4 Topologia Dinamica

Durante lo sviluppo del software, si è presentata la necessità di simulare accuratamente i movimenti dei satelliti nello spazio. Da qui nasce il concetto di **Multi-configuration**, che rielabora il concetto di topologia come una sequenza di configurazioni indipendenti. Questa reinterpretazione è fondamentale poiché consente di modellare il moto dei satelliti come una successione di **frame** temporali, analogamente ai fotogrammi di un video, dove il rapido susseguirsi delle immagini genera la percezione del movimento.



**Figura 4.7.** Rappresentazione dell’insieme di tutte le configurazioni

Definiamo due istanti temporali,  $i$  e  $f$ , che delimitano l’intervallo temporale di simulazione, introduciamo inoltre  $\Delta t$  parametro che rappresenta l’incremento temporale con cui avanzare da  $i$  a  $f$ , il numero totale di configurazioni necessarie per coprire l’intero intervallo è dato da:

$$N = \left\lfloor \frac{f - i}{\Delta t} \right\rfloor \quad (4.5)$$

Introduciamo  $SAT_j$  l’insieme di tutti i satelliti Starlink attivi, definiamo quindi il sottoinsieme  $C_j \subseteq SAT_j$  come l’insieme dei satelliti che costituiscono la configurazione topologica, nel modo seguente:

$$C_j = \{s_k \in SAT_j \mid \varphi_{s_k} \geq \varphi_{\max}\} \quad (4.6)$$

dove  $j$  non rappresenta un singolo istante, ma l'indice della configurazione determinato matematicamente come:

$$j = \left\lfloor \frac{t}{\Delta t} \right\rfloor \quad (4.7)$$

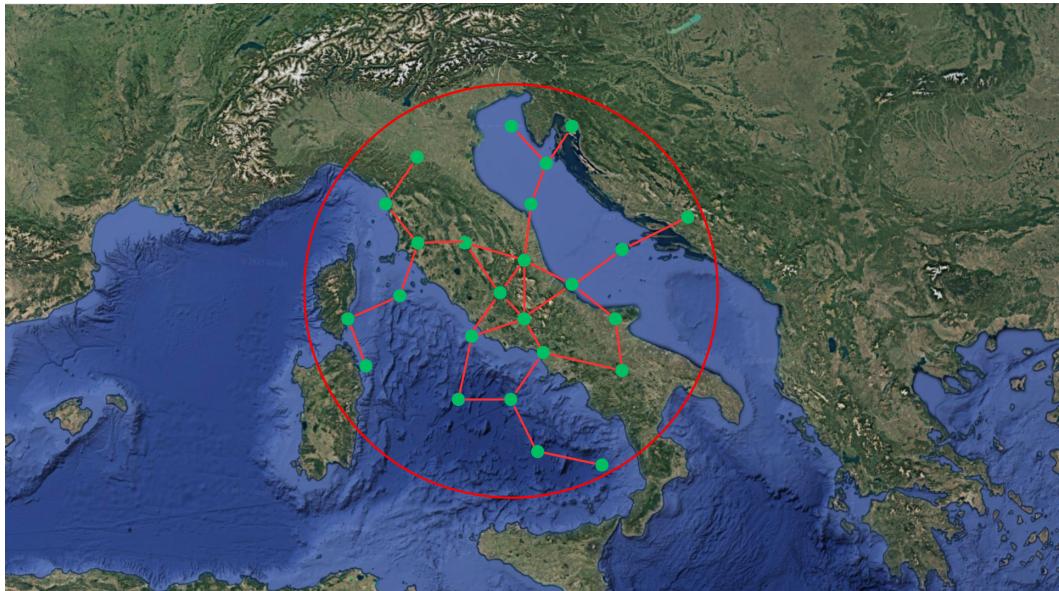
$t_i$  è l'istante temporale normalizzato rispetto all'intervallo di aggiornamento  $\Delta t$  (ad esempio, 30 secondi). Questo metodo permette di individuare l'insieme dei satelliti che, in un dato istante  $t_i$ , garantiscono copertura alla GS. Durante la simulazione, è sufficiente caricare la configurazione corrispondente all'indice  $i$ , che dipende dal tempo del simulatore consentendo di avanzare con passi di ampiezza  $\Delta t$ .

#### 4.4.1 Neighborhood

Riprendendo i concetti introdotti nella sottosezione 4.3.3, li contestualizziamo ora rispetto alla singola area di copertura dell'utente, denominata cupola. Le relazioni intra-satellitari definite nell'insieme 4.3 vengono modificate come segue:

$$R_j = \{(s_h, s_k) \mid \forall s_h, s_k \in SAT_j, \varphi_{s_h} \geq \varphi_{max}, \varphi_{s_k} \geq \varphi_{max}, \text{Distance}(s_h, s_k) \leq r_{max}\} \quad (4.8)$$

Questo insieme consente di modellare le relazioni che i satelliti instaurano all'interno della cupola dell'osservatore.



**Figura 4.8.** Rappresentazione delle relazioni intra-satellitari

Nel contesto della simulazione, tali relazioni evolvono dinamicamente, modificandosi di configurazione in configurazione. Questo aspetto consente di elevare la simulazione dello scambio di task a un livello superiore: si possono osservare situazioni in cui alcuni satelliti, inizialmente centrali nella topologia, perdono progressivamente questo ruolo, riducendo il traffico dati gestito. Di conseguenza, le connessioni tra satelliti

non sono statiche, ma si ridefiniscono nel tempo. Possiamo dunque categorizzare i vicini che un satellite incontra nel corso della sua operatività in base alle tipologie già descritte nella sottosezione 2.4.2.

**Gestione della Latenza** Per una gestione efficace delle comunicazioni tra satelliti, è fondamentale disporre delle informazioni relative ai loro vettori di posizione e alle distanze reciproche in ogni istante.

In questa nuova definizione, non ci limitiamo a rappresentare le relazioni come coppie  $(s_h, s_k)$ , ma, estendiamo la struttura includendo anche la latenza di comunicazione, che dipende direttamente dalla distanza tra i satelliti. Di conseguenza, ogni relazione viene descritta da una tupla contenente i due satelliti e la latenza stimata.

La latenza può essere calcolata assumendo che il segnale viaggi alla velocità della luce nel vuoto ( $c = 299,792,458 \text{ m/s}$ ).

---

**Algorithm 1** Calcolo della latenza basata sulla distanza

---

**Require:**  $d$ : distanza tra i satelliti in km

**Ensure:**  $\lambda$ : latenza in secondi

- |   |                              |
|---|------------------------------|
| 1: $c \leftarrow 299792458$                           | ▷ Velocità della luce in m/s |
| 2: $d_m \leftarrow d \times 1000$                     | ▷ Conversione da km a metri  |
| 3: $\lambda \leftarrow d_m/c$ <b>return</b> $\lambda$ |                              |
- 

La relazione tra satelliti viene ridefinita come segue:

$$\begin{aligned}
 R_j = \{(s_h, s_k, \lambda) \mid & s_h, s_k \in SAT_j \mid \varphi_{s_h} \geq \varphi_{\max}, \varphi_{s_k} \geq \varphi_{\max}, \\
 & \text{Distance}(s_h, s_k) \leq r_{\max}, \\
 & \lambda = \frac{\text{Distance}(s_h, s_k)}{c}\}
 \end{aligned} \tag{4.9}$$

Dove  $\lambda$  rappresenta la latenza in secondi, calcolata in base alla distanza tra i satelliti, valore che verrà preso in considerazione dalle funzioni viste nel DTS-TMAX nella sezione 3.3.2.

#### 4.4.2 AP-SAT

Prima di poter effettuare una simulazione è necessario ridefinire la scelta degli AP-SAT, ovvero i punti di accesso attraverso i quali i task possono entrare nella rete satellitare. A differenza del modello adottato nella ricerca presentata nella sottosezione 3.3.1, si è deciso di selezionare gli access point seguendo un criterio differente.

L'insieme dei satelliti nella cupola visibile dall'utente è definito come  $T_j$ , tenendo conto esclusivamente quelli che si stanno avvicinando all'osservatore, escludendo quelli che hanno già superato la loro **Max-El**. Questo approccio permette di evitare di selezionare satelliti che stanno già allontanandosi e che dunque avrebbero un tempo di visibilità residuo ridotto. Per determinare se un satellite si sta avvicinando o allontanando, si sfrutta la conoscenza della sua velocità relativa all'osservatore. Il seguente pseudocodice implementa il filtro per selezionare solo i satelliti che si stanno muovendo nella direzione dell'utente:

---

**Algorithm 2** Filtraggio dei satelliti in avvicinamento

---

**Require:** Satellite  $s$ , Istante  $t$

**Ensure:** Booleano: **true** se il satellite si avvicina, **false** altrimenti

```

1:  $sys \leftarrow getSystemFromSat(s, t)$   $\triangleright$  Ottieni il sistema di riferimento del satellite  
rispetto all'osservatore a tempo  $t$ 
2:  $v \leftarrow sys.velocity$   $\triangleright$  Velocità del satellite
3:  $r \leftarrow sys.position.km$   $\triangleright$  Posizione relativa del satellite rispetto all'osservatore
4:  $r_{unit} \leftarrow r / \|r\|$   $\triangleright$  Calcolo del vettore unitario della direzione di  $r$ 
5:  $v_{rel} \leftarrow v \cdot r_{unit}$   $\triangleright$  Prodotto scalare tra velocità e direzione di  $r$ 
6: if  $v_{rel} < 0$  then
7:   return true  $\triangleright$  Il satellite si sta avvicinando
8: else
9:   return false  $\triangleright$  Il satellite si sta allontanando
10: end if

```

---

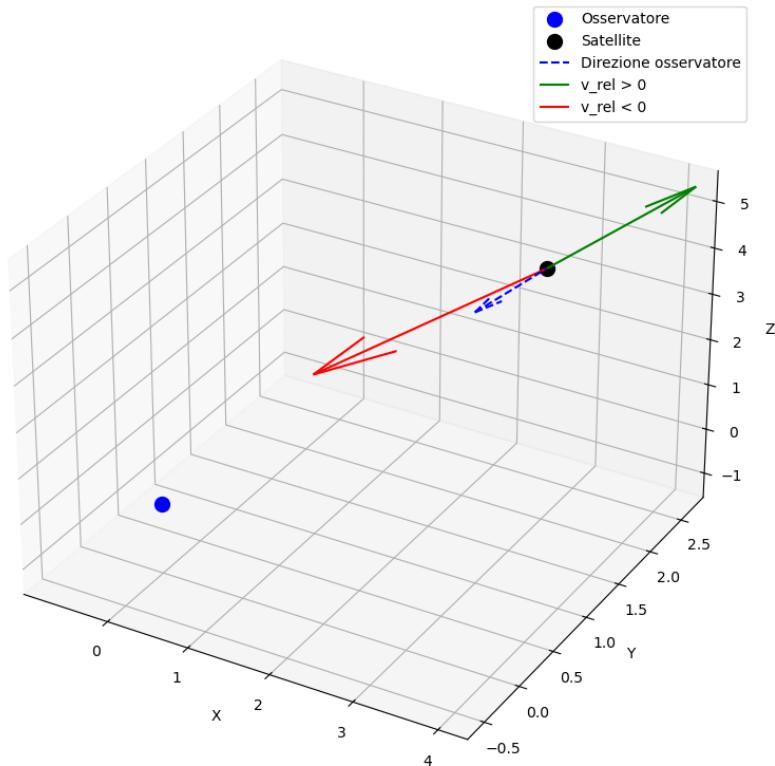
Il ruolo del vettore unitario  $r_{unit}$  è fondamentale per determinare la direzione relativa del satellite rispetto all'osservatore. Esso si ottiene normalizzando il vettore posizione  $r$ , dividendo ciascuna delle sue componenti per la sua norma (modulo):

$$r_{unit} = \frac{r}{\|r\|} \quad (4.10)$$

Questo passaggio serve a ottenere un vettore con la stessa direzione di  $r$  ma con modulo unitario, utile per calcolare la proiezione della velocità lungo questa direzione. Il valore  $v_{rel}$  viene ottenuto tramite il prodotto scalare tra la velocità del satellite e il vettore unitario  $r_{unit}$ :

$$v_{rel} = \mathbf{v} \cdot \mathbf{r}_{unit} \quad (4.11)$$

Il prodotto scalare restituisce la componente della velocità del satellite nella direzione di  $r_{unit}$ . Se  $v_{rel}$  è negativo, come mostrato in figura 4.9 dal vettore rosso, significa che il satellite si sta avvicinando all'osservatore. Viceversa, se  $v_{rel}$  è positivo, colore verde, il satellite si sta allontanando. Questo criterio consente di selezionare dinamicamente i satelliti che rappresentano validi candidati per fungere da *access point*.



**Figura 4.9.** Illustrazione della componente  $v_{rel}$  di un satellite rispetto ad un Osservatore.

### Gestione dei casi particolari

Nel processo di creazione della topologia, sono emersi alcuni casi particolari che hanno richiesto un'approfondimento. Tutti i calcoli relativi alla costruzione della rete satellitare sono stati effettuati assumendo come riferimento geografico la città di Roma, garantendo così una copertura completa per l'osservatore. Tuttavia, questa condizione non è universalmente valida: non tutte le aree geografiche presentano una copertura satellitare tale da assicurare elevate prestazioni nello scambio dei task.

In particolare, posizionando l'osservatore in prossimità dei poli, emergono criticità dovute alla configurazione orbitale della costellazione. Come illustrato nella figura 4.10 di riferimento, ponendo l'osservatore a Oulu, in Finlandia ( $65.0121^\circ$  N,  $25.4651^\circ$  E), si osserva una significativa riduzione della copertura, fino al punto in cui non è possibile assegnare il numero minimo di *access point* richiesti per la simulazione.

Per affrontare questa problematica, è stato adottato un criterio adattivo nella selezione degli *access point*. Se il filtro implementato nell'algoritmo 2 restituisce un numero di access point inferiore alla soglia minima necessaria per la simulazione, si procede con una riduzione adattativa, dimezzando la quantità di access point richiesti. Questo approccio, garantisce, la possibilità di eseguire la simulazione anche in condizioni di copertura limitata, pur tenendo conto delle conseguenti implicazioni sulle prestazioni del sistema.

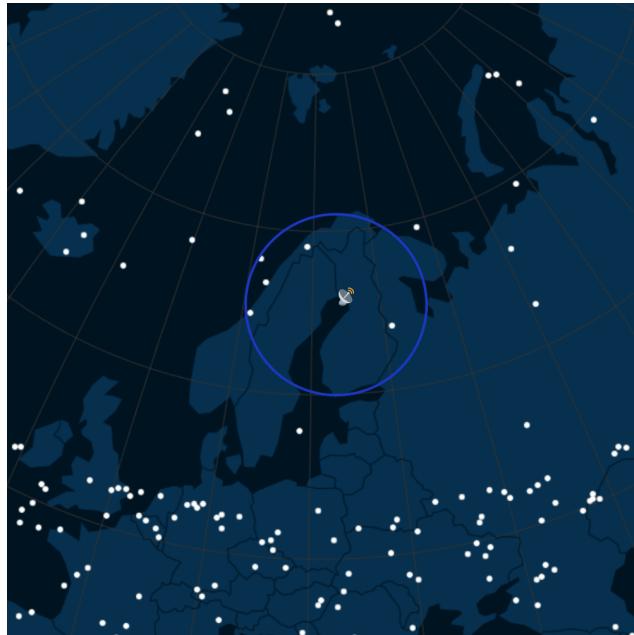
---

#### **Algorithm 3** Verifica e adattamento del numero di access point

---

**Require:**  $T_i, n\_Access\_Points$

- 1: **if**  $n\_Access\_Points > |T_i|$  **then**
  - 2:      $n\_Access\_Points \leftarrow \lfloor |T_i|/2 \rfloor$                $\triangleright$  Aggiorna il numero di Access Point
  - 3: **end if**
  - 4:  $selectedAccessPoints \leftarrow$  seleziona i primi  $n\_Access\_Points$  elementi da  $T_i$        $\triangleright$   
Seleziona gli Access Point richiesti
  - 5: **return**  $selectedAccessPoints$
-



**Figura 4.10.** Illustrazione del caso particolare centrato in Oulu, Finlandia.

#### 4.4.3 Buffer Zone

Per migliorare la rappresentazione delle relazioni tra satelliti nella simulazione, si è introdotta una zona di buffer che estende l'area di analisi oltre la cupola principale. Dato che i satelliti sono in costante movimento, diventa fondamentale studiare il loro comportamento anche nelle regioni limite della cupola. Non è realistico assumere che le interazioni tra satelliti si verifichino esclusivamente all'interno della cupola sopra l'osservatore, come mostrato dall'insieme 4.3 i satelliti a livello globale sono interconnessi in ogni momento.

L'inclusione di questa zona cuscinetto, che prende il nome di "Buffer", consente di modellare in modo più affidabile le interazioni inter-satellitari. Tuttavia, i satelliti che transitano nella zona di buffer senza entrare nella cupola principale non assumono un ruolo primario nella simulazione. Piuttosto, vengono considerati come elementi di supporto per analizzare la redistribuzione dei task da parte dell'algoritmo decentralizzato.

L'adozione di questo approccio ha portato a un aumento significativo del numero di satelliti considerati nella simulazione: da un intervallo iniziale di 25-30 satelliti si è passati a 60-65 contemporaneamente in orbita sopra l'area di studio.

Si mantiene un valore massimo di elevazione della cupola pari a  $\varphi_{max} = 40^\circ$ , e si introduce un intervallo supplementare  $\varphi_{buffer} = 15^\circ$ . Di conseguenza, vengono considerati tutti i satelliti con angolo di elevazione superiore alla soglia  $\varphi_{dome}$ , definita nella formula 4.12.

**Algoritmo di Selezione Satelliti** Introduzione dell'algoritmo `getAllSatOnMe`, algoritmo 4, con lo scopo di individuare e selezionare i satelliti visibili sopra una determinata posizione di osservazione. L'input dell'algoritmo include l'angolo di elevazione massima  $\varphi_{max}$ , una soglia di buffer  $\varphi_{buffer}$  per definire una seconda lista di satelliti e il numero di punti di accesso richiesti  $Num_{Access\_point}$ .

L'algoritmo processa i dati orbitali in formato TLE (Two-Line Elements) per calcolare la posizione e l'altitudine dei satelliti rispetto all'osservatore. Ogni satellite viene classificato in due liste:

- *satellites\_dome*: contiene i satelliti con altitudine superiore alla soglia  $\varphi_{max}$ .
- *satellites\_buffer*: contiene i satelliti con altitudine compresa tra  $\varphi_{dome}$  e  $\varphi_{max}$ .

Entrambe le liste vengono ordinate in base alla distanza della GS rispetto al satellite. Successivamente, l'algoritmo verifica se il numero di punti di accesso richiesti supera la disponibilità di satelliti nella lista principale. In tal caso, il numero di punti di accesso viene ridotto della metà per garantire una selezione fattibile.

Infine, i satelliti vengono iterati e filtrati attraverso una funzione che verifica se i satelliti si stanno avvicinando all'osservatore o meno, inserendoli nella lista finale degli `acc_point` oppure nella lista di `dome`. La funzione `filterSatellitesInView` viene rivista meglio nella sotto sezione 5. Il risultato finale è composto da tre insiemi di dati:

- *acc\_points*: i satelliti selezionati come punti di accesso nell'istante  $i$ .
- *dome*: i satelliti presenti nella cupola all'istante  $i$ .
- *sat\_sort\_buff*: i satelliti presenti nella zona di buffer nell'istante  $i$ .

---

**Algorithm 4** getAllSatOnMe

---

**Input:**  $t, \varphi_{max}, \varphi_{buffer}, NumAccess\_point$   
**Output:**  $acc\_points, dome, sat\_sort\_buff$

```

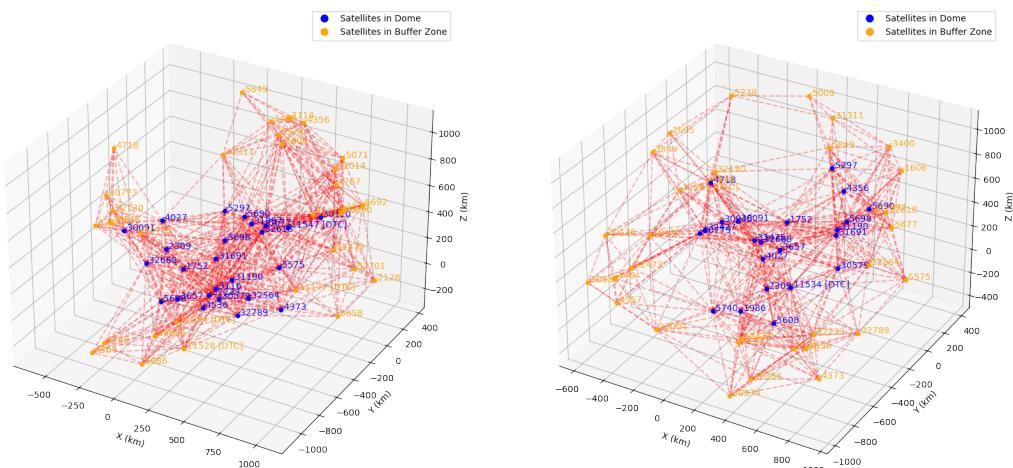
 $\varphi_{dome} \leftarrow \varphi_{max} - \varphi_{buffer}$ 
 $satellites\_dome \leftarrow [], satellites\_buffer \leftarrow []$ 
 $tle\_data \leftarrow TLE\_DATA$ 
for  $i \leftarrow 0$  to  $len(tle\_data)$  step 3 do
     $name \leftarrow tle\_data[i]$ 
     $line1 \leftarrow tle\_data[i + 1]$ 
     $line2 \leftarrow tle\_data[i + 2]$ 
     $satellite \leftarrow EarthSatellite(line1, line2, name, ts)$ 
     $syst \leftarrow getSystemFromSat(satellite, t)$ 
     $alt, az, distance \leftarrow syst.altaz()$ 
    if  $alt.degrees > \varphi_{dome}$  then
        if  $alt.degrees > \varphi_{max}$  then
             $satellites\_dome.append((satellite, distance.km))$ 
        else
             $satellites\_buffer.append((satellite, distance.km))$ 
        end if
    end if
    end for
 $sat\_sort\_dome \leftarrow sorted(satellites\_dome, key = lambda x: x[1])$ 
 $sat\_sort\_buff \leftarrow sorted(satellites\_buffer, key = lambda x: x[1])$ 
 $counter \leftarrow 0, acc\_points \leftarrow [], dome \leftarrow []$ 
if  $NumAccess\_point > len(sat\_sort\_dome)$  then
     $NumAccess\_point \leftarrow len(sat\_sort\_dome) // 2$ 
    print("WARNING: Not enough satellites to satisfy the request.")
end if
for  $s \in sat\_sort\_dome$  do
    if  $counter < NumAccess\_point \wedge filterSatellitesInView(s[0], t)$  then
         $acc\_points.append(s)$ 
         $counter \leftarrow counter + 1$ 
    else
         $dome.append(s)$ 
    end if
end for
return  $acc\_points, dome, sat\_sort\_buff$ 

```

---

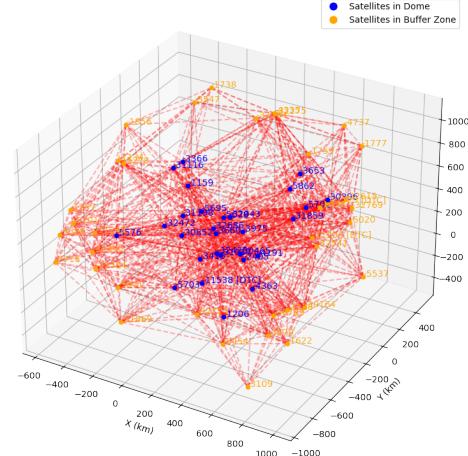


**Figura 4.11.** Rappresentazione della zona di Buffer



**Figura 4.12.** Topologia completa Starlink istante 1

**Figura 4.13.** Topologia completa Starlink istante 2



**Figura 4.14.** Topologia completa Starlink istante 3

$$\varphi_{dome} = \varphi_{max} - \varphi_{buffer} \quad (4.12)$$

L'insieme dei satelliti inclusi nella topologia è dunque cambiato, e viene ridefinito come segue:

$$T_j = \{s_k \in S_j \mid \varphi_{s_k} \geq \varphi_{dome}\} \quad (4.13)$$

$$R_j = \{(s_h, s_k) \mid s_h, s_k \in S_j, \varphi_{s_h} \geq \varphi_{dome}, \varphi_{s_k} \geq \varphi_{dome}, \text{Distance}(s_h, s_k) \leq r_{max}\} \quad (4.14)$$

Nei plot mostrati nelle figure 4.12, 4.13 e 4.4.3 è rappresentata l'evoluzione della topologia, comprensiva della zona di buffer, in tre distinti istanti temporali. Si osserva come la struttura della topologia si modifichi nel tempo.

## 4.5 Dalla Teoria alla Computazione: Elaborazione dei Dati

In questa sezione analizzeremo quali calcoli vengono effettuati a livello computazionale.

### 4.5.1 Pre-elaborazione dei Dati

Durante la fase di simulazione, è emerso che, nonostante il numero di satelliti da considerare fosse circa 60 per ogni configurazione (dato che includiamo quelli nella Dome e nella Bufferzone), i calcoli eseguiti in tempo reale risultavano comunque troppo onerosi.

Per ottimizzare il processo, si è deciso di precalcolare le posizioni dei satelliti e le relative metriche, come la distanza dagli utenti e la latenza. Questo approccio permette di caricare i dati direttamente dalla memoria al momento del bisogno, riducendo il carico computazionale.

Si è scelto di utilizzare un file JSON per memorizzare tutte le informazioni necessarie. Il file è strutturato in due sezioni principali:

- Una sezione generale che contiene informazioni di base sulla simulazione.
- Una sezione dedicata alle configurazioni satellitari.

La prima sezione del JSON contiene informazioni come il tempo iniziale, l'intervallo di simulazione, la durata totale e la posizione dell'osservatore:

```
{  
    "t0": "<inizio_simulazione>",  
    "interval": <intervallo_secondi>,  
    "total_seconds": <durata_simulazione>,  
    "observer_position": "<posizione_osservatore>",  
    "configurations": [<lista_configurazioni>]  
}
```

Dove:

- **t0** indica il tempo iniziale della simulazione.
- **interval** rappresenta il passo di avanzamento  $\Delta t$  in secondi.
- **total\_seconds** è la durata complessiva della simulazione.
- **observer\_position** indica la posizione geografica da cui vengono effettuati i calcoli.

La seconda sezione del JSON contiene le configurazioni satellitari. Ogni configurazione è un dizionario che include:

```
{
    "time": "<tempo_configurazione>",
    "configuration": [
        {
            "satellite": "<nome_satellite>",
            "distance_from_user": <distanza>,
            "is_access_point": <true/false>,
            "elev_angle": <angolo_elevazione>,
            "TLE-DATA": [
                {
                    "name": "<nome_satellite>",
                    "line1": "<TLE_linea1>",
                    "line2": "<TLE_linea2>"
                }
            ],
            "life": {
                "AOS": "<timestamp_inizio_visibilità>",
                "Max-El": "<timestamp_massima_elevazione>",
                "LOS": "<timestamp_fine_visibilità>",
                "life_seconds": <durata_visibilità>,
                "time_until_set_seconds": <tempo_alla_perdita_di_segnale>
            },
            "neighbors": [
                {
                    "name": "<nome_satellite_vicino>",
                    "distance": <distanza>,
                    "latency": <latenza>
                },
                {
                    "name": "<nome_altro_satellite>",
                    "distance": <distanza>,
                    "latency": <latenza>
                }
            ]
        }
    ]
}
```

In questo modo, le informazioni necessarie vengono caricate direttamente dal file JSON senza dover essere ricalcolate in tempo reale, migliorando così l'efficienza della simulazione.

### 4.5.2 Intersezioni delle Configurazioni

Il modello di riferimento discusso nella Sezione 4.4 presenta alcune lacune che devono essere colmate. In particolare, non è sufficiente passare da una configurazione all'altra per dare l'impressione che i satelliti si stiano effettivamente muovendo. Per affrontare questa problematica, è stato sviluppato un metodo per determinare l'intersezione tra due configurazioni consecutive.

Introduciamo **Servers** l'insieme che include tutti i satelliti che sono stati almeno una volta considerati parte della rete topologica. All'avvio della simulazione viene caricata la configurazione iniziale (configurazione 0), l'insieme **Servers** viene popolato con tutti i satelliti che, nell'istante definito da tale configurazione, sorvolano l'area di interesse.

Il problema principale emerge nel passaggio tra configurazioni successive ( $i \rightarrow i + 1$  con  $i > 0$ ), nel quale si possono individuare tre sottoinsiemi distinti:

- **New\_server**: comprende i satelliti che non erano presenti nella configurazione  $i$ , e che appaiono nella configurazione  $i + 1$ . Poiché questi satelliti sono appena entrati nel sistema, non richiedono aggiornamenti. Essi verranno inclusi nella topologia a partire dalla configurazione  $i + 1$  e saranno disponibili per servire la rete nell'esecuzione dei task.
- **Intersection**: include i satelliti presenti nelle configurazioni  $i$  e  $i + 1$ . I satelliti presenti in questo insieme svolgono un ruolo cruciale nella rete, poiché centrali nella topologia tendono a sostenere un carico elevato. I passaggi per gestire questo insieme vengono affrontati nell'algoritmo 5, dove:
  - `elev_angle` : Angolo di elevazione del satellite rispetto l'osservatore.
  - `orbital_sunset` : tempo rimanente prima del raggiungimento della condizione di LOS.
  - `is_acc_point` : Flag che indica se un satellite funge da access point.
- **Old\_server**: rappresenta l'insieme dei satelliti che facevano parte della configurazione  $i$  ma che non sono più presenti nella configurazione  $i + 1$ . Questi satelliti non richiedono ulteriori aggiornamenti, in quanto non sono più considerati attivi nel sistema.

---

**Algorithm 5** Aggiornamento Server Intersection

---

```
Input: old_servers, new_servers, acc_point
Output: intersection                                ▷ Dizionari per i risultati
intersection ← {name : server | name ∈ old_servers ∧ name ∈ new_servers} ▷
Aggiornamento attributi dell'intersezione
for (name, server) ∈ intersection do
    server.elev_angle ← new_servers[name].elev_angle
    server.orbitalSunset ← new_servers[name].orbitalSunset
    if name ∈ acc_point then                      ▷ Aggiornamento flag acc_point
        server.is_acc_point ← True
    else
        server.is_acc_point ← False
    end if
end for
```

---

Questo approccio consente una gestione più efficace delle transizioni tra configurazioni, garantendo che i satelliti vengano correttamente aggiornati e mantenendo coerente la topologia del sistema.



# Capitolo 5

## Conclusioni

La presente ricerca ha esplorato il tema del Satellite Edge Computing (SEC), con particolare attenzione alla modellazione e simulazione del processo di handover. L'obiettivo è stato definire un modello capace di ricostruire con precisione la struttura dinamica della rete satellitare, abilitando la simulazione dello scambio di dati tra nodi nel contesto dell'Edge Computing.

### 5.1 Innovatività della ricerca

La soluzione proposta integra dati orbitali in formato Two-Line Elements (TLE) e impiega il modello di propagazione SGP4 per stimare con accuratezza la posizione e il movimento dei satelliti. La metodologia adottata ha permesso la costruzione di una topologia dinamica, capace di adattarsi alla continua evoluzione delle relazioni tra i satelliti, ottimizzando il processo di selezione dei nodi per la comunicazione e la distribuzione del carico computazionale.

Uno dei principali contributi di questa ricerca è stato lo sviluppo di una topologia basata sui reali spostamenti orbitali dei satelliti e sulla loro posizione rispetto all'osservatore. L'introduzione della zona di buffer e del concetto di vicinato satellitare ha migliorato la rappresentazione delle relazioni di prossimità tra i nodi, consentendo una modellazione più accurata dell'interazione tra i satelliti.

La transizione da un modello concettuale, in cui ogni nodo aveva al massimo quattro vicini, a un modello basato su un numero variabile di vicini, ha permesso di ampliare il numero di nodi con cui instaurare uno scambio di dati.

I risultati ottenuti hanno dimostrato che il modello è in grado di rappresentare con precisione la topologia di rete in diversi contesti orbitali, fornendo una piattaforma affidabile per la simulazione di scenari realistici di Satellite Edge Computing. Tut-

tavia, la complessità del sistema impone alcune limitazioni, come l'incremento del carico computazionale in presenza di costellazioni di grandi dimensioni.

## 5.2 Sviluppi futuri

Le prospettive future di questa ricerca prevedono l'estensione della simulazione a costellazioni di tipo Mega-constellation Networks (MCNs), con l'obiettivo di valutare l'efficienza del modello su larga scala e di migliorare l'ottimizzazione delle risorse computazionali distribuite.

Questo lavoro rappresenta un primo passo verso la progettazione di infrastrutture satellitari intelligenti, capaci di supportare le esigenze emergenti nel contesto dell'Internet of Things (IoT) e delle comunicazioni globali, offrendo soluzioni innovative per la gestione del traffico dati nello spazio.



# Bibliografia

- [1] David Vallado and Paul Crawford. *SGP4 Orbit Determination*.
- [2] David Vallado, Paul Crawford, Ricahrd Hujak, and T.S. Kelso. *Revisiting Spacetrack Report #3*.
- [3] European Space Agency. Eduspace: Le orbite dei satelliti. [https://www.esa.int/SPECIALS/Eduspace\\_IT/SEMRCWZRAOG\\_0.html](https://www.esa.int/SPECIALS/Eduspace_IT/SEMRCWZRAOG_0.html), 2025. Accessed on January 22, 2025.
- [4] European Space Agency. Types of orbits. [https://www.esa.int/Enabling\\_Support/Space\\_Transportation/Types\\_of\\_orbits](https://www.esa.int/Enabling_Support/Space_Transportation/Types_of_orbits), 2025. Accessed on January 27, 2025.
- [5] T.S. Kelso. Tle format documentation, 2025. Accessed: 2025-02-15.
- [6] Wikipedia contributors. Two-line element set, 2025. Accessed: 2025-02-15.
- [7] Emiliano Casalicchio and Danilo Magliarisi. Decentralized task scheduling in satellite edge computing. In *2024 9th International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 154–161, 2024.
- [8] Akram Al-Hourani. A tractable approach for predicting pass duration in dense satellite networks. *IEEE Communications Letters*, 25(8):2698–2702, 2021.
- [9] Shkelzen Cakaj. The parameters comparison of the “starlink” leo satellites constellation for different orbital shells. *Frontiers in Communications and Networks*, 2, 2021.
- [10] Aizaz U. Chaudhry and Halim Yanikomeroglu. Laser intersatellite links in a starlink constellation: A classification and analysis. *IEEE Vehicular Technology Magazine*, 16(2):48–56, 2021.
- [11] Quan Chen, Giovanni Giambene, Lei Yang, Chengguang Fan, and Xiaoqian Chen. Analysis of inter-satellite link paths for leo mega-constellation networks. *IEEE Transactions on Vehicular Technology*, 70(3):2743–2755, 2021.

- [12] Ruoqi Deng, Boya Di, Hongliang Zhang, Linling Kuang, and Lingyang Song. Ultra-dense leo satellite constellations: How many leo satellites do we need? *IEEE Transactions on Wireless Communications*, 20(8):4843–4857, 2021.
- [13] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. Mendoza Montoya, J. C. Merlano Duncan, D. Spano, S. Chatzinotas, S. Kisseeleff, J. Que-rol, L. Lei, T. X. Vu, and G. Goussetis. Satellite communications in the new space era: A survey and future challenges, 2020.
- [14] Israel Leyva-Mayorga, Beatriz Soret, Maik Röper, Dirk Wübben, Bho Matthiesen, Armin Dekorsy, and Petar Popovski. Leo small-satellite constellations for 5g and beyond-5g communications. *IEEE Access*, 8:184955–184964, 01 2020.
- [15] Angelo Podda, Sergio Casu, Antonio Coppola, Fabio Protopapa, Andrea Lazzareschi Sergiusti, Tonino Pisanu, Enrico Urru, Luca Schirru, Pierluigi Ortu, Francesco Gaudiomonte, Germano Bianchi, Claudio Bortolotti, Mauro Roma, Giuseppe Pupillo, Luca Lama, Federico Perini, Marco Schiaffino, Andrea Maccaferri, Giovanni Naldi, Andrea Mattana, Pierluigi di Lizia, Giovanni Purpura, and Mauro Massari. Exploitation of bi-static radar architectures for leo space debris surveying and tracking: the birales/blralet project. In *2020 IEEE Radar Conference (RadarConf20)*, pages 1–6, 2020.
- [16] Yanbo Wei, Huaijian Li, and Xiaojing Du. An efficient leo global navigation constellation design based on walker constellation. In *2020 IEEE Computing, Communications and IoT Applications (ComComAp)*, pages 1–6, 2020.
- [17] Spacex satellite internet project status. <http://cis471.blogspot.com/2017/08/spacex-satellite-internet-project-status.html>, 2017. Accessed: 2025-01-29.
- [18] Blesson Varghese, Nan Wang, Sakil Barbuiya, Peter Kilpatrick, and Dimitrios S. Nikolopoulos. Challenges and opportunities in edge computing. In *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, pages 20–26, 2016.
- [19] MARIO COMINI. Orbit determination with the simplified general perturbation model. 2015.
- [20] Shkelzen Cakaj, Bexhet Kamo, Vladi Koliçi, and Olimpjon Shurdi. The range and horizon plane simulation for ground stations of low earth orbiting (leo) satellites. *Int. J. Commun. Netw. Syst. Sci.*, 4(9):585–589, 2011.
- [21] Wei Dong and Zhao Chang-yin. An accuracy analysis of the sgp4/sdp4 model. *Chinese Astronomy and Astrophysics*, 34(1):69–76, 2010.

- [22] R. Wang, J. Liu, and Q.M. Zhang. Propagation errors analysis of tle data. *Advances in Space Research*, 43(7):1065–1069, 2009.
- [23] D. C. Beste. Design of satellite constellations for optimal continuous coverage. *IEEE Transactions on Aerospace and Electronic Systems*, AES-14(3):466–473, 1978.