Gunnar Miller

Student Number: 300213623

University of the Fraser Valley

[gunnar.miller@student.ufv.ca](mailto:gunnar.miller@student.ufv.ca)

**Interstellar Rocketry Fuel Efficiency Project**

**Technical Background:**

Managing fuel and fuel consumption is likely to be one of the single biggest hurdles for long-range space exploration, informing almost every aspect of ship design and mission planning. Because interstellar space is vast, uninhabited and almost completely empty, ships will need to carry all of their fuel with them. This imposes severe limitations because in addition to accelerating the ship itself, the engine will also need to accelerate all the unburned fuel. Moreover, ships that wish to explore new star systems need to carry not only the fuel necessary to take them there, but also fuel to decelerate at the end of the trip; space doesn’t allow many opportunities for braking.

In rocketry in particular, the fuel needs to be able to provide the spacecraft with not only kinetic energy, but also momentum: the backward momentum of the fuel is always equal to the forward momentum of the craft. Since momentum is the product of mass and velocity, having a fuel with a high exhaust velocity (the speed at which it leaves the craft) is crucial to save fuel mass. Chemical fuels can’t achieve high enough exhaust velocities to be practical for interstellar travel, so new propulsion methods will be required. An inspiration for this project was one such proposal, Nuclear Pulse Propulsion, which carries fuel in the form of small atomic bombs and reflects the light and debris they emit off a specially designed plate to propel the craft forward. Ultimately, however, despite its novel details, this form of propulsion still follows the same basic rules of rocketry, so a simulation at this level doesn’t need to be specific to it. The remainder of the paper (and the simulation itself) simply cover the general case where the craft is some sort of rocket (albeit with an extremely high exhaust velocity).

The ability of a rocket to maneuver (accelerate, decelerate and turn) is governed by the Tsiolkovsky Rocket Equation:

Where mf is the total mass of the rocket and fuel, m0 is the “dry mass” (the mass with no fuel), Δv is the change in velocity and ve is the exhaust velocity. This means, for example, that a craft wishing to travel at the exhaust velocity of its engine must carry fuel mass of roughly 2.7 times the dry mass of the ship. If it wishes to decelerate at the end of its voyage, its fuel needs increase *again* by that factor of 2.7, for a total mass of fuel equal to 7.8 times the dry mass. If it instead wanted to travel at double the speed, it would need 7.82 or nearly 55 times the dry mass. Since most conceivable engine designs produce exhaust velocities that are only a fraction of the speed of light, this means that traveling even to our closest stellar neighbors will require either an inordinate amount of time (in many cases longer than a human lifetime) or an impractical amount of fuel.

One partial solution to this problem involves using stationary facilities in or near our solar system to launch a ship at high velocity before it begins using rocket population. This would reduce the delta-V needed to reach its desired speed and could thus potentially significantly reduce fuel requirements. However, the fuel needed to decelerate at the end of the trip would remain unchanged. There are several possible schemes for how a ship could be launched like this–the program includes suggestions for two of them–but the details don’t especially matter to the fuel economy of the ship, all that matters is how much speed they can impart.

**Design Approach:**

My overall approach was to let the user simulate one trip at a time, giving them as much fine control as possible over each factor of the voyage: spacecraft, destination, speed and so on. Most of the factors that affected fuel efficiency were completely intrinsic to the overall simulation concept, but a few others I added in separately. In rough order of importance, they were as follows:

1. Speed of the trip: the fuel mass required by the Tsiolkovsky rocket equation depends *heavily* on the speed achieved. Faster speeds are important to make trips in reasonable times, but use enormously more fuel. The user doesn’t directly choose the speed, but does choose the distance and time (which ultimately determine the speed.
2. Spacecraft Dry Mass: the fuel used is directly proportional to the dry mass of the spacecraft. Doubling the mass of the craft doubles the fuel use.
3. Reserve Fuel: reserve fuel (fuel not used for primary acceleration and deceleration) effectively increases the dry mass of the craft.
4. Launch Velocity: a high launch velocity lowers the amount of fuel needed for acceleration, potentially resulting in significant savings.
5. Hazards: this was the most speculative category, as no good information exists on the realities on interstellar travel. Of the two hazards incorporated, course corrections always use fuel, while drag zones can actually save fuel (at the cost of making the trip take longer). Travelling to regions with higher density means encountering more hazards, potentially impacting fuel use.
6. Maximum acceleration: this factor proved to be almost entirely unimportant. In theory, allowing the craft to reach higher speeds more quickly, a craft can make the same length trip with a slightly lower top speed. In practice, fuel constraints meant that the acceleration/deceleration phases were always short compared to the overall length of the trip. In a craft without the limitation of a rocket—especially on that could reach relativistic speeds, this would be a *much* more important factor to the overall speed of the trip.

Note that the speed category includes two separate parameters that impact fuel use (trip distance and time) and the hazards category includes three: thus the model has nine factors in total. The entries and mechanics of the hazards category had to be essentially made up: course corrections are something that could certainly be required, but they aren’t likely to be common (since we can see our destinations in advance and know how they’re moving relative to us). Drag zones are entirely speculative: most interstellar space seems to be very, very empty: for something like this to occur, there would need to be significant clusters of higher density matter that were still too diffuse to block light from the target star.

In approaching the code base, I quickly realized that many different variables would need to be included in various calculations, oftentimes in ways that weren’t obvious until I dug into the code. To that end, I found it useful to create a single large class—the Trip class—to contain most of the variables pertinent to a single voyage. Each voyage creates one object of the Trip class, which is overwritten on the next voyage. The two smaller pieces that the user could interact with separately, and which could persist through multiple voyages were implemented in the form of the Destination and Spaceship structs. Each Trip object included one of each as attributes, but inventories of each struct were also stored separately for the user to build up and choose across multiple trips.

**Highlights:**

The most interesting parts of the code are the customization options for ships and destinations and the hazard handling. The customization options allow the user to design their own ships and destinations that persist after the end of the trip and are available for later use.

A screenshot of a computer program

Description automatically generated

The hazard handling code does several things. First, it randomly populates the length of the voyage with hazards, weighted so that more are likely to be found later in the voyage (since those are the parts of the trip that would be least-known in advance). Second, it interrupts the cruising phase each time a hazard is encountered, tracking the distance and time where each occur. Third, it resolves the effects of the hazard on the ship’s speed and fuel reserves. In the case of drag zones, the code calculates whether or not the ship can “afford” to speed back up again: that is, whether doing so would leave the ship too little fuel to decelerate at the end of the journey.

A screenshot of a computer screen

Description automatically generated

**Constraints:**

In terms of project concept, the biggest constraint was simply the fact that much of the subject matter is speculative. We understand the physics behind rocket propulsion very well, but none of the engines that could achieve this kind of trip have ever been tested, and our entire knowledge of the space itself comes from observations here on Earth. We can’t be sure what factors may or may not impact fuel use for this sort of travel, because we haven’t been able to try it yet. In particular, the entire implementation of hazards is largely just based on very loose guesses.

On a technical side, I had to scale back some of the original plans for my code, specifically those which involved incorporating relativistic effects. There are two reasons for this. First, in attempting to keep the parameters of the simulation as close as possible to the limits of proven technology, I rendered relativistic speeds unrealistic; a user can set up a trip that reaches such speed, but it will require some combination of extreme launch speeds and ridiculously impractical amounts of fuel. Even if a ship were *launched* at a velocity of 0.4c (where relativistic effects are *just barely* starting to be significant), it would require an astonishing 54 times its dry mass just to slow down at the end. The second reason for declining to include relativistic effects was simply a practical matter of time: they would have required additions to a significant fraction of the Trip class methods and the required changes would likely have propagated much farther throughout the code.

One final constraint is that despite the theme of the project being fuel efficiency, I ultimately didn’t find a good way to incorporate a statistic of fuel efficiency into the results reporting. The trouble is that (unlike with cars and planes) there isn’t any standard way to report fuel efficiency for speculative long-ranged spacecraft. I could have made up a metric (e.g. tons per light-year), but each trip is sufficiently different that I didn’t feel the comparison between them would be especially informative. So I simply reported the total mass of fuel used by each craft, as well as the mass of fuel used specifically for handling hazards.

**References for Scientific/Technical Background:**

Tsiolkovsky Rocket Equation:

<https://en.wikipedia.org/wiki/Tsiolkovsky_rocket_equation>

Nuclear Pulse Propulsion:

<https://en.wikipedia.org/wiki/Nuclear_pulse_propulsion>

Interstellar Travel:

<https://en.wikipedia.org/wiki/Interstellar_travel>