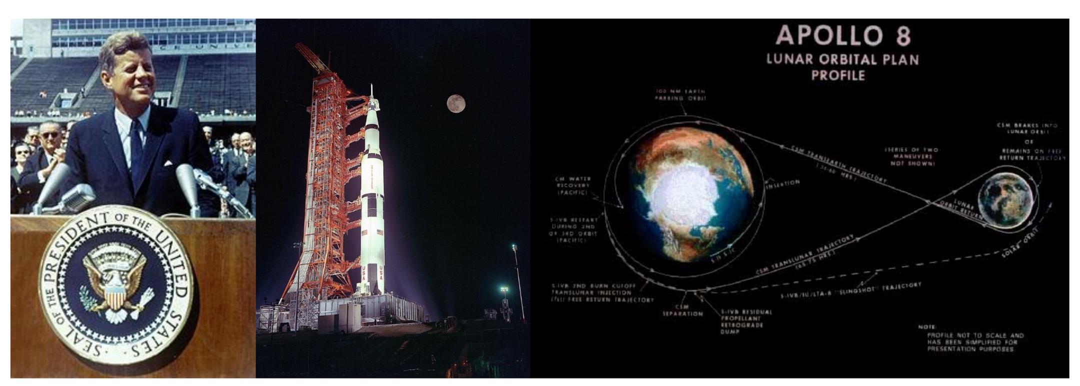
To the Moon and Back—Simulating the Trajectory of a Multi-Stage Rocket Similar to Saturn V in an Apollo 8 Mission Analogue

Kirk Long—PHYS 325—Boise State University

Introduction

In John F. Kennedy's famous "moon speech", he declared that "we choose to go to the moon in this decade and do the other things, not because they are easy, but because they are hard." Landing on the moon was a capstone achievement for both the United States and humanity as a whole, and to this day remains the farthest living humans have ventured from home. Solving these kinds of problems is incredibly interesting and important—if our species has any hope of surviving on a cosmic scale we must be capable of shedding the confines of our blue marble. Apollo 8 was the final test before the moon landing, and was the first time humanity proved we could venture out beyond the Earth's gravity well. In its simplest form (which is what I modelled), generating an orbital profile for a mission like this involves simulating the motion of three bodies and the gravitational interactions between them. I created a crude, two-dimensional simulation that models the motion of the Earth, the moon, and a rocket similar to the real Saturn V rocket that first took humans around the moon in late 1968.



L to R: John F. Kennedy during his famous "moon speech" (1). Apollo 8 on the launch pad (2). A simplified orbital profile for the real Apollo 8 mission (3). Image credits: NASA.

Theory

The gravitational interaction between two bodies (where masses are relatively small and simplified Newtonian physics are thus valid) is given by:

$$|F| = G\frac{m_1 m_2}{r^2}$$

In the case where one body is significantly less massive than the other, the acceleration will be towards the larger body and can be modelled using just the mass of the larger body:

$$a = -G\frac{M}{r^2}\hat{r}$$

This equation allows us to simply model the orbits of both the moon and the rocket while it is not under thrust.

While the rocket is under thrust, however, its velocity vector is undergoing an additional change due to Newton's third law, and this "delta V" is given by the Tsiolkovsky rocket equation:

$$\Delta v = v_e \ln\left(\frac{m_0}{m_f}\right)$$

Where v_e is the exhaust velocity and $\frac{m_0}{m_f}$ is the ratio of initial to final mass, or how much fuel was burned. The real Saturn V was about 90% propellant on the launch pad, and in my simulation the rocket was close to this ratio as well.

By itself the three-body problem is not solvable analytically, but it is even more difficult to solve when one of the bodies is capable of powered flight. This is a system of differential equations that requires a solution via numerical methods, where the accuracy of said solution is highly dependent upon the size of time step.

Methodology

The equation for gravitational acceleration can be transformed into two ordinary differential equations and placed in a Cartesian coordinate system. For the moon we only need to consider the gravitational interaction between it and the Earth, where the Earth is much more massive:

$$v_x = \frac{dx}{dt} \qquad v_y = \frac{dy}{dt}$$

$$\frac{dv_x}{dt} = \frac{d^2x}{dt^2} = -G\frac{M_e x}{|r|^3} \qquad \frac{dv_y}{dt} = \frac{d^2y}{dt^2} = -G\frac{M_e y}{|r|^3}$$

Where r is the distance from the moon to Earth, and x and y are the moon's coordinates. For the rocket we need to consider the pull of both the Earth and the moon:

$$v_x = \frac{dx}{dt} \qquad v_y = \frac{dy}{dt}$$

$$\frac{dv_x}{dt} = \frac{d^2x}{dt^2} = -G\frac{M_e x_r}{|r_e|^3} - G\frac{M_m(x_r - x_m)}{|r_m|^3} \qquad \frac{dv_y}{dt} = \frac{d^2y}{dt^2} = -G\frac{M_e y_r}{|r_e|^3} - G\frac{M_m(y_r - y_m)}{|r_m|^3}$$

Where r again is the distance (the subscript denotes whether it is the rocket-Earth or rocket-moon distance) and x and y are the rocket's or moon's coordinates. These equations were solved numerically using an explicit implementation of fourth order Runge-Kutta, where r is a vector containing the positions and velocities of both the spacecraft and the moon:

$$k_1 = \Delta t \times f(r, t)$$
$$k_2 = \Delta t \times f\left(r + \frac{k_1}{2}, t + \frac{\Delta t}{2}\right)$$
$$k_3 = \Delta t \times f\left(r + \frac{k_2}{2}, t + \frac{\Delta t}{2}\right)$$
$$k_4 = \Delta t \times f(r + k_3, t + \Delta t)$$
$$r += \frac{(k_1 + 2k_2 + 2k_3 + k_4)}{6}$$

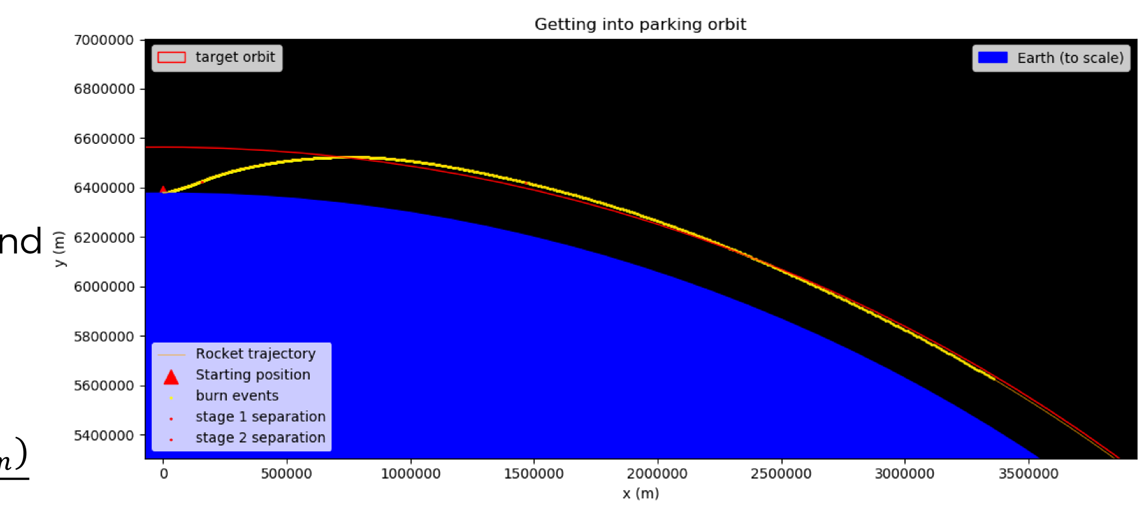
The computer performed these operations from an initial time to a final time with a set step size, and after each step the coordinates of both the rocket and the moon were recorded. While the rocket was under thrust there was an additional check implemented, which updated the vehicle's velocity according to the Tsiolkovsky rocket equation before the rest of the algorithm was performed. Note that in this case the function does not depend on t, but it is left here to better express the more general Runge-Kutta method. While the rocket was burning fuel this process ran once a second, and while it was drifting 10,000 operations were performed over a specified time interval. No adaptive step size was explicitly implemented, but when the rocket was close to the Earth or the moon I performed said 10,000 operations over a shorter total interval than when the rocket was coasting between the two bodies. The error here scales at each step with the fourth power of the time interval (and accumulates the longer the program runs), so it is very important to choose a sufficiently small step size.

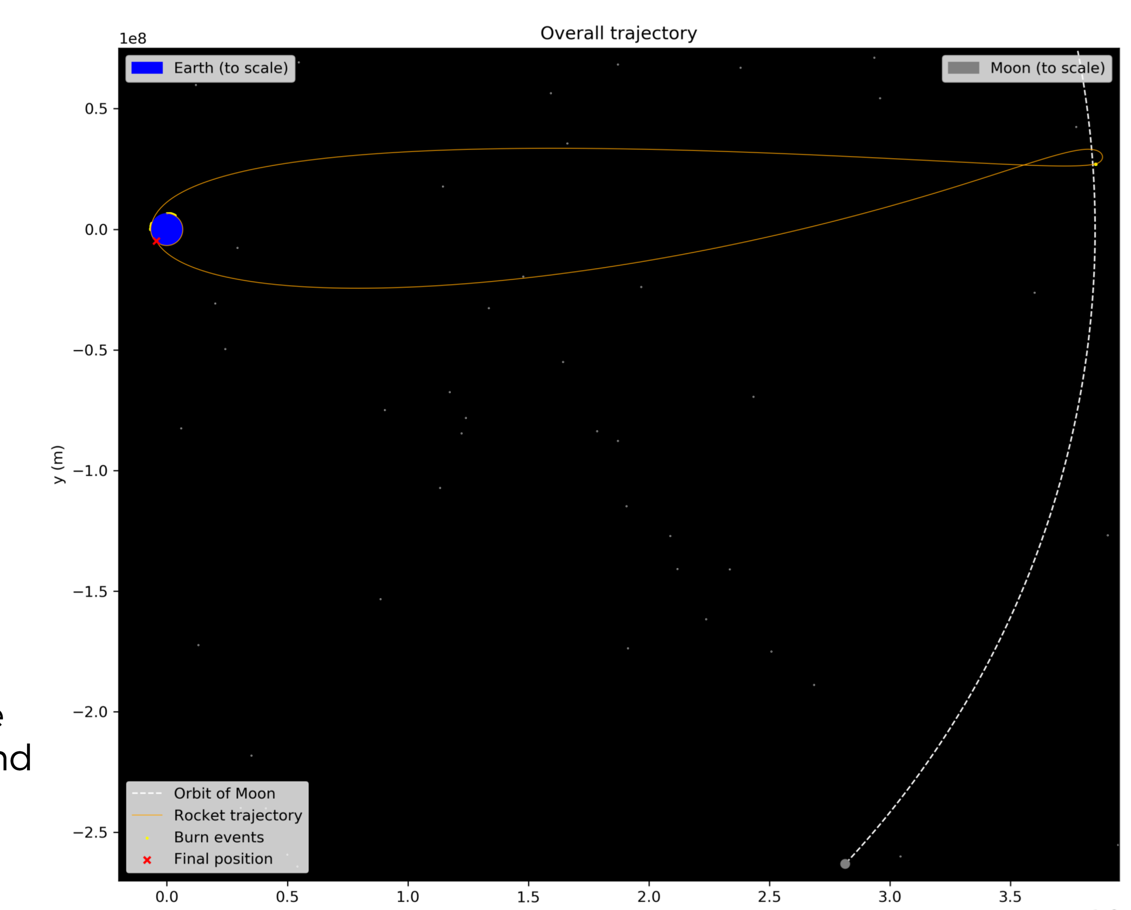
This was the easiest part of the program, however. Controlling the rocket's trajectory required implementing a rigorous "flight computer," which controlled the direction thrust was applied. While the rocket's altitude was less than 5 km it burned "straight up" (perpendicular to Earth's surface), and above this height it entered a slow pitch program that eventually aligned the thrust vector with the tangential vector, so that most of the rocket's energy was devoted into achieving the tangential velocity necessary to keep it continually falling yet "missing" the Earth. The real Saturn V entered a parking orbit at a height of 186 km (above the Earth's surface) and required a tangential velocity of about 8 km/s. My "flight computer" was able to achieve this same orbit with relatively high accuracy (see results).

After achieving this desired parking orbit I initiated the translunar injection burn. The simulated journey to the moon took a little more than two days (similar to Apollo 8). Once near the moon an opposing burn was initiated to ensure that the rocket would "slingshot" around the moon instead of escaping into deep space. After this gravity assist the rocket returned back to Earth, a total journey time of about six days (which is very similar to the time spent in space by Apollo 8, which did not land on the moon but instead followed a more flyby-like trajectory akin to the one implemented here).

Results

Below are two graphical representations of the rocket's trajectory—the first shows the initial climb into space (the most computationally difficult and intensive portion of the simulation) while the second shows the overall path of the rocket and moon over the six day mission time period.





Discussion and Analysis

The orbital trajectory calculated by my program is physically plausible and matches pretty well with the timescale and orbit of the Apollo 8 lunar mission. However, this simulation is a very idealized case and has the following limitations:

1. The Earth, moon, and rocket were all modelled as point masses with no physical dimensions.
2. There was no account for drag on the rocket when leaving the atmosphere, although for the entire first stage burn I used the published value for the specific impulse (and thus exhaust velocity) at sea level.
3. Although the moon/Earth system is relatively planar, there is still a z component that is not accounted for in this idealized two-dimensional simulation.
4. To make the math easier when under burn the time step was always one second, when it probably should be smaller in these spaces where the rate of change is highest.
5. Despite these idealizations, my simulated rocket required slightly more fuel mass than the actual Saturn V needed, although I found some variance in published historical numbers. I believe this error is largely attributed to the choice of initial pitch angle and the fact that—because I modelled the rocket as a point mass—the simulation had to expend extra fuel to pitch the rocket over when in real life gravity turns the rocket by generating torque on the rocket body.

Conclusions

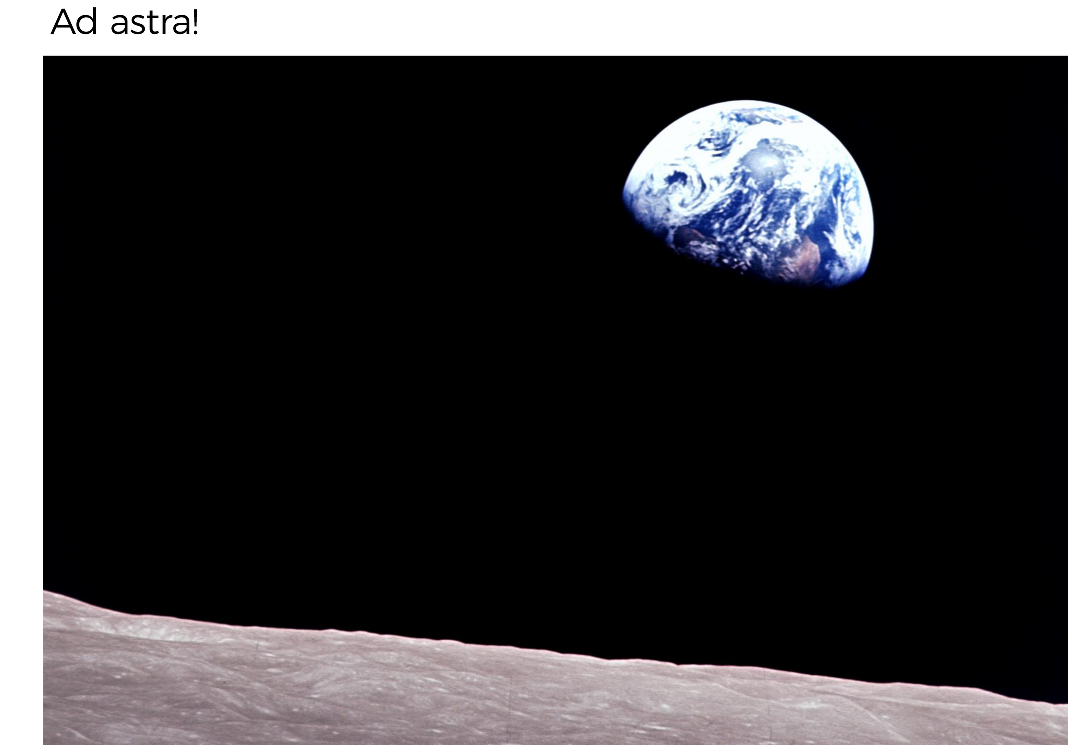
Rocket science is hard.

I have certainly gained newfound respect for the many scientists and engineers who first calculated these trajectories to get us to the moon. One interesting comparison of note—although my simulation is certainly less intense than the models NASA generated in the 60's, it is probably roughly within an order of magnitude or two in terms of computational operations required. On average it took my computer between 30 seconds and a minute to plot the entire trajectory start to finish, where just 50 years ago the same computations would have taken an entire building of "supercomputers" significantly more time–and some of those early "computers" (as portrayed most famously in the movie *Hidden Figures*) were people!

I have also learned that simulating these kinds of problems isn't something only NASA is capable of, as my program generated results that are consistent with historical ones. Given more time I am confident that I could fix the problems outlined in the discussion section and gain results that are even more accurate. After all, NASA employed thousands of incredibly gifted and talented people for nearly a decade to do this kind of thing, and I'm just a lone physics major with limited Python experience who spent a few weeks doing this.

I expect that I could also modify my program to simulate other lunar mission trajectories, including the landings of the later Apollo missions. It would also be relatively easy to modify the program to calculate orbital trajectories to other bodies in the solar system, assuming non-relativistic physics were still at play.

Ad astra!



"Earthrise"—taken by astronaut Bill Anders aboard Apollo 8. Image credit: NASA.

References and Acknowledgements

Many thanks to Dr. Olga Goulko, whose expertise and class made this possible! I must also acknowledge the Python programming language, including the matplotlib, numpy, and scipy modules.

Newman, M. (2013). *Computational Physics.* Scotts Valley, CA: CreateSpace Independent Publishing Platform

Apollo 8: The second mission. (n.d.). Retrieved from https://history.nasa.gov/SP-4029/Apollo_08a_Summary.htm

Saturn V Press kit. (1968). Retrieved from https://history.msfc.nasa.gov/saturn_apollo/saturnv_press_kit.html