GUI User Guide

Running the GUI

Initialization

To run the GUI, execute the following code in your python environment:

```
pip install your_path_of_whl_file
```

Then, in your python environment import warwick_pmsc_skylab and run the code:

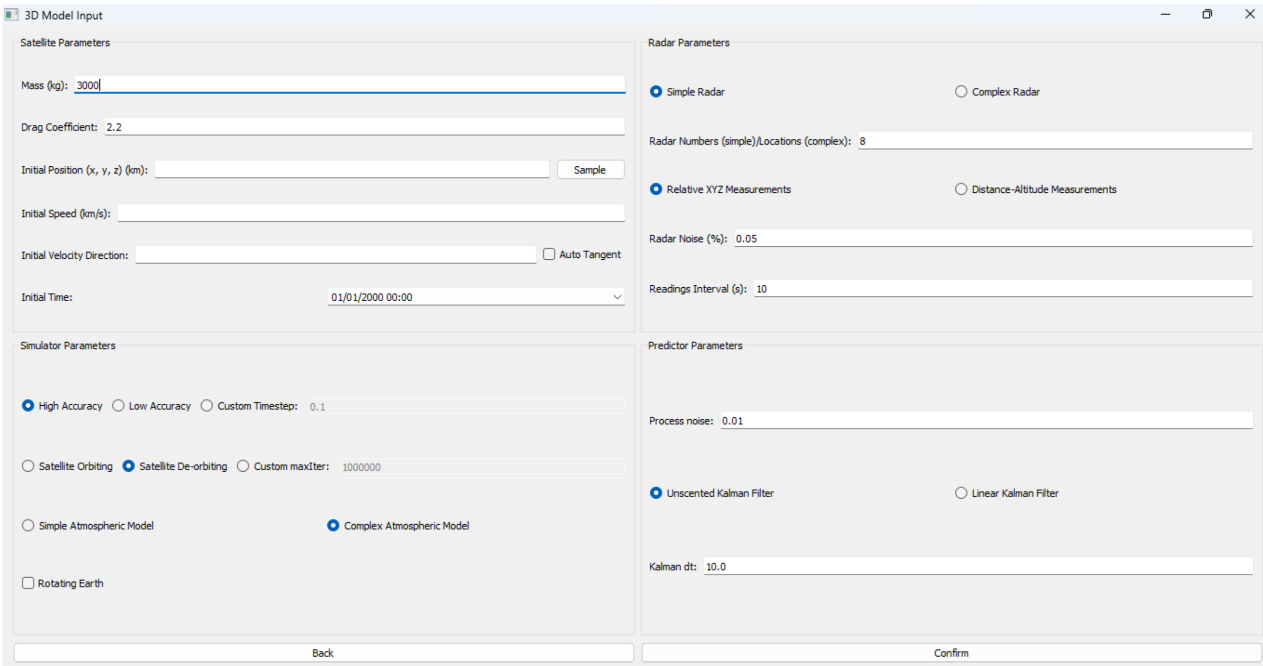
```
warwick_pmsc_skylab.run_GUI()
```

When the code is run, an interactive window will open, looking like this:

Here, the user can specify which model they want to use. This is either the assumption of a two-dimensional Earth or a three-dimensional Earth.

3D Model GUI

If ‘3D version’ is chosen, the GUI will now look like:



The following sections go over what each field means, going block per block.

Satellite Parameters

This block outlines the parameters of the satellite orbiting the earth. Most satellite parameters are self-explanatory. The ‘sample’ option on initial position randomly chooses an initial position for the satellite such that it is 408km from the surface of the Earth (which is a realistic initial height of a satellite).

Initial satellite speed should be input as a scalar value corresponding to the tangential speed of the satellite in km/s. This speed is combined with the ‘initial velocity direction’ which is a *unit* vector, tangential to the initial position. The user can specify such a vector themselves, but an easier option is to use the ‘auto-tangent’ option which generates such a unit, tangent velocity vector.

TIP: A good initial speed is $7kms^{-1}$.

Simulator Parameters

This next section outlines how the orbit is simulated. The first parameter is accuracy of the simulated data. The 'High Accuracy' option results in a smaller timestep of dt = 0.1, resulting in more simulated position data. The 'Low Accuracy' option results in a larger timestep of dt = 1, leading to faster runtime but less reliable results. The user can also specify their own time step with the 'Custom Timestep' option, depending on their individual accuracy preferences.

The 'Satellite Orbiting/De-orbiting' parameter ensures that the simulation does not run for longer than it should. It sets the maximum number of iterations of simulated data to be 100,000 or 1,000,000 respectively. The user should specify 'Satellite De-orbiting' first to see how long the simulation takes. If it runs in a reasonable time, it means that the satellite is de-orbiting and crashes. If the simulation is taking longer than expected, the satellite is likely to be orbiting the Earth and so maximum iterations should be reduced so unnecessary position data is not simulated and run time is reduced. Alternatively, the user can directly specify the maximum number of iterations they are willing to run by choosing 'custom'.

If the option 'Rotating Earth' is selected, then the Earth is assumed to be rotating around its axis, with one full rotation corresponding to a stellar day.

Radar Parameters

The 'Simple Radar' option automatically creates radar stations along the equator, equally spaced. The number of radars can be specified in the 'Radar Numbers/Locations (complex)' field.

If the 'Complex Radar' button is chosen, then the user can specify exact locations of radar stations in the same field. It is important that these radar stations are located on the surface of the earth. This input must be in the form of an array with each element being the $(x,\ y,\ z)$ position a radar relative to the centre of the earth. For example, if we have three radars placed at co-ordinates $(-1,\ 0, 0)$, $(0,\ 1, 0)$ and $(1,\ 0, 0)$ then the input to the field should be $\big[[-1,\ 0,\ 0]\,,\ [0\ ,\ 1,\ 0]\,,\ [1,\ 0,\ 0]\big]$.

If 'Relative XYZ measurements' is selected, radar readings are measured in $(x,\ y,\ z)$positions *relative* to the radars. If 'Distance-Altitude Measurements' is selected, then radar readings of satellite position are in distance, altitude readings (see the project report for an explanation of these coordinate systems).

The input to ‘Radar Noise (%)’ describes how much noise should be added to the measurement data in each coordinate component when a radar reads the position of a satellite. This percentage is a percentage of the absolute value of each component, and it denotes the variance of the added (Gaussian) noise in each component.

‘Reading Interval’ denotes how often radar measurements should be recorded. For example, a reading interval of 10 implies that radar measurements are taken every 10 seconds.

Predictor Parameters

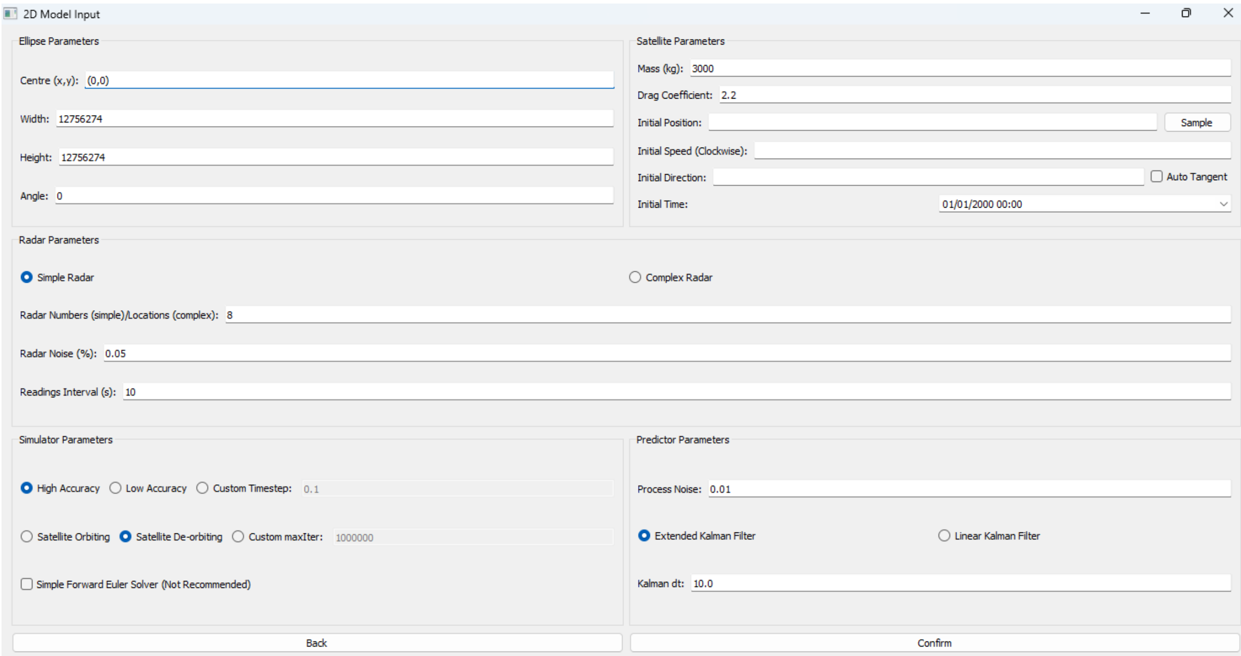
Process noise is a parameter that sets how accurate the Kalman filter position predictions should be (see report for explanation).

An option to select what type of Kalman filter (‘Unscented or ‘Linear’) to use in the predictor is available.

‘Kalman dt’ denotes the timestep that the predictions take from one state to the next state.

2D Model GUI

If 2D model is selected on the initial screen, then the GUI will look like:



Except for the ellipse parameters section, all the buttons shown here perform the same function as they do for the 3D model. The extra ellipse section allows the user to specify the dimensions of their ellipse, although using the given values is recommended. Note that unlike for the 3D case, distance units should be in meters and speeds in m/s. Auto Tangent doesn't specify an explicit direction, but instead specifies a flag to the backend which runs the tangent computation automatically there.

Visualization Window

Description

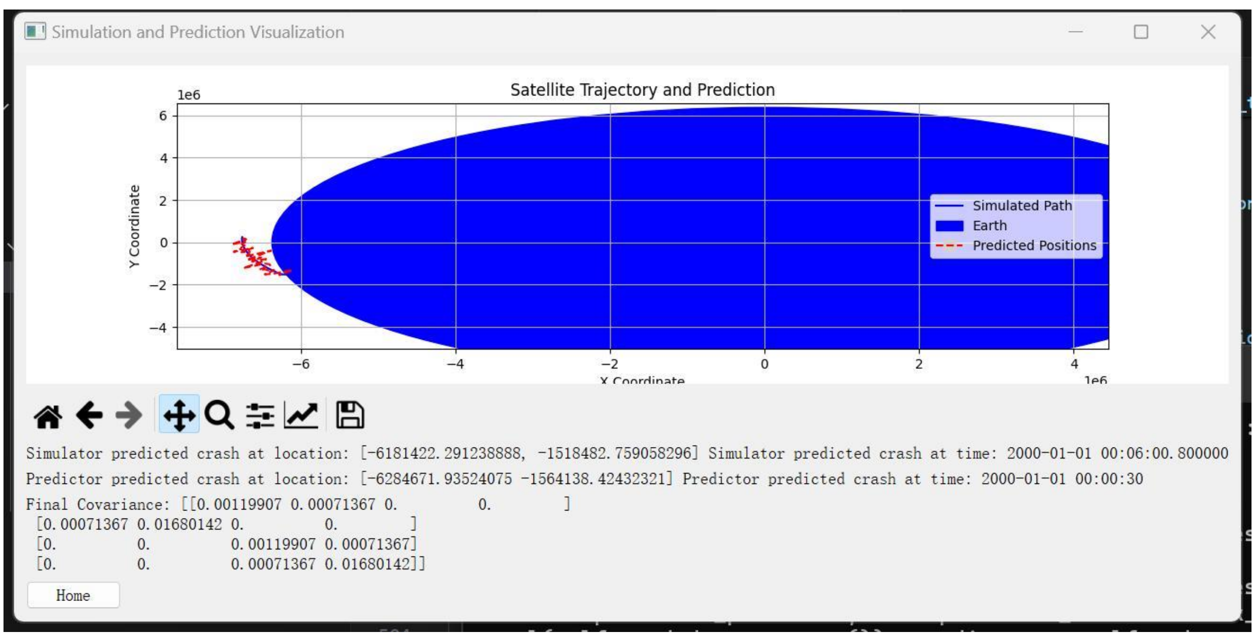
This window displays the simulation and prediction results.

Features

- **Plot Area**: Displays the simulation and prediction paths of the satellite and relative earth model.
- **Toolbar**: Allows interaction with the plot, with Initialize, retreat, forward, moving, zooming, configuration plots and save output.
- **Simulation Values**:
  - **Simulated Crash Location**: Displays the simulated path location from simulator.
  - **Simulated Crash Time**: Displays the simulated crash time from the simulator.
- **Prediction Values**:
  - **Predicted Crash Location**: Displays the predicted path location from predictor.
  - **Predicted Crash Time**: Displays the predicted crash time from the predictor.
  - **Final Covariance:** Displays final co-variance of the state space at final predicted point.
- **Back Button**: Returns to main window

2D Visualization

Visualize the 2D model results.

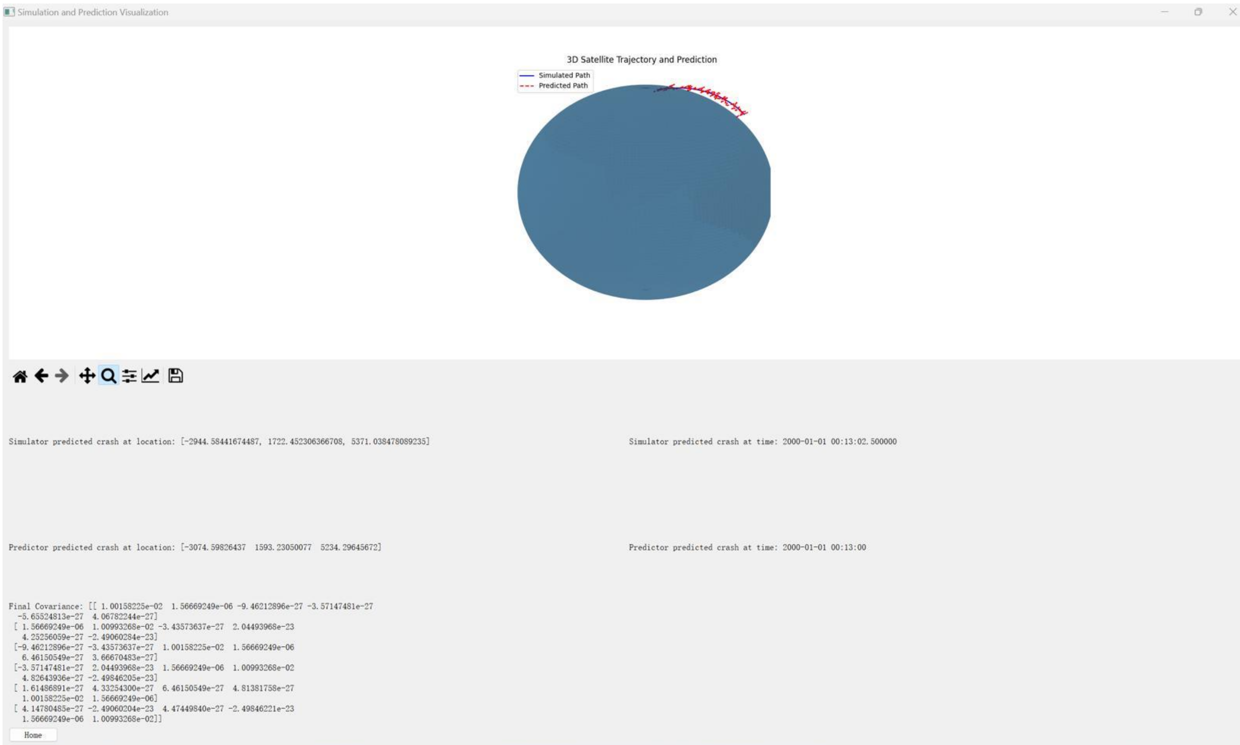


Extra Feature

- **Predicted path**: predicted path comes with grey squares which indicate error bars.

3D Visualization

Visualize the 3D model results.



Extra Feature

- **Predicted Path**: Crashing point is visible while zooming in.