CHAPTER I
PROPOSAL OF IDEAS

1.1 General Description of the problem

Currently, Low Earth Orbit (LEO) is one of the most popular orbits for operating satellites due to its lower complexity, minimal latency, and greater bandwidth. LEO refers to the lowest region in which satellites can operate relative to the earth's surface [1]. This is because the altitude of LEO is below 2,000 km, with most satellites positioned between 500 and 1,600 km above the earth [1]. Additionally, deploying satellites in LEO requires less energy, and they do not need powerful signal amplifiers to transmit data effectively.

Currently, about 4 million people live in the Arctic region at latitude 66.34 ° N [2]. The capacity of the LEO satellite system, especially the constellations in inclined orbits, is evenly distributed across all longitudes and offers symmetrical coverage around the equator, making it possible to reach the poles such as the arctic region, providing global coverage. Even the system on the Geostationary Earth Orbit (GEO) satellite cannot provide such a service feature [3].

Achieving global coverage through LEO satellites requires deploying a large number of satellites, typically ranging from a few hundred to thousands of nodes. Current research and development efforts focus on creating mega-constellations, where LEO satellites are interconnected via Inter-Satellite Optical Links (ISLs) [4]. Companies such as SpaceX (Starlink), OneWeb, Amazon Kuiper, Telesat, and LeoSat have already begun implementing these technologies [4]. To effectively design and manage these large-scale constellations, advanced software tools are essential. These tools help design, simulate orbits, optimize trajectories, and ensure consistent global coverage, thereby emphasizing the need for reliable orbit design software [5].

1.2 Problem Analysis

These challenges are not merely technical but also involve interconnected operational, economic, and regulatory considerations. Therefore, this problem analysis is crucial during the initial design phase, where architectural decisions and mitigation strategies must be made to ensure mission success. This analysis includes an evaluation of constraints such as power limitations, thermal management requirements, collision risks, and the complexity of inter-satellite communication, all of which have a direct impact on system performance and reliability [6] [7].

By understanding the interplay between these factors, we can develop robust and efficient solutions capable of adapting to the dynamic LEO environment. The aspects encompassing challenges in designing LEO satellite applications can be considered through complex factors that impact various case studies, as follows.

1.2.1 Technical Aspects

The design of LEO satellite applications involves several technical aspects. This requires an accurate mathematical model to describe the orbital dynamics of the satellite, considering various factors such as gravitational forces, atmospheric drag, and inter-satellite interactions [8]. LEO satellites orbit the Earth in the exosphere at a relatively close distance to the earth's surface, which results in gravitational forces significantly affecting the lifespan of LEO satellites. Each satellite has a circular orbit defined by the equation $r = R + h$, where R represents the radius of the earth [9].

The distance between the Ground Station (GS) and the satellite is a crucial factor in calculating communication latency [5]. The selection of programming languages and development frameworks must also be considered, focusing on aspects such as performance and scalability of the application [10]. Furthermore, integration with a database for storing and managing simulation data, as well as the programming of algorithms for real-time orbit visualization, will enhance the quality and utility of the application [11] [12].

1.2.2 Technology Aspects

The design of LEO satellite simulation applications presents several challenges that require the careful application of various technological aspects to ensure accuracy and reliability [13]. One major challenge is selecting appropriate programming languages capable of handling large-scale data processing, which is essential for simulating complex satellite networks [14].

Another challenge is to create an intuitive user experience which displays two-dimensional visualization, utilizing advanced graphics technologies like OpenGL or similar frameworks to enable analysis of satellite movements in a simplified yet effective manner [15].

1.2.3 Legal Aspects

In designing a LEO satellite application, it is crucial to consider and ensure compliance with applicable regulations throughout the entire development process. The landscape of satellite communication regulations has significantly evolved, recognizing the role of satellite communication systems as a primary service in governmental affairs year by year [16]. Therefore, a comprehensive understanding of these regulatory frameworks is essential to avoid potential legal challenges and operational constraints. Adherence to these standards guarantees the application's legal and operational viability in the long term.

This includes the meticulous protection of intellectual property rights, such as copyrights and patents, which are vital for the unique designs and proprietary algorithms used in the application. Moreover, a thorough examination of international regulations concerning satellite technology and broader space activities is indispensable. These regulations can have a significant impact on the application's operational scope, its permissible uses, and its overall deployment strategy [17]. By integrating these legal considerations from the outset, the project can proceed with confidence and a clear path forward.

1.3 Analysis of Available Solution

As the demand for satellite-based services increases, so does the need for tools that can simulate, predict, and visualize LEO satellite orbits [4]. These tools allow users to understand satellite behavior, plan missions, and monitor real-time satellite positions.

There are several existing software solutions for orbit simulation, such as AGI's System Tool Kit (STK), NASA's General Mission Analysis Tool (GMAT), Simplified General Perturbations 4 (SGP4), and other professional tools. These systems generally share several common features, making them powerful tools for simulating and analyzing satellite orbits.

1.3.1 Simplified General Perturbations 4 (SGP4)

The Simplified SGP4 algorithm is a widely used technique for simulating the orbit of satellites near Earth by utilizing known orbital elements to forecast their future position and velocity [5]. Developed in the 1970s, this method is recognized for being relatively straightforward, efficient, and accurate [18]. It is specifically designed to work with orbital data in the Two-Line Element (TLE) format, provided by the North American Aerospace Defense Command (NORAD) [18].

The satellite position coordinates generated by the SGP4 algorithm are typically presented in a geocentric coordinate system known as the True Equator Mean Equinox (TEME) system. This is an Earth-Centered Inertial (ECI) system that is fixed at the Earth's center of mass and does not rotate, referencing the celestial sphere. However, ground communication stations located on the Earth's surface use a geodetic coordinate system, which rotates with the planet. Therefore, it is necessary to convert TEME coordinates into geodetic coordinates to accurately determine the relative distance and latency between satellites and ground stations (GS) [5].

Despite its advantages, SGP4 has several weaknesses. It relies heavily on TLE data, which can be imprecise or outdated, leading to potential errors in position predictions [18]. The algorithm simplifies perturbative effects, which may result in inaccuracies, especially for satellites in orbits influenced by significant factors such as atmospheric drag or gravitational interactions with other celestial bodies [19]. Additionally, SGP4's output in the TEME coordinate system necessitates conversion to geodetic coordinates, which can introduce further inaccuracies [5]. Over extended periods, particularly for satellites in LEO, the algorithm may struggle with accurately predicting positions due to orbital decay and atmospheric effects [18]. Furthermore, SGP4 does not adapt to real-time environmental changes or unexpected satellite maneuvers, making it less reliable for high-precision applications. These limitations underscore the importance of utilizing SGP4 in conjunction with more sophisticated methods and updated data to ensure greater accuracy and reliability in satellite tracking and mission planning.

1.3.2 System Tool Kit (STK) by AGI/Ansys

AGI's Systems Tool Kit (STK) is a software suite designed for modeling and analyzing the performance of complex systems in space, air, land, and sea [20]. STK excels in providing high-fidelity simulations of satellite orbits, mission planning, and scenario analysis [20]. It incorporates various propagation algorithms, including SGP4 and more sophisticated numerical integration techniques, allowing users to analyze the effects of environmental factors such as atmospheric drag and gravitational perturbations [20].

STK also offers advanced visualization capabilities, enabling users to create detailed Three-Dimensional (3D) models of satellite trajectories and ground station coverage [20]. Despite its strengths, STK can be complex to use and may require significant computational resources for large-scale simulations [21]. Additionally, the software's licensing costs may pose a barrier for some users, limiting its accessibility [21].

1.3.3 NASA's General Mission Analysis Tool (GMAT)

NASA's GMAT is an open-source software application designed for trajectory optimization and mission analysis [21]. GMAT supports various numerical integration techniques, enabling users to simulate and predict satellite orbits with high accuracy [21].

The tool provides a flexible framework for modeling complex missions, including interplanetary trajectories and satellite constellations [21]. Its user-friendly interface and extensive documentation make it accessible to a wide range of users, from academic researchers to industry professionals [21]. However, GMAT's reliance on accurate input data and models means that it is also susceptible to inaccuracies if the initial parameters are not carefully chosen [22].

1.3.4 General Perturbation Methods

General Perturbation methods encompass a variety of techniques used to simulate satellite orbits by accounting for perturbative forces acting on the satellites, such as gravitational influences from other celestial bodies, atmospheric drag, and solar radiation pressure [23]. While general perturbation methods can yield high accuracy in predicting satellite positions over extended periods, they often require significant computational power and can be complex to implement [23].

The accuracy of these methods also depends heavily on the precision of the input orbital elements and environmental models [18]. As such, they are typically used in scenarios where high fidelity is critical but may not be as accessible for general users without a strong background in astrodynamics [18].

1.4 Objectives

The satellite industry continues to experience significant growth each year, with Low Earth Orbit (LEO) satellites being a key contributor to this trend. LEO satellites are preferred due to their relatively lower complexity, minimal latency, and higher bandwidth capabilities. The process of orbit design involves numerous components and parameters that influence the accuracy of satellite positioning and the overall performance of the orbit during operational missions.

Given the complexity of these parameters, a dedicated platform is essential to support the design of LEO satellite orbits. Such platforms simplify the design process for users, although existing platforms on the market come with their own strengths and limitations. To make these goals operational and verifiable, this study adopts the following quantitative acceptance thresholds for preliminary-design fidelity, schedule prediction, and usability. Achieving these thresholds constitutes successful fulfillment of the objectives:

1. **Propagation Accuracy:** When compared with NASA GMAT over a 1-hour horizon, the geodetic position error shall not exceed **50 km** with a worst-case limit of **60 km**, acknowledging inherent differences between analytical (J_2-averaged) and numerical (RK4) propagators.

2. **Analytical Cross-Check:** When validated against hand calculations or closed-form benchmarks, the equatorial geodetic position error shall not exceed **12 km**.

3. **Constellation Placement:** For both Train and Walker-Δ configurations, the RAAN and true anomaly placement errors shall not exceed **0.05**° per satellite.

4. **Coverage Accuracy:** The ground footprint radius error shall not exceed **10 km** across representative altitudes (500–2000 km) and beamwidths.

5. **Link Budget Accuracy:** When compared with manual calculations, differences shall not exceed **0.2 dB** for C/N ratio and link margin, with received power differences not exceeding **0.5 dB** (using consistent units: dBW or dBm).

6. **Access Schedule Accuracy:** The pass count must match reference values exactly, with first-access timing errors not exceeding **15 seconds** and mean pass duration errors not exceeding **15 seconds**.

7. **UI/UX Reliability:** The user interface shall achieve a task success rate of at least **99%** across a minimum of 5 trials per action and 2 modern browsers (e.g., Chrome, Firefox), with zero blocking crashes.

1.5 Problem Limitations

The development of the LEO Satellite Orbit Design Application, although designed as a comprehensive and user-friendly platform, is subject to a number of inherent limitations. These limitations are crucial for defining the scope of the project and for setting realistic expectations regarding its capabilities and accuracy. Acknowledging these constraints allows for a focused and manageable development process, ensuring that the final application performs reliably within its specified boundaries.

The primary constraints are related to the complexity of the LEO environment, which requires certain simplifying assumptions. For instance, the application's accuracy may be affected by the simplification of atmospheric drag models or other subtle gravitational perturbations [24]. These acknowledged limitations are vital for informing users about the application's intended use-cases and its performance in various operational scenarios, thereby preventing misuse and misinterpretation of simulation results.

- **Practical and resource limitations**, while the application aims to be more accessible than high-fidelity professional software such as STK or GMAT, its performance and features are constrained by significant computational resource requirements [10]. This includes the need for powerful hardware to support large-scale simulations, which represents a practical limitation compared to expensive commercial solutions [10].

- **Accuracy and data limitations**, the application has inherent limitations regarding the accuracy of its orbit calculations. This accuracy is affected by dynamic space environmental conditions (such as atmospheric variations, solar radiation, and gravitational perturbations) and the reliance on input data precision [13]. The mathematical models employed may not fully capture all of these complex effects, resulting in limited real-time accuracy [10].