Discrete-Time Trajectory Optimization In Space: Low-Thrust Satellites, Orbit Lowering Maneuvers, And Fuel Constraints

Vasanth Gogineni (student)[1 *]

1. Greenwood High International School, Bangalore, Karnataka, 560087, India

* Corresponding author email: vhpgoginen@gmail.com

Abstract

The scarcity of resources in spacecraft makes some missions and maneuvers quite challenging. However, many of these missions have an abundance of another factor: time. Low-thrust spacecraft can follow trajectories that guide them to their destination using very little fuel. Instead of burning sharply to make their orbit smaller, for example, they burn fuel slowly but for a longer time. An algorithm can be developed that allows the optimization of thrust profiles of low-thrust satellites around central bodies(planets or stars), minimizing fuel use for the reduction of orbit size around the central body. Using fixed time horizons, the dynamic simultaneous equation algorithm in this paper develops thrust profiles that allow low-thrust trajectory optimization using discrete time. The paper uniquely explores the development and experimentation of this algorithm in Python code for orbit-lowering maneuvers, using the GEKKO optimization suite. It was concluded that the thrust profiles demonstrated that the combination of prograde and retrograde burns provided optimum fuel use, satellites further away from central bodies require more fuel, and satellites with lower maximum power require more complex prograde and retrograde maneuvers than higher-power satellites. This translated to the more complex maneuvers requiring more iterations and solution time in the algorithm.

Keywords

Physics and Astronomy; Computational Physics; Trajectory Optimization; Optimal Control; Mechanics

Introduction

Traditionally, spacecraft have had large engines burning fuel at a high rate. However, the increasing length and distance of each space mission with space agencies pushing the boundaries of exploration may increase costs and create a need for a sustainable fuel source. With space colonization and other ambitious goals for the future of space exploration, it is imperative that space exploration does not pollute the Earth, deplete resources, or be infeasible due to a lack of fuel efficiency. Low-thrust engines can provide the answer to this problem, as they are more efficient and can be fueled by a sustainable source. For example, ion-thrusters have shown fuel efficiencies of over 90%, making the total cost of fuel cheaper and the mission more affordable.[1]

Therefore, in order for space missions to be successful, it is vital that low-thrust maneuvers and trajectories minimize the use of fuel to avoid defeating the purpose of efficiency. Trajectory optimization involves finding the best sequence, direction, and magnitude of burns in order to minimize fuel use. This is done by experimenting with thrust profiles to find the one that uses the least fuel.

In the past, a great amount of research has been conducted to find the best method for trajectory optimization. Optimization techniques for spacecraft trajectories include continuous dynamics.[2] Here, the state variables of the system change instantaneously and not at divided intervals of time. This is opposed to the discrete dynamics used in the algorithm in this paper, which uses a finite number of time points instead. Therefore, the continuous dynamics method is far more computationally demanding, although it is more accurate. For this reason, discrete dynamics are used in the algorithm in this paper.

Given a system with a central star, a low thrust spacecraft can burn fuel to escape a planet it may be orbiting and further control its thrust to reach a certain minimum distance to the star using the least fuel possible. This fairly complex problem needs to be broken down in order to build an algorithm capable of handling several similar situations in space.

First, the orbital mechanics were simulated using VPython,[3] a simulation tool in Python. Next, using an optimization tool named GEKKO,[4] the equations verified and applied in the simulation were then adapted to suit the fuel-constrained optimization problem. The new GEKKO model was manipulated to handle various satellites and planets as needed and was finally used to minimize fuel use.[4] With the rapid growth of the Python programming language, this algorithm in Python code is unique and relevant due to the growing need for wider access to optimization tools to calculate optimum satellite ephemeris around Earth or other parameters.

This complex system needs to consider the fuel constraints of satellites, the trajectory of satellites, planet positions, and other factors. This algorithm, thus, provides a quick yet rigorous method of estimating optimum thrust profiles to plan a space mission for, for example, an electric-propulsion spacecraft.

The algorithm used in this paper is a dynamic optimization algorithm that uses discrete time for trajectory optimization. It involves simultaneous equations that define the orbital mechanics. This paper uniquely and specifically examines a new discrete-time and simultaneous differential equation trajectory optimization algorithm in terms of orbit-lowering maneuvers.

Methods

Problem Statement

I. The first part of the problem involves defining the equations of orbital mechanics to later derive equations suitable for the optimization problem.

$$F = \frac{Gm_1m_2}{R^2}$$

$$p = mv$$

$$\frac{dr_x}{dt} = p_x/m_2, \frac{dr_y}{dt} = p_y/m_2$$

$$\frac{dp_x}{dt} = F_x + T_x, \frac{dp_y}{dt} = F_y + T_y$$

Simulating this situation in VPython for all planets centered around the sun,[3] we can verify the functionality of these basic equations as shown in Figure 1.

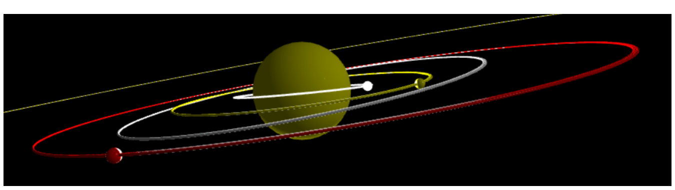


Figure 1. Image of the simulation of the orbital mechanics of all planets and the Sun in VPython.

The above screenshot of the working simulation allows us to derive further equations for discrete time.

II. The equations for the mathematical optimization problem can now be derived in order to function simultaneously in discretized time.

Splitting F into components for a mathematical simulation:

$$\vec{F} = \frac{-Gm_1m_2}{R^3} \times \vec{r} \rightarrow$$

$$\vec{Fx} = \frac{-Gm_1m_2}{R^3}\times rx \, , \vec{Fy} = \frac{-Gm_1m_2}{R^3}\times ry \, , \vec{Fz} = \frac{-Gm_1m_2}{R^3}\times rz$$

Let $\mu = Gm_1$

If so, $\mu = 2.959\times10^{-4} AU^3/day^2$

Now, $\vec{F} = \frac{-Gm_1m_2}{R^3}\times rx$

Converting force to acceleration:

$$ax = \frac{Fx}{m_2} = \frac{-\mu}{R^3}\times rx \, , ay = \frac{Fy}{m_2} = \frac{-\mu}{R^3}\times ry \, , az = \frac{Fz}{m_2} = \frac{-\mu}{R^3}\times rz$$

T is defined as the ratio of engine thrust to mass

$$|v| = \sqrt{(vx^2 + vy^2 + vz^2)}$$

$$T_x =- T\times\frac{v_x}{|v|} \, , T_y =- T\times\frac{v_y}{|v|} \, , T_z =- T\times\frac{v_z}{|v|}$$

Therefore, the governing differential equations are as follows:

$$\frac{d}{dt}vx = ax \, , \frac{d}{dt}vy = ay \, , \frac{d}{dt}vz = az$$

$$\frac{d}{dt}rx = vx \, , \frac{d}{dt}ry = vy \, , \frac{d}{dt}rz = vz$$

by definition.

III. The **objective function** needs to integrate the thrust with respect to time to obtain the fuel use. This fuel use needs to be minimized for making this low-thrust fuel optimization algorithm. The thrust here is taken as a function of time.

$$min\int_{0}^{t} T(t)dt$$

IV. Constants and initial state

$$r_x(0) = \ -25182934840.587124$$

$$r_y(0) = 132794579028.3616$$

$$p_x(0) = m_1 \times -29802.266813363603$$

$$p_y(0) = m_1 \times -4860.043346427637$$

$$m_1 = 1.989\times10^{30} \text{ [MASS OF SUN]}$$

$$m_2 = 500$$

$$T_{max} = 2\times10^{-5}$$

IV. The **constraints** of the algorithm include the maximum magnitude of thrust and the final orbit radius.

T = Manipulated instantaneous thrust

R = distance of the satellite from the barycenter

f = Final time point

x = required radius of the orbit around the central body

T_{max} = magnitude of maximum instantaneous thrust delivered by the satellite

t = time

$$R(t) = \sqrt{(r_x)^2 + (r_y)^2 + (r_z)^2}$$

$$R(f) < x$$

$$|T| < T_{max}$$

Assumptions

Throughout the problem statement, it has been assumed that certain conditions are true for the simplification of the problem and due to the large scale of the solar system.

A. It must be noted that a key assumption in this algorithm is the proportionality of thrust and fuel use. They are considered proportional due to the high efficiency of low-thrust engines and because the energy loss due to mass and other factors is significantly lower compared to the values for larger propulsion systems.

B. Another important assumption to simplify the problem is that the sun is at the barycenter. This is assumed to be true due to the relative closeness of the sun to the barycenter considering the scale of the solar system.

Computational Experiment Setup

The mathematical form of the optimization problem in the previous section is quite complex to convert to python code to use the GEKKO optimization package.[4] Therefore, we can simplify the conversion by first defining an orbit in GEKKO and then introducing a test case for the optimization problem.[4]

The Python code for the simulations and optimization algorithms can be found in this link: https://github.com/Vasanth-Gogineni/Discrete-Time-Trajectory-Optimization-In-Space.

I. Defining and simulating an orbit

The solver is tuned to sequential simulation independent of a manipulated variable. The time in the GEKKO model is discretized and a time horizon of 1 year is defined.[4] The timescale is initialized to 1/1/2010. Using the Skyfield API file ‘de421.bsp’ to obtain ephemeris data and state vectors,[5] the initial conditions of the orbit are defined.

We can set the number of nodes in each time segment to obtain greater precision. Then, implementing only the 'governing differential equations' mentioned in the previous section, we have a system of equations that simulate the gravitational effect of the Sun on the Earth, producing the orbit shown in Figure 2.

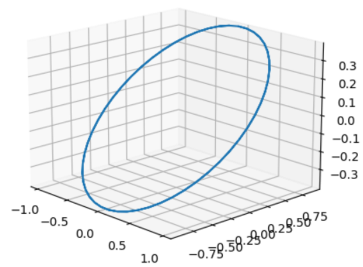


Figure 2. The graph shows the trajectory or orbit of a satellite defined and simulated in GEKKO.

However, underlying intermediate equations are needed to define ax, ay, and az. These intermediate equations are essentially equations that dynamically assign values to explicitly defined intermediate variables using updated values after every iteration to reduce the problem complexity.

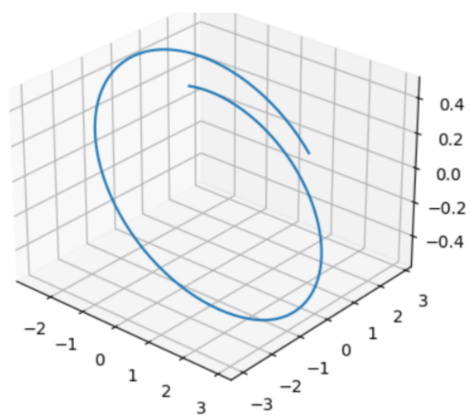
The variables ax, ay, and az should be intermediate variables dynamically assigned values using the equations in the Problem Statement. However, these require R also to be defined, and so the R(t) equation should also be an intermediate.

II. Introducing objective function for optimization in GEKKO.[4]

Applying the structure of the algorithm from the orbit simulation, the solver is tuned to dynamic simultaneous equation solving for control applications. This now allows the manipulated variable(MV), fuel/thrust(used interchangeably), to be introduced. The MV has been constrained with a maximum and minimum thrust suitable for low-thrust engines($2*10^5$ AU/day^2). The lower bound is negated to provide reverse thrust.

The GEKKO model required the objective function to be defined in a split form.[4] With integrated thrust set as a variable, its differential equation assigned it the absolute value of the instantaneous thrust to signify fuel use. Next, a zero-filled array with its last element with value '1' was pointwise multiplied with the integrated thrust function and summed to extract the value of the final and total fuel used. This optimized fuel value and trajectory were plotted using MATPLOTLIB.[6]

To verify the functionality of the trajectory optimization in the algorithm, an experiment with the asteroid Ceres' initial state vectors from NASA's Horizons system and the sun as the central body was performed,[7] as seen in Figures 3-5. The final orbit radius was set to 1.8 AU.



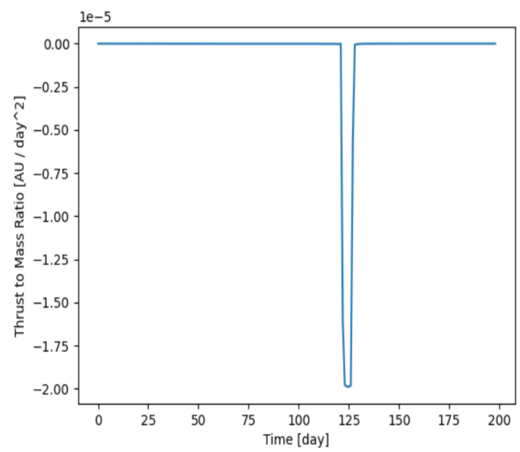


Figure 3 and Figure 4. Figure 3 on the left shows the trajectory of the satellite departing from Ceres. Figure 4 on the right shows the thrust profile or the fuel use with time for the optimized trajectory.

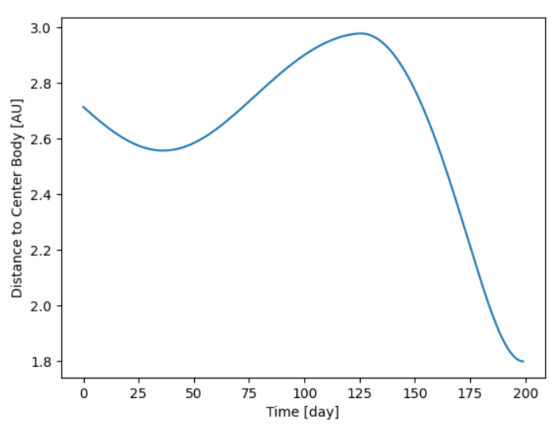


Figure 5. The figure shows the distance to the Sun as time progresses. Note that the final distance in the graph is 1.8 AU.

Here, to change the orbit, R(distance between central body and planet) in Figure 5 was given an inequality constraint to define the radius of the new orbit. Overall, the introduction of the optimization required a complex definition of the objective function and constraints.

Graphs

There are 3 types of graphs produced by the algorithm using MATPLOTLIB,[6] as seen in Figures 3-5. The first graph is a 3D graph showing the exact trajectory of the satellite around the Sun, which is located at (0,0,0). The second graph shows the thrust profile or the thrust-to-mass ratio throughout the time horizon, which is eventually integrated to find the total thrust or fuel use. The third graph shows the position relative to or the distance from the Sun. The 3 graphs will be identified as A, B, and C respectively in the set of graphs for an experiment.

Experiment

The optimization algorithm was first tested to ensure its accuracy. With no thrust acting in Figure 2, the orbit and its inclination is plotted in a 3-dimensional graph. The orbit is elliptical and verifies the accuracy of the algorithm in simulating ephemeris. The plot and data showed that the elliptical orbit makes the Earth move a mean distance of around 1 AU away from the barycenter, which is true by definition of the 'AU'.

The optimization algorithm will consider the following test cases:

I. Different satellites around the Sun

II. Effect of maximum thrust on thrust profile

III. Effect of prograde and retrograde burns

I. Different Satellites around the Sun(Sun as a central body)

In this experiment, the central body is the Sun. Satellites originating from different planets are tested, assuming the satellite has left the planet's gravitational field already. The satellite and Sun system is then used with the optimization algorithm to reduce the orbit size of the satellite around the Sun.

For this experiment, the algorithm has been set up with 0.7 AU as the maximum final displacement from the sun. Since an orbit with an average radius of 0.7 AU is larger than or similar to that of Mercury and Venus, only satellites or spacecraft originating from Earth, Mars, Jupiter, and Saturn are considered to test a consistent type of maneuver(larger to smaller orbit).

An important parameter to note is the absolute value of thrust available. The value is 2×10^{-5} AU.

First, a system with the Sun and a satellite originating from the earth was tested as shown in Figures 1A, 1B, and 1C.

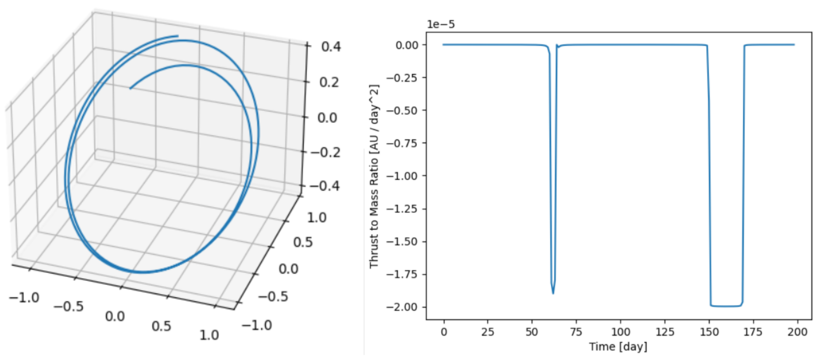


Figure 1A and Figure 1B. Figure 1A(left) shows the trajectory of the satellite departing from Earth in Experiment 1. Figure 1B(right) shows the thrust profile of the satellite departing from Earth in Experiment 1.

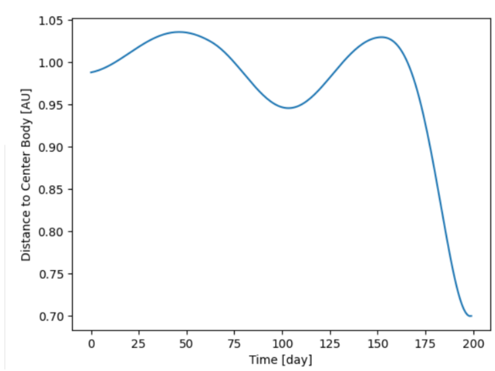


Figure 1C. Figure 1C shows the distance to the Sun of the satellite departing from Earth in Experiment 1.

Further exploring, a system with the Sun and satellite originating from Mars can be tested as in Figure 2A, 2B, and 2C. The initial state vectors can be adjusted using the Skyfield API to test satellites from Jupiter and Saturn.[5]

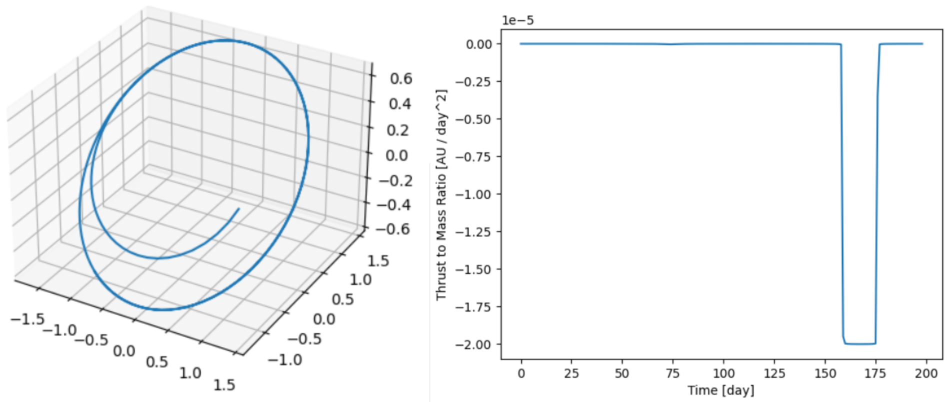


Figure 2A and Figure 2B. Figure 2A(left) shows the trajectory of the satellite departing from Mars in Experiment 1. Figure 2B(right) shows the thrust profile of the satellite departing from Mars in Experiment 1.

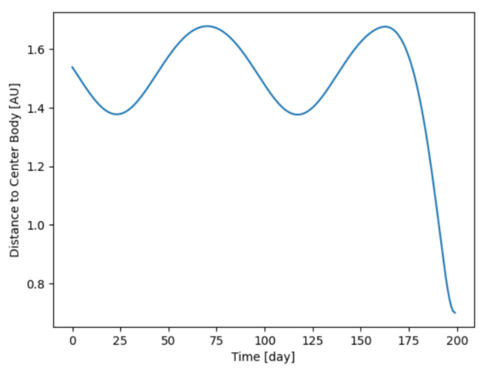


Figure 2C. Figure 2C shows the distance to the Sun of the satellite departing from Mars in Experiment 1.

Figures 3A, 3B, and 3C show a Jupiter Satellite and Sun system.

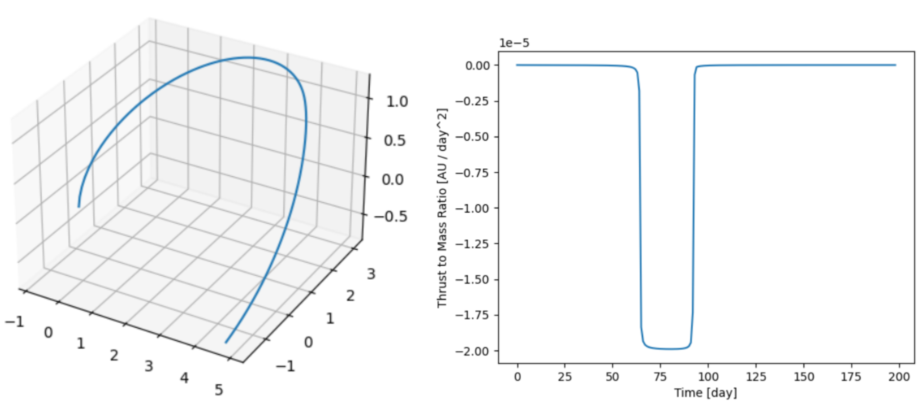


Figure 3A and Figure 3B. Figure 3A(left) shows the trajectory of the satellite departing from Jupiter in Experiment 1. Figure 3B(right) shows the thrust profile of the satellite departing from Jupiter in Experiment 1.

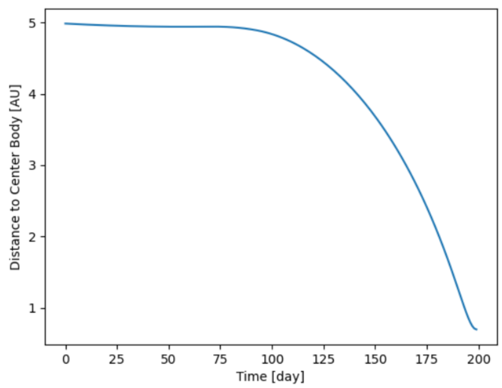


Figure 3C. Figure 3C shows the distance to the Sun of the satellite departing from Jupiter in Experiment 1.

Figures 4A, 4B, and 4C show a Saturn Satellite and Sun system.

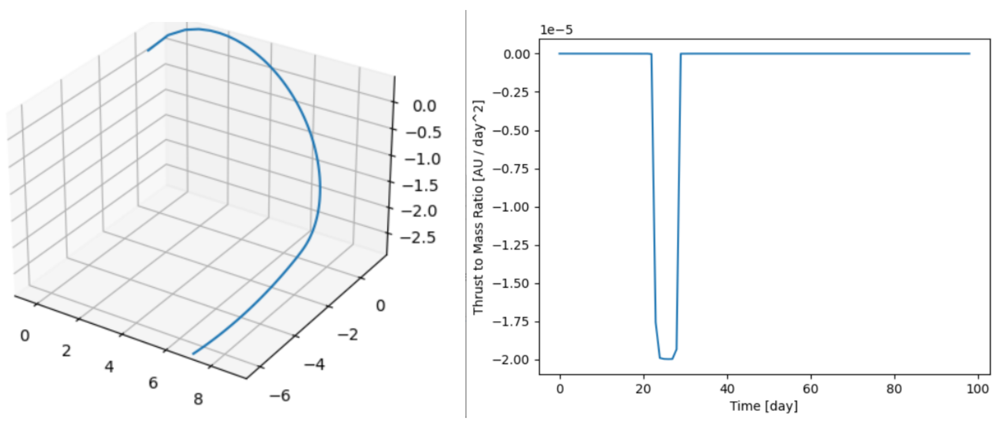


Figure 4A and Figure 4B. Figure 4A(left) shows the trajectory of the satellite departing from Saturn in Experiment 1. Figure 4B(right) shows the thrust profile of the satellite departing from Saturn in Experiment 1.

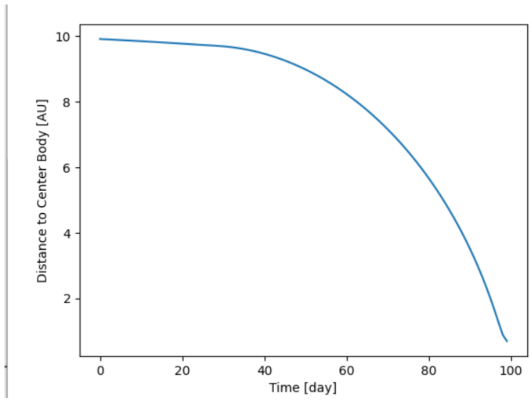


Figure 4C. Figure 4C shows the distance to the Sun of the satellite departing from Saturn in Experiment 1.

The results of the experiment from the solver are summarized in Table 1.

Table 1. Table 1 below summarizes the results from the solver for Experiment 1. It shows the solution time, total thrust used, and number of iterations.

Data Table for Experiment 1

Planet from which satellite originated	Solution Time(secs)	Objective/Total thrust (AU/day^2)	No. of iterations
Earth	95.5246000000006	1.607075469841769E-003	364
Mars	61.0000999999975	2.496310652516179E-003	241
Jupiter	8.30809999999838	4.060410390841822E-003	35
Saturn	272.912699999986	3.386803406157031E-003	1972

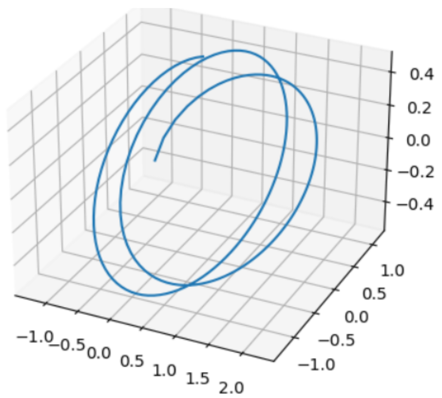
II. Effect of maximum thrust on thrust profile

The next experiment involved testing the effect of the maximum magnitude of instantaneous thrust(T_{max}) on the optimum thrust profile, the objective function(minimized thrust), solution time, and the number of iterations to reach the solution.

The parameters used included a time horizon of 3 years and 200 total time points within that time horizon. The maximum magnitude of instantaneous thrust values that were tested were

1*10^{-5}AU/day^2, 2*10^{-5}AU/day^2, 3*10^{-5} AU/day^2, and 4*10^{-5}AU/day^2. The planet of origin of the satellite was kept constant as Earth with the desired orbit radius as 0.3 AU.

The graphs for T_{max} =1*10^{-5}AU/day^2 are shown in Figures 5A, 5B, and 5C.



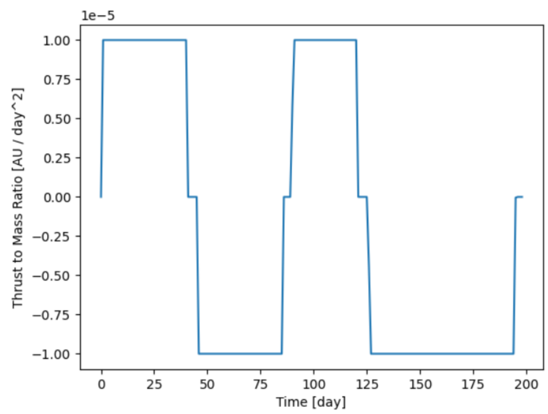


Figure 5A and Figure 5B. Figure 5A(left) shows the trajectory of the satellite with Tmax=1*10^{-5} in Experiment 2. Figure 5B(right) shows the thrust profile of the satellite with Tmax=1*10^{-5} in Experiment 2.

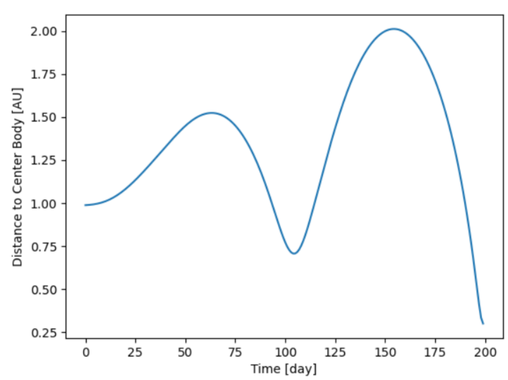


Figure 5C. Figure 5C shows the distance to the Sun of the satellite with Tmax=1*10^{-5} in Experiment 2.

The graphs for T_{max} =2*10^{-5}AU/day^2 are shown in Figures 6A, 6B, and 6C.

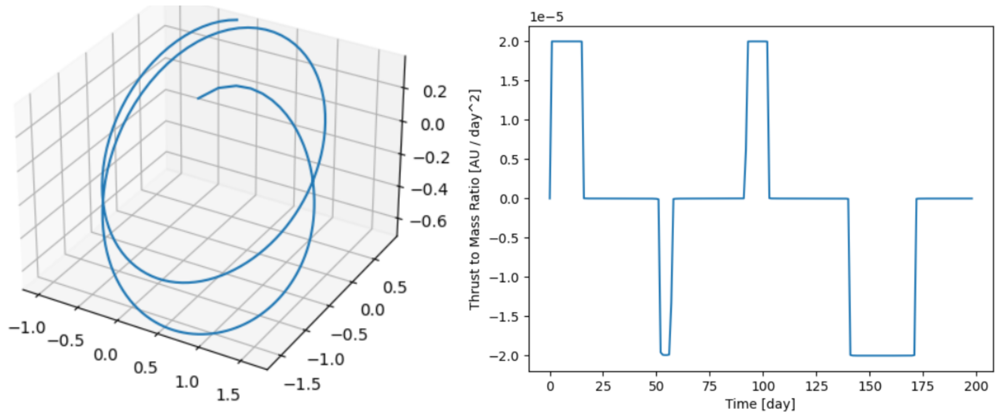


Figure 6A and Figure 6B. Figure 6A(left) shows the trajectory of the satellite with Tmax=2*10^{-5} in Experiment 2. Figure 6B(right) shows the thrust profile of the satellite with Tmax=2*10^{-5} in Experiment 2.

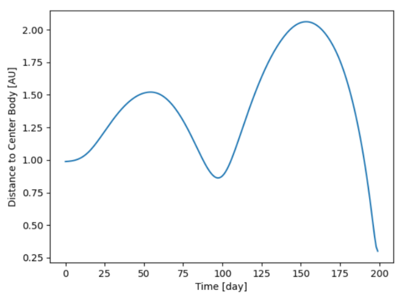


Figure 6C. Figure 6C shows the distance to the Sun of the satellite with Tmax=2*10^{-5} in Experiment 2.

The graphs for T_{max} =3*10^{-5}AU/day^2 are shown in Figures 7A, 7B, and 7C.

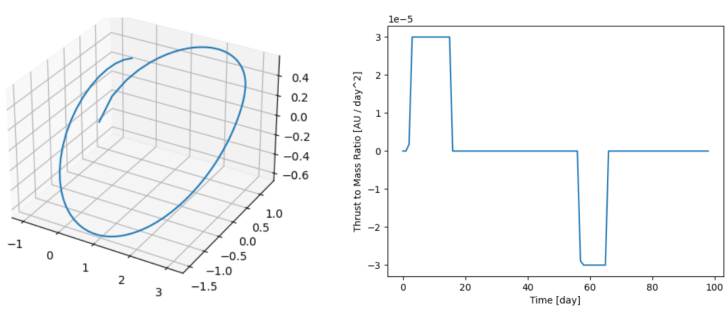


Figure 7A and Figure 7B. Figure 7A(left) shows the trajectory of the satellite with Tmax=3*10^{-5} in Experiment 2. Figure 7B(right) shows the thrust profile of the satellite with Tmax=3*10^{-5} in Experiment 2.

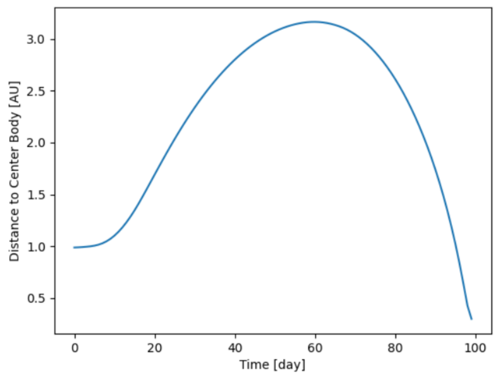


Figure 7C. Figure 7C shows the distance to the Sun of the satellite with Tmax=3*10^{-5} in Experiment 2.

The graphs for T_{max} =4*10^{-5}AU/day^{2} are shown in Figures 8A, 8B, and 8C.

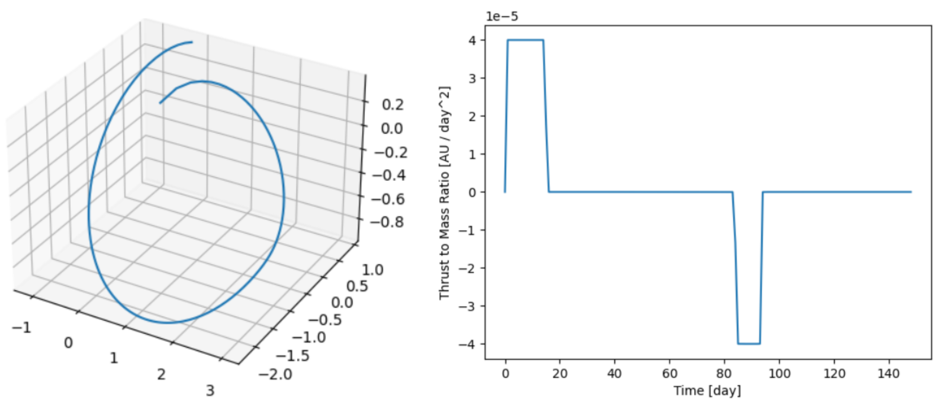


Figure 8A and Figure 8B. Figure 8A(left) shows the trajectory of the satellite with Tmax=4*10^{-5} in Experiment 2. Figure 8B(right) shows the thrust profile of the satellite with Tmax=4*10^{-5} in Experiment 2.

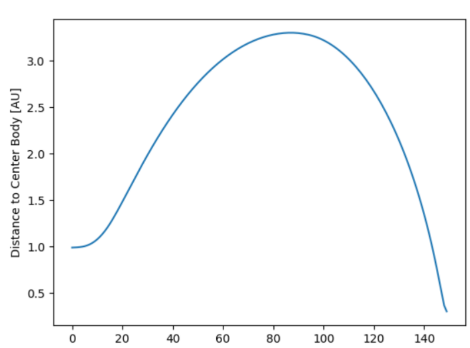


Figure 8C. Figure 8C shows the distance to the Sun of the satellite with Tmax=4*10^{-5} in Experiment 2.

The results from the solver are summarized in Table 2.

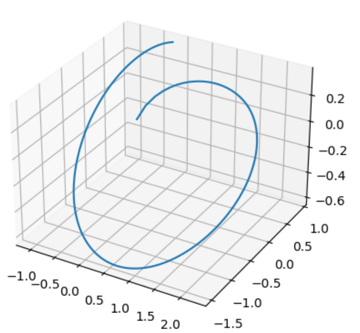
Table 2. Table 2 below summarizes the results from the solver for Experiment 2. It shows the solution time, total thrust used, and the number of iterations.

Data Table for Experiment 2

T_{max}	Solution Time(secs)	Objective/Total thrust (AU/day^2)	No. of iterations
1*10^{-5}	269.367300000013	9.838975205602179E-003	1096
2*10^{-5}	109.026799999992	6.799494899789606E-003	425
3*10^{-5}	206.828299999994	7.287932388867752E-003	1535
4*10^{-5}	21.8685000000114	6.968041141756977E-003	122

III. Effect of prograde and retrograde

Considering an Earth Satellite and Sun system with a satellite departing from Earth at Earth's initial position relative to the barycenter, the trajectory of the satellite to an orbit around the Sun can be fuel-optimized. By constraining the final displacement(R) from the barycenter to be the radius of the orbit, the algorithm produces an optimum thrust profile using 200-time points over a time horizon of 2 years. The value of 'R' is set to 0.2 AU. The result of this experiment is shown in Figures 9A, 9B, and 9C.



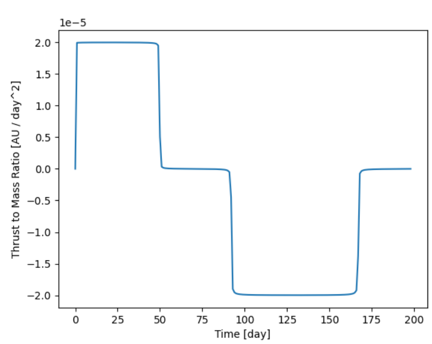


Figure 9A and Figure 9B. Figure 9A(left) shows the trajectory of the satellite departing from Earth and using prograde and retrograde burns in Experiment 3. Figure 9B(right) shows the thrust profile of the satellite departing from Earth and using prograde and retrograde burns in Experiment 3.

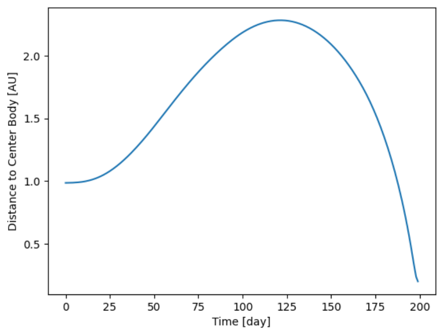


Figure 9C. Figure 9C shows the distance to the Sun of the satellite departing from Earth and using prograde and retrograde burns in Experiment 3.

The thrust to mass ratio-time graph in Figure 9B shows the thrust profile of the satellite described above. By integrating the thrust shown here, the solver calculates a thrust value. Experimenting with various such thrust profiles, the solver has singled out the above fuel-minimized profile.

Results and Discussion

Discussion of Experiment 3

The graph in Figure 9A shows the trajectory of the satellite to the new orbit. The problem setup in the script is a satellite that just left earth that needs to get within 0.2 AU of the sun. The solver is able to do this by first burning prograde to actually move away from the sun initially. Once the satellite is farther away from the sun, relative to where it started on earth, it is moving much slower. Therefore the second thrust maneuver, which is retrograde, is more efficient. The result is the satellite makes a pass within 0.2 AU of the sun at the end of the 2-year time horizon.

It can be verified from this experiment that a prograde burn increases the velocity of the satellite in orbit, causing an increase in orbit radius. On the other hand, a retrograde burn slows down the satellite, causing the Sun's gravity to pull the satellite inwards to a smaller orbit.

It can also be concluded that a prograde burn to increase orbit size increases the efficiency of the maneuver. A continuous retrograde burn would reduce the speed but would require continuous fuel use. However, a prolonged prograde burn increases the radius of the orbit.

$$F = \frac{mv^2}{r}$$

The centripetal force equation above shows that the gravitational force on the satellite reduces as the orbit radius increases, as $F \propto 1/r$.[8] This also means the velocity of the satellite in the orbit reduces, as $F \propto v^2$. Therefore, the prograde burn assists the reduction of velocity required to reduce the orbit size. This more efficient reduction of speed assisted by Earth's gravity makes the retrograde burn more effective, as less speed needs to be reduced to reduce the orbit size.

The theory and discussion described above in this subsection will be used in the following discussions.

Discussion of Experiment 1

As can be seen in Table 1, the fuel use or total thrust increases continuously as we go from closer to distant planets from the Sun. This is quite intuitive because a satellite further away from the Sun would be expected to use more fuel to reach an orbit closer to the Sun. In the following discussion, the data row for Saturn is anomalous as discussed later in this section.

However, from the thrust profile graphs, it is evident that the duration of the burn has increased from the Earth to the Jupiter graphs. This indicates a greater fuel use or greater thrust required for satellites further from the Sun to reduce the speed to a value that allows the gravity of the Sun to pull the satellite into the desired orbit. The greater thrust is required for satellites further away from the Sun because the satellite needs to reduce its speed by a greater amount to reach the new orbit within the time horizon, as the gravitational force would be greater with a lower speed. The greater force would pull the slowed satellite toward the new orbit faster than satellites that burned retrograde for less time.

The first 2 sets of graphs(Earth and Mars) show a movement away from and toward the Sun in the distance to the Sun graphs. The next 2 sets of graphs(Jupiter and Saturn) show a continuous movement towards the Sun. These 2 types of graphs indicate that the solution or trajectory is simpler for the graphs for Earth satellites and Mars satellites. This explains why the number of iterations required and solution time is greater for the satellites closer to the Sun(Earth and Mars). However, they all show retrograde burns, indicating the reduction of speed to use a gravity assist from the Sun towards the new orbit.

However, the data row for Saturn is anomalous as it shows a lower total thrust, but greater solution time, and iterations compared to Jupiter. This breaks the trend and is mainly due to the alteration and restriction of the time horizon and time points, which was done due to the infeasibilities of the problem setup for Saturn for 2000 maximum iterations and 200 time points. This restricted problem setup forced the solver to use a greater total thrust in order to reduce the speed faster to achieve the required orbit within the time horizon. It can be predicted, however, that a satellite from Saturn too would take more fuel than from a closer planet, such as Jupiter. This is because the graphs are in the same shape and follow the same type of trajectory and fuel use as the other planets. Therefore, it can be expected that a satellite from Saturn would also follow the trend.

Discussion of Experiment 2

Experiment 2 examined the effect of T_{max} on the optimum thrust profile. From Table 2, it is evident that all values of T_{max} tested yielded values of the objective function of the same order of magnitude. Therefore, there is not much of a clear observation from the objective function values only. The graphs must also be considered. However, it should be noted that the objective function value for $T_{max}=1*10^{-5}$ is greater than that of the other T_{max} values. This indicates that an extremely low thrust reduces the efficiency of the maneuver noticeably in terms of fuel use. Further support is provided by the fact that the other objective function outputs in the table are all roughly the same, around the value $7*10^{-3}$. This means extremely low thrust burns may also be less efficient compared to moderately low-thrust burns for orbit-change maneuvers(inferred for larger to smaller orbit only from the scope of this experiment).

Note, again, that $3*10^{-5}$ is an anomalous data row due to the infeasibilities described in the previous subsection. However, as described in the previous subsection, it can be predicted to follow the same trend without the infeasibilities. The $4*10^{-5}$ data was tested with 140 time points to reduce infeasibilities, so the objective function may not be accurate. However, the graphs produced provide an accurate shape to demonstrate the basic maneuver, giving a trend described below.

Moving onto the graphs, there is a noticeable trend or difference between lower thrust satellites and higher thrust satellites. Two general groups are identified. Each group's graphs follow the same pattern of trajectory, thrust profile, and displacement from the Sun.

Group 1: Figure 5A and Figure 6A are grouped together since they use 2 prograde and 2 retrograde burns to achieve the new orbit, as shown by Figure 5B and Figure 6B. This is verified from Figure 5C and 6C respectively.

Group 2: Figure 7A and Figure 8A are grouped together since they each use 1 prograde and 1 retrograde burn to achieve the new orbit, as shown by Figure 7B and Figure 8B. This is verified from Figure 7C and 8C respectively.

The Group 1 maneuvers involve 2 revolutions around the sun, during which the satellite used a first prograde burn to raise its orbit as in the trajectory graphs. This effectively reduces the speed of the satellite automatically due to the relation between orbit radius and velocity discussed in the Discussion of Experiment 3. The subsequent retrograde burn further reduces the orbital velocity till a point where Sun's gravity starts pulling the satellite towards it. This is repeated again in the second revolution, reducing the speed until the Sun's gravity pulls the satellite into the orbit of the required radius.

The Group 2 maneuvers use only 1 revolution around the Sun. In this revolution, they burn prograde followed by retrograde for reasons mentioned in the previous paragraph and in the Discussion of Experiment 3. However, they only burn prograde and retrograde once. This difference can be attributed to the fact that the lower-thrust Group 1 satellites cannot produce enough thrust in the same period of time as higher-thrust Group 2 satellites. This explains why Group 1 satellites needed 2 revolutions with 2 retrograde and prograde burns in order to take the Sun's assistance twice. The higher-thrust satellites could produce enough thrust to complete the maneuver within 1 revolution, as they could reduce the speed by the required amount till the Sun pulled the satellite into the required orbit. The lower-thrust satellites needed an extra gravity assist.

In each of the groups, however, the lower T_{max} values have a longer solution time and take more iterations to reach the desired orbit. This can be attributed to the fact that, for the same pattern or type of trajectory(the differentiator between the groups), the higher T_{max} satellites would be able to burn fuel to achieve the thrust required in a shorter period of time. Therefore, the number of iterations required in the solver, and hence the time taken to complete the maneuver, would all be lesser.

It is important not to confuse the time horizon(number of days) with the time points. The time points are the number of points used to divide the time horizon for the solver to calculate positions and equations periodically in every iteration.

Conclusion

By testing an algorithm for trajectory optimization for fuel use in discrete time, it can be concluded that maneuvers using a prograde and a retrograde burn optimize fuel use for a maneuver that reduces the radius of the satellite orbit around a massive central body. It can also be concluded that the further away a satellite is from the Sun, the greater total thrust is required during a retrograde burn to use the gravity assist of the Sun to bring the satellite to a smaller, desired orbit. Moreover, from Experiment 2, it is evident that lower thrust satellites follow a different optimized thrust profile due to their inability to reduce or increase speed as quickly as higher thrust satellites. The optimized thrust profiles involve the use of gravity assists from the central body.

In terms of the algorithm performance and solution parameters, the simpler or less varied maneuvers used a lower number of iterations and solution time. These simpler maneuvers were characteristic of higher thrust satellites and satellites further away from the central body, as evident from the 2 types of graphs in Experiment 1 and 2.

The research can be used to study orbit lowering maneuvers, the characteristics of discrete optimization in terms of astrodynamic applications, and as a tool to generate optimum thrust profiles. The algorithm serves as the basis and has the potential to determine the time feasibility for satellites aiming to be sent on a long-term or short-term mission towards the Sun, or another central body. It provides easy determination of the feasibility of missions in terms of fuel availability and provides access to simulating and optimizing a wide range of situations by changing the algorithm parameters and variables.

References

1. Zona, K. NASA - Ion Propulsion: Farther, Faster, Cheaper. https://www.nasa.gov/centers/glenn/technology/Ion_Propulsion1.html (accessed Jan 2, 2022).

2. Morante, D.; Sanjurjo Rivo, M.; Soler, M. A Survey On Low-Thrust Trajectory Optimization Approaches. *Aerospace* 2021, *8* (3), 88.

3. VPython. https://vpython.org/ (accessed Aug 9, 2021).

4. Beal, L.; Hill, D.; Martin, R.; Hedengren, J. GEKKO Optimization Suite. *Processes* 2018, *6* (8), 106.

5. Skyfield — documentation. http://rhodesmill.org/skyfield/ (accessed Aug 9, 2021).

6. Hunter, J. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* 2007, *9* (3), 90-95.

7. Horizons System. https://ssd.jpl.nasa.gov/horizons/app.html#/ (accessed Aug 9, 2021).

8. Bhattacharjee, P. A Novel Discussion On Centripetal Force. *Revista Brasileira de Ensino de Física* 2018, *41* (1).