Justin Pajak, Patrick Creaven, Carter Goldman

Professor Morrison

CSE 20312

Monday, October 19, 2020

Preternship Project Proposal

**Table of Contents**

**Introduction Statement** ……………………………………………………………………….. 2

**Definitions**…………….. …………………………………………………………………….. 2-3

**Design Considerations** ………………………………………………………………...…….. 4-5

**Implementation of Data Structures Course Concepts**……………………………………….. 6

**Goals and Timelines**………………………………………………………………………….. 7-8

**References**………………………………………………………………………………………. 9

**Introduction Statement**

For our project, we plan to build a tool that can calculate the latency of communication between two points in a satellite network, which includes a group of ground stations and a constellation of satellites in geostationary earth orbit (GEO). The tool will take in parameters on the orbit of satellites in a given constellation and the location of ground stations, and it will create new objects to keep track of those parameters for each satellite and ground station. Then, it will calculate the distance between the ground station and each satellite and the distance for viable paths between satellites, and from that, it will calculate the time taken for the information to travel between any two points within the network of ground stations and satellites. It will create a graph-like object with the ground station and satellite objects as the nodes. From this, we will then be able to use Dijkstra's Algorithm to find the lowest-latency path between two points within the network.

**Definitions**

* **Satellite**: A machine placed in an orbit around the Earth, moon, or another celestial body in order to collect information or serve as a far-distance communication device.
* **Latency**: The amount of time it takes for a ground station to communicate with a satellite and then for the satellite to send a return signal back to the ground station.
* **Bandwidth**: The amount of data can be received, processed, and transmitted by a satellite in a given a certain amount of time
* **LEO**: Low Earth Orbit. LEO satellites orbit from 160 kilometers to 2,000 kilometers above the Earth.
* **GEO**: Geostationary Earth Orbit. GEO satellites orbit at 35,786 kilometers above the equator. They move in the same direction and at the same velocity as the Earth rotates on its axis. This causes them to appear fixed in the sky to an observer on the ground.
* **Ground Station**: A facility that is capable of sending and receiving signals to and from satellites in Earth’s orbit. They communicate with these satellites through radio signals.
* **Satellite Constellation**: A group of satellites in Earth’s orbit that work together in order to create a large area on Earth that can communicate with one another by bouncing signals between the satellites.

**Design Considerations**

1. **Assumptions:** 
   1. It is assumed that the orbit of the satellite will be circular, as opposed to elliptical.
   2. All satellites within the same constellation follow the same trajectory in orbit, only differing in the times in which they pass the ground station.
   3. Constellations will be in GEO, meaning that their positions relative to ground stations will remain unchanging over time.
   4. Only factoring in speed of light for latency; the time in which the data is processed will be assumed to be zero.
2. **Project Requirements**
   1. The final product will allow the end user to calculate the time it will take for a data transfer to occur between two points within a satellite network, using the most efficient route possible. This calculation will be based upon the latency from the speed of light between the two points.
   2. If time and complexity allows, the final product will also take into account data size and bandwidth when performing calculations, and it will calculate the most efficient path for all data packets, in order of priority.
3. **Initial Risk and Alternatives:**
   1. Due to the way in which we are gathering the data for the locations of the satellites, the calculation between the satellite and the ground station may prove to be more extraneous than initially thought. We are working on mitigating this risk by doing additional research about how the distances are calculated using geocentric coordinates.
   2. Possibility that our scope is not large enough to fulfill the 4 week, 8 hours/week requirement. To work around this risk, we will continuously attempt to add more and more to the scope of the project, while also staying within the restraints of time and complexity (i.e. create an LEO constellation after a successful GEO, or factoring data size and bandwidth into the latency calculations).
4. **Description of Artifacts**
   1. We have multiple risks due to schedule constraints. First and foremost, we will be performing the work for Week 1 in the second half of the week due to the Data Structures Exam 2 on October 21st, 2020.
   2. Our requirements are compatible with our stakeholder’s expectations. Mr. Mallinger’s initial expectations are that we create a tool to analyze the latency of satellite communication due to distance. Beyond that, his expectations are very open-ended, allowing us to tinker with ideas and create our own objectives to achieve.
   3. Currently, we have no concrete way of gathering and inputting orbit and location data for our program. While we are currently looking into a few sources, we need to determine which source is the best for our purposes.
   4. We do not yet know how to determine whether or not two satellites can successfully communicate with each other based solely on orbital data. More research is required to figure out that relationship and to figure out viable graph edges.
5. **Required Items**
   1. Satellite class
   2. Ground station class
   3. Adjacency Matrix
   4. Satellite and ground station research material and data

**Implementation of Data Structures Course Concepts**

Data Structures To Implement:

* **Objects:** We will initially create a class for both the Satellites and the Ground Stations. These classes will contain several class members. For example, both of these classes will contain a double x value, y value, and z value in order to store their locations relative to the center of the Earth. These values will be useful in calculating the distance between an individual satellite and a ground station in order to determine what is the latency when they communicate.
* **Graphs:** We plan on creating a graph data structure that will be templated with the Satellite object we will create. This will allow us to represent a constellation of satellites and the paths data can travel between each satellite, weighted by the latency of those paths. We plan on using an Adjacency Matrix to implement our Graph data structure because our constellation will not be large. Therefore, we do not have to worry about space complexity and can achieve time to determine whether a valid communication path exists between two satellites. With this information, we will be able to utilize Dijkstra’s Algorithm to efficiently calculate the minimum latency between two points in the constellation.
* **Priority Queue:** If time allows, we may enhance our project so that the latency calculation takes the size of the data transferred into account. We can use a priority queue data structure for this in order to demonstrate how data packets with the shortest latency time based on bandwidth calculations will be transmitted first between satellites and ground stations. This data structure will use insertion and deletion.

**Goals and Timeline**

|  |  |  |
| --- | --- | --- |
| **Task** | **Time** | **Individual vs Collab** |
| **Week of 10/19** |  |  |
| Perform general research | 4 hours | Collaborative |
| Create and test Satellite Class | 3 hours | Justin |
| Create and test Ground Station Class | 3 hours | Patrick |
| Create Functions to calculate satellite position based on orbital data | 3 hours | Carter |
| Weekly Memorandum | 30 minutes | Collaborative |
| Weekly Code Review | 30 minutes | Collaborative |
| **Week of 10/26** |  |  |
| Develop basic main program and makefile | 4 hours | TBD |
| Create functions in satellite class that allow basic communication between one satellite and one ground station | 4 hours | TBD |
| Create functions in ground station class that allow basic communication between one satellite and one ground station | 4 hours | TBD |
| Additional Research | 2 hours | Collaborative |
| Debug and Discussion Time | 1 hour | Collaborative |
| Weekly Memorandum | 30 minutes | Collaborative |
| Weekly Code Review | 30 minutes | Collaborative |
| **Week of 11/2** |  |  |
| Create Graph Class | 2 hours | TBD |
| Implement Graph class to work with Satellite objects | 2 hours | TBD |
| Implement Graph class to work with Ground Station objects | 2 hours | TBD |
| Create and implement Shortest Path algorithm | 4 hours | TBD |
| Project Scope Considerations | 1 hour | Collaborative |
| Debug and Discussion Time | 1 hour | Collaborative |
| Weekly Memorandum | 30 minutes | Collaborative |
| Weekly Code Review | 30 minutes | Collaborative |
| **Week of 11/9** |  |  |
| New Features TBD based on scope considerations (LEO constellation, data processing time taken into account for latency, etc.) | 6 hours | Collaborative |
| Debug and Discussion Time | 1 hour | Collaborative |
| Weekly Memorandum | 30 minutes | Collaborative |
| Weekly Code Review | 30 minutes | Collaborative |
| **Week of 11/16** |  |  |
| Weekly Memorandum | 30 minutes | Collaborative |
| Weekly Code Review | 30 minutes | Collaborative |
| Final Debugging and Project Report | 7 hours | Collaborative |

**References**

1. <https://satoms.com/satellite-latency/>
2. <https://www.satellitetoday.com/telecom/2009/09/01/minimizing-latency-in-satellite-networks/>