VPython Modeling

1.1 What is Computational Science?

Computational science involves using mathematical models and computer programs to improve our understanding of the way the world works. There are three parts to the process:

- Find a complete mathematical description of the system that will include all the behavior you are interested in.
- Construct an efficient computer code that solves these equations
- Extract results and predictions that can be compared with real world, with experiments or with other scientific theories.

These steps are repeated to refine both the computer program and our understanding of the problem.

You now should have the skills to create the more complex program that will simulate an object under the influence of general gravitation.

VPython simulation of Apollo Flight
(General Gravitation and Keplers Laws)

This laboratory introduces gravitational motion and explains with the help of examples, motion under the gravitational force. You should have completed the Vpython tutorial

1 Newton's Law of Gravitation

Before we start the simulation we need to understand the concept of gravitation force given by Sir Isaac Newton in 1665.

Newton was the first to show that the force that holds the Moon in its orbit is the same force that makes an apple fall. Newton concluded that not only does the Earth attract an apple and the Moon but every body in universe attracts every other body, this phenomena of bodies to move towards each other is called *gravitation*. Quantitatively, Newton proposed a law, which is famously called as *Newton's Law of Gravitation*. This law states that every particle attracts any other particle with a gravitational force whose magnitude is given by

$$F = \frac{Gm_1m_2}{r^2}\hat{\mathbf{r}}$$

here m_1 and m_2 are the mass of the particles, $r = r_2 – r_1$ is the distance between them, r_{12} is a vector of unit length pointing from r_1 to r_2 and G is the gravitational constant, whose value is known to be

$$G = 6.67 \times 10^{-11}\ \text{N m}^2/\text{kg}^2$$

The force F acting on the two particles is opposite in direction and equal in magnitude.

2 Simulating Flight Path of Apollo 13

To help us visualize and make this tutorial more interesting we will simulate the flight path of Apollo 13 when subjected to the Earth's gravitational force. We will need to calculate the parking orbit (orbit of the space craft to revolve around the Earth) of Apollo 13.

For successfully implementing the simulation we will need to know the acceleration, position and velocity of Apollo 13, which can be respectively found by equation (1.1) and the equations of motion learnt in the projectile tutorial.

2.1 Initial attributes and display window

In every computer we need to specify some basic variables without which a computer program fails to run. In this and the next section we will fill in these attributes,

```
from visual import *

G=6.67e-11
dt=10
```

Here `G` is the gravitational constant and `dt` is the time interval.

In VPython we can display various windows, but initially there is one Visual display window named scene. Any objects we begin creating will by default go into the scene. In this program we embellish the display window by modifying the attributes with the help of the display function.

```
scene = display(title = 'Earth and Apollo 13',width = 800, height = 600)
```

Here we create a visual display window 800 by 600, with 'Earth and Apollo 13' in the title bar and with a background color of cyan filling the window. Various other attributes can also be specified. For more information visit the documentation of VPython in www.vpython.org

2.2 Attributes of Apollo 13 and Earth

Now we specify the attributes concerning the earth and apollo 13 needed for the simulation. In order to simplify the problem we will assume that the Apollo spacecraft starts in a circular orbit at a height of 173 km above the earth. Setup the initial position of the Apollo so it is located 173 km above the earth in the x direction. From newton’s laws derive the required velocity for the spacecraft so it will remain in a stable circular orbit.

This is done as:

```
###################################################
#   Declaration of Variables                      #
###################################################

#Declaring variables for earth
earth= sphere(pos=(0,0,0), color=color.green)
earth.mass=5.97e24  #mass of the earth
earth.p = vector(0,0,0)*earth.mass
earth.radius = 6378.5e3  #radius of the earth

#Declaring variables for apollo
apollo=sphere(pos=(??????,0,0),radius = 100e3,color=color.red)
apollo.mass = 31.3e3 #mass of apollocraft
apollo.p=vector(0,???????,0)*apollo.mass
apollo.radius= 100e3

#Creating the trails for graphing the orbital path of Apollo
apollo.orbit=curve(pos=[apollo.pos],color=apollo.color,radius = 1e4)
```

2.3 Labeling the Objects

VPython allows us to label objects in the display window with the label() command. In our program we label Earth and Apollo in the following manner.

```
#Labelling Earth and Apollo
earth.name = label(pos = earth.pos,text='Earth',xoffset = 10, yoffset = 10, space =
earth.radius, height = 10, border = 6)
apollo.name = label(pos = apollo.pos,text='Apollo13',xoffset = 10, yoffset = 10,
space = apollo.radius, height = 10, border = 6)
```

A unique feature of the label object is that several attributes are given in terms of screen pixels instead of the usual "world-space" coordinates. For example the height of the text is given in pixels, with the result the text remains readable even when the object is moved far away.

2.4 Acceleration of Apollo 13

To calculate the position and velocity of Apollo 13 as it moves around the earth we will use a momentum formulation to determine how the motion of the spacecraft evolves over time.

The momentum of an object is determined by the forces on it, which in turn depends on its position compared to other objects. For example the force of gravity between two objects in space is given by equation (1). The change in momentum of the body can be calculated by

$$F = \frac{Gm_{earth}m_{apollo}}{r^2}\hat{\mathbf{r}}$$

$$\Delta p = Fdt = \frac{Gm_{earth}m_{apollo}}{r^2} * dt\hat{\mathbf{r}} = \frac{Gm_{earth}m_{apollo}}{r^3} * dt\mathbf{r}$$

In Vpython we can calculate the acceleration of the two planets can be calculated as:

```
####################################################
#   Loop for moving the Apollo 13 around earth     #
####################################################

while (1==1):
    rate(1000)
    # Calculating the change in p of apollo
    dist = apollo.pos - earth.pos
    force = 6.67e-11 * apollo.mass * earth.mass * dist / mag(dist)**3
    apollo.p = apollo.p - force*dt
```

Here `mag` gives the magnitude of the distance between the two objects.

In our program we will need to insert this in the while loop part of the program because the acceleration of Apollo 13 will constantly change with respect to time. The condition `while (1==1):` specifies the simulation to continue forever, as the condition 1==1 can never become false. We can exit the program by just closing down the display window.

2.5 Position calculations

We will determine the new position of Apollo 13 at some later point in time. This is done in the form of

```
    #Calculating the updated position of the Apollo
   apollo.pos = apollo.pos + apollo.p/apollo.mass * dt
```

We also need to update the orbit curve and the label:

```
   apollo.orbit.append(pos=apollo.pos)
   apollo.name.pos = apollo.pos
```

Now we have managed to complete our simulation of Apollo 13, around the earth, due to earth's gravitational force.

NOTE: We have ignored the force exerted by Apollo 13 on the earth because of its small magnitude. This is fine for a spacecraft, but in general we cannot ignore the forces moons exert on planets and vice versa.

Exercise:

Modify your program so that it accounts for the change in momentum of the earth due to the Apollo spacecraft. To test your program, increase the mass of the Apollo spacecraft to 31.3e23. If your program is correct, the two should orbit the center of mass of the system.

3 Gravitational Interaction between Three or more Bodies

The situation for with gravitational interactions between more than two objects/planets is more complicated than that for two planets. We need to use the principle of superposition of gravitational forces

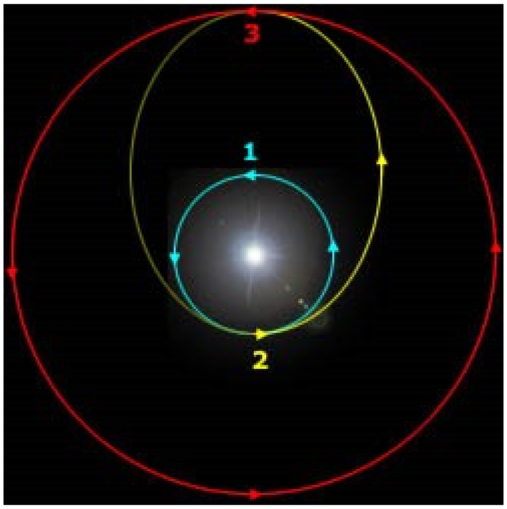
$$F_1 = F_{12}+F_{13}+F_{14}+\ldots+ F_{1n} \qquad (2)$$

Here F_1 is the net force on planet 1 and F_{1n} is the force exerted on particle 1 by particle n, which is calculated by the gravitational force formula given in equation (1).

4 The Earth, Apollo 13 and the Moon

Now we will take our simulation a step further and will now try to simulate how Apollo 13 escaped from it's parking orbit and went to Moon.

After Apollo 13 was launched, it orbited the earth 1.5 times, which is roughly 2.5hrs. after being launched from earth. At this time the translunar injection(TLI) took place. The translunar injection (ignition of engines designed to escape earth) accelerates the spacecraft such that its velocity changes from its initial value to some new velocity. The direction of velocity will be determined by the current orientation of the spacecraft. It is therefore important to initiate the TLI in precisely the correct moment to put Apollo 13 on the correct path to the moon. (As shown in the picture at right)



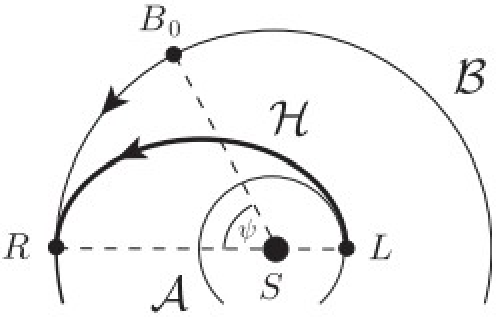
Hohmann transfer orbit

In astronautics and aerospace engineering, the **Hohmann transfer orbit** is an orbital maneuver that, under standard assumption, moves a spacecraft from one circular orbit to another using two engine impulses. This maneuver was named after Walter Hohmann, the German scientist who published it in 1925.

The Hohmann transfer orbit is one half of an elliptic orbit that touches both the orbit that one wishes to leave (labeled *1* on diagram) and the orbit that one wishes to reach (*3* on diagram). The transfer (*2* on diagram) is initiated by firing the spacecraft's engine in order to accelerate it so that it will follow the elliptical orbit. When the spacecraft has reached its destination orbit, it has slowed down to a speed not only lower than the speed in the original circular orbit, but even lower than required for the new circular orbit; the engine is fired again to accelerate it again, to that required speed.

Since this transfer requires two powerful bursts, it can not be applied with a low-thrust engine. With that, the circular orbit can be gradually enlarged.

Hohmann transfer orbits also work to bring a spacecraft from a higher orbit into a lower one – in this case, the spacecraft's engine is fired in the opposite direction to its current path, causing it to drop into the elliptical transfer orbit, and fired again in the lower orbit to brake it to the correct speed for that lower orbit.



Two planets move on the circular orbits A and B. A spacecraft is required to depart from one planet and rendezvous with the other planet at some point of its orbit. The Hohmann orbit H achieves this with the least expenditure of fuel.

In our analysis we will neglect the time during which the rocket engines are firing so that the engines are regarded as delivering an impulse to the spacecraft, resulting in a sudden change of velocity. After the initial firing impulse, the spacecraft is assumed to move freely under the Sun’s gravitation until it reaches the orbit of the second planet, when a second firing impulse is required to circularize the orbit. This is called a **two-impulse transfer.**

If the two firings produce velocity changes of $|\Delta V^A|$ and $|\Delta V^B|$ respectively, then the quantity Q that must be minimized if the least fuel is to be used is

$$Q = |\Delta \boldsymbol{v}^A| + |\Delta \boldsymbol{v}^B|$$

The orbit that connects the two planetary orbits and minimizes Q is called the **Hohmann transfer orbit** and is shown. It has its perihelion at the lift-off point L and its aphelion at the rendezvous point R. It is not at all obvious that this is the optimal orbit; and a nice proof was published by Prussing in the Journal of Guidance Vol 15 No. 4 . However, it is quite easy to find the properties of this orbital transfer.

Since the perihelial and aphelial distances in the Hohmann orbit are A and B (the radii of the orbits of Earth and Jupiter), it follows that

$$A = a(1 - e), \qquad B = a(1 + e),$$

so that the geometrical parameters of the orbit are given by

$$a = \tfrac{1}{2}(B + A), \qquad e = \frac{B - A}{B + A}.$$

The angular momentum L of the orbit is then given by the L-formula to be

$$L^2 = \frac{\gamma b^2}{a} = \gamma\left(1 - e^2\right)a = \frac{\gamma B A}{B + A}$$

where $\gamma = M_0 G$.

From L we can find the **speed** V^L of the spacecraft just after the lift-off firing, and the **speed** V^R at the rendezvous point just before the second firing. These are

$$V^L = \left(\frac{2\gamma B}{A(B + A)}\right)^{1/2}, \qquad V^R = \left(\frac{2\gamma A}{B(B + A)}\right)^{1/2}$$

The **travel time** T, which is half the period of the Hohmann orbit, is given by

$$T^2 = \frac{\pi^2 a^3}{\gamma} = \frac{\pi^2 (B + A)^3}{8\gamma}$$

Finally, in order to rendezvous with Jupiter, the lift-off must take place when Earth and Jupiter have the correct relative positions, so that Jupiter arrives at the meeting point at the right time. Since the speed of Jupiter is $(\gamma/B)^{1/2}$ and the travel time is now known, the angle ψ in the figure must be

Numerical results for the Earth–Jupiter transfer

In astronomical units, $G = 4\pi^2$, $A = 1$ AU and, for Jupiter, $B = 5.2$ AU. A speed of 1 AU per year is 4.74 km per second. Simple calculations then give:

(i) The travel time is 2.73 years, or 997 days.

(i) V^L is 8.14 AU per year, which is 38.6 km per second. This is the speed the spacecraft must have after the lift-off firing.

(i) V^R is 1.56 AU per year, which is 7.4 km per second. This is the speed with which the spacecraft arrives at Jupiter before the second firing.

(i) The angle ψ at lift-off must be 83°.

The speeds V^L and V^R should be compared with the speeds of Earth and Jupiter in their orbits. These are 29.8 km/sec and 13.1 km/sec respectively. Thus the first firing must boost the speed of the spacecraft from 29.8 to 38.6 km/sec, and the second firing must boost the speed from 7.4 to 13.1 km/sec. The sum of these speed increments, 14.5 km/sec, is greater than the speed increment needed (12.4 km/sec) to escape from the Earth's orbit to infinity. Thus it takes more fuel to transfer a spacecraft from Earth's orbit to Jupiter's orbit than it does to escape from the solar system altogether!

Calculate the minimum change in velocity the Apollo spacecraft needs in order to transfer from the low earth orbit you calculated earlier to lunar orbit.

In this exercise we will simulate this tranlunar injection (TLI) and Apollo 13 path to the moon and make it transit to the moon and enter a stable orbit.

To compute the path for the passage from the earth's parking orbit to the moon's orbit, we use the initial condition of the spacecraft in earth orbit as a was animated in the first part of this tutorial. We will apply the first delta v which will transfer the apollo spacecraft from earth orbit to lunar orbit. Once the spacecraft reaches its point of closest approach to the moon, we will apply the second delta v, to put the spacecraft in a stable orbit.

5 Logical Reasoning of the Simulation

Now we will briefly touch the logical reasoning steps to the simulation:

1. We will need to know the parameters of the moon, time taken to complete one orbit and angle at which the TLI takes place.
2. We will need to calculate the Apollo 13 Δp due to gravitational force from the earth and moon.
3. We will need to keep a check on the angle for the TLI, as at that specified angle the velocity and radius of the spacecraft so the first delta v can be applied at the appropriate time. But please note this method of increasing the velocity of the spacecraft is used to make the simulation easy to understand. In reality the spacecraft gradually increases it's velocity to the specified velocity.

4. We monitor the angle between the spacecraft and the moon and determine when to apply the second delta v to park the spacecraft in a stable orbit around the moon. At the appropriate position, we apply the second thrust to circularize the orbit of the spacecraft around the moon.

6 Begin with the Simulation

```
#Declaring variables for Moon
moon=sphere(pos=(384e6,0,0),color=color.black)
moon.velocity=vector(0,0,0)
moon.mass=7.3480e22
moon.radius=2.238e6
```

6.2 Apollo Acceleration

With the introduction of the moon into the simulation, the Apollo 13 experiences gravitational from earth and moon. So the net force on Apollo 13 is,

$$F_{apollo} = F_{moon}+F_{earth}$$

Be sure to update your code to reflect the new net force.

6.4 Applying the TLI

This part of the simulation is similar to the previous simulation where we only had the Earth and Apollo 13. The only difference is we keep a check on the angle Apollo 13 is making and when this angle equals the angle for TLI, we switch to the TLI part of the simulation. We perform this check by using an if statement of VPython and checking when the angle of Apollo becomes equal to angle of TLI.

Follow the Logical reasoning explained earlier and implement the TLI thrust at the appropriate angle. The simulation should execute the second delta v if the x position of the craft ever gets much larger that the x position of the moon and the spacecraft is at the appropriate angle.

7 Conclusion

We have simulated the motion of Apollo 13 around the Earth and performing a Translunar Injection to escape the Earth's force and travel towards the Moon. The equations of motion have been used to calculate the updated velocity and position of Apollo 13.

<u>Now:</u>

- Make the moon orbit around the earth. The only numbers you may have in your program are G, the moons semimajor axis 384,748 km and eccentricity: 0.0549006 and the masses and radii of the earth and moon! Remember that the earth and moon orbit around the center of mass of the earth moon system. Have both the moon and the earth leave a track indicating their path. How does the earth move due to the effect of the moon?
- Recalculate the required delta v and angles to make the Apollo spacecraft undergo a transfer orbit to the moon while the moon is moving. (Give a detailed derivation of how you obtained the launch angle. Hint since the speed of the moon is $(\gamma/B)^{1/2}$ and the travel time can be found from Kepler's 3rd law, the angle ψ can be found!)
- After 2 days orbiting the moon, have the Apollo craft transfer back to earth orbit.