Evolving Rocket Control Systems for Launches to Orbital Flight
Literature Survey
CMPLXSYS 425
April 19th, 2018

Noah Bearman - nbearman@umich.edu
Owen Cannon - cannono@umich.edu
Joseph Donohue - donohuej@umich.edu
Divyansh Saini - divyansh@umich.edu

Problem Background

Unsurprisingly, ascent trajectories of rockets are complicated to compute and difficult to optimize. To begin with, specification and performance of a rocket's trajectory are dependent on its mission parameters and design objectives. In a basic sense, a good ascent trajectory is one that reaches the desired orbit with minimal fuel consumption and with acceptable loading on the rocket's structure.

In this project, we explore how a genetic algorithm can be used to evolve a control system to bring a rocket to orbit along an optimal trajectory. The definition of an optimal trajectory, as well as how to assign fitness to individuals to optimize a trajectory, is something that has been discussed at length by researchers in the aerospace field. Genetic algorithms are by no means the only method of computing and optimizing rocket trajectories and this paper will present some of the other ways this problem has been approached.

Our Simulator: The Kerbal Space Program

The Kerbal Space Program (KSP) is a video game and space flight simulator in which you can create spacecraft and fly them into orbit and venture to other planets. The game boasts realistic aerodynamics and orbital physics. There are different game modes that require players to get funding and develop technology to build a spacecraft. For our experiment, however, we used the sandbox mode which allowed us to build a rocket without obstacles. In sandbox mode, after designing and building a rocket, it is possible to launch it or attempt to pilot it into space, orbit, or other planets.

Despite being a video game, Kerbal Space Program does a good job of modeling real world physics. There is an extensive PDF[4] describing the different physics simulated in the game. For our purpose, the relevant physics simulations are aerodynamics, orbital mechanics, and planetary constants. The game allows a player to design a ship with each added part contributing to total drag and mass. Air drag is related to the geometry of the object via the drag coefficient, the fluid properties of the atmosphere, and the fluid flow over the object. As is the case in the physical world, only the frontal area of components in the direction of travel are taken into account when calculating air drag. For example, when traveling sideways the drag coefficient is different then when traveling straight. In addition to a variable drag coefficient, KSP also simulates the density of the atmosphere. Therefore, as the spaceship travels higher the air drag is reduced due to a thinning atmosphere. These factors make for a realistic simulation of aerodynamics. The game also uses the rocket's velocity, direction and the planet's gravitational constant to calculate orbital trajectories, orbital velocities, periods, and other relevant information. The realistic physics simulations of KSP were determined to be provide a satisfactory environment to evolve control systems and flight paths that would be analogous to practical real-world solutions.

Defining an Optimal Trajectory

There are two main forces acting against a rocket as it ascends into space. The first is gravity, which seeks to pull the rocket back to earth. The second is aerodynamic drag, which acts against the rocket's direction of motion. Minimizing the sum of these two forces will result in an optimal launch trajectory.

Normally, optimizing an ascent trajectory requires knowledge of the desired end state of the rocket. This would mean aiming for a specific rocket orbit and minimizing the gravity and air drag forces between the orbital injection point and the launch pad. Different methods of determining the optimal trajectory of a rocket going from one point to another predetermined point are discussed later. Genetic algorithms, however, depend upon evolutionary principles and therefore rely on rocket success, failure, mutation, and rewards to move towards an optimal solution. Since failure in the early stage of evolution could mean crashing on the launch pad and success could mean the rocket achieving an infinite number of altitudes or orbits, we sought different ways to define a good rocket trajectory that would allow our rockets to move toward an optimal solution.

Looking at real rockets, there are several characteristics that successful ascent trajectories have. First, ideal trajectories end with the rocket reaching orbital velocity, altitude, and orientation just as the rocket finishes its burn.[1] For the purposes of a genetic algorithm, then, a good trajectory could potentially be defined as one that reaches any orbit with a relatively stable orientation. Secondly, aerodynamic forces are kept small by keeping the rocket at a zero angle of attack.[2] This reduces the sectional area of the rocket normal to the direction of fluid flow. Third, ideal trajectories utilize a pitch over maneuver, where the rocket turns from a vertical trajectory to a more horizontal trajectory. This reduces the component of the gravitational force that is directly opposing the rocket's thrust, thereby increasing overall efficiency.[3]

Methods of Trajectory Optimization

---

[1] *Launch Vehicle Ascent Trajectories and Sequencing.* Space Daily, June 2016
[2] Ibid.
[3] KSP Physics https://github.com/mhoram-kerbin/ksp-physics-documentation/releases/tag/v1.3

There have been many methods developed to calculate and optimize rocket trajectories. In M.H. Reilly's "Equations of Powered Ascent and Orbit Trajectory," trajectories are computed in three stages. First, rocket equations of motion are analyzed using an earth-fixed frame of reference for the portion of flight when aerodynamic effects are important. Next, this information is transformed to an inertial reference frame which takes into account the rotation of the earth. Finally, the inertial velocities for the portion of flight when aerodynamic forces are negligible are computed.[4]

More recently, particle swarm optimization methods have been applied to the trajectory optimization problem. Particle swarm algorithms take their inspiration from the behavior of organisms in nature like ants and birds, which travel in flocks (or colonies) and tend to bias their movement toward fitter members of the group.[5] This method has been successful in finding optimal trajectories for three stage launch vehicles.[6]

For NASA's now-cancelled Ares I launch vehicle, designed to carry the Orion crew capsule, engineers simplified the optimization process by constraining the search space to two parameters: pitch rate and launch azimuth.[7] Pitch rate refers to the rocket's turn to a more horizontal flight path after a short vertical rise up from the launch pad. Launch azimuth refers to the angle of the vehicle's orbital plane with respect to true north adjusted for the rotation of the earth. These two parameters affect the trajectory speed, altitude, flight path angle, and velocity heading.

With the principles from these methods in mind, we developed our evolutionary optimization algorithm. To do this, we sought a measure of trajectory performance that could serve as a fitness function for our algorithm. Initially we used a value called 'delta v' to measure the performance of our rockets. Delta v is very commonly used in flight dynamics and is a measure of the impulse required for a specific maneuver. Since all rockets accelerate by exploiting the law of conservation of momentum, delta v corresponds directly to the amount of fuel or energy required to perform a maneuver. Our idea was to compare the maximum delta v available of the rocket on the launchpad to the optimal delta v of an orbit attained when the simulation ended. After attempting to implement this it became clear that calculating the optimal delta v of an orbit is a very challenging problem and would lead to evolving a solution to closely match the methods described above rather than searching for potentially unique trajectories. Instead, we chose to base our fitness off of the eccentricity of the rocket's trajectory and the altitude that it attained to evolve a control system that follows an optimal path.

A Previous use of Genetic Algorithms for Trajectory Optimization

In a paper published by the American Institute for Aeronautics and Astronautics, researchers used a genetic algorithm to evolve interplanetary trajectories via a patched-conic

---

[4] M.H. Reilly, *Equations of Powered Ascent and Orbit Trajectory*. U.S. Naval Research Laboratory, 1979
[5] "Particle Swarm Optimization", cleveralgorithms.com, accessed April 2018.
[6] Pontani and Cecchetti, *Ascent Trajectories of Multistage Launch Vehicles: Numerical Optimization with Second-Order Conditions Verification.* University La Sapienza, September 2013.
[7] Dukeman and Hill, *Rapid Trajectory Optimization for the Ares I Launch vehicle.* NASA, March 2018

analysis. In this method, each celestial body is assigned a sphere of influence, modeling the trajectory in stages where the object traveling interacts only with the body whose sphere of influence it exists in. The paper found that the genetic algorithm outperformed grid-search, a method commonly used for trajectory planning to Mars.[8] Our findings with respect to our ascent trajectory algorithm finding local minima and eliminating higher-order unviable spaces was consistent with the results of this paper. We were able to use the findings related to crossover dynamics to better constrain our population mechanics.

[8] P. Cage, I. Kroo, and R. Braun. "Interplanetary trajectory optimization using a genetic algorithm", Astrodynamics Conference, Guidance, Navigation, and Control and Co-located Conferences