INSTITUTO SUPERIOR TÉCNICO

Atmospheric Models: Satellite Decay Time Program: User Manual

Students:

António José Domingos Reis (102473)
Ricardo Gandra de Sousa (102498)
Tiago André da Silva Ruge (102551)
João P. Veloso Onofre de Carvalho (102686)
Eduardo De Almeida Helena (102793)
Tomás Gomes Coelho (102805)
Gonçalo J. R. Bessa da Silva (102995)
Fernando Meneses Vicente (103048)
Álvaro G. Silva Vilela Caridade (103526)
João Nuno Rodrigues Alegrete (103676)
João Bessa e Silva Machado Vilaça (103966)

Course: *Coupled Phenomena* – Professor: *Paulo S. Gil*

Introduction

The program we're presenting has the purpose of computing the decay time of a satellite in an initial orbit.
This is related to our theme (Atmosphere Models) since the aerodynamic forces responsible for "slowing down" the satellite depend directly on the characteristics of the atmosphere at the considered trajectory. Those characteristics such as density and temperature can be computed by the models we present in this report. This way, in the program, the user will have the opportunity to observe the differences in the output between the different models and how the decay time of the satellite varies from one to another. We decided to use Python as the Language used to implement this idea.

Required Python Packages

In order to run our program, make sure to install the following Python packages using the following command in your terminal:

```
pip install numpy matplotlib
```

If pip is not recognized by your terminal, it means that you didn't add Python to the local path during installation. We suggest the following website to fix the problem: `https://builtin.com/software-engineering-perspectives/pip-command-not-found`

Configuration

To configure the program, edit the **input.txt** file with any text editor. In this file, you're gonna need to provide the data of the satellite you want to analyze and the orbit parameters.

The different Density Models available are:

- Student Model - Model with several approximations, g constant, deduced in the theoretical Reports by the students.
- US1976 - U.S. Standard Atmosphere 1976
- CIRA - COSPAR International Reference Atmosphere (CIRA-86)
- GRAM99 - with the possibility to choose the thermospheric model (Jacchia or MET)

Specify the following parameters:

- `CD`: Drag coefficient
- `m (Kg)`: Mass of the satellite in kilograms
- `d (diameter m)`: Diameter of the satellite in meters
- `Apoapsis Altitude (Km)`: Apoapsis altitude in kilometers
- `Periapsis Altitude (Km)`: Periapsis altitude in kilometers
- `Method (StudentModel, US1976 , CIRA, Jacchia, MET)`: Choose the analysis method
- `Model Temperature (For Jacchia and MET) (Kelvin)`: Model temperature in Kelvin

For Jacchia, the exospheric temperatures available are 600 K, 1500 K, 2400 K, and 600 K, 1400 K, 2200 K for MET.

Note: The satellite is considered as a sphere, so only its diameter is needed.

Example Usage

Here is an example of how to fill in the information in the `input.txt` file:

```
CD: 2.2
m (Kg): 100
d (diameter m): 1
Apoapsis Altitude (Km): 939
Periapsis Altitude (Km): 500
Method (StudentModel, US1976 , CIRA, Jacchia, MET): MET
Model Temperature (For Jacchia and MET) (Kelvin): 2200
For Jacchia, Temperatures available are 600, 1500, 2400 K
For MET, Temperatures available are 600, 1400, 2200 K
```

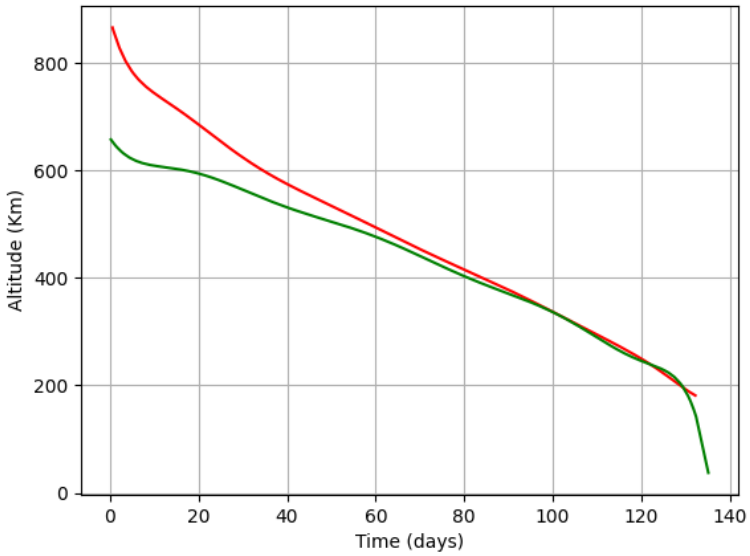
Running the Program

Execute the main Python script `main.py` to run the program. Ensure that you have the required Python packages installed.

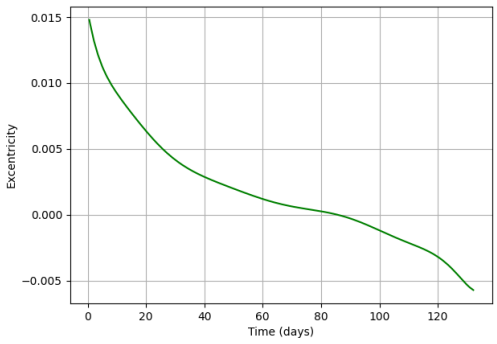
In the case you've inputted something wrong, the console will prompt an error message telling you what went wrong.An example of an error message follows:

```
R_a not specified. Please specify the parameters in the input.txt file.
Press Enter to close the terminal...939
```

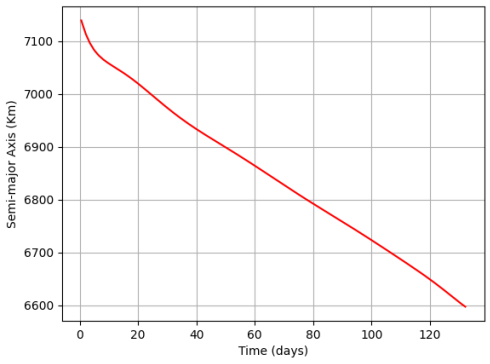
For every Method, except StudentModel, the program will output 3 different graphics, The Apogee and Perigee plotted with time, the Semi-Major Axis (a) plotted with time, and eccentricity (e) plotted with time.



(a) Apogee and Perigee Altitudes with time



(b) Eccentricity with time



(c) Semi-Major Axis with time

For the StudentModel case, our software exhibits a notably rapid satellite descent, where eccentricity (e) and major semi-axis (a) analysis is irrelevant. Only one image is generated, showcasing the satellite's altitude variation over time, with a time scale measured in hours instead of days.

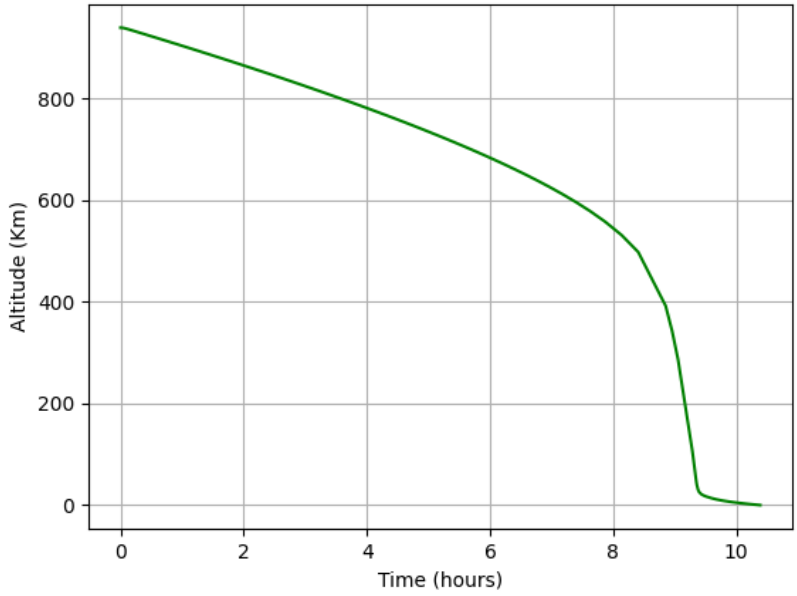


Figure 2: Apogee and Perigee Altitudes

The program closes when the graphics are closed, if there's any error, you just need to press space to close the terminal. For new simulations, run main.py with different inputs in .txt document.