A blue and white logo

Description automatically generated

***School of Mechanical & Manufacturing Engineering (SMME),***

***National University of Science and Technology (NUST),***

***Sector H-12, Islamabad***

PROJECT REPORT

Satellite Tracker

19-Nov-2024

**Name:** Zahika Ihsan

**CMS Id:** 461202

**Semester:** 03

**Course Code:** CS-223

**Submitted To:** Miss Laiba Waheed

**Introduction:**

In this project, I aim to develop a Satellite Tracker using a Python program. Tracking and visualization satellites are very useful in space research and navigation. The capability to predict and visualize with reasonable accuracy the path of a satellite is very important for many applications in aerospace. This project will demonstrate how Python libraries like Skyfield, NumPy, and Matplotlib can be used to create a system that tracks a satellite in the sky. From Celestrak, using Two-Line Element data, the system will track the trajectory of the ISS relative to an observer's location in Wah Cantt, Pakistan over a specified time.

**Theory:**

1. **Two-Line Element (TLE) Data**:

TLE data is a set of parameters describing the orbit of a satellite. It is used to calculate the satellite's position and velocity in space. The Skyfield library can interpret this data to provide accurate satellite tracking information. TLEs are typically updated periodically to account for changes in the satellite’s orbit.

1. **Satellite Position Calculation**:

Using orbital mechanics and trigonometric calculations, the Skyfield library computes the satellite's position relative to an observer on Earth. This involves coordinate transformations from Earth-centered inertial (ECI) frames to local horizontal frames defined by altitude and azimuth.

1. **Observer's Location and Altitude-Azimuth System**:

The observer's location on Earth is specified using latitude and longitude coordinates. The altitude-azimuth system is then used to describe the satellite’s position in the sky:

* **Altitude**: The angle above the horizon, measured in degrees (0° at the horizon, 90° directly overhead).
* **Azimuth**: Horizontal direction of an object in the sky (degrees)

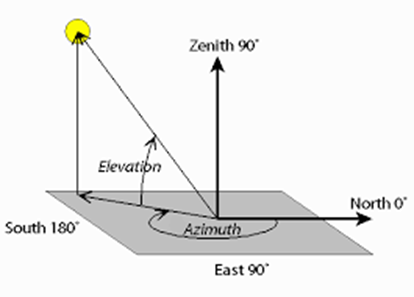


Figure 1: Azimuth and Elevation (Altitude)

1. **Polar Plot**:

A polar plot is used to visualize the satellite's trajectory in the sky. The azimuth angle is plotted in a circular coordinate system, and the altitude is converted into a radial distance from the center, where lower altitudes are farther from the center and higher altitudes are closer.

**Procedure:**

1. **Importing Libraries**: The code will import essential Python libraries:

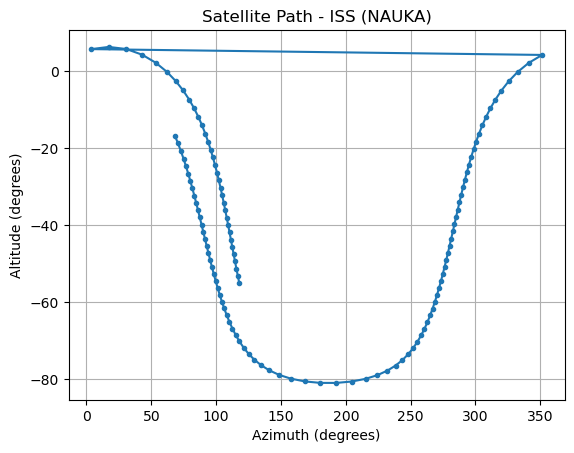
NumPy for numerical operations.

Matplotlib for plotting graphs.

Skyfield for astronomical computations and satellite tracking.

1. **Loading TLE Data:** The TLE data for the ISS is loaded from Celestrak using Skyfield's load.tle function. The ISS is identified and selected from the data.
2. **Setting up Time Range:** A time range of three hours is defined using the timescale object from Skyfield. The time range is set in UTC format, starting February 21, 2024 at 2:00 AM.
3. **Setting up observer's location:** The geographical location of an observer is defined by using Skyfield's Topos class that takes the latitude and longitude of Wah Cantt, Pakistan.
4. **Calculating Satellite Position:** In this function, satellite altitude and azimuth relative to the observer are computed at each time step. The altaz() method is used to get the altitude and azimuth in degrees, which then gets appended to the respective lists.
5. **Plotting Satellite Path:** Now, let's plot the satellite's path in two different ways:

**Azimuth vs. Altitude Plot:** This plot represents the path of the satellite in the sky with time, azimuth on the horizontal axis, and altitude on the vertical axis.



**Polar Chart:** The polar plot plots the trajectory of the satellite in this polar plane. This allows better visualization for the observer to understand in which direction and at what elevation the ISS is. Altitude is shifted within the radial limits so that 90° is at the center and 0° at the extreme edge.

A polar chart with numbers and points

Description automatically generated

1. **Visibility Check:** The visibility of ISS is determined by checking whether altitude is more than 0°, that is if it is above the horizon. Only those visible points are plotted on polar chart.

**Conclusion:**

This project successfully demonstrates how to track and visualize the trajectory of the International Space Station by using Python. With TLE data and astronomical libraries like Skyfield, one can compute and graph the path of the satellite accurately. The two visual representations-one, a 2D plot and another, a polar chart-offer simple intuitive insights about the satellite's movement in the sky. This process is highly applicable for amateur astronomers, satellite communication enthusiasts, and aerospace researchers who need reliable tracking methods.

**References:**

1. Angelina Tsuboi. "Python for Aerospace." *YouTube*, uploaded by Angelina Tsuboi, Sep 4, 2024. Available at: <https://youtu.be/V4jXVrUJsfM?si=MSIM5FHetdwM-Ous>.
2. "NORAD Two-Line Element Sets - Stations." Celestrak. Available at: <https://celestrak.com/NORAD/elements/stations.txt>.