Visualiser - De-Orbiting Satellite Predictor

Jack Roberts

April 30, 2025

1 Data visualisation

Data visualisation is the art of representing data such that it maximises the quality to quantity ratio. In other words, display the smallest quantity of data that explains as close to 100 percent of our results as possible. The project proposal, handed in during term 1 outlined the visualiser as a visual representation of the predictor (Extended Kalman Filter) and will display temporal as well as spatial data regarding the trajectory of the satellite. There are many data visualisation tools to accomplish this, such as Unity (C#), matplotlib (Python), pyQtGraph (Python) or Tableau.

2 Methods

2.1 A change in software

Unity and C# was the software planned for use for the visualiser. It had a built in Graphical User Interface (GUI) and was able to convert large amounts of data into highly customisable graphs. Upon reflection, and some guidance from the academics, it became apparent that using as many as three languages to complete this project was too much. Moreover, the transfer of data was proving to be difficult as running a python venv inside of C# and converting its output to C# arrays was a significant bottleneck. Instead a Pythonic approach became the apparent option.
The visualiser now runs inside python, running the *pyQtGraph* module. This module builds on its parent module *pyQt* for creating python GUIs by adding graphical support such as plotting, adding legends, multiple subplots and dynamic plotting, where the figure could update as it receives updates from the predictor in real time. This was an excellent choice as this module allowed the project to be kept in python, a language everybody in the group understood, without sacrificing dynamic updates from our predictor in real time which is what was needed.

2.2 Requirements

The following python packages are required for the visualiser and are NOT standard packages such as numpy:

- PyQt5
- pyqtgraph

The file *requirements.txt* contains all required packages which can be run in the terminal using:

$$\text{python3 -m pip install -r requirements.txt} \tag{1}$$

2.3 GUI Layout

The main window is the class **MasterWindow**. This window collates the other windows into one and provides the ability to navigate through them. To initialise the GUI, this class is run and loads the window **MainMenu** containing a welcome message and links to each of the other windows. The links are to the actual simulation, instructions on how to use the GUI, credits, and an exit button. Each subsequent page contains a **Navbar** class that sits at the top of the window, allowing for safe return to the main menu. The simulation, a class named **Graphs** is an umbrella window for the entire project. Within the **Graphs** class sits a script which displays the initial input data (x and y), and contains an update function which removes the oldest datapoint and replaces it with the incoming new datapoint, maintaining the initial input list length. Each plot in the window is allocated a space and passed into another class called **Plot**. Each instance of **Plot** is responsible for a given subplot in the window, controlling its initial plot and subsequent updates.

2.4 Customisation

Under the *partials* directory lies the file *global_settings.json*, containing universal information regarding the GUI such as font style, background and font colour and window size. By placing all universal variables in a single json file, changing a variable that is widely used becomes easy. Moreover, within the profiles directory sits the individual profiles for each plot which can be passed into each instance of **Plot**, containing details such as plot and label titles and styles, line details and booleans for legends and grids. This separation of interests de-clutters the complex code and improves readability.

3 Plots

There will be IDK plots, each displaying vital information representing the current state of both the satellite and the Kalman Filter's prediction of its predicted landing site.

4 Results