# PH2150 Project: Solar System Simulation

Candidate Number: 1707736

December 8, 2016

**Abstract**

I have created a program with a focus on our solar system. The main menu has four useful buttons: one linking to this report, one which opens a simulation of how the Earth-like planets move around the sun, one which opens a second menu providing information on each of the planets in our solar system and another opening a window which shows the planets to scale. When opening the Earth-like planets simulation, a GUI opens with sliders which can be moved to change variables regarding each planet. The "Show Solar System" button then shows how the solar system evolves over time with these variables in place.

# 1 Planetary Orbits

## 1.1 Newton's Laws

Planetary orbits follow Newton's Laws of Motion and Newton's Universal Law of Gravitation, according to equation (1):

$$\vec{F} = \frac{GmM}{|\vec{x}|^2}\hat{x} \tag{1}$$

Using Newton's Third Law ($\vec{F} = m\vec{a}$), equation (2) can be found:

$$\ddot{\vec{x}} = \frac{-GM}{|\vec{x}|^2}\hat{x} \tag{2}$$

If you consider this for N objects, the force has to be summed for each individual object. For a particular object $i$ with a mass of $m_i$ and a position $x_i$, it is possible to obtain equation (3):

$$\ddot{\vec{x}} = -\sum_{j=1}^{N} \frac{Gm_j}{|\vec{x_i} - \vec{x_j}|^3}(\vec{x_i} - \vec{x_j})(1 - \delta_{ij}) \tag{3}$$

## 1.2 Euler's Method

Equation (3) cannot be solved analytically, so must be solved numerically. One method to do this is the Euler method. This uses known initial conditions for a known differential equation. At the starting position, the gradient of the curve is calculated. The gradient is then extrapolated to the next point, as determined by the number of calculations you wish to make. The gradient for this value of $x$ can then be calculated and extrapolated. This process continues to give an approximation of the curve. The smaller the step-up value, the closer the approximated values will be to the true curve.

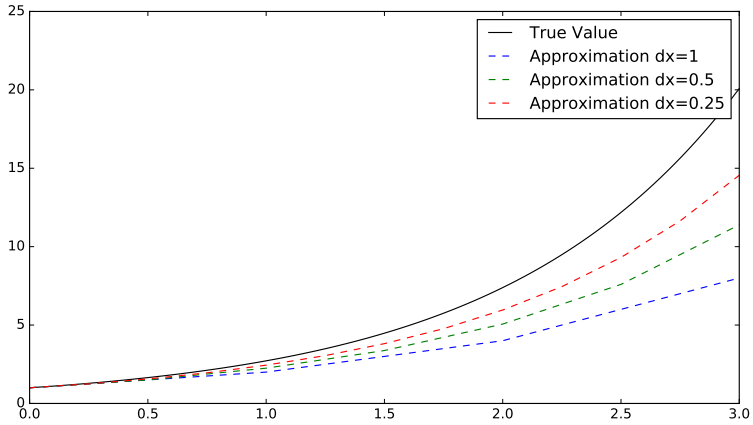


Figure 1: *An example of how the Euler method can be used to approximate a curve.*

# 2 How to Use the Program

## 2.1 Running the Program

In order to run this program, the following Python 2.7.11 packages are required:

- pyqt 4.11.4-7
- numpy 1.10.4-4
- scipy 0.17.1-3

All of these are available through the Enthought Canopy distribution, however pyqt is not a default package so needs to be installed manually through the package manager. Turtle was another package used, but does not appear in the package manager so I assume it is installed in Python by default.

Once you have all of these installed, run Solar_System.py through either Canopy or the terminal on your PC. The following menu should appear:

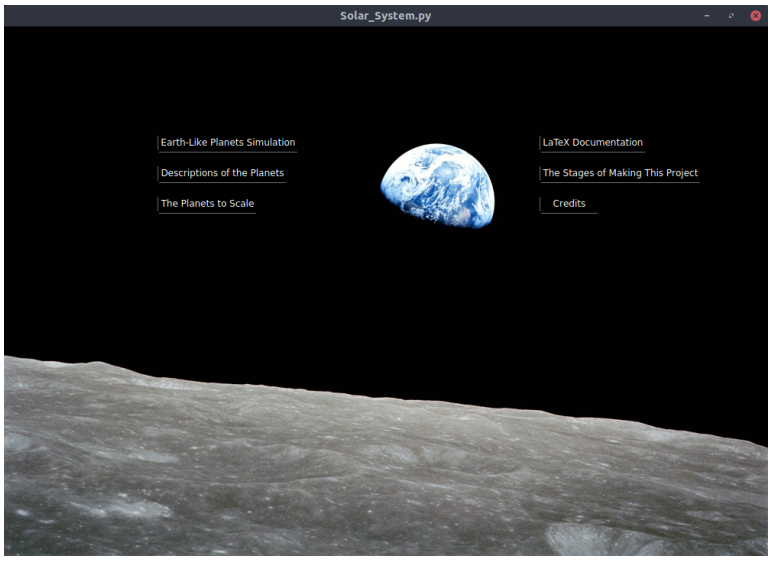


Figure 2: *The main menu of the program, as will appear when the program is run.[1]*

## 2.2 Earth-Like Planets Simulation

Pressing this button will open a menu consisting of sliders and two buttons, as shown below:

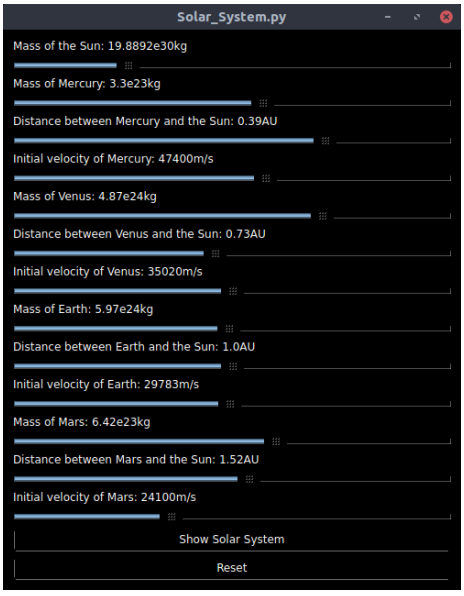


Figure 3: *Pressing "Earth-Like Planets Simulation" opens this window. The sliders can then be moved to change the values of the variables.*

The sliders are automatically set to the actual values for each variable[2], which is indicated in the respective label. Each label refers to the slider below. The sliders can then be moved, which will cause the values in the label to update to the new slider value. I opted for sliders rather than input boxes so that the variables cannot be changed massively, but instead can be changed to reasonable amounts. The sliders can be set back to their default values by using the"Reset" button. The "Show Solar System" button will then plot the paths of each planet according to the variables determined by the sliders[1]. A few combinations can be seen below:

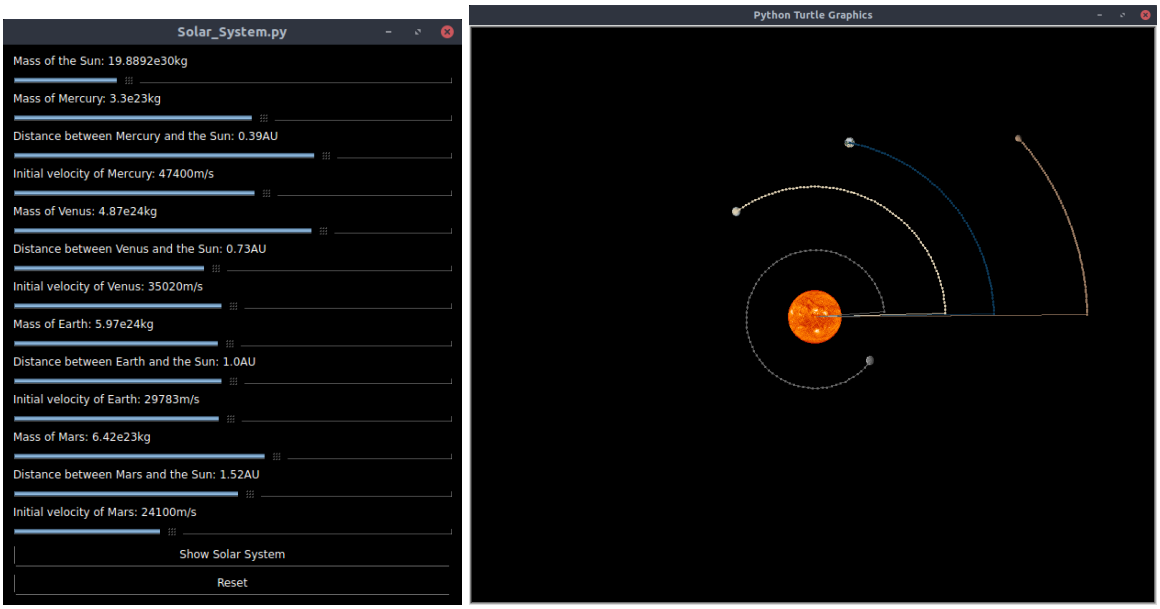


Figure 4: *Pressing "Show Solar System" with the sliders on the default values will begin the simulation, which looks like this.[3] [4]*

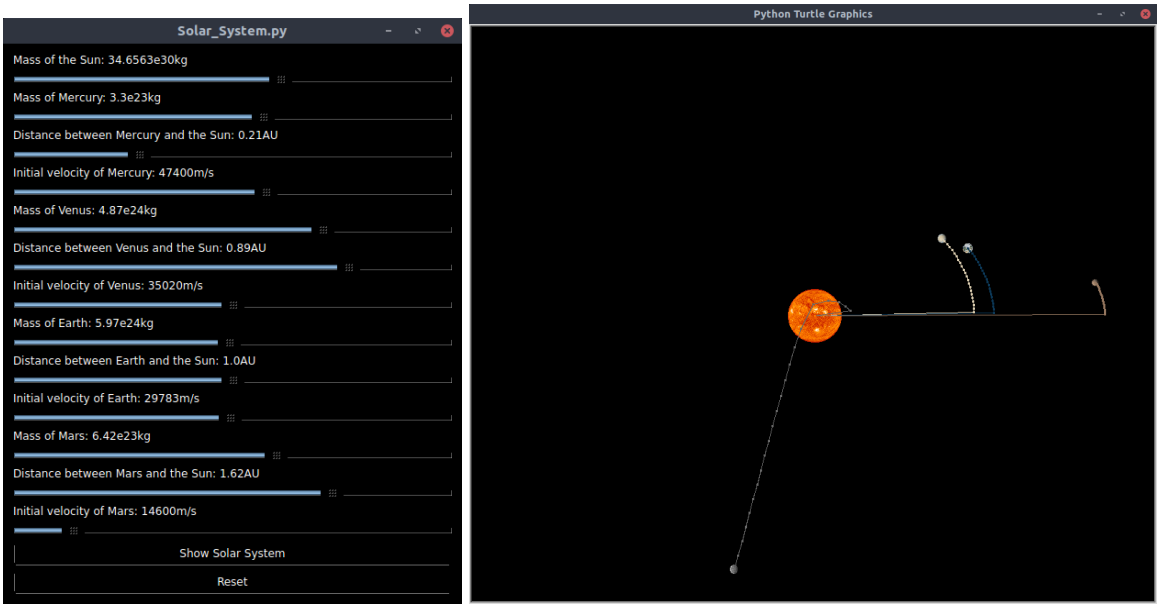


Figure 5: *Changing the sliders to these values and pressing "Show Solar System" significantly changes the trajectories of all planets.[3] [4]*

One issue with the program is that after the first plot is created, the "Show Solar System" button has to be pressed twice to create any subsequent plots. I tried to fix this using several different methods, however my efforts were in vain since nothing appeared to fix this problem. Given more time it would perhaps be possible, but I decided to focus on the rest of the program.

Closing the plot is also required before a new plot can be displayed.

The images for the sun and the planets were all taken from online. They are not to scale since this would make the planets nearly impossible to see, but I have made the planets significantly smaller than the sun to honor the differences in size. The colors of the paths were each selected based off of the images used, since I wanted the color scheme to mimic the colors of the planets.

[1]Note: the GUI will lock while the program is running. In order to move the sliders or open a different option from the main menu, the plot needs to be closed first.

Each dot plotted is approximately one Earth day[2].

I originally toyed with the idea of having an image of stars as the background of the plot[3], however I felt it was rather distracting so opted for the black background instead.

## 2.3 Descriptions of the Planets

Opening the descriptions of the planets will show the following menu:

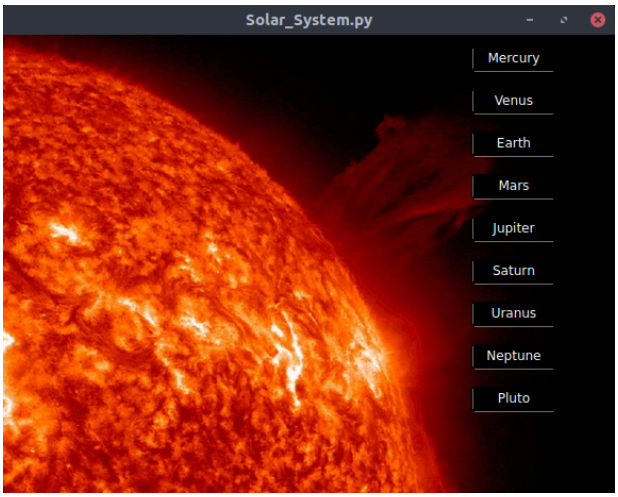


Figure 6: *Planet description menu.[5]*

Each of these buttons will then open a window with a picture of the selected planet, a description of the planet and a weblink to more information. The weblink is also the source of the information shown in the window. Obviously, an internet connection is required to open the weblink. The window for Uranus is shown below:

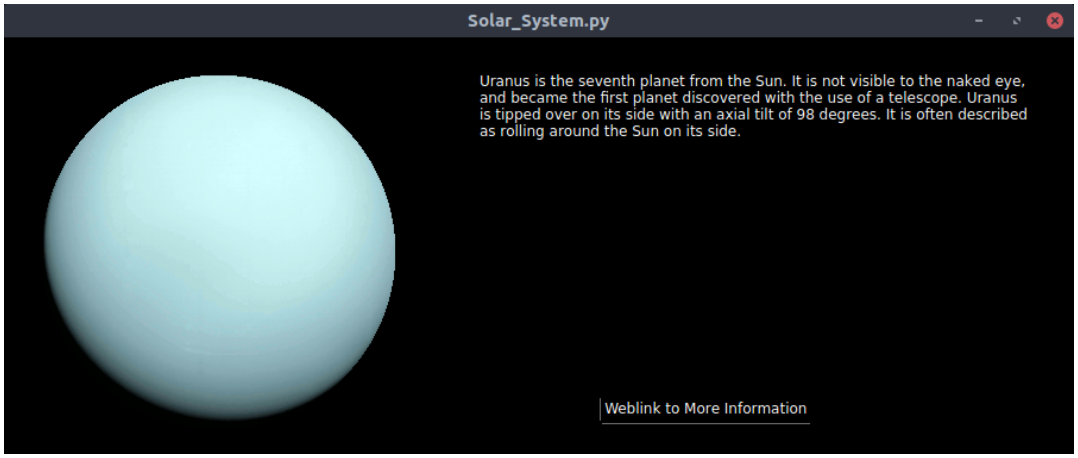


Figure 7: *Choosing Uranus from the menu in Figure 6 will show this window.[4] [2]*

## 2.4 The Planets to Scale

This opens a window with images of the planets to approximate scale. The sizes are based off the image for the Earth being 12x12 pixels, with each planet then be scaled using approximate multiples of this. The colors for the labels were taken from the images used. Clearly the distances between the planets are not to scale here.

[2]Approximately since 365 dots are plotted for one complete orbit of the Earth.
[3]This would have been done using screen.bgpic()

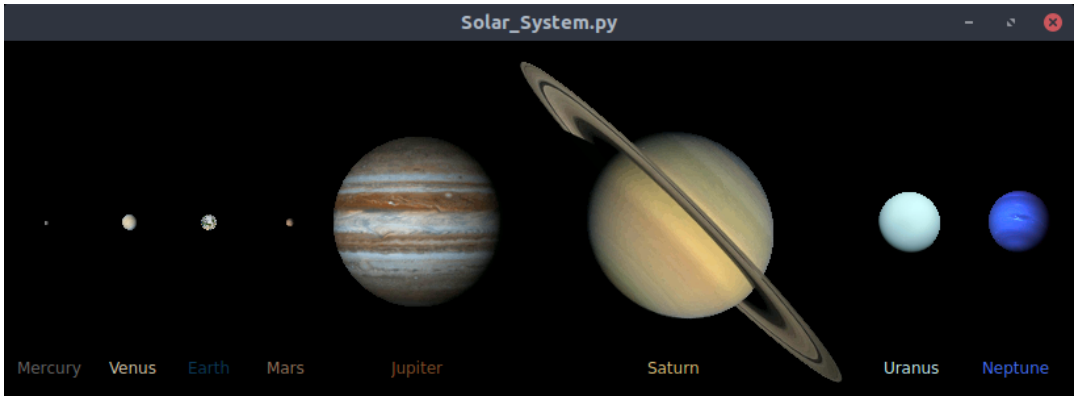


Figure 8: *"The Planets to Scale" opens this window, which shows approximations of the sizes of the planets with respect to each other.[4]*

I decided to add this window since I had spent a lot of time making the images to scale and finding the colors to associate with them. I did this so that I could create a simulation similar to the one for the Earth-like planets, but for all eight planets of our solar system. This however required a lot of computational power, so could not be run on my computer.

### 2.5 Other Options

The "LaTeX Documentation" button in the main menu clearly links to this document, as you probably found out by pressing it to get here! The "Stages of Making This Project" was something I added for humor, so I shall let you discover that on your own. I also added some "Credits", as I felt there should be a place within the program that gives thanks to the sources of the images and information. Clicking on the credits button will show this window:

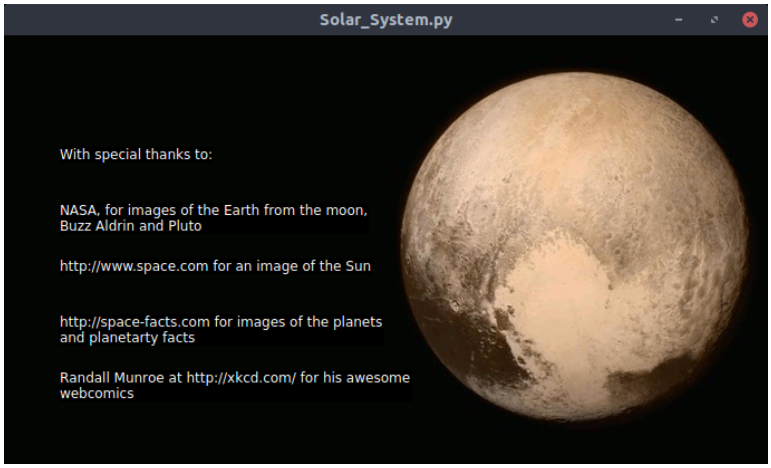


Figure 9: *"Credits" shows a window with special thanks to the sources of my images and information.[7]*

## 3 Improvements

As mentioned in section 2.2, the "Show Solar System" button has to be pressed twice after the first plot in order to create a new simulation. This is something I would have liked to be able to fix, and given more time I think it may have been possible.

In section 2.4 I mentioned another modification I would have liked to make: a simulation of the entire solar system. However, this was not possible due the computational power required.

Being able to spend more time on the project would have also meant I could have created a more interactive section for the details of each planet. This could have therefore also been potentially more interactive, which I feel would increase the appeal of the program.

Another improvement that could be made given more time would be using the Runge-Kutter numerical method to find the new positions of the planets. This method would give a better approximation for larger time steps (the time step here was a day, so pretty big!).

# References


[1] NASA "Apollo 8 Mission image, Earth over the horizon of the moon" Date Accessed: 8 Dec. 2016. https://images.nasa.gov/#/details-as08-14-2383.html

[2] Chris Jones "Planet Facts" Date Accessed: 8 Dec. 2016 http://space-facts.com/planets/

[3] RoyalBlueIV "Sun Transparent Background" Date Accessed: 8 Dec. 2016. http://royalblueiv.deviantart.com/art/Sun-Transparent-Background-Sun-world-408332527

[4] Chris Jones "Transparent Planet Pictures" Date Accessed: 8 Dec. 2016. http://space-facts.com/transparent-planet-pictures/

[5] Credit given to NASA/SOHO Date Accessed: 8 Dec. 2016.http://www.space.com/11663-comet-hits-sun-solar-explosion.html

[6] Date Accessed: 8 Dec. 2016 http://pasaulis.lrytas.lt/rytai-vakarai/rusai-eme-abejoti-ar-jav-astronautai-tikrai-buvo-menulyje.htm

[7] Credit to New Horizons Date Accessed: 8 Dec. 2016 http://i.imgur.com/62b2y9Z.png

[8] Randall Munroe (various links were used, all of which can be found in the code) Date Accessed: 8 Dec. 2016 http://xkcd.com/