Project DOTCOM

Deep-Space Orbital Telecommunications
Project Final Report (PFR)

Monday, May 3rd, 2021

Customers

<table>
<thead>
<tr>
<th>Affiliation</th>
<th>Customer Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Atomics</td>
<td>Benjamin Krajt</td>
</tr>
<tr>
<td>General Atomics</td>
<td>Tommy Romano</td>
</tr>
<tr>
<td>General Atomics</td>
<td>Eric Natchitagall</td>
</tr>
<tr>
<td>General Atomics</td>
<td>James Harper</td>
</tr>
</tbody>
</table>

Advisor
Dr. Jade Morton

Team Members

| Name: Tristan Liu          | Email: trli7263@colorado.edu | Phone: 703-963-3109 | Position: Systems Engineer |
| Name: Hunter Rohlman       | Email: huro6781@colorado.edu | Phone: 720-353-0471 | Position: Network Protocol Integration Lead |
| Name: Jennifer Gurtler     | Email: jegu5171@colorado.edu | Phone: 720-454-7668 | Position: Testing & Safety Lead |
| Name: Douglas Brough       | Email: dobr2274@colorado.edu | Phone: 303-656-5890 | Position: Manufacturing Lead |
| Name: Buck Guthrie         | Email: jagu2282@colorado.edu | Phone: 303-726-4187 | Position: Financial Lead |
| Name: Caelan Maitland      | Email: cama2924@colorado.edu | Phone: 612-819-5460 | Position: Modeling/SysML Lead |
| Name: N. Sebastian Damm    | Email: nida5622@colorado.edu | Phone: 805-807-7975 | Position: Modeling Integration Lead |
| Name: Conner Lewis         | Email: cole8123@colorado.edu | Phone: 720-243-9029 | Position: Network Architecture Lead |
| Name: Forrest Jordan       | Email: fojo3298@colorado.edu | Phone: 970-237-0994 | Position: Network Protocol Lead |
| Name: Sam Taylor           | Email: sata5601@colorado.edu | Phone: 303-883-8722 | Position: Project Manager |
Contents

1 Project Purpose
   Sam Taylor & Tristan Liu 7

2 Project Objectives and Functional Requirements
   Forrest Jordan & Sebastian Damm 7
   2.0.1 Functional Requirements ................................................. 7
   2.1 Concept of Operations (CONOPS) ........................................... 8
       2.1.1 System CONOPS ......................................................... 8
       2.1.2 Customer Deliverables ............................................... 8
   2.2 Project Levels of Success ............................................... 9
   2.3 Functional Block Diagram (FBD) ........................................... 10

3 Final Design
   Conner Lewis, Hunter Rohlman, Forrest Jordan, Buck Guthrie, Sam Taylor, Doug
   Brough, & Tristan Liu 11
   3.1 Satellite Constellation Architecture ....................................... 14
   3.2 Network Protocol Final Design ............................................. 15
   3.3 Spectrum Allocation and Link Budgets .................................... 15
   3.4 Ground Station Final Design .............................................. 16
   3.5 Hardware Final Design ..................................................... 16

4 Manufacturing
   Conner Lewis, Hunter Rohlman, Forrest Jordan, Tristan Liu, Douglas Brough, & Buck
   Guthrie 18
   4.1 Architecture Model Software ................................................. 18
       4.1.1 STK Architecture Models ............................................ 18
   4.2 Network Capacity Model Software ........................................ 18
       4.2.1 Introduction to the Network Capacity Model (NCP) .............. 18
       4.2.2 NCP Development ...................................................... 19
       4.2.3 Outcome of NCP Development ....................................... 21
       4.2.4 Challenges of NCP Development and Project Integration ....... 21
   4.3 Link Budget Software ....................................................... 22
       4.3.1 RF Link Budget Model ................................................. 22
       4.3.2 RF Power Optimization Test and Results .......................... 24
       4.3.3 Laser Link Model ..................................................... 25
   4.4 Hardware Manufacturing .................................................... 26

5 Verification and Validation
   Caelan Maitland, Buck Guthrie, Hunter Rohlman, Forrest Jordan, Conner Lewis &
   Jennifer Gurtler 27
   5.1 Validation of the Network Capacity Model ............................... 27
       5.1.1 NCP Results Verification ............................................ 28
   5.2 Verification of Network Architecture Model ............................ 28
       5.2.1 Matlab Coverage and Connectivity Tests ......................... 28
   5.3 Verification of the Link Budget Model ................................... 32
   5.4 Verification via Model-Based Systems Engineering and SysML ....... 32
       5.4.1 Block Definition Modeling .......................................... 33
       5.4.2 Internal Block Modeling .............................................. 33
       5.4.3 State Machine Modeling .............................................. 34
       5.4.4 Use Case Modeling .................................................... 35
       5.4.5 Requirement Tracing & Satisfaction ................................. 35

6 Risk Assessment, Mitigation, and Realization
   Hunter Rohlman 36
   6.1 Modeling Risks ............................................................. 37
6.2 Inaccessibility Risks................................................................. 38
6.3 Additional Risks ........................................................................ 40
6.4 Realization of Risks ................................................................. 41

7 Project Planning
Sam Taylor, Buck Guthrie & Doug Brough 41
7.1 Organizational Chart ............................................................... 41
7.2 Work Breakdown Structure ..................................................... 42
7.3 Work Plan .............................................................................. 43
7.4 Cost Plan .............................................................................. 45
7.5 Test Plan .............................................................................. 46

8 Lessons Learned
Sam Taylor, Tristan Liu 47
8.1 Systems Engineering ............................................................... 47
8.2 Project Management ............................................................... 48

9 Individual Report Contributions 49

10 Appendix .............................................................................. 53
10.1 Network Capacity Model MATLAB Code ................................. 53
10.2 Constellation Design Figures .................................................. 62
10.3 Hardware Figures .................................................................. 70
10.4 SysML Diagrams ................................................................... 73
10.5 RF and Laser Link Budget Information and Equations ............ 78
10.6 RF Link Model MATLAB Code .............................................. 80
10.7 Laser Link MATLAB Code ...................................................... 84
10.8 System Architecture Trade Study ........................................... 86
  10.8.1 Trade Study ................................................................. 87
10.9 Network Protocol ................................................................. 88
  10.9.1 Conceptual Designs Considered ........................................ 88
  10.9.2 Trade Study ................................................................. 92
10.10 Extraterrestrial Relay Station .................................................. 96
  10.10.1 Conceptual Designs Considered ........................................ 96
  10.10.2 Trade Study ................................................................. 98
10.11 Wireless Communication Spectrum ....................................... 99
  10.11.1 Conceptual Designs Considered ........................................ 99
  10.11.2 Trade Study ................................................................. 100
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Full-system CONOPS</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>DOTCOM Project Levels of Success</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>Functional Block Diagram - Data Flow</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>Illustration of the orbital height above the moon, along with the percent</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>coverage of one satellite</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Illustration of the baseline orbit configuration around the moon</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>Hardware test bed schematic</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>Single Network Node FBD</td>
<td>19</td>
</tr>
<tr>
<td>8</td>
<td>Three Node Network FBD</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>Node Performance Through Multiple Tests</td>
<td>21</td>
</tr>
<tr>
<td>10</td>
<td>Power Optimization Plot for Ground to Satellite</td>
<td>25</td>
</tr>
<tr>
<td>11</td>
<td>Power Optimization Plot for Satellite to Satellite</td>
<td>25</td>
</tr>
<tr>
<td>12</td>
<td>Hardware test bed schematic</td>
<td>27</td>
</tr>
<tr>
<td>13</td>
<td>Hardware test stand setup</td>
<td>27</td>
</tr>
<tr>
<td>14</td>
<td>Water Tank System Diagram</td>
<td>29</td>
</tr>
<tr>
<td>15</td>
<td>Illustration of the minimum elevation required for connectivity in the</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>architecture model</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Illustration of two satellites connecting at maximum separation</td>
<td>31</td>
</tr>
<tr>
<td>17</td>
<td>An example of the block definition diagram that shows the network structure</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>of the DOTCOM networks</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>An example of an internal block diagram that shows internal structure of the</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>Lunar Satellite</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>An example of a state machine diagram that shows the communication system</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>in the Lunar Satellite</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>An example of a use case diagram that shows the system of Lunar Satellite</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>interacts with the Lunar Ground Stations and the Lunar Orbit/Surface</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Vehicles.</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>An example of a requirements matrix showing system elements satisfying</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>several design requirements within the SysML workspace.</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Risk Matrix of Modeling Risks</td>
<td>37</td>
</tr>
<tr>
<td>23</td>
<td>Risk Matrix of Inaccessibility Risks</td>
<td>39</td>
</tr>
<tr>
<td>24</td>
<td>Risk Matrix of Additional Risks</td>
<td>40</td>
</tr>
<tr>
<td>25</td>
<td>Organizational Chart</td>
<td>42</td>
</tr>
<tr>
<td>26</td>
<td>Work Breakdown Structure</td>
<td>43</td>
</tr>
<tr>
<td>27</td>
<td>Simplified Work Plan</td>
<td>44</td>
</tr>
<tr>
<td>28</td>
<td>System Architecture Work Plan</td>
<td>44</td>
</tr>
<tr>
<td>29</td>
<td>MBSE Work Plan</td>
<td>45</td>
</tr>
<tr>
<td>30</td>
<td>Hardware Work Plan</td>
<td>45</td>
</tr>
<tr>
<td>31</td>
<td>Final Cost Overview</td>
<td>46</td>
</tr>
<tr>
<td>32</td>
<td>Image of the constellation design around the moon</td>
<td>62</td>
</tr>
<tr>
<td>33</td>
<td>Image of the groundtracks of the baseline constellation design as well as</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>the distribution of the satellites</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>Level 1 Hardware Design</td>
<td>70</td>
</tr>
<tr>
<td>35</td>
<td>Level 2 Hardware Design</td>
<td>70</td>
</tr>
<tr>
<td>36</td>
<td>Level 3 Hardware Design</td>
<td>71</td>
</tr>
<tr>
<td>37</td>
<td>Level 4 Hardware Design</td>
<td>71</td>
</tr>
<tr>
<td>38</td>
<td>Synchronous Transmission Latency Calculations</td>
<td>72</td>
</tr>
<tr>
<td>39</td>
<td>Asynchronous Transmission Latency Calculations</td>
<td>72</td>
</tr>
<tr>
<td>40</td>
<td>Earth Ground Station IBD</td>
<td>73</td>
</tr>
<tr>
<td>41</td>
<td>Earth Ground Station State Machine</td>
<td>73</td>
</tr>
<tr>
<td>42</td>
<td>Possible Earth Satellite IBD</td>
<td>74</td>
</tr>
<tr>
<td>43</td>
<td>Possible Earth Satellite State Machine</td>
<td>74</td>
</tr>
<tr>
<td>44</td>
<td>Earth Ground Station IBD</td>
<td>75</td>
</tr>
<tr>
<td>45</td>
<td>Lunar Ground Station IBD</td>
<td>75</td>
</tr>
<tr>
<td>46</td>
<td>Lunar Satellite IBD</td>
<td>76</td>
</tr>
<tr>
<td>47</td>
<td>Lunar Satellite Sate Machine</td>
<td>76</td>
</tr>
<tr>
<td>48</td>
<td>Example Lunar Ground Vehicle IBD</td>
<td>77</td>
</tr>
<tr>
<td>49</td>
<td>Example Lunar Orbital Vehicle IBD</td>
<td>77</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
<td>Reference</td>
</tr>
<tr>
<td>------</td>
<td>---------------------------------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>50</td>
<td>BER Curve: Found in Space Mission Engineering: the New SMAD</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>Optical Communications Power Received Equations 1</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>Optical Communications Power Received Equations 2</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>Laser Link between Earth and Moon from Deep Space Optical Communications</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>Illustration of each orbital height, high, medium, and low, over the moon along with their coverage percentages</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>Basic IP Protocol Stack [1]</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>Basic IP Network [1]</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>Visualization of the DTN Stack [1]</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>DTN Network Visualization [1]</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>End-to-end IP vs DTN in a Variable Connectivity Environment [NASA.gov]</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>Illustration of Satellite Relay Data Transfer Path</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>Illustration of Ground Relay Data Transfer Path</td>
<td></td>
</tr>
</tbody>
</table>
List of Tables

1. Table of Functional Requirements ............................................. 7
2. Inter-satellite RF Communications Link Budget .......................... 16
3. Satellite to Lunar Surface RF Communications Link Budget ........... 16
4. Inter-satellite Communications Link Budget .............................. 23
5. Satellite to Lunar Surface Communications Link Budget ............... 24
6. This table lists the percentages of satellite coverage levels available to points on the lunar surface over the course of one orbital period for the seven constellations tested. ........ 30
7. This table lists the percentages of satellite coverage levels available to points in lunar orbit over the course of one orbital period for the seven constellations tested. ........ 31
8. ISL patch antenna parameters for TSX-TDX boresight link margin from the Antenna Trade Study [18] .................................................. 32
9. Modeling Risk Descriptions .................................................. 37
10. Modeling Risk Mitigation Strategies and Results ....................... 38
11. Inaccessibility Risk Descriptions .......................................... 38
12. Inaccessibility Risk Mitigation Strategies and Results ............... 39
13. Additional Risk Descriptions .............................................. 40
14. Additional Risk Mitigation Strategies and Results .................... 41
15. Detailed Cost Breakdown .................................................. 45
16. Individual Report Contributions .......................................... 49
17. 6/6/4 satellite orbit characteristics ...................................... 63
18. 12/6/4 satellite orbit characteristics .................................... 63
19. 12/12/10 satellite orbit characteristics .................................. 64
20. 15/15/6 satellite orbit characteristics .................................... 64
21. 15/15/2 satellite orbit characteristics .................................... 65
22. 18/6/4 satellite orbit characteristics ..................................... 65
23. 24/6/4 satellite orbit characteristics ..................................... 66
24. High-Alitude Constellation Pros and Cons .............................. 86
25. Low-Alitude Constellation Pros and Cons ............................... 87
26. Medium-Altiude Constellation Pros and Cons ......................... 87
27. Value assignments for Coverage .......................................... 88
28. Value assignments for Link Budget Requirements ...................... 88
29. Value assignments for Failure Vulnerability ............................ 88
30. Satellite Constellation Design Trade Study ............................. 88
31. TCP/IP Pros and Cons .................................................. 90
32. DTN Protocol Pros and Cons ............................................. 91
33. Value assignments for degree of robustness ............................ 92
34. Value assignments for latency resolution ............................... 92
35. Value assignments for network congestion ............................. 93
36. Value assignments for network complexity ............................. 93
37. Value assignments for scalability ....................................... 93
38. Value assignments for protocol compatibility ........................ 94
39. Support Satellite Relay Pros and Cons ................................ 97
40. Ground Station Relay Pros and Cons .................................. 98
41. Value assignments for Accessibility ..................................... 98
42. Value assignments for Cost ............................................... 98
43. Value assignments for Data Specs ....................................... 98
44. Extraterrestrial Relay Station Trade Study ............................ 99
45. RF Communications Pros and Cons ..................................... 99
46. Laser Communications Pros and Cons ................................ 100
47. Value assignments for Bandwidth and Frequency ..................... 100
48. Value assignments for Power Requirements .......................... 101
49. Value assignments for Range and Coverage .......................... 101
50. Communications Method Trade Study Surface To Constellation .... 101
51. Communications Method Trade Study Interplanetary ................. 101
## Nomenclature

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BER</td>
<td>Bit error rate</td>
</tr>
<tr>
<td>CONOPS</td>
<td>Concept of operations</td>
</tr>
<tr>
<td>DBP</td>
<td>Delay-bandwidth product</td>
</tr>
<tr>
<td>DOTCOM</td>
<td>Deep-space orbital telecommunications</td>
</tr>
<tr>
<td>DTN</td>
<td>Delay-tolerant network</td>
</tr>
<tr>
<td>EDRS</td>
<td>European Data Relay System</td>
</tr>
<tr>
<td>FBD</td>
<td>Functional block diagram</td>
</tr>
<tr>
<td>FTC/ITU</td>
<td>Federal Trade Commission / International Telecommunication Union</td>
</tr>
<tr>
<td>GA</td>
<td>General Atomics</td>
</tr>
<tr>
<td>IP</td>
<td>Internet protocol</td>
</tr>
<tr>
<td>JPL</td>
<td>Jet Propulsion Lab</td>
</tr>
<tr>
<td>LCT</td>
<td>Laser Communication Terminal</td>
</tr>
<tr>
<td>LEO</td>
<td>Low Earth Orbit</td>
</tr>
<tr>
<td>MBSE</td>
<td>Model-Based Systems Engineering</td>
</tr>
<tr>
<td>SysML</td>
<td>Systems Modeling Language</td>
</tr>
<tr>
<td>BDD</td>
<td>Block Definition Diagram</td>
</tr>
<tr>
<td>IBD</td>
<td>Internal Block Diagram</td>
</tr>
<tr>
<td>OISL</td>
<td>Optical Inter-satellite Link</td>
</tr>
<tr>
<td>RF</td>
<td>Radio-frequency communication</td>
</tr>
<tr>
<td>RRT</td>
<td>Round trip times</td>
</tr>
<tr>
<td>SCPS-TP</td>
<td>Space Communications Protocol Standard - Transport Protocol</td>
</tr>
<tr>
<td>ION</td>
<td>Interplanetary Overlay Network</td>
</tr>
<tr>
<td>CFDP</td>
<td>Consultative Committee for Space Data Systems - File Delivery Protocol</td>
</tr>
<tr>
<td>PYION</td>
<td>Python Interface for ION</td>
</tr>
<tr>
<td>RPi</td>
<td>Raspberry Pi</td>
</tr>
</tbody>
</table>
1 Project Purpose
Sam Taylor & Tristan Liu

The colonization of extraterrestrial planets was once a concept reserved for one’s dreams and imagination, illustrated explicitly under the label of science fiction. However, with recent advances in technology, not only has the concept started to look feasible, but plans are already being put in place. But, as with any project this large and complex, success comes with a variety of new problems that need to be solved. One of which is a reliable communications network with complete coverage across the extraterrestrial body and a direct line to earth. That’s where DOTCOM comes in.

The purpose of project DOTCOM is to design, test, and verify a conceptual interstellar communications network within the scope of the Earth-Moon system. Project DOTCOM is a unique project, as its primary focus on research and software modeling differentiates itself from the majority of ASEN4018 projects. Project DOTCOM seeks to optimize its design choices to create a comprehensive network concept with the capacity to meet requirements on telecommunications coverage, data transmission rates, existing infrastructure compatibility, and support for vehicle controls and communications at locations within the deep-space environment as outlined in our design requirements. These design choices include: (1) the orbital and positional determinations for satellite constellations, (2) the choice of location for inter-planetary relay stations, (3) the spectrum and methods to allow for communications, (4) the data transmission protocol used to combat obstacles posed by the space environment, and (5) a hardware testbed comprised of a simulated network architecture illustrating the capabilities of communications software for purposes in network protocol proof-of-concept. The work done by project DOTCOM will prove to have numerous applications within the realm of development for satellite constellations, deep-space communications capabilities, and communication architecture design strategies as they relate to planetary expansion as a whole.

2 Project Objectives and Functional Requirements
Forrest Jordan & Sebastian Damm

The project objectives and functional requirements section describes how this project has accomplished the design problem and how success was defined. The operation of the system is described along with an illustration of the CONOPS showing the network architecture and link types between network sections and customer deliverables. Next we define what success looks like for this project, including levels of success and what subsections of each level this project was able to accomplish as well as a description of the project deliverables to the client. Finally, a functional block diagram of the entire system demonstrates the major elements of the functioning system and the critical interactions of project elements.

2.0.1 Functional Requirements

<table>
<thead>
<tr>
<th>Functional Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FR 1</td>
<td>Communication architecture must be capable of transmitting and receiving data simultaneously and non-simultaneously (store-and-forward) between the Earth-Moon system.</td>
</tr>
<tr>
<td>FR 2</td>
<td>Satellite constellations around the Moon must be able to provide communication and vehicle control capabilities on their surfaces and in their orbits.</td>
</tr>
<tr>
<td>FR 3</td>
<td>Communication network must ensure safety of and be collaborative with existing and future communications infrastructure.</td>
</tr>
</tbody>
</table>

Table 1: Table of Functional Requirements

Functional Requirement 1 is the primary driver of project DOTCOM. It’s source is the communication architecture created that allows for the Earth and Moon communications to be capable of connecting to one another.

Functional Requirement 2 is the command and control aspects of the network communications project. The ability to connect various orbital and surface vehicles is a fundamental requirement for project DOTCOM. These constellations shall supplement future interplanetary missions.
The motivation behind Functional Requirement 3 is General Atomics' contracts for many deep-space satellite and communications applications, which project DOTCOM’s network must be able to interface with using existing network protocols and architectures.

2.1 Concept of Operations (CONOPS)

2.1.1 System CONOPS

Figure 1: Full-system CONOPS

Figure 1 describes the full-scale operation of the system. The CONOPS depicts a simplified illustration of the web-like nature of the network nodes and the connections between them. The network consists of four types of nodes, ground stations, constellation satellites, surface vehicles, and orbital vehicles, with the anatomy of their connections shown in the CONOPS. Signals can originate or terminate at ground stations or vehicles, and can be distributed anywhere in the network.

The hardware portion of this project will serve as a small scale analog of the system described in Figure 1. Each of the nodes will be represented by a Raspberry Pi, and will be linked via wired connections. By simulating the light delays associated with each link and running the network software on the Raspberry Pis, this hardware analog will provide information on the latencies associated with the real system as well as where network congestion may arise. The hardware analog will consist of four levels of success, shown in Appendix Figures 34-37, each building off of the last.

2.1.2 Customer Deliverables

Since this project is mostly system modeling in scope, the final deliverable for our client is the Model-Based Systems Engineering (MBSE) software package which encompasses the critical project elements which are made up of the Network Protocol, System Link Budgets, Relay Stations and Satellite Constellation. MBSE is a package that contains all the structural and behavioral elements modeled by this project as well as the interactions between critical project elements. Furthermore, MBSE is used to verify and validate the final design by tracing design requirements throughout the system and produced satisfy, derive and refine matrices that show which elements are responsible to satisfy requirements. This allows our client to understand how our system meets the functional requirements prescribed by them and therefore they are able to model that system within their own MBSE program.
2.2 Project Levels of Success

Figure 2 breaks down all project deliverables into four distinct levels of success. Level 1 deliverables are network performance requirements that the network design must meet. Level 2 success contains hardware deliverables for a DTN protocol proof of concept and network prototype. Level 3 deliverables takes the DOTCOM network out of the theoretical realm and begins to derive requirements for individual network components. Level 4 deliverables encompass program planning and system deployment studies.

<table>
<thead>
<tr>
<th>Level 1</th>
<th>Design Meets Performance Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>99% coverage of Lunar Surface</td>
</tr>
<tr>
<td></td>
<td>99% coverage in Lunar Orbit</td>
</tr>
<tr>
<td></td>
<td>Ability to command lunar surface vehicles</td>
</tr>
<tr>
<td></td>
<td>Ability to command lunar orbiting vehicles</td>
</tr>
<tr>
<td></td>
<td>Simultaneous coms support for 5 surface locations</td>
</tr>
<tr>
<td></td>
<td>Simultaneous coms for 10 orbital vehicles</td>
</tr>
<tr>
<td></td>
<td>Store and forward support for 10 surface locations</td>
</tr>
<tr>
<td></td>
<td>Store and forward support for 20 orbital vehicles</td>
</tr>
<tr>
<td></td>
<td>Earth to Moon data rate exceeds 500 Mbps</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 2</th>
<th>Hardware Proof of Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acquire ION software</td>
</tr>
<tr>
<td></td>
<td>Load ION software on Raspberry PI units</td>
</tr>
<tr>
<td></td>
<td>Send Messages between nodes using ION</td>
</tr>
<tr>
<td></td>
<td>Software/Hardware latency test</td>
</tr>
<tr>
<td></td>
<td>Network capacity validation test</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 3</th>
<th>Required Satellite and Ground Station Specs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Transmission power requirements</td>
</tr>
<tr>
<td></td>
<td>Link budgets</td>
</tr>
<tr>
<td></td>
<td>Thermal requirements</td>
</tr>
<tr>
<td></td>
<td>Pointing accuracy</td>
</tr>
<tr>
<td></td>
<td>Propulsion budgets</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 4</th>
<th>Exploration of Cost and Mission timeline</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cost of each unit</td>
</tr>
<tr>
<td></td>
<td>Cost of deployment</td>
</tr>
<tr>
<td></td>
<td>Expected lifespan</td>
</tr>
<tr>
<td></td>
<td>Study of mission timeline</td>
</tr>
<tr>
<td></td>
<td>Resiliency study</td>
</tr>
</tbody>
</table>

Figure 2: DOTCOM Project Levels of Success

The DOTCOM project was successful in completing all level 1, most level 2, and some level 3 deliverables. Due to time constraints, no level 4 deliverables were attempted. Overall, project DOTCOM has produced a design of a lunar communications network that meets all performance requirements specified in table 1.
2.3 Functional Block Diagram (FBD)

Our functional block diagram depicts the manner in which data will flow to construct a validated model of the whole system, with input parameters on the left, predictive models in the middle as acquired software and mathematical simulations, and the desired output parameters on the right which will be used to satisfy our design requirements. The major project elements of DOTCOM are depicted in the FBD and color coded in reference to the legend. As seen in the diagram, our predictive models will be used with one another to output our given performance parameters.

To best quantify the ability of our design communications medium to propagate in space, we will be designating frequencies for inter-stellar communications between ground stations on different planetary bodies, as well as for satellite constellations that communicate with ground stations and certain locations in each planetary body’s environment. Using these metrics combined with characteristic parameters (system attenuation due to atmosphere and space loss, as well as node-wise available power) we are constructing link budgets for constellation and relay station systems within our network. These link budgets will provide insight into network performance characteristics, which include bit error rate (BER), percent losses, and available bandwidth per node.

We will also be employing the NASA DTN Developmental Kit, an open-source software package created with the intent to model certain communications network configurations in space (shown in the blue oval). Each simulated node (satellite, ground station, vehicle, or otherwise) operates with on-board software capable of handling DTN protocols for transmission. In the DevKit, we will be creating scenarios to examine the data path in space. By incorporating BER and percent losses as determined by our link budgets, the DevKit will be used to created matrices prescribed for certain transmission orientations, the contents of which include...
The yellow and red blocks – relay station and satellite constellation properties, respectively – make up the physical network architecture that DOTCOM seeks to optimize. The preliminary input parameters will be used to create network orientation models in STK with regards to the numbers and locations of ground stations and constellation satellites. This orientation will allow us to create a network connection map in MATLAB using STK positional outputs, where we can analyze the most suitable connection at any time to transmit data through the network with maximum speed. Hence, the network connection map be integral in determining the reliability of data throughput. The network orientation model will address the 99 telecommunications coverage in the Lunar environment design requirement, as well as assist with the creation of the network capacity model.

The network capacity model will take inputs in the form of node-wise bandwidth, storage capacity, and positional outputs from the network connection map. In turn, it will provide an estimation of the maximum data stream that can be sent through the network, for the purpose of mitigating congestion in our final network’s architectural design. The network capacity and network emulation – stemming from the network capacity model and NASA DevKit, respectively – will be key input parameters for our hardware test bed. With this hardware simulation, open source DTN transmission software will run on Ubuntu onboard a multi-node Raspberry Pi network. This test bed will simulate latencies resulting from the space environment and will be physically oriented as a scaled-down version of our network architecture, and will be used to validate the use of DTN in scaled spacial models and to help us determine the data latency resulting from internal hardware operations.

The output parameters shown on the right will be utilized at the end of the May 2021 to model our overall network concept in one cohesive block diagram. Project DOTCOM will use Papyrus, a model-based systems engineering software, to depict the system model in its entirety and meet our customer requirements for deliverables. When a MVP is created in Papyrus to represent the flow of information as depicted in the FBD, DOTCOM will seek to optimize the overall network by adjusting input parameter ranges to better match our output parameters to the requirements assigned by General Atomics.

3 Final Design

Conner Lewis, Hunter Rohlman, Forrest Jordan, Buck Guthrie, Sam Taylor, Doug Brough, & Tristan Liu

All design requirements, listed below, were given by the customer, General Atomics. The given design requirements were then grouped into functional requirements determined by team DOTCOM. The design requirements allowed the DOTCOM network architecture to be scalable and testable. Additionally, multiple sub-routines exist for the network, and these were addressed through the use of model-based systems engineering. Included in the list of these requirements are multiple transmission rate requirements, coverage requirements, communication protocols, and compatibility requirements.

FR 1 *Satellite constellations around the Moon must be able to provide communication and vehicle control capabilities on their surfaces and in their orbits.*

**Motivation:** The ability to connect various orbital and surface vehicles is a fundamental requirement for project DOTCOM. These constellations shall supplement future interplanetary missions.

**Verification:** Multiple nodes in the communications constellation network will be illustrated to be capable of interacting with one another through simulated modelling.

DR 1.1 *The mission shall provide real time (light-time delay with additional delay less than 1 second) relay of data from the Lunar environment to Earth-based communication networks.*

**Motivation:** In order to improve upon already existing deep-space networks, our protocol must be able to relay information based on the data rates specified by GA.

**Verification:** Link budgets shall assist in the theoretical modeling of the mission.

DR 1.2 *The mission shall provide global surface telecommunications coverage (>99%) of the Lunar surface.*
**Motivation:** Future Lunar mission locations are not designated, and thus this mission must leave the question of mission surface position open to future possibilities by providing global coverage of their surfaces.

**Verification:** Satellite architecture with respect to ground coverage will be scaled using a model to approximate real surface coverage.

**DR 1.3** *The mission shall provide simultaneous telecommunications availability (>99%) of the Lunar surface.*

**Motivation:** Simultaneous telecommunications on the surfaces is required to supplement future missions.

**Verification:** This requirement can be verified through simulation. If the simulation identifies >99% of points on the body’s surface as having a satellite overhead at all times, then the requirement is met.

**DR 1.4** *The mission shall provide simultaneous telecommunications availability (>99%) of the Lunar and Martian orbit.*

**Motivation:** Simultaneous telecommunications in the orbits is required to supplement future missions.

**Verification:** This requirement will be modeled and verified through the use of computer-aided modelling.

**DR 1.5** *The mission shall provide the ability to relay data between mission segments (Earth/Lunar/Martian communication networks).*

**Motivation:** The relay data between the Earth and Moon communication networks are applied for the mission.

**Verification:** The relay data for communication networks can be verified by computer-aided modelling.

**DR 1.6** *The mission shall allow for high speed data transmission between the Moon and Earth, with threshold and objective rates being 500 Mbps and 5 Gbps, respectively.*

**Motivation:** Requirements specified by General Atomics. This will serve as validation that the communication speeds between the Earth and the Moon are fast enough and sufficiently large for the type of data being transmitted.

**Verification:** Link budgets will be used as a theoretical baseline for additional disruption tolerances. Network architecture will be illustrated using computer-aided modelling.

**FR 2** *Communication architecture must be capable of collecting and linking simultaneous and non-simultaneous communications data between the Earth and Moon.*

**Motivation:** This is one of the leading requirements for project DOTCOM. There must be a communication architecture created that allows for the Earth and Moon communications to be capable of connecting to one another.

**Verification:** A model-based systems engineering approach will be taken to ensure sufficient modeling of this network’s architecture.

**DR 2.1** *The mission shall provide the ability to command Lunar and Martian surface vehicles.*

**Motivation:** Commanding surface vehicles is a necessity for any Lunar or Martian surface scientific exploration.

**Verification:** Will be verified through simulation using a MBSE approach.

**DR 2.2** *The mission shall provide the ability to command Lunar and Martian orbiting vehicles*
**Motivation:** Commanding orbiting vehicles is necessary for keeping the communication network running, as well as supporting other satellites observing the Lunar and Martian areas.

**Verification:** Will be verified through a simulation using a MBSE approach.

**DR 2.3** *The mission shall provide simultaneous communication support to 5 or more locations on the Lunar Surface and 10 or more locations on the Martian Surface.*

**Motivation:** Communicating with the surfaces was one of the main requirements given by General Atomics, so that the network can provide telecommunications to crews on the surface.

**Verification:** Will be verified through a simulation using a MBSE approach.

**DR 2.4** *The mission shall provide non-simultaneous store and forward (within six hours of transmission) communication support of 10 or more locations on the Lunar Surface and 20 or more locations on the Martian Surface.*

**Motivation:** Larger data packets are required to be sent/received in a timely manner so that there are efficient surface communications outside of telecommunications.

**Verification:** Will be verified through a simulation using a MBSE approach.

**DR 2.5** *The mission shall provide simultaneous communication support of 10 or more space vehicles in Lunar Orbit and 15 or more space vehicles in Martian Orbit.*

**Motivation:** Required to support vehicle telecommunications, command, and control in orbits.

**Verification:** Will be verified through a simulation using a MBSE approach.

**DR 2.6** *The mission shall provide non-simultaneous store and forward (within six hours of transmission) communication support of 20 or more space vehicles in Lunar Orbit and 30 or greater space vehicles in Martian Orbit.*

**Motivation:** Required to support the non-telecommunications side of the network for relaying larger data files from orbit.

**Verification:** Will be verified through a simulation using a MBSE approach.

**FR 3** *Communication network must ensure safety of and be collaborative with existing and future communications infrastructure.*

**Motivations:** Because General Atomics is contracted for many deep-space satellite and communications applications, project DOTCOM’s network must be able to interface with existing network protocols and architectures.

**Verification:** Will be verified by paper study along with results from a simulation generated by a MBSE platform.

**DR 3.1** *The mission shall utilize existing NASA communications infrastructure wherever possible.*

**Motivation:** Meets the requirement of being collaborative with existing infrastructure, allows team DOTCOM to focus more on integration rather than creation.

**Verification:** If used, the infrastructure will be simulated in the MBSE analysis.

**DR 3.2** *The mission shall not interfere with the operational capabilities of the Lunar Gateway.*

**Motivation:** The Lunar Gateway will be launched soon and project DOTCOM must not interfere with its orbit trajectory or transmission quality to ensure the safety of future infrastructures.

**Verification:** Orbit trajectory will be verified through a mathematical analysis, while the transmission quality will be verified through a link budget/signal analysis.

**DR 3.3** *The mission shall be capable of supporting Lunar Gateway communications signals*

**Motivation:** Project DOTCOM must also be able to communicate with the Lunar Gateway so that there is collaboration between team DOTCOM and the Lunar Gateway.
**Verification:** Verified through the MBSE simulation as well as a link budget/signal analysis.

**DR 3.4** The mission shall be capable of interfacing with government and/or commercial communications infrastructure.

**Motivation:** Most communications infrastructure in space and deep space is government owned or operated, requiring project DOTCOM to be able to communicate with these infrastructures.

**Verification:** Verified through the MBSE simulation as well as a link budget/signal analysis.

**DR 3.5** The mission shall comply with FTC/ITU regulations.

**Motivation:** Project DOTCOM will be sending/receiving transmissions, some of which may be sensitive information. Therefore project DOTCOM must comply with the government restrictions and regulations on how to send/receive signals.

**Verification:** Verified through the MBSE simulation as well as a link budget/signal analysis.

### 3.1 Satellite Constellation Architecture

The physical network architecture must be designed such that signals can be distributed throughout it as defined by the project’s requirements. These include persistent coverage of >99% of the surface of a planetary body. Thus, a satellite constellation is required about the body in order to provide this coverage.

A satellite constellation with this objective has numerous design components, among the most fundamental of which are the number of satellites used and the geometry of their orbits. For the near-complete coverage stated in the project’s requirements, at least one satellite in the constellation must be in the sky over >99% of points on the surface at all times.

As determined in the satellite constellation trade study detailed in the appendix, the constellation for signal distribution in the lunar environment will operate at approximately 5,509 km above the lunar surface and will consist of six satellites. The orbits are oriented in what is known as a Walker-Delta pattern, using circular orbits of uniform inclination. Illustrations of this altitude and configuration are shown in Figures 4 and 5.

![Figure 4: Illustration of the orbital height above the moon, along with the percent coverage of one satellite](image)

5,509 km — 38%
Figure 5: Illustration of the baseline orbit configuration around the moon

Walker (1977) describes this as a 6/6/4 pattern, consisting of 6 satellites in 6 orbits, with their Right Ascensions of Ascending Nodes (RAANs) being separated by $360/6 = 60^\circ$, and the satellites’ phases being staggered by $4\times360/6 = 240^\circ$. Walker notes that the optimal inclination for this pattern is 53.1°. This pattern is very similar to that used by the original 24 GPS satellites, albeit with fewer satellites per plane, and has been numerically verified by Walker as optimized for constant global coverage, satisfying the requirements for >99% persistent coverage. Images of this constellation in 3D as well as its groundtracks can be found in the appendix.

This constellation was implemented using AGI STK, which output the positions of each satellite in the constellation over time. Using this information, the connection windows, describing which links are viable between which satellites and when, as well as the coverage map, containing points in the environment and whether or not they are able to link into the network, were calculated and implemented by the overall network simulation. The details of this process are described in the Manufacturing and Verification and Validation sections.

3.2 Network Protocol Final Design

Delay Tolerant Networking is used to direct and control data flow through the DOTCOM network. Specifically, we will implement NASA’s ION network protocol solution for deep space networks. DTN outperforms other network protocols because it was developed with the space environment in mind. The space communications environment is characterized by long round trip data times, connectivity losses, and asymmetric data rates. DTN’s solution to these challenges is a store and forward technique that optimizes data flow in this complex and challenging environment. The utilization of DTN throughout the DOTCOM network enables the success of the project and leads into the satisfaction of functional requirement 2. Functional requirement 2 outlines the need for endpoint support on the lunar surface and in the lunar orbit. The Network Capacity Model, which was created to optimize a DTN network, analyzed the network of 55 nodes by simulating the flow of data throughout all the nodes. In order to satisfy functional requirement 2, the model determined a data rate of 35.4 Mbps that was necessary across all constellation satellites which was then validated as feasible in the link budget model. Completing these two steps, alongside a full system model in the Network Capacity Model, verifies function requirement 2’s endpoint support requirements.

3.3 Spectrum Allocation and Link Budgets

RF communications will be utilized for short distance inter-satellite and satellite to ground data relay, and a laser (optical) link will be used for long distance interplanetary data relay. Laser communications provides drastically higher data transfer rates and is the main reason why it will be used in the Earth to Moon communication link to achieve the required 500 Mbps data transfer rate between the two planetary bodies. RF is more reliable and has a much larger beam width which increases coverage and will be used for inter-satellite and satellite to lunar ground communications. The transmit frequency used throughout the system is the Ka-band (26 GHz). This frequency was chosen because the Ka-band has a large bandwidth and has been determined to be an ideal frequency for future lunar communications [23]. The frequency used for the laser link communications is 193.4 THz, which is a wavelength of a 1550 nm. Below are the link budgets for
the inter-satellite links and satellite to lunar ground links. A more detailed explanation of the variables that were used in the link budget calculations can be found in the Manufacturing section under RF Link Budget Model.

<table>
<thead>
<tr>
<th>Transmit Power</th>
<th>Pt</th>
<th>43</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range (Worse Case Separation)</td>
<td>r</td>
<td>11024</td>
<td>km</td>
</tr>
<tr>
<td>Transmit Frequency</td>
<td>$f_t$</td>
<td>26</td>
<td>GHz</td>
</tr>
<tr>
<td>ISL Receiver Noise Temp</td>
<td>Ts</td>
<td>700</td>
<td>K</td>
</tr>
<tr>
<td>Data Rate</td>
<td>DataRate</td>
<td>36</td>
<td>Mbps</td>
</tr>
<tr>
<td>Antenna Gain</td>
<td>$G_t$</td>
<td>40.8</td>
<td>dB</td>
</tr>
<tr>
<td>Required Energy per bit to Noise ratio</td>
<td>$\frac{E_b}{N_0}$</td>
<td>11</td>
<td>dB</td>
</tr>
<tr>
<td>Required Design Margin</td>
<td>RDM</td>
<td>3</td>
<td>dB</td>
</tr>
<tr>
<td>Signal to Noise Ratio</td>
<td>SNo</td>
<td>99.2</td>
<td>dB</td>
</tr>
<tr>
<td>Received Power</td>
<td>Pr</td>
<td>-141</td>
<td>dB</td>
</tr>
<tr>
<td>Free Space Loss</td>
<td>Lsp</td>
<td>201.6</td>
<td>dB</td>
</tr>
<tr>
<td>Link Margin</td>
<td>LM</td>
<td>13.7</td>
<td>dB</td>
</tr>
</tbody>
</table>

Table 2: Inter-satellite RF Communications Link Budget

<table>
<thead>
<tr>
<th>Transmit Power</th>
<th>Pt</th>
<th>43</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range (Worse Case)</td>
<td>r</td>
<td>6885</td>
<td>km</td>
</tr>
<tr>
<td>Transmit Frequency</td>
<td>$f_t$</td>
<td>26</td>
<td>GHz</td>
</tr>
<tr>
<td>ISL Receiver Noise Temp</td>
<td>Ts</td>
<td>300</td>
<td>K</td>
</tr>
<tr>
<td>Data Rate</td>
<td>DataRate</td>
<td>36</td>
<td>Mbps</td>
</tr>
<tr>
<td>Antenna Gain</td>
<td>$G_t$</td>
<td>40.8</td>
<td>dB</td>
</tr>
<tr>
<td>Required Energy per bit to Noise ratio</td>
<td>$\frac{E_b}{N_0}$</td>
<td>11</td>
<td>dB</td>
</tr>
<tr>
<td>Required Design Margin</td>
<td>RDM</td>
<td>3</td>
<td>dB</td>
</tr>
<tr>
<td>Signal to Noise Ratio</td>
<td>SNo</td>
<td>99.2</td>
<td>dB</td>
</tr>
<tr>
<td>Received Power</td>
<td>Pr</td>
<td>-141</td>
<td>dB</td>
</tr>
<tr>
<td>Free Space Loss</td>
<td>Lsp</td>
<td>197.5</td>
<td>dB</td>
</tr>
<tr>
<td>Link Margin</td>
<td>LM</td>
<td>13.7</td>
<td>dB</td>
</tr>
</tbody>
</table>

Table 3: Satellite to Lunar Surface RF Communications Link Budget

3.4 Ground Station Final Design

A ground station interplanetary relay will be used for project DOTCOM. This decision was heavily based on the accessibility to upgrade or maintain the hardware. Along with accessibility, the team decided that the cost was very important and that using the ground station relay was significantly less expensive to design, manufacture, and maintain. Additionally, there were no significant differences in data quality/specifications, which allowed the decision to be based on the other important aspects mentioned above.

3.5 Hardware Final Design

The hardware design consists of 5 major elements. The first two elements are the Raspberry Pi devices and the ION software implemented across all Raspberry Pi nodes. The ION software is an open-source software provided by JPL, this software is in charge of all network protocol operations. Then links between these nodes will be created using a home modem with WiFi. These links will have associated delays hard coded into the nodes to reflect the light time delay between said nodes. For example, if a link was made between a ground station on Earth, and a ground station on The Moon, the associated delay would be that distance divided by the speed of light. The fourth element of the hardware design is the monitors that will be displaying the output of the Raspberry Pi’s. Finally, a software called Wireshark will be used to analyze the data traffic being sent over the home modem.
This design will be implemented to find reliable values for hardware latencies that can be applied to the team’s MBSE simulation to verify data rate requirements are being met. Along with generating values to be used in the final simulation, this test will verify the team’s network capacity model is working as expected.

![Diagram of the hardware test bed schematic](image)

Figure 6: Hardware test bed schematic

The group will be sending messages synchronously and asynchronously between the two nodes depicted in Figure 12. After using the ION software to perform the network protocol operations, the team can generate timestamps for when data is entering, leaving, and being stored in a node, by using Wireshark. Using these parameters the hardware latencies can be calculated by simple subtraction. Further, these latencies can be entered into the MBSE simulation as hard-coded latency values that will verify the teams design is meeting data rate requirements. After a hardware latency value is converged on throughout testing, the team will verify the network congestion model created is working as expected. This will be done by finding the amount of data stored in a node throughout the test, and verifying that that amount is not higher than it’s capacity. By verifying the storage on each node is not over it’s capacity, the network congestion model can be verified by the hardware testing process.

Team DOTCOM’s hardware test aimed to achieve two goals. The first goal was to calculate a value for hardware latency. The latency calculations include reception, storage, and forward latency values. As described before, there are two software implementations that must occur to be able to calculate these values. First, the ION software package, which is a flight ready, open source software, using DTN procedures, which will be the network protocol. This software package enables data to be sent throughout nodes, however it does not provide timestamps or any analytical tools for the message. Another software package must be installed to easily and correctly interface with the ION protocol. This next package is called Wireshark, which is a software package that analyzes the data being sent over a network. Once a transmission is made throughout nodes, timestamps should be seen by Wireshark and lead to the calculation of hardware latency. When implementing this software package the team was not able to correctly configure Wireshark onto the Raspberry Pi’s. This lead to the team falling back on the first goal and to instead provide a proof of concept of the ION-DTN software. The team was able to send messages synchronously and asynchronously, shown in the demonstration video attached to the appendix. This completed a full level 1 success for Team DOTCOM, and a partial level 2 success.

The second goal of team DOTCOM’s hardware test is to verify the network congestion model created by the team. To meet this goal the team would have to find the storage of each node, over time, throughout each test of the respective configurations. These data storage values will be crosschecked with what the created model predicts, ensuring its validity. Due to the complications from integrating Wireshark onto the network, the messages could not be analyzed. This made this goal unfeasible for Team DOTCOM, and was subsequently abandoned.
4 Manufacturing

Conner Lewis, Hunter Rohlman, Forrest Jordan, Tristan Liu, Douglas Brough, & Buck Guthrie

4.1 Architecture Model Software

As the system architecture of multiple satellites in orbit around the moon is designed to operate entirely in space, it is not feasible to create a useful, physical analog of it as a part of this project. Thus, modeling and testing for this portion of the project were entirely software-based.

4.1.1 STK Architecture Models

The first segment of the architecture software was the model of the constellation around the moon created in STK. This STK model was used for both visualizations as well as the source of the ephemeris data for each satellite, which was essential in subsequent models and tests to provide the physical location of each satellite across the time steps considered. This project ultimately modeled and tested seven different constellation designs in order to characterize their attributes, particularly performance and cost. Each of the constellation configurations were designed according to [9] and were modeled in separate scenarios in STK.

Analysis of the constellation design began with the definition of each of the individual satellites’ orbits. With \( \Omega \) referring to the Right Ascention of the Ascending Node of an orbit and \( \nu \) referring to its true anomaly, [9] gives the following description of a constellation configuration T/P/F, where T refers to the number of satellites, P refers to the number of orbit planes, and F refers to the satellite spacing:

\[
\Delta \Omega = \frac{360^\circ}{P} \\
\Delta \nu = F \times \Delta \Omega
\]

Applying the above to a 6/6/4 configuration gives \( \Delta \Omega \) of 60° and \( \Delta \nu \) of 240°, and thus gives the following orbital parameters to the six satellites in the baseline configuration.

<table>
<thead>
<tr>
<th>Satellite</th>
<th>( \Omega (^\circ) )</th>
<th>( \nu (^\circ) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>120</td>
</tr>
</tbody>
</table>

STK’s built-in tools were used to specify the semi-major axis, eccentricity, inclination, right ascendion of ascending node, and true anomaly of each orbit. The period over which the model was evaluated was ultimately one orbital period in order to best express the coverage of the constellation with the most accuracy, this being 10.38 hours at a time step of 60 seconds. Satellite-by-satellite details of each configuration can be found in the appendix. Once the models were complete, the ephemeris for each satellite was exported for use in subsequent models and tests.

The software manufacturing for the system architecture model also consisted of the coverage and connectivity tests, which were carried out in Matlab. The details of these tests are described in the Verification and Validation section.

4.2 Network Capacity Model Software

4.2.1 Introduction to the Network Capacity Model (NCP)

The purpose of the network capacity model is to provide a tool to analyze network traffic and congestion in the DOTCOM network. This analysis is necessary in order to meet design requirements 2.1.1 through 2.2.2. These requirements define the type and number of network endpoints that are to be supported by the final design. Endpoints in this case refer to lunar rovers, science satellites, human space vehicles, or any other device sending or receiving communications via the network. Each endpoint will have specific
data requirements that govern network traffic. In a DTN network, each relay node receives, stores, and then forwards data on through the network. Problems begin to arise when a node’s internal storage fills up and can no longer receive incoming data packets. When this happens, the network is considered congested. Network congestion is undesirable as it can cause data transmissions to fail. An analogous situation can be considered involving water tanks. Picture each node of a DTN network as a water tank with constant volume. Let the water represent bits of data. The purpose of the network of tanks is to move water from several sources into a local water tower. In this network, water gets pumped between intermediary tanks at varying rates. When the first tank begins to overflow, the water network has reached its capacity. This causes water loss and is analogous to failed transmissions in a DTN network.

As a major goal of DOTCOM is verifying the support of up to 35 endpoints, it is important to develop a method to test this. This model will determine basic satellite performance requirements for the network architecture to remain functional at all times. As it is impossible to model this through a hardware demonstration the team determined that a software model would be best suited for the job.

### 4.2.2 NCP Development

The network capacity model handled this task by utilizing information on connection windows from the architecture model alongside known space vehicle data rates to analyze the flow of network data in the architecture design. The principles behind the model is as follows.

To simplify the model two assumptions must be made. Data is assumed to be a continuous quantity, and data transfer between nodes occurs instantaneously. In reality, digital data is a discreet quantity and travel delays exist in the data transfer process. Making these assumptions allows for the derivation of a connected set of differential equations that dictate the flow of data through a network.

Figure 7 represents a single node in a DTN network. It has two incoming data connections (a1 and a2) and one outbound connection (a3). Note that a1, a2, and a3 are data transfer rates. We define the total data stored inside the node as m. The rate of change of data in the node is the difference in the inbound and outbound rates, given in equation 16.

![Figure 7: Single Network Node FBD](image)

$$\frac{dm}{dt} = a1 + a2 - a3 \tag{1}$$

Solving this rate equation for m yields equation 2, a function for the amount of data in the node with respect to time.

$$m(t) = \int_{0}^{t} (a1 + a2 - a3) dt \tag{2}$$

Now we connect two other nodes to create a small network. Node two will receive data from node one. Node 3 will receive and send data to and from from node one, and will also receive data from node two. Figure
8 is a visual of how these nodes are connected. This network scenario yields a system of three equations that defines the rate of change of data in each node. The full system is given in equation 3.

$$\frac{dm_1}{dt} = a_4 - a_1 - a_3$$  \hspace{1cm} (3)
$$\frac{dm_2}{dt} = a_1 - a_2$$
$$\frac{dm_3}{dt} = a_2 + a_3 - a_4$$

We can define a matrix containing all of the relevant data transfer rates as $A$. Element $A(1,3)$ corresponds to the rate of data transfer from node 1 to node 3 ($a_3$). $A(2,2)$ indicates the rate of data being internally generated at node 2 (zero in this case). The $A$ matrix for the described scenario is given in equation 4.

$$A = \begin{bmatrix} 0 & a_1 & a_3 \\ 0 & 0 & a_2 \\ a_4 & 0 & 0 \end{bmatrix}$$  \hspace{1cm} (4)

To capture the network connections, we define a similar matrix $C$. Each element in $C$ will equal one if nodes are connected or zero if nodes are not. For example, if $C(1,2) = 1$ then node one is connected and sending data to node 2. If $C(3,1) = 0$ then node 3 is not sending data to node 1. Equation 5 shows the $C$ matrix for this scenario.

$$C = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$  \hspace{1cm} (5)

The rate of change of data in each node of the network is given by the diagonals of the $\frac{dM}{dt}$ matrix in equation 6.

$$\frac{dM}{dt} = C^tA - AC^t$$  \hspace{1cm} (6)

Note that the diagonals of $\frac{dM}{dt}$ in equation 6 match the original system of equations defined in eq. 3. $M(t)$ for each node is found by integrating equation 6. Once the amount of data in each node over time is known, determining if the network is over saturated or at capacity is straightforward. We check that the maximum stored data in each node is less than its respective storage capacity. If so, the network will function as designed.
4.2.3 Outcome of NCP Development

The Network Capacity model was successfully written and deployed for testing purposes. The model analyzed a system of 6 relay satellites, 4 ground stations, 15 surface vehicles, and 30 orbiting vehicles for a total of 55 nodes. Using common communications data rates for lunar vehicles, the network simulated the performance of the network utilizing multiple (up to 100) different data rate capabilities onboard the relay satellites. After optimizing the performance of the network, the model determined the optimal data rate of 35.4 Mbps. A visual of the tests are shown in figure 9 below.

![Node Performance Through Multiple Tests](image)

Figure 9: Node Performance Through Multiple Tests

Figure 9 shows the variation in node to node performance based on a varying of relay satellite data rates. Understanding the details of all nodes’ performance is not important but it is important to note that nodes 20-25 are the relay satellites in question and their performance dictates the success of the test.

4.2.4 Challenges of NCP Development and Project Integration

Although the NCP was successfully developed, this model did not come without its difficulties. Creating a complex simulation of 55 nodes was no small feat. Being able to properly utilize a large matrix of connection windows and data rates would often create small errors due to a single wrong number in a loop that would propagate throughout the rest of the model. This led to many hours being sunk into debugging that could have otherwise been used to further the capabilities of the model. However, overcoming this obstacle was simple as it was entirely based on time spent developing the NCP. Additionally, this model was very hardware intensive and initially required a very long period of time (several minutes per operation) and testing was difficult. This also required an intensive process of improving the model’s performance which could have again been used to further the capabilities of the model. Overall, the challenges that were encountered were expected and handled without jeopardizing the deployment of the model for DOTCOM.

After going through the process of running multiple tests using the Network Capacity Model, the data rate of 35.4 Mbps was incorporated into the link budget models developed. Additionally, this model is pulling information from the architecture model in STK to form the connection map. Combining the functionality of the NCP with other developed models led to a fully functioning system that satisfied the design requirements for DOTCOM.
4.3 Link Budget Software

A key aspect to the success of a communications architecture is the link budgets. The link budgets account for all of the communications gains and losses and determine what kind of communication parameters are needed to support data relay between two locations in the system. The main goal of determining the link budgets for different links in the system is to make sure the network is able to achieve the required data rates provided by the network capacity model and the functional requirements. Radio Frequency (RF) and Laser link budget software models have been developed to determine the necessary parameters to achieve successful links between the Earth and the Moon, the lunar surface and orbiting satellites, and for intersatellite communications. These models were developed in MATLAB and the code can be found in the Appendix. A successful link is dictated by a positive link margin, which is one of the outputs of each of the models.

4.3.1 RF Link Budget Model

The RF link budget model is a collection of equations that have been found through research to determine the link parameters for communication links in the DOTCOM network. Most of the Link Budget equations were adapted from the AMSat link budget calculator developed by Jan King [31]. This tool was used in the past by the Hermes CubSat Project which was sponsored by the Colorado Space Grant Consortium. For the RF link budget model, certain input parameters like data rate and antenna size are taken from the Network Capacity and STK models to ensure the network meets certain design requirements. The main goal of the RF link budget model is to ensure successful communications throughout the system by inputting parameters for some of the boundary cases like maximum range between constellation satellites and max distance from the lunar surface. Many variables go into the calculation of the link margin including: range, transmit frequency, transmit power, antenna diameter, antenna efficiency, system noise temperature, Required Design Margin (RDM), Required Data Rate, and Energy Per bit to Noise Ratio ($E_b/N_0$). Some of these variables are determined by the satellite architecture and network capacity model. Those variables are the antenna diameter, the distance (range) between satellites in orbit and the satellites to the lunar surface, as well as the necessary data rates to ensure the network is not over capacity. The other parameters are design variables that are carefully determined through research to show what could be expected for different link budgets in the DOTCOM network.

The RF link margin is calculated by taking the difference between the actual power to noise ratio and the minimum power to noise ratio. The $E_b/N_0$ is assumed to be the minimum power to noise ratio and is based on the modulation type and desired Bit Error Rate (BER). It is assumed that BPSK modulation is used for all RF links in the system because BPSK has a high bandwidth efficiency. Based on the plot of the BER curves as a function of $E_b/N_0$, which can be found in Figure 50 in the Appendix, the Energy per bit to Noise ratio should be approximately 11 dB to achieve a BER between 1E-6 and 1E-7.

The range used in the link budget is taken from the satellite configuration and the maximum distances are used to ensure all communication configurations can be supported. The maximum distance between the satellite constellation and the lunar surface is 6885 km and the maximum distance between adjacent satellites is 11024 km.

The transmit frequency used throughout the RF portion of the system is the Ka-band (26 GHz). This frequency was chosen because the Ka-band has a large bandwidth and has been determined to be an ideal frequency for future lunar communications [23]. One of the functional requirements of the project is to be able to support and communicate with existing/future infrastructure. To be compatible with the Lunar Gateway the system must be able to use frequencies between 22.55-23.15 GHz and 25.5-27 GHz based on the International Communication System Interoperability Standards provided by NASA [30]. The network uses the Ka-band frequencies to be able to communicate with the Lunar Gateway in the future.

The transmit power used throughout the satellite configuration is 43 W. This was determined through the power optimization test which is described in the next section.

A main assumption in the link budget model is that the satellites are equip with parabolic dish antennas. The model takes in the antenna efficiency and antenna diameter and calculates the antenna gain. If other types of antennas need to be used or analyzed, their gain values can be written into the script and override the gain calculation. For this link budget model it is assumed that the satellite antenna diameter is 0.5 m which allows for total coverage of the lunar surface based on the satellite configuration. It is also important to note that the antenna gain in the system is quite high for a communications satellite. As seen in Table 4 the antenna gain is 40.8 dB which means that the antenna is highly directive. The same gain could be
achieved by an array of antennas on the satellite or the proposed 0.5 m diameter dish antenna. In addition, it is assumed that the antenna efficiency is 65%. This is an estimate on the lower end as efficiencies could increase to about 75% to 80% and would produce a higher link margin.

Another assumption for the link model is that the receiver system temperature of the satellites and ground nodes is a constant value for all lunar antenna and a constant value for satellite antenna. A value of 700 K is used to estimate the inter-satellite antenna system noise temperature [20] and a value of 300 K is used for receiving antenna system temperature for sky facing antenna and satellites pointing at the moon [18] [21]. There are a lot of variables that contribute to the calculation of the receiver system noise temperature including antenna temperature, system line temperature, noise temperature of the low noise amplifier and the second stage amplifier, as well as the feed line coefficient. For the purposes of this model, the system noise temperatures were not calculated and did not include the latter variables listed but were simply values found from research used as baseline estimations of the system noise temperature for the link budget calculations.

The Required Design Margin (RDM) was a value provided by the customer, General Atomics, and was determined to be 3 dB. This value acts as a safety margin to ensure the communication links can be made.

The data rate used for the links was 36 Mbps which was calculated through the Network Capacity Model to ensure the network remains under capacity.

Another key assumption of the RF link model is that the main loss is due to free space because the RF links in the network are for inter-satellite and satellite to lunar ground communications, therefore no atmospheric losses need to be accounted for in the link calculations. Free space loss is normally the biggest loss factor in a link budget calculation which is why it is considered an assumption to be the main loss. Additional losses which need to be considered in the link budget calculations are the antenna line loss and pointing loss. Ideally, if the transmit antenna is directly pointed at the receiving antenna and perfectly aligned then there should be no pointing loss but this is never the case in reality. The antenna pointing loss is a function of the antenna pointing offset and the antenna beamwidth. It is assumed that the pointing loss for the communications satellites in the system to be a set value of 1 dB, which is a good estimation based on a communications protocol study that analyzed the Link Budget of a VORSAT Satellite [22]. One aspect that was not considered in the link budgets was Forward Error Correction (FEC). If FEC is applied it can lower the signal to noise ratio that is required to get a specific BER. By adding half rate FEC to the data stream using convolutional coding, the decrease in BER at the output of the receiver is equal to a 3 dB increase in signal to noise ratio [32]. Therefore, if FEC is implemented the Link Margin could be improved by about 3 dB. Overall, all of the variables discussed were used to create link budgets for the DOTCOM network. A summary of the different link budgets in the system are listed again below in Table 4 and 5 for easy reference.

<table>
<thead>
<tr>
<th>Transmit Power</th>
<th>Pt</th>
<th>43</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range (Worse Case Separation)</td>
<td>r</td>
<td>11024</td>
<td>km</td>
</tr>
<tr>
<td>Transmit Frequency</td>
<td>ft</td>
<td>26</td>
<td>GHz</td>
</tr>
<tr>
<td>ISL Receiver Noise Temp</td>
<td>Ts</td>
<td>700</td>
<td>K</td>
</tr>
<tr>
<td>Data Rate</td>
<td>DataRate</td>
<td>36</td>
<td>Mbps</td>
</tr>
<tr>
<td>Antenna Gain</td>
<td>Gt</td>
<td>40.8</td>
<td>dB</td>
</tr>
<tr>
<td>Required Energy per bit to Noise ratio</td>
<td>$\frac{E_b}{N_0}$</td>
<td>11</td>
<td>dB</td>
</tr>
<tr>
<td>Required Design Margin</td>
<td>RDM</td>
<td>3</td>
<td>dB</td>
</tr>
<tr>
<td>Signal to Noise Ratio</td>
<td>SNo</td>
<td>95.5</td>
<td>dB</td>
</tr>
<tr>
<td>Received Power</td>
<td>Pr</td>
<td>-145</td>
<td>dB</td>
</tr>
<tr>
<td>Free Space Loss</td>
<td>Lsp</td>
<td>201.6</td>
<td>dB</td>
</tr>
<tr>
<td>Link Margin</td>
<td>LM</td>
<td>5.96</td>
<td>dB</td>
</tr>
</tbody>
</table>

Table 4: Inter-satellite Communications Link Budget
### 4.3.2 RF Power Optimization Test and Results

One of the tests that was conducted using the link budget model was the power optimization test. This test was done to determine the minimum power for different links in the system based on varying certain input parameters. To conduct this test, the link model was solved for the transmit power, meaning the transmit power was made a function of all of the other link parameters. Then, the link margin was held constant (at 5 dB then at 10 dB) and the antenna size was varied to identify what the minimum transmit power was for certain links in the system. The data rate used for the links was 50 Mbps even though the minimum data rate for the network to remain under capacity is 35.4 Mbps. This was to create a factor of safety in the test and ensure that the required data rate was achievable. The two links that were analyzed in this optimization test were the lunar ground to satellite link and satellite to satellite link. The distance between the two communication nodes were taken from the STK simulation. For both cases the worst case scenario, maximum range, is used. The goal of this test is to determine the minimum transmit power for these different nodes in the system.

The first link that was analyzed was the lunar ground to orbiting satellite link. In this link, the receive antenna diameter was fixed at a constant value of 0.5 m which assures that we are able to achieve 99% coverage of the lunar surface based on the satellite constellation. The transmit antenna diameter was varied between 0.1 and 1 m as these are within reasonable ranges for a lunar ground based antenna [24]. The results of this test reveal that with a transmit antenna diameter of 0.2 m the minimum power required to achieve a 5 dB link margin is 15 W. To achieve a more reliable and robust link margin of 10 dB, the minimum transmit power would be 42.8 W. The second link that was analyzed was an inter-satellite communications link. The receiver antenna diameter was held constant at 0.5 m and the transmit diameter varied. To achieve a link margin of 5 dB, with a 0.5 m diameter transmit antenna, the minimum transmit power is 38 W. To achieve a more reliable link with a 10 dB link margin, the required transmit power would be 124 W. Based on these tests, without exceeding a 1 m diameter for the lunar ground station or a 0.5 m diameter for inter-satellite communications, the minimum transmit power required to support the RF links in the system is 42.8 W which allows for an reliable link margin of 5 dB for certain links as well as a 10 dB link margin for other links. This value of 42.8 W transmission power is within a reasonable range as GPS satellites use 45 W of transmission power [25]. The plots of antenna diameter vs transmission power produced for these two tests can be seen in Figure 10 and 11.
4.3.3 Laser Link Model

The laser link model is essentially a baseline estimation of the power received and link margin for the optical link between the Earth and moon ground stations. Through research it was determined that one of the only ways to calculate the laser link margin is to use an equation that represents the link margin as the ratio of the received power and required power. The power received equations were provided by General Atomics and can be found in Figure 51 and 52 in the Appendix. The power received term is a function of the transmit power, antenna gain, free space loss, pointing error loss, antenna efficiencies, and atmospheric losses. The transmit power used for the optical link is 5 W which is on the lower side of transmission power used for optical communications [28]. 10 W of transmission power could be implemented but the low end was used for the link margin calculation as it could be increased if needed [16]. The transmit and receiver antenna diameter are both taken to be 1 m which makes an antenna gain of 126 dB which is within a reasonable range as a study on optical laser cross-links used for satellite communications denotes an antenna gain for a satellite system to be 113.51 dB [29]. The DOTCOM network is using ground stations which can support much larger antenna (much higher gain) compared to the satellite antenna noted in this study so the antenna size and gain values used and calculated in this model are reasonable. The free space loss between the Earth and the Moon is estimated to be -309.8 dB assuming the distance between the Earth and the Moon to be 384,000 km. Other estimated values for the optical link are the transmitter pointing error and the transmitter and receiver optical efficiencies. The transmitter pointing error is assumed to be -2 dB based on the estimations presented in a Link Summary Table between the Earth and Moon found in the Deep Space Optical Communications
The Link Summary Table can be seen in Figure 53 in the Appendix. The transmitter and receiver optical efficiencies can range between 50% and 80%. It is assumed that the optical efficiencies are 70% for this model. It is important to note that even when using the worse case optical efficiency of 50% the link margin for the optical link is still positive. The final variable in the link margin calculation that needed to be estimated is the atmospheric losses. The atmospheric effects and losses will vary depending on location and because there is no specific ground location used for this laser link margin calculation, it is assumed that the atmospheric losses are 5 dB. This value could be higher or lower depending on the atmospheric conditions. The atmospheric loss could be as high as 20 dB and still produce a positive link margin.

The required power term in this calculation is a function of the frequency, data rate, receiver sensitivity (Number of photons per bit) which is associated with the BER, and Planck’s constant [26] [27]. The wavelength used for the laser link in the system is 1550 nm. After research it was determined that this wavelength is an ideal wavelength for an optical communication links due to the minimal atmospheric attenuation at this wavelength [16]. The data rate used was 500 Mbps, the minimum required data rate for Earth to Moon communications based on the functional requirements of the project. For this link it is assumed that the modulation type is OOK NRZ (On-Off Keying Non-Return-to-Zero). The receiver sensitivity used was 70 photons per bit which corresponds to a BER of 1E-9 which is a function of the modulation type. The results of this model estimation show in order to achieve a data rate of 500 Mbps, the required power is -83.4 dBW and the link margin is 21 dB. The laser link Model MATLAB code can be found in the Appendix.

### 4.4 Hardware Manufacturing

The hardware test bed was constructed in order to both satisfy the ASEN curriculum’s need for a hardware component, as well as serve as a proof-of-concept for the ION-DTN open-source software. The word manufactured here is interpreted as defining the construction of the hardware test bed system. As mentioned in the final design section, the hardware test bed consisted of two Raspberry Pi 4 node, each equipped with ION-DTN. JPL associate Scott Burleigh is credited with the development of this software suite, which was downloaded from sourceforge.net. The Raspberry Pis were loaded with Ubuntu 18.04 server operating systems in order for the ION-DTN software to have a virtual machine platform to operate on. Two keyboards, two computer monitors, as well as two HDMI cables were used in testing and configuring the system in Douglas Brough’s living room.

Interpreting both the operating capabilities of the Raspberry Pis as well as proper installation and deployment of ION-DTN served to be a significant challenge. The hardware subteam was unaware of the knowledge needed to interpret the ION-DTN user guide, as the guide was made specifically for it. It was unclear in the ION-DTN documentation which Ubuntu server version was used, and how specifically to download and use the ION-DTN packages on the Ubuntu command-line interface. Through testing the network connection of the Raspberry Pis using both HDMI cables as well as over wireless internet connection, we concluded that the ION-DTN software allowed the Raspberry Pis to best connect to the ION network via wireless connection. As little documentation exists for sending and receiving messages as well as designating node points, the team ran into roadblocks when using the commands “bpsource” and “bpsink”, which allowed the team to specify transmission starting and ending points through recognition of IP addresses on the network. However, in the end the system proof-of-concept was validated, and the SFR demonstration video gave an overview of the visual process. Additional visuals of the hardware functional block diagram as well as the test stand setup is given below:
5 Verification and Validation

Caelan Maitland, Buck Guthrie, Hunter Rohlman, Forrest Jordan, Conner Lewis & Jennifer Gurtler

Project DOTCOM’s final deliverables to General Atomics are based around producing a fully-realized network architecture which a real-world network can be built around. As such, verification and validation of the system model and network architecture will be done in two parts. The team will use both a Network Capacity Model and a full model of the system’s networked nodes and connections built in SysML in order to validate both the DOTCOM network architecture and network protocol, while ensuring that the final network design indeed satisfies all requirements of the system.

5.1 Validation of the Network Capacity Model

The network capacity model makes two core assumptions. These are that data transfer is instantaneous between nodes and that data volume is handled as a continuous variable. The average light delay between
the Earth and the Moon is 1.3 seconds. If not handled correctly, this delay could potentially break the instantaneous data transfer assumption. To determine what impact this delay could have, let's look at an analogous situation involving water movement and a storage tank.

Let tank 1 be an infinite source of water for tank 2. Tank 1 can release water at a rate of 10 gal/min. It takes water 5 minutes to travel from tank 1 to Tank 2. Tank 2 can release water at a rate of 5 gal/min. Figure 14 is a diagram of this two tank system.

If tank 1 and 2 are both set to continuously release water, tank 2 will eventually fill up and overflow. This is due to the larger flow rate that tank 1 is able to release. The designers of the tank system know this and build a float valve that shuts off the water supply from tank 1 when tank 2 fills up. Unfortunately, the designers forgot about all the water still traveling down the pipe and tank 2 overflows anyway. A better solution would be to design a shutoff based on the amount of water leaving tank 5 rather than the amount in tank 2 at any given time. By tracking how much water gets introduced into the system by tank 1, and assuming that it instantaneously travels into tank 2, the designers could have avoided a spill. This is the approach that the DOTCOM network capacity model uses to ensure nodes can handle the required data load without overflowing. A key component to the success of this approach is the assumption that data transmission is instantaneous.

The second core assumption to the network capacity model is that data volume is a continuous quantity. In reality, digital data is a discreet set of ones and zeros. Data bounces around the network in bit sized packages. By treating data volume as a continuous variable, we assume that the data stored in each node can take on any value greater than or equal to zero. This assumption only becomes problematic at extremely low data rates on the order of bits/second. On the Mb/s and Gb/s range that DOTCOM operates under, there is enough resolution to justify the continuity assumption.

5.1.1 NCP Results Verification

The Network Capacity Model testing was conducted entirely in MATLAB. As described in section 4.3.2, the baseline mathematics behind the analysis of the network is equation 6 which simply utilizes the connection window and data rate matrices of the network. The test comes in when the data rate of the constellation relay satellites, or the satellites that manage all data transmission throughout the lunar network, are changed to optimize the required data rates and minimize the on-board power consumption. The test operates by running the calculation process of the NCP multiple (up to 100) times with a large vector of varying data rates to determine the turning point where the network shifts from being over capacity to under capacity.

The method for determining whether the network falls into the category of over capacity or under capacity is also quite simple. The model examines the internal storages of each constellation satellite and requires each one to return to approximately zero once they complete their connection window with the lunar ground station. If the data stored does not return to zero, the nodes will slowly stockpile data until there is saturation in their storages. After running multiple iterations of the test and measuring the internal storage of those constellation satellites, the Network Capacity Model determined that the minimum data rate for the constellation satellites to keep the network under capacity was 35.4 Mbps. This data rate holds significant value for the overall success of the project. Firstly, this result is utilized in the teams Link Budget Model to verify that this result is manageable and feasible on a satellite (this will be discussed in the next section). Additionally, this test alongside the Link Budget Model’s verification of this test, allowed the team to satisfy functional requirements two which detailed the support of surface and orbital vehicles around the moon.

5.2 Verification of Network Architecture Model

5.2.1 Matlab Coverage and Connectivity Tests

The following three tests of the architecture model were manufactured and carried out in Matlab. They strictly considered the geometry of the system, particularly signal range and elevation limitations, to determine which nodes of the network are able to connect to which others and when. It is important to note that only line-of-sight to any satellite was considered as criteria for connection feasibility, and time required for attitude control and pointing the satellite’s antenna were not taken into account.

The first test assessed the ability of points on the surface of the moon to connect into the network in order to determine if the system is able to fulfil the 99% surface coverage specified in Functional Requirement 1. The test separated the moon’s surface into a 1° square grid of 180x360 units, the vertices of which were used as the points from which connectivity was assessed. In this scenario, on the surface of the moon, the primary
Figure 14: Water Tank System Diagram
restriction is the elevation of the satellites, as the satellites go over the horizon before they go out of RF range. It should be noted that this model assumes the moon is a sphere, both with regard to the shape of the entire body as well as the topography of the surface, and the minimum elevation of a satellite to be eligible to connect was set at 5° as shown in Figure 15. This is not an all-encompassing guideline, and receivers in a deep crater or the shadow of a mountain may have poorer connectivity than described by this model. The test noted how many satellites in the constellation a particular point is able to connect to at each time step, information which was then reported in the form of the coverage multiple chart shown in Figure 6.

Figure 15: Illustration of the minimum elevation required for connectivity in the architecture model

<table>
<thead>
<tr>
<th>Configuration</th>
<th>0x</th>
<th>1x</th>
<th>2x</th>
<th>3x</th>
<th>4x</th>
<th>5x</th>
<th>6x</th>
<th>7x</th>
<th>8x</th>
<th>9x</th>
<th>10x</th>
<th>11x</th>
<th>12x</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/6/4</td>
<td>0</td>
<td>14.63</td>
<td>66.78</td>
<td>18.48</td>
<td>0.109</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12/6/4</td>
<td>0</td>
<td>0</td>
<td>5.528</td>
<td>16.65</td>
<td>44.98</td>
<td>24.03</td>
<td>8.82</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12/12/10</td>
<td>0</td>
<td>0</td>
<td>0.077</td>
<td>18.94</td>
<td>49.35</td>
<td>30.17</td>
<td>1.458</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15/15/6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.338</td>
<td>16.66</td>
<td>51.2</td>
<td>24.86</td>
<td>5.849</td>
<td>0.09</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15/15/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.048</td>
<td>20.73</td>
<td>44.71</td>
<td>30.75</td>
<td>3.738</td>
<td>0.034</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18/6/4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.304</td>
<td>15.75</td>
<td>48.53</td>
<td>25.87</td>
<td>5.581</td>
<td>0.746</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24/6/4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.957</td>
<td>10.09</td>
<td>17.14</td>
<td>24.16</td>
<td>19.17</td>
<td>24.71</td>
<td>1.571</td>
<td>0.202</td>
</tr>
</tbody>
</table>

Table 6: This table lists the percentages of satellite coverage levels available to points on the lunar surface over the course of one orbital period for the seven constellations tested.

This first test had very favorable results. As shown in the table, no constellation had any points with no coverage across an entire orbital period, equating to persistent 100% coverage of the lunar surface at the resolution and elevation limitations described above, and fulfillment of the surface coverage requirement specified in Functional Requirement 1.

The second test was very similar to the first, this time assessing the ability of points in lunar orbit to connect into the network in order to determine if the system is able to fulfill the 99% orbital coverage specified in Functional Requirement 1. This test separated lunar orbit into a spherical shell grid, with 2° resolution in the angular directions and a 195.9 km resolution in the radial direction. This shell began at 150 km altitude and concluded at 9757 km altitude, the altitude of the GEO-equivalent orbit assessed in the altitude trade study. The size of this shell grid was made with as large a resolution as possible without having the program exceed its RAM limitations, though this was ultimately a much lower resolution than originally planned. Fortunately, the purpose of this test was only to provide a general idea of coverage distribution, and it would be trivial to modify the code to analyze a smaller portion of the system, such as a smaller time period or altitude range, at a much higher resolution. The coverage multiple chart for this test is shown in Table 7.
Table 7: This table lists the percentages of satellite coverage levels available to points in lunar orbit over the course of one orbital period for the seven constellations tested.

The results of the second test were very favorable. As shown in the table, even the baseline constellation, the smallest, was able to provide at least single coverage to every point in lunar orbit as defined above. This equates to 100% coverage of lunar orbit, fulfilling Functional Requirement 1. However, this does not translate to a flawless system.

Each of the constellations have a particular redundancy built in that both expands the potential capacity of the network as well as safeguards against the failure of individual satellites, which is a persistent threat in real-world spaceflight. For each constellation in Tables 6 and 7, the smallest coverage multiple for which the constellation has a non-zero percentage represents the minimum number of satellites the constellation can lose before it begins to lose coverage. For example, in Table 7, the 24/6/4 constellation has no less than 5x coverage for all points, meaning even in a worst-case scenario, the constellation would have to lose 5 satellites before its coverage becomes compromised.

The third and final test determined the connection windows between all satellites in the constellation. The purpose of this test is to assess whether or not the system is able to provide continuous connection to nodes as specified in Functional Requirement 2. For a point to be eligible for continuous connection, it must be able to connect into the network at all times, and therefore it must never be fully isolated from any other point in the network.

Like the previous two tests, this test’s assessment was primarily geometric, noting that two satellites are able to connect if they are within range and not blocked by the moon. A diagram of the maximum connection distance is shown in Figure 16.

The results of this test were very promising, as they demonstrated that adjacent satellites are always able to connect. This means that, combined with the 100% coverage results from the first two tests, no part of

<table>
<thead>
<tr>
<th>Configuration</th>
<th>0x</th>
<th>1x</th>
<th>2x</th>
<th>3x</th>
<th>4x</th>
<th>5x</th>
<th>6x</th>
<th>7x</th>
<th>8x</th>
<th>9x</th>
<th>10x</th>
<th>11x</th>
<th>12x</th>
<th>13x</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/6/4</td>
<td>0</td>
<td>2.235</td>
<td>28.45</td>
<td>36.55</td>
<td>12.76</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12/6/4</td>
<td>0</td>
<td>0</td>
<td>0.854</td>
<td>2.352</td>
<td>9.255</td>
<td>9.49</td>
<td>77.85</td>
<td>0.001</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12/12/10</td>
<td>0</td>
<td>0</td>
<td>0.051</td>
<td>2.735</td>
<td>9.185</td>
<td>27.27</td>
<td>45.92</td>
<td>14.69</td>
<td>0.146</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15/15/6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.242</td>
<td>2.412</td>
<td>10.92</td>
<td>16.93</td>
<td>32.46</td>
<td>27.61</td>
<td>7.965</td>
<td>1.498</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15/15/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.034</td>
<td>3.094</td>
<td>8.396</td>
<td>19.33</td>
<td>32.26</td>
<td>27.43</td>
<td>9.106</td>
<td>0.355</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18/6/4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.126</td>
<td>2.739</td>
<td>10.94</td>
<td>12.43</td>
<td>21.94</td>
<td>28.71</td>
<td>16.11</td>
<td>1.555</td>
<td>2.544</td>
<td>0</td>
</tr>
<tr>
<td>24/6/4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.454</td>
<td>1.462</td>
<td>2.614</td>
<td>4.44</td>
<td>4.72</td>
<td>7.621</td>
<td>5.963</td>
<td>72.73</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Figure 16: Illustration of two satellites connecting at maximum separation
the network is ever isolated, and any two endpoints are always eligible for continuous connection regardless of their location. However, it should be noted that the system could not support continuous connection between all endpoints simultaneously, only that geographic location is not a factor when determining if an endpoint is eligible for continuous connection.

5.3 Verification of the Link Budget Model

The Link Budget Model developed in MATLAB was tested against a test case link budget from an intersatellite communications trade study on satellite antennas[18]. After extensive research, we determined this sources would be best to compare against the link budget model because the antenna trade study provided the most in depth input parameters out of any other source. The input parameters used for the link budget validation test can be seen in Table 8. The link model was tested against this use case by inputting the same parameters of range, frequency, transmit power, receiver system temperature, data rate, required $E_b/N_0$, and antenna gain, into the link budget model and then comparing the link margin calculated by the link budget model to that of the trade study. Based on these input parameters the link budget model produced a similar result with a link margin of 35.99 dB which is only 2 dB different than the link margin calculated in the test case of 37.99 dB [18]. The cause of this slight difference may be in the rounding of certain values throughout calculations. This small difference in the link margin verifies that the RF link model is accurate and has produced similar results to that of other link budget calculations.

<table>
<thead>
<tr>
<th>$C/N_0$</th>
<th>Bit rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_0$</td>
<td>TX Carrier Center Frequency</td>
</tr>
<tr>
<td>$B$</td>
<td>TX Carrier Bandwidth</td>
</tr>
<tr>
<td>$G$</td>
<td>ISL Antenna Gain</td>
</tr>
<tr>
<td>$P$</td>
<td>Transmitter Power</td>
</tr>
<tr>
<td>$R$</td>
<td>Worst-Case Separation (Range)</td>
</tr>
<tr>
<td>$T$</td>
<td>ISL Receiver Noise Temperature</td>
</tr>
</tbody>
</table>

Table 8: ISL patch antenna parameters for TSX-TDX boresight link margin from the Antenna Trade Study [18]

5.4 Verification via Model-Based Systems Engineering and SysML

As both a response to the futuristic time frame and highly model-oriented nature of this project and as a way to manage the larger scale of Team DOTCOM’s testing and verification requirements spread out over several software applications, as well as per directives from General Atomics, the principles of model-based systems engineering (MBSE) were applied to Team DOTCOM’s project. MBSE is a formalized methodology that is used to support the requirements, design, analysis, verification, and validation associated with the development of complex systems, in which a large focus is put on development of subsystem models that integrate within a larger model of the system as a whole. In order to do this, the team used the Systems Modeling Language (SysML), which is a general-purpose system architecture modeling language designed to support MBSE principles, and one of its many associated software compilers, named Papyrus. As of the time of this final review, Team DOTCOM’s systems engineering subteam has completed structural and behavioral modeling of the DOTCOM network design. DOTCOM’s structural modeling includes modeling data connections between network nodes, such as the laser link between the terrestrial and Lunar ground stations, as well as the internal structure of network nodes, such as data connections between a satellite antenna and its signal processing hardware. The behavioral modeling of the DOTCOM network includes the modeling of use-case interfaces with users of the network, as well as the the actions of system nodes such as network satellites. Finally, this behavioral and structural modeling of the system is used to perform requirement tracing and verification of system functions. Explanations of this modeling in greater detail are given in the subsections below.
5.4.1 Block Definition Modeling

Block definition modeling, and its related block definition diagram, is a method of structural modeling that shows an overview of system nodes and the relationships between components. System nodes such as ground stations or satellites are represented by blocks containing parts such as antennae or central processing units, and the data transmission relationships between these blocks are then drawn. In the case of the DOTCOM network, block definition diagrams are being used in order to represent the system nodes under either the terrestrial or Lunar mission segments, as well as a larger view of the data connections between nodes within all mission segments.

Figure 17: An example of the block definition diagram that shows the network structure of the DOTCOM networks.

5.4.2 Internal Block Modeling

Internal block modeling, as the name suggests, is a structural model of the connections and parts that make up a single system node. In the case of Team DOTCOM’s network, internal block diagrams are used to show the encapsulated contents of satellites, ground stations, and other system vehicles, including signal processing units, storage that facilitates delay-tolerant networking protocols, and the interfaces between said internal structure.
5.4.3 State Machine Modeling

State machine modeling allows for the modeling of network node behavior. Said modeling is accomplished via use of states, which represent conditions or situations a node may encounter within its operation. When the node reaches this condition, it performs an activity or somehow interact with the rest of the system. A state machine diagram is then a behavioral diagram that shows the sequence of states the node goes through while performing its duties, and additionally what actions it may take when it reaches them. Some of these states have been left in a high-level format within the DOTCOM modeling workspace, as they rely on established elements of the completed network that are beyond the team’s scope. Such an example of this would be how the satellite actually decides to fire its maneuvering thrusters in order to move into position to send and receive communication data.
5.4.4 Use Case Modeling

Use case modeling, in the context of SysML, is used to show how the system interacts with external users of the network. This allows the subject system’s interactions to be described in a non-technical, graphical representation for stakeholders, including those without a technical understanding of the project or a background in SysML development. Use case diagrams show these user relationships in the context of those inside or outside the network, and thus can be applied to software and hardware systems, as well as persons or organizations using the network.

![Use Case Diagram](image)

Figure 20: An example of a use case diagram that shows the system of Lunar Satellite interacts with the Lunar Ground Stations and the Lunar Orbit/Surface Vehicles.

5.4.5 Requirement Tracing & Satisfaction

Successful modeling of the behavioral and structural sides of the network allow the team to perform requirement tracing and satisfaction steps, which is achieved by the use of «Satisfy» associations within the SysML workspace. These associations create a hard-coded relationship between the requirement being satisfied and the specific network element which fulfills the requirement. Linking a network element to a requirement through use of the «Satisfy» association also allows this relationship to be shown visually via a requirement satisfaction matrix, which enables this connection to be displayed and checked off in an easily understandable tabular format.
This visual matrix representation between a series of requirements and the network elements that satisfy them is shown above in Fig. 21. As an example, since Design Requirement 1.1 (shown as R-005 within the SysML naming scheme) specifies that the network must feature transmit capability between different planetary mission segments, the DOTCOM network has been designed in such a way that ground stations on the Earth and Moon will be able to support this transmission via a laser communications link that enables high-speed data transfer. Each network node additionally possesses internal features and behavior modeling that enable outbound data transmission capability, first from satellites to ground stations and then from ground station to ground station between mission segments, in order to satisfy other network requirements for simultaneous and non-simultaneous transmission to network nodes. This data transmission is done with full store-and-forward signal protocol via node properties such as the attributes *Antenna* and *OutboundSignalProcessing*, which can be seen in Appendix 10.4, Fig. 45, which allows DOTCOM to then complete this «Satisfy» relationship, tying these features of the system to the completion of the requirement.
### 6.1 Modeling Risks

<table>
<thead>
<tr>
<th>Risk</th>
<th>Description</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Capacity Model</td>
<td>The network capacity model is based upon assumptions of constant data flow rate and does not consider additional factors for transmission such as on board processing and read or write speeds.</td>
<td>The network capacity model will be utilized to verify that the network architecture is valid by showing how data traverses the network. Inaccuracies in the model will result in lower fidelity data and a poorer representation of the overall architecture. This will reduce the ability of the model to verify the DOTCOM design claims.</td>
</tr>
<tr>
<td>Inaccurate Assumptions</td>
<td>Every model DOTCOM creates operates on some foundational assumptions. The risk comes from the potential that these assumptions do not apply under specific circumstances when analyzing and modeling the network architecture.</td>
<td>Systematic errors may occur in the model in question. This creates a discrepancy between the true network and the model. Additionally, this will degrade the credibility of the data supplied by this model to verify design requirements.</td>
</tr>
<tr>
<td>MBSE Integration</td>
<td>Model-Based Systems Engineering is the system utilized to combine the created models to validate that design requirements are met. This requires one cohesive model in order to be effective. The members working on MBSE are using different computers and may face difficulties combing their respective models to create the overall cohesive model the system.</td>
<td>As the finalized MBSE model is the primary deliverable for DOTCOM, this risk directly effects the outcome of the spring semester deliverables. It is likely that a general model could still be generated in separate pieces, however this does not completely satisfy the requirements set by General Atomic.</td>
</tr>
</tbody>
</table>

Table 9: Modeling Risk Descriptions

![Risk Matrix of Modeling Risks](image)

Figure 22: Risk Matrix of Modeling Risks
The network capacity model underwent multiple iterations and improvements. This allowed the model to slowly cover more gray areas in the design plan as well as reduce the inaccuracies in the fundamental math. However, the risk was not completely mitigated as the model is still incapable of accounting for hardware latency as well as still being unverified by the hardware test bed.

Throughout the model creation process the team kept a close eye on the equations utilized and the underlying assumptions of each one. This is the only true method for mitigating the detrimental effects of each assumption. DOTCOM ensured that each model utilized the appropriate assumptions through thorough research and careful implementation. DOTCOM is confident that all models have accurate assumptions that do not take away from the legitimacy of their results.

The MBSE team will continuously compared models as well as tested the compatibility of said models throughout the spring semester. This worked greatly as the MBSE team managed to successfully create a project space that contained the results of the models as well as verified requirements.

<table>
<thead>
<tr>
<th>Risk</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Capacity Model</td>
<td>The network capacity model underwent multiple iterations and improvements. This allowed the model to slowly cover more gray areas in the design plan as well as reduce the inaccuracies in the fundamental math. However, the risk was not completely mitigated as the model is still incapable of accounting for hardware latency as well as still being unverified by the hardware test bed.</td>
</tr>
<tr>
<td>Inaccurate Assumptions</td>
<td>Throughout the model creation process the team kept a close eye on the equations utilized and the underlying assumptions of each one. This is the only true method for mitigating the detrimental effects of each assumption. DOTCOM ensured that each model utilized the appropriate assumptions through thorough research and careful implementation. DOTCOM is confident that all models have accurate assumptions that do not take away from the legitimacy of their results.</td>
</tr>
<tr>
<td>MBSE Integration</td>
<td>The MBSE team will continuously compared models as well as tested the compatibility of said models throughout the spring semester. This worked greatly as the MBSE team managed to successfully create a project space that contained the results of the models as well as verified requirements.</td>
</tr>
</tbody>
</table>

Table 10: Modeling Risk Mitigation Strategies and Results

### 6.2 Inaccessibility Risks

<table>
<thead>
<tr>
<th>Risk</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside Architecture</td>
<td>DOTCOM is required to utilize outside architecture like Lunar Gateway to maximize the capabilities and cooperative ability with others. This allows DOTCOM to ensure continuous operational ability.</td>
</tr>
<tr>
<td>Information</td>
<td>In order to properly utilize this architecture, information regarding it must be available online. As most of the infrastructure in space has documentation regarding it this risk carries low weight, however for any architecture with insufficient documentation it would be impossible to properly utilize it in DOTCOM.</td>
</tr>
<tr>
<td>COVID Building Restrictions</td>
<td>COVID-19 has been imposing restrictions to building access which includes construction and testing facilities. There is a significant chance that these restrictions would not be lifted in the near future. These facilities were utilized by DOTCOM to construct and test the hardware component of the design.</td>
</tr>
<tr>
<td></td>
<td>If COVID-19 restrictions are not lifted, the team would be unable to utilize the high-end facilities offered by CU. This would drastically increase the difficulty of hardware construction and testing as the team would have to procure the necessary equipment. This may also impact the budget for the team.</td>
</tr>
</tbody>
</table>

Table 11: Inaccessibility Risk Descriptions
As with many other aspects of DOTCOM, the team ensured thorough research is conducted regarding all outside architecture. If little information was found, the team would utilize connections through General Atomic to ensure that enough information is found to allow for compatibility between DOTCOM, Lunar Gateway, and others. Through the team’s research, the information required was found and the risk was completely mitigated.

Unfortunately, this risk was not controllable by the team. To prepare for the situation where these restrictions were still in place, DOTCOM moved forward under the assumption that the entirety of the hardware construction and testing will need to be done locally and not incorporate the facilities offered by CU into the test plans. This process allowed DOTCOM to successfully mitigate this risk as it was very possible to construct this hardware outside of restricted buildings.
6.3 Additional Risks

<table>
<thead>
<tr>
<th>Risk</th>
<th>Description</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raspberry Pi ION Integration</td>
<td>The purpose of the hardware test bed was to properly emulate the overall network architecture for DOTCOM. As the architecture is utilizing delay tolerant networking (DTN), ION must be integrated as it implements the DTN capability into the Raspberry Pi’s.</td>
<td>Without ION integration the hardware does not accurately resemble the network architecture. This also implies that the hardware cannot be used to validate the functionality of the network capacity model. This would drastically affect the quality of the final deliverables of DOTCOM.</td>
</tr>
<tr>
<td>DOTCOM Project Complexity</td>
<td>DOTCOM is a multi-faceted project. With much emphasis on research and modelling, as well as hardware and testing, there are many deliverables required in the final design. The team will design a network architecture, create multiple software models, test these models through a hardware component, as well as create a research paper discussing the team’s finding. This requires a very large time commitment from each team member. The risk comes from the not insignificant chance that all of these deliverables cannot be completed.</td>
<td>As there is a limited amount of time in a semester, the team may be unable to thoroughly form the research paper component of the design. This will not greatly affect the final outcome of DOTCOM however it will still reduce the overall quality of the deliverables.</td>
</tr>
</tbody>
</table>

Table 13: Additional Risk Descriptions

Figure 24: Risk Matrix of Additional Risks
The hardware team heavily utilized available resources. This included, but was not limited to, in depth online documentation of this software and engineers working under Professor Morton who have experience with implementing this software. After extensive work in reaching out to these resources, the team successfully integrated ION into the RPI setup. However, the hardware test bed was still unable to create a DTN setup that would be replicable in the Network Capacity Model. This unfortunately means that the model will go without proper validation. Overall, the risk was mostly mitigated, however the complexity of this task proved greater than initially expected.

The entirety of the team put forth as much time as necessary to ensure a high quality final product, however limited individual time and semester length still resulted in the inability to reach the desired thoroughness in some aspects of the project deliverables.

<table>
<thead>
<tr>
<th>Risk</th>
<th>Mitigation Strategies and Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raspberry Pi ION Integration</td>
<td>The hardware team heavily utilized available resources. This included, but was not limited to, in depth online documentation of this software and engineers working under Professor Morton who have experience with implementing this software. After extensive work in reaching out to these resources, the team successfully integrated ION into the RPI setup. However, the hardware test bed was still unable to create a DTN setup that would be replicable in the Network Capacity Model. This unfortunately means that the model will go without proper validation. Overall, the risk was mostly mitigated, however the complexity of this task proved greater than initially expected.</td>
</tr>
<tr>
<td>DOTCOM Project Complexity</td>
<td>The entirety of the team put forth as much time as necessary to ensure a high quality final product, however limited individual time and semester length still resulted in the inability to reach the desired thoroughness in some aspects of the project deliverables.</td>
</tr>
</tbody>
</table>

Table 14: Additional Risk Mitigation Strategies and Results

6.4 Realization of Risks

A major goal of DOTCOM was to understand and prevent the risks to the success of this project as much as possible. However, there were still impacts on the projects due to these risks regardless of the mitigation strategies employed. The largest effects due to the risks throughout this project came from the Network Capacity Model and the Raspberry Pi ION Integration risks. As mentioned above, the Network Capacity Model relied heavily on outside verification such as the hardware test bed or literature based on DTN networks for comparison. However, DOTCOM fell back on using the hardware test bed as a proof of concept which disabled its ability to be used to verify the accuracy of the Network Capacity Model. Additionally, outside literature approaches the modeling of DTN networks with a Markov-Chain model which proved to be too taxing to create after the constructing the initial Network Capacity Model. Both of these risks combined could lead to significant uncertainty regarding the results of the model. However, due to the simple nature of the underlying principles of the model as well as the produced data rate that falls within expectations, it can be assumed that the risk of inaccuracy is minimal.

Additional risks were realized during the timeline of this project, however they proved to have little effect on the final product the team delivered. Initially, covid-19 restrictions were predicted to heavily affect the ability to construct the hardware test bed, however the final test bed design did not rely on any equipment that the team was unable to utilize due to these restrictions. Further, the team was able to access all the necessary information regarding outside architecture such as Lunar Gateway, mitigating this issue. Lastly, the MBSE team was able to successfully integrate the models into one workspace, which was a fundamental aspect of the project. Overall, the effects of risks on the project were successfully mitigated in most cases. The Network Capacity Model and hardware test bed were the only situations where there was uncertainty introduced due to failing to completely mitigate the risks involved. However, taking the effects of the risks into consideration, the impact due to failing to mitigate risks in these specific scenarios did not take away from the project’s success.

7 Project Planning

Sam Taylor, Buck Guthrie & Doug Brough

7.1 Organizational Chart

The organizational breakdown of DOTCOM is illustrated in Figure 25. DOTCOM was split into the following four sub-teams:

- System Architecture–determined all quantitative characteristics of the system and its various parts.
- Modeling–became fluent in MBSE in order to verify and illustrate the entire system.
- Network Protocol–determined most efficient data flow throughout system.
- Hardware Integration—integrated hardware tests into the project to provide physical proof-of-concept of the system.

It can be seen that some group members who are not leading a sub-team have other leadership roles that apply to the entire project. This was done to ensure that all group members had equal opportunity in research/design as well as leadership.

![Organizational Chart](image)

Figure 25: Organizational Chart

### 7.2 Work Breakdown Structure

The Work Breakdown Structure is shown below in Figure 26. All tasks were first grouped into general categories based on the type of work that had to be done, resulting in 3 primary sections (called sub-teams): system architecture, modeling, and hardware design. These are illustrated by the blue boxes. Tasks were then grouped into the relevant sub-systems they were representing, illustrated by the purple boxes. All purple boxes are critical project elements, exclusively focused on by 1-2 team members and integrated together via the sub-team lead. Last, the yellow boxes display the objective of the key tasks each sub-team must complete in order for the project to be successful.
### 7.3 Work Plan

A simplified Work Plan is shown below as a Gantt chart in Figure 27. It is organized by color in the columns on the left as follows:

- **Dark Blue**–key aspects of DOTCOM (blue boxes in Work Breakdown Structure).
- **Light Blue**–sub-teams/main work categories within each key aspect of DOTCOM (purple boxes in WBS).

As can be seen, almost all sub-teams worked independently from the start. All system architecture work was done between week 5 and week 8 in order to satisfy internal requirements and the modeling group’s requirements of entering data into their validation software. Meanwhile, hardware components were purchased within the first two weeks, where all designing, manufacturing, and testing was completed within the following 6 weeks. This project’s testing, modeling, and results took a total of 10 weeks to complete. Key due dates for deliverables to the PAB are highlighted at the top with a yellow star.
A full Work Plan for each individual sub-team is provided below. The expected time of completion for each task is illustrated by the black outline overlaying the colored bar for each task. The margin is then illustrated by the continuation of the bar without the black outline. It was desired to make the margin of each task the same length that the task was expected to take. However, due to the high amount of work per person required for this project, that was commonly not the case. Often, margins of tasks that were expected to take longer than one week were reduced to one week, and in some cases, margins were reduced to two days. The critical path is illustrated by the red arrows, and consists of all hardware network demonstration tasks followed by MBSE’s parametric modeling and activity modeling, which were done in parallel.

**Figure 27: Simplified Work Plan**

**Figure 28: System Architecture Work Plan**
### 7.4 Cost Plan

The cost breakdown for Project DOTCOM can be seen in Table 15. The main components include the Raspberry Pis, Monitors, and associated connection cables. Project DOTCOM’s overall hardware and final project costs are $645.83. It is important to note that this final cost is well below the allocated budget of $5,000. A general overview of the different components and how they relate to the overall cost can be seen in Figure 31.

<table>
<thead>
<tr>
<th>Item (Total Number Needed)</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raspberry Pi 4 (3)</td>
<td>$116.94</td>
</tr>
<tr>
<td>Monitors (2)</td>
<td>$209.98</td>
</tr>
<tr>
<td>Keyboards (2)</td>
<td>$119.98</td>
</tr>
<tr>
<td>SD Cards (3)</td>
<td>$59.97</td>
</tr>
<tr>
<td>Ethernet Cables (2)</td>
<td>$62.98</td>
</tr>
<tr>
<td>HDMI Cords (2)</td>
<td>$59.98</td>
</tr>
<tr>
<td>Power Cables (2)</td>
<td>$16.00</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>$645.83</strong></td>
</tr>
</tbody>
</table>

Table 15: Detailed Cost Breakdown
7.5 Test Plan

- Hardware Test

The Hardware test was carried out in a living room of one of the members of Team DOTCOM. This test used the 5 major elements previously described, Raspberry Pi’s, ION-DTN software, Ethernet cords, computer monitors, and finally the Wireshark software. The only test that was conducted was in fact a proof of concept rather than a true test. In this test the team sent messages through the ION-DTN network and could visually confirm the message was being sent through the output on the monitor. Since the team spent approximately 2 and 1/2 months simply trying to configure the ION-DTN software, there was not enough time to troubleshoot why Wireshark was not working. This led the team to not be able to conduct the network capacity validation test, as there were no means to analyze the amount of data being sent in the network without Wireshark.

- System Architecture Model

The tests of the system architecture consisted of three parts developed and carried out in Matlab. The tests imported satellite data from STK, then conducted numerical tests on the geometry of the system at each point in time to determine connectivity between various parts of the system. The first test assessed connection between constellation satellites and surface endpoints, the second assessed connection between constellation satellites and orbital endpoints, and the third assessed connection amongst constellation satellites. The combination of this data serves as a map of both connectivity and potential data paths supplied by our network.

- Link Budget Verification and Power Optimization

The Link Budget Model developed in MATLAB was tested against a test case link budget from a inter-satellite communications trade study on satellite antennas. After research, we determined this sources would be best to compare against the link budget model because the antenna trade study provided the most in depth input parameters out of any other source. In addition, a power optimization test was carried out using the Link Budget Model to understand the minimum transmission power required throughout the satellite system.
• Network Capacity Model

The network capacity test was designed to validate the ability of the DOTCOM network to support the required amount of network endpoints. These endpoints were simulated to be various rovers, human occupied surface bases, and imaging satellites in the lunar environment. The test found that a data relay speed of at least 35.4 mb/s between relay satellites was required to keep the network operating under capacity.

• MBSE

The MBSE is using the SysML to create the system that verifies the requirement that GA provided. The structural and behavioral modeling is used to performed the requirement tracing and verification of system functions. The requirements that GA provided for this project is applied to the requirement diagram. After the requirement diagrams is done, the structural diagrams are designed for the DOTCOM network. The behavioral diagrams is designed for the communication behavior that each blocks is performed in the Block Definition Diagram. The requirement tracing is tracing the elements that satisfies the requirements which is present in requirement diagram.

8 Lessons Learned

Sam Taylor, Tristan Liu

8.1 Systems Engineering

The systems engineering approach taken for this project was guided by the systems engineering V. Due to the nature of this project, each subsystem required flexibility and coordination in order to fully integrate and optimize the overall system. The conceptual network specifications as designated by General Atomics at the beginning of September established our initial scope, to which we created our initial Concept of Operations and High Level Design. The team further moved to define the project by rescoping the system to focus on Earth and the Moon in our Detailed Design and finalize our predicted verification and validation of each subsystem model. Our software development and hardware design stage required the team to coordinate subsystem models to ensure that they both optimized the particular topic as well as complemented the overall system design. Through the preliminary testing and system integration of our several parallel but in-depth subsystems, we verified that the integration satisfied our detailed design from the CDR. At present, we have integrated all subsystem models and used the hardware demonstration to satisfy our system requirements for the network, and in the coming weeks the team will be creating a user guide for the MBSE system integration as well as a technical paper detailing our final design decisions and rationale to deliver to our customer.

During the critical design phase, we predicted that the main systems engineering issues would involve the MBSE cross team integration, as well as the integration of our hardware testbed with our network capacity model. We addressed this risk as being rare, however the MBSE integration proved to be a difficult endeavor due to the limited literature regarding its usage for a project of our scope. However, our customers were always available for guidance on this software, and their experience with a similar software helped to give the MBSE subteam a working framework for integrating our subsystems and tracing our project requirements. A key challenge was integrating the system data budget model into the MBSE software, and the team was able to use values outputted to a text file from the MATLAB system model to trace our requirements within the MBSE environment. We also experienced scheduling constraints with the hardware DTN integration, to which the risk of failure was assessed as likely. Though the hardware subteam successfully sent messages between the two Raspberry Pi nodes, we have been unable to integrate Wireshark to analyze the network traffic over the simulated network for use in the Network Capacity Model. However, the Network Capacity Model proved to be sufficiently robust for our uses in evaluating the bottlenecks in our network transmissions, so despite our possible risk evaluation we were successful in overcoming the requirement for integration.

Ultimately, the lessons learned at the systems level came down to 3 key takeaways. At the end of the day, project DOTCOM is a research-heavy modelling project which will lay the foundation for General Atomics and its partner organizations to begin integrating a robust communications system in the Earth-Moon system. Because so much of our project was theoretical and model-based, a project of this nature requires key trades to be done on the research level, as opposed to the engineering approach taken with physical manufacturing assignments. Since we also needed to develop the architecture for the overall network, we needed assurance that our key trades for satellite architecture, interplanetary communications capabilities, compatibility and
connectability, and transmission routing procedures, such that they were not only optimal for their respective system focus, but also complementary to the overall infrastructure of the network. Additionally, limiting our scope for the project helped the team to narrow our focus and fully satisfy the needs for the Earth-Moon system, and the reduction in scope was crucial for both time management but also cross-team communication and assurance of compatible integration.

8.2 Project Management

From a project management perspective, there were three key lessons learned as the year progressed. The first lesson can be summarized as: the bigger the project, the longer planning takes. As most projects throughout the undergraduate aerospace program are under a month or two, diving into a year long project came with unfamiliar tasks and timelines. Planning for a project in the past was typically done primarily by the professor teaching the class, with students contributions typically taking less than ten minutes and minimal thought. However, it was very unexpected just how much the planning section increased in both time and effort relative to past experience. The first issue was determining exactly what tasks had to be completed to result in the high level systems that needed to be developed. Due to lack of experience in many of the sub-systems required within the project, lots of initial research had to be completed before the project could even start. The second issue was developing a timeline for each individual task that would result in the project being completed on time. This was difficult for a few reasons. The first was tied back to our initial issue, where not only were we unfamiliar with the steps it would take to complete each section of the project, but with the aforementioned lack of experience, it was even more unknown how long each task would take. The second difficulty was given our estimations in the amount of work it would take to complete each task determining how many people had to work on each task in order to get everything done by the time the project needed to be completed. Finally, the importance of completing everything previously mentioned in a detailed and accurate manner was highlighted by our third and final issue making a Gantt Chart. The chart provided a great outline for the entire project to ensure we knew what we had to do and make sure it was done on time. However, it required the tasks and timelines to be accurate from the start, or else it would be rendered useless. Making it proved to take much longer than anyone anticipated, which caused both panic and inconvenience at the time of its creation. We recommend future seniors give themselves plenty of time to create their Gantt chart and work hard in their initial research to develop the most accurate initial plan they can, as it will pay off tremendously throughout the project. The second lesson learned was to use given resources as much as possible. In the early stages of the project, many team members were faced with tasks they were relatively unfamiliar with. Additionally, it was proving to be quite difficult to find the information necessary to complete the tasks. This resulted in our group becoming behind schedule, as we continued to scour the web for what we needed to find. Eventually, we reached out to our customer, who quickly introduced us to experts within their company who could help us. They were incredibly insightful, which resulted not only in us completing the tasks we were stuck on, but doing so in a quick manner that resulted in us catching up and getting back on schedule. We encourage future seniors not to shy away from reaching out for help from their given resources in the future any time they get stuck. It is not a sign of incompetence as they may believe, but in fact a sign of maturity, as everyone eventually learns that it is always best to work with others. The third and final lesson learned was that there is no such thing as over-communication. Over the course of our first few presentations to the PAB, we were receiving feedback that none of the professors were completely understanding what our project was about, which made it near impossible for them to understand any of the additional details we were presenting. First, we worked with our advisor to try and restructure our presentation to make it easier to understand, but the results stayed the same. It was then recommended to us that we reach out to each individual member to the PAB to discuss our project directly and answer all the questions they had that were preventing them from understanding our project. The impact was game changing, as PAB members expressed both upon conclusion of our 1-on-1 meetings and throughout the remaining presentations how much more they understood the project, which allowed them to focus more on the details of what we were working on. We encourage future seniors to try and meet with each PAB member after their first few presentations if there seems to be any confusion regarding their project, as it has a very positive impact on both the team and the PAB members that lasts throughout the rest of the year.
## 9 Individual Report Contributions

<table>
<thead>
<tr>
<th>Team Member</th>
<th>Completed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tristan Liu</td>
<td>Project Purpose, Final Design (Hardware), Manufacturing (Hardware), Lessons Learned (Systems Engineering)</td>
</tr>
<tr>
<td>Buck Guthrie</td>
<td>Final Design (Spectrum Allocation and Link Budgets), Link Budget Software, Cost Plan</td>
</tr>
<tr>
<td>Jennifer Gurtler</td>
<td>Verification &amp; Validation (MBSE/SysML), Test Plan(MBSE)</td>
</tr>
<tr>
<td>Sebastian Damm</td>
<td>Project Purpose and Functional Requirements, Concept of Operations, Customer Deliverables</td>
</tr>
<tr>
<td>Hunter Rohlman</td>
<td>Final Design (Network Protocol), Network Capacity Model Software, NCP Results Verification, Risk Assessment</td>
</tr>
<tr>
<td>Douglas Brough</td>
<td>Final Design (Hardware), Manufacturing (Hardware), Project Planning (Test Plan)</td>
</tr>
<tr>
<td>Caelan Maitland</td>
<td>Verification &amp; Validation (MBSE/SysML)</td>
</tr>
<tr>
<td>Conner Lewis</td>
<td>CONOPS, Final Design (Constellation Architecture), Manufacturing (Architecture Model Software), Verification and Validation (Network Architecture Model)</td>
</tr>
<tr>
<td>Sam Taylor</td>
<td>Project Purpose, Final Design (Overview), Project Planning (OC, WBS, WP), Lessons Learned (Project Management)</td>
</tr>
</tbody>
</table>

Table 16: Individual Report Contributions
References


10 Appendix

10.1 Network Capacity Model MATLAB Code

```matlab
%% Network Capacity Model
% Created by: Hunter Rohlman and Forrest Jordan
% Last Modified: 04/29/2021

% Model Description:
% The Network Capacity Model was developed to analyze the network traffic
% of a DTN network and simulate the flow of data to find an optimal data
% rate for relay satellites managing the network. The process boils down
% to creating two matrices. One of these is a data rate matrix while the
% other is a connection window matrix. Together, these matrices yield a
% data transfer at any node of the network. The model will utilize the
% designation of over or under capacity for each satellite data rate until
% the minimum (or optimal) data rate is found. This data rate is the main
% output of this model. If you’re curious about the minimum data rate, it
% is stored as "min_data_rate" and is listed in bits/s (divide by 10^6 to
% convert to Mbits/s). Keep in mind that bits and bytes are not the same
% and that there are 8 bits in a byte.

%% Initialization
% Values such as orbital period and number of surface vehicles are input
% here. These values will be referenced throughout the runtime of this
% code.

clear
close all
clc
tic

do_plots = 1; % create plots of node capacities
capacity_plot = 1; % Plot the capacitance optimization (not necessary as
% output in command window will show the result of this
save_plots = 0; % Set this to 1 to save plots as png’s, 0 to not save

t_step = 1; % seconds
t_end = 86400; % seconds
period = 55376; % seconds
num_sats = 6; % number of constellation sats in scenario, used for
% connection functions
num_sv = 15; % number of surface vehicles
under_capacity = []; % used for optimization plot
nodes = 55;
num_routes = 2;

sat_datarate = linspace(28,38,10)*10^6; % for varying data rates in our test
num_tests = length(sat_datarate); % runs a test for each input data rate in
% the sat_datarate vector
dtcount = 1; % for finding min data rate

%% Data Rates (A) matrix
% The following code creates what is referred to as the "A" matrix. This
% matrix contains the data rates between all nodes represented in the
% network as well as internal data rates for generating data at our end
% points (i.e. the surface or orbiting vehicles). This matrix will be
% combined with the connection windows matrix to simulate the flow of data
% throughout the network. Additionally, this matrix is structured such
% that the path of data from the Earth to the Moon is in the first page of
% this matrix while the path from the Moon to the Earth is in the second
```

Page 53
% page.  (The difference between these two paths is that certain nodes will
% not be generating data depending on the direction, such as the surface
% vehicles on path 1.)

for tests = 1:num_tests

A = zeros(nodes,nodes,num_routes);  % 1 refers to the path from earth to moon, 2 is
% opposite path

% internal generated data
%% may need to divide these data rates by 2 since they will be generated
%% on both paths
total_earth_outbound = 2*10^-3*5 + 10*10^-3*6 + 100*10^-3*2 + 1.5 * 10^-6 * 2 + 20*10^-3 * 20 +
1.5*10^-6*10;
A(1,1,1) = total_earth_outbound/3;
A(2,2,1) = total_earth_outbound/3;
A(3,3,1) = total_earth_outbound/3;
A(4,4,2) = 0;

% data rates between nodes
A(1:3,4,:) = 500*10^6;  % gs on earth to gs on moon
A(4,1:3,:) = 500*10^6;  % gs on moon to gs on earth

A(4,20:25,:) = sat_datarate(tests);  % from gs to orbiting sats
A(20:25,4,:) = sat_datarate(tests);  % from orbiting sats to gs

% To stay under capacity this value has to be locked, if all satellite
% data rates increase the system is never under capacity as they’re
% just spam sending information to each other. It can be lowered as
% needed though.
A(20:25,20:25,:) = sat_datarate(tests)/5;  % relay sat to relay sat

% define surface vehicles here (nodes 5—19)
for i = 5:9
A(i,i,2) = 20*10^-3;  % lunar rover
A(i,20:25,2) = 20*10^-3;  % lunar rover
A(20:25,i,1) = sat_datarate(tests);  % lunar rover
end

for i = 10:15
A(i,i,2) = 10*10^-3;  % voice comms
A(i,20:25,2) = 10*10^-3;  % voice comms
A(20:25,i,1) = sat_datarate(tests);  % voice comms
end

for i = 16:17
A(i,i,2) = 1.5*10^6;  % high data rate rover
A(i,20:25,2) = 1.5*10^6;  % high data rate rover
A(20:25,i,1) = sat_datarate(tests);  % high data rate rover
end

for i = 18:19
A(i,i,2) = 1.5*10^6;  % video comm
A(i,20:25,2) = 1.5*10^6;  % video comm
A(20:25,i,1) = sat_datarate(tests);  % video comm
end

for i = 20:25
A(i,i,2) = 20*10^-3;  % orbiter
A(i,20:25,2) = 20*10^-3;
A(20:25,i,1) = sat_datarate(tests);
end

for i = 26:35
A(i,i,2) = 1.5*10^-6;  % high data rate orbiter
A(i,20:25,2) = 1.5*10^-6;
end

for i = 36:40
A(i,i,2) = 1.5*10^-6;  % high data rate orbiter
A(i,20:25,2) = 1.5*10^-6;
\begin{verbatim}
A(20:25, i, 1) = sat_datarate(tests);
end
for i = 41:50
    A(i, i, 2) = 20*10^3;  % orbiter
    A(i,20:25, 2) = 20*10^3;
    A(20:25, i, 1) = sat_datarate(tests);
end
for i = 51:55
    A(i, i, 2) = 1.5*10^6;  % high data rate orbiter
    A(i,20:25, 2) = 1.5*10^6;
    A(20:25, i, 1) = sat_datarate(tests);
end

%%% Connection Matrix (C) setup

% The "C" matrix, or the connection matrix, is a matrix full of either 1's
% or 0's. A 1 refers to a node being connected to the other node while a 0
% shows a disconnect. These values are autogenerated for the ground
% stations on each body while the connection windows for any satellites or
% surface vehicles are pulled directly from a text file that is generated
% in our STK simulation of an identical system. These values will be used
% in determining the flow of data through the network.

ind = t_end/t_step;  % max index for connection functions
C = zeros(nodes, nodes, num_routes);

%continuous connections and endpoint generation

for n = 1:nodes
    C(n, n, :) = 1;
end

% Orbiting Satellite and SV Connections

sat_1 = read_connection('Sat1Con.txt');
sat_2 = read_connection('Sat2Con.txt');
sat_3 = read_connection('Sat3Con.txt');
sat_4 = read_connection('Sat4Con.txt');
sat_5 = read_connection('Sat5Con.txt');
sat_6 = read_connection('Sat6Con.txt');

orb_1 = read_connection2('Sat1Orb.txt');
orb_2 = read_connection2('Sat2Orb.txt');
orb_3 = read_connection2('Sat3Orb.txt');
orb_4 = read_connection2('Sat4Orb.txt');
orb_5 = read_connection2('Sat5Orb.txt');
orb_6 = read_connection2('Sat6Orb.txt');

% Earth -> Moon
% We want to disconnect certain nodes depending on the direction of data.
% In this case, the moon ground station should not be sending data to the
% earth ground station since data is traversing from the earth to the moon,
% not from the moon to the earth. Similarly, the ground vehicles and
% orbiting satellites should not be sending data towards the earth.
C(4,1:3,1) = 0;  % moon gs to earth gs on route 1
C(20:25,4,1) = 0;  % orbiting sats to moon gs on route 1
C(5:19,20:25,1) = 0;  % ground vehicles to orbiting sats on route 1
C(26:55,20:25,1) = 0;

% Moon -> Earth
% The opposite is true here, data should flow freely from the moon to the
% earth while no data should flow from the earth to the moon.
\end{verbatim}
for n = 1:3
    C(n,n,2) = 0; % internal generations on earth gs
end

C(1:3,4,2) = 0; % earth gs to moon gs not connected
C(4:20:25,2) = 0; % moon gs not connected to orbiting sats here
C(20:25,5:19,2) = 0; % orbiting sats not connected to ground vehicles here
C(20:25,26:55,2) = 0; % orbiting sats not connected to ground vehicles here

%% Connection Functions
% The connection functions are a way to create connection windows for the
% ground stations on each body as well as the connections between the moon
% ground station and orbiting satellites. This is accomplished through the
% sat_connection_generate function as well as information regarding the
% period, time step, and delay. For more information regarding the
% function's notation, view the sat_connection_generate.m file.

delay_amount = period/num_sv;
delay_amount2 = period/num_sats;

for n = 1:3
    gstogs(n,:) = sat_connection_generate(86400,28800,(n-1)*28800,t_end,t_step); % moon gs is connected to earth gs
end

% Connection functions for each orbiting sat to moon gs
for i = 1:6
    sat2gs(i,:) = sat_connection_generate(period,period/6,(i-1)*delay_amount2,t_end,t_step);
end

%% Determining data stored
% This is the most complex process in this program. However, it can be
% simplified down to the fundamental process of multiplying the A and C
% matrices together to determine the data rate at each node at each time
% step. What this while loop does is continuously iterate through each
% time step in our orbital period and monitor the data flow at each step.
% At the end of the process, the changes in the internal satellite storages
% are summed and compiled into a matrix called msave. msave1 refers to
% path 1 storage, msave2 refers to path 2 storage, and msave3 refers to
% total storage (a summation of both).

m = zeros(nodes,nodes,num_routes);
dm_save = zeros(1,nodes,num_routes);
index = 2;
t_save = 0;
t = t_step;

counttest = 1;
while t < t_end
    A_adj = A;
    idx = index-1;

    % Setting up connection windows between the satellites and surface
    % vehicles nodes 5–19 connecting to nodes 20–25
    % orbiting sat connections
    row = 2;
    while sat_1(row,1) <= t
        row = row+1;
    end

    % Sat 20
    for x = 1:6
        if x ~= 1

Page 56
C(20, (19+x), :) = sat_1((row-1), (1+x));
end
end
% this for loop connects the sats to surface vehicles (for paths 1 and
% 2 respectively)
for x = 1:15
    C(20, (4+x), 1) = sat_1((row-1), (7+x));
    C((4+x), 20, 2) = sat_1((row-1), (7+x));
end
for x = 1:30
    C(20, (25+x), 1) = orb_1((row-1), (7+x));
    C((25+x), 20, 2) = orb_1((row-1), (7+x));
end
% Sat 21
for x = 1:6
    if x ~= 2
        C(21, (19+x), :) = sat_2((row-1), (1+x));
    end
end
for x = 1:15
    C(21, (4+x), 1) = sat_2((row-1), (7+x));
    C((4+x), 21, 2) = sat_2((row-1), (7+x));
end
for x = 1:30
    C(21, (25+x), 1) = orb_2((row-1), (7+x));
    C((25+x), 21, 2) = orb_2((row-1), (7+x));
end
% Sat 22
for x = 1:6
    if x ~= 3
        C(22, (19+x), :) = sat_3((row-1), (1+x));
    end
end
for x = 1:15
    C(22, (4+x), 1) = sat_3((row-1), (7+x));
    C((4+x), 22, 2) = sat_3((row-1), (7+x));
end
for x = 1:30
    C(22, (25+x), 1) = orb_3((row-1), (7+x));
    C((25+x), 22, 2) = orb_3((row-1), (7+x));
end
% Sat 23
for x = 1:6
    if x ~= 4
        C(23, (19+x), :) = sat_4((row-1), (1+x));
    end
end
for x = 1:15
    C(23, (4+x), 1) = sat_4((row-1), (7+x));
    C((4+x), 23, 2) = sat_4((row-1), (7+x));
end
for x = 1:30
    C(23, (25+x), 1) = orb_4((row-1), (7+x));
    C((25+x), 23, 2) = orb_4((row-1), (7+x));
end
% Sat 24
for x = 1:6
    if x ~= 5
        C(24, (19+x), :) = sat_5((row-1), (1+x));
    end
end
for x = 1:15
    C(24, (4+x), 1) = sat_5((row-1), (7+x));
    C((4+x), 24, 2) = sat_5((row-1), (7+x));
end
for x = 1:30
    C(24,(25+x) ,1) = orb_5((row−1),(7+x));
    C((25+x) ,24 ,2) = orb_5((row−1),(7+x));
end

% Sat 25
for x = 1:6
    if x ~= 6
        C(25,(19+x) ,:) = sat_6((row−1),(1+x));
    end
end
for x = 1:15
    C(4,(4+x) ,1) = sat_6((row−1),(7+x));
    C((4+x) ,4 ,2) = sat_6((row−1),(7+x));
end
for x = 1:30
    C(25,(25+x) ,1) = orb_6((row−1),(7+x));
    C((25+x) ,25 ,2) = orb_6((row−1),(7+x));
end

% Setting up the connections from the earth gs to moon gs
for i = 1:3
    C(4,i ,1) = 0;
    C(i ,4 ,2) = 0;
    C(4,i ,2) = gstogs(i , idx);
    C(i ,4 ,1) = gstogs(i , idx);
end

% Setting up the connections from the orbiting sats to moon gs and visa versa
for i = 1:6
    C(i+19,4,2) = sat2gs(i , idx);
    C(4,i+19,1) = sat2gs(i , idx);
    if C(i+19,4,2) == 1 %only 1 satellite connected to moon gs at a time
        C(i+19,20:25,2) = 0;
    end
end
C_out = C;

% This disconnects the nodes from themselves across all routes
for n = 1:nodes
    C_out(n,n,:) = 0;
end

% check data rate doesn’t exceed data in node
% adjust data rate if necessary
for i = 1:num_routes % run through each route possible
    rate_out_max = A(:,i)‘*C_out(:,i)’;
    for a = 1:nodes
        if m(a,a,i)<(rate_out_max(a,a)*t_step)
            A_req = m(a,a,i)/t_step;
            for b = 1:nodes
                pct = A(a,b,i)*C_out(a,b,i)/rate_out_max(a,a);
                if a ~= b
                    A_adj(a,b,i) = A_req*pct;
                end
            end
        end
    end
end

% Test Metrics
% for i = 1:2
%    node_in(:,i) = C(:,i)’ * A_adj(:,i);
%    node_out(:,i) = A_adj(:,i) * C_out(:,i)’;
% end
% for a = 1:nodes
% in_save(a, index, tests) = node_in(a,a,1);
% out_save(a, index, tests) = node_out(a,a,1);
% in_save2(a, index, tests) = node_in(a,a,2);
% out_save2(a, index, tests) = node_out(a,a,2);
% end

for i = 1:2
    dm(:,:,: , i ) = C(:,:,: , i )' * A_adj(:,:,: , i ) - A_adj(:,:,: , i ) * C_out(:,:,: , i );
end
m = m + dm*t_step;

for a = 1:nodes
    % dm_save1(a, index, tests) = dm(a,a,1);
    % dm_save2(a, index, tests) = dm(a,a,2);
    % m_save1(a, index) = m(a,a,1);
    % m_save22(a, index) = m(a,a,2);
end

t_save(index) = t;
index = index+1;
t = t+t_step;
end
t_save = t_save/60;  %convert time to min
m_save1(:, :, tests) = m_save11;
m_save2(:, :, tests) = m_save22;
m_save3 = m_save1+m_save2;

% We are looping through each set of total data for the sat
% nodes. We are using the indices of the max values in for each nodes data
% to determine if the data returns to ~0 after the max. If this happens,
% count is increased by 1 and the if statement for that node will never run
% again to ensure there isn’t double counting. Then if the count is 5 (or
% 6 once I figure out why node 20 feels so special), the under_capacity
% value is set to 1 to signify an under capacity network.

count = 0;
ifzero0 = 0;
ifzero = 0;  % These are set to 1 to terminate the if statements after one
            % case of returning to 0 is determined
ifzero2 = 0;
ifzero3 = 0;
ifzero4 = 0;
ifzero5 = 0;

[-, max20] = max(m_save3(20, :, tests));  % Finds index of max value
[-, max21] = max(m_save3(21, :, tests));  % Finds index of max value
[-, max22] = max(m_save3(22, :, tests));
[-, max23] = max(m_save3(23, :, tests));
[-, max24] = max(m_save3(24, :, tests));
[-, max25] = max(m_save3(25, :, tests));

for i = 10:length(m_save3)−10
    if i > max20
        if ifzero0 == 0
            if m_save3(20,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
    if i > max21
        if ifzero == 0
            if m_save3(21,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
    if i > max22
        if ifzero2 == 0
            if m_save3(22,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
    if i > max23
        if ifzero3 == 0
            if m_save3(23,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
    if i > max24
        if ifzero4 == 0
            if m_save3(24,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
    if i > max25
        if ifzero5 == 0
            if m_save3(25,i,tests) <= 2e9
                count = count + 1;
            end
        end
    end
end
ifzero0 = 1;
end
end
end

if i > max21
    if ifzero == 0
        if m_save3(21,i,tests) <= 2e9
            count = count + 1;
            ifzero = 1;
        end
    end
end

if i > max22
    if ifzero2 == 0
        if m_save3(22,i,tests) <= 2e9
            count = count + 1;
            ifzero2 = 1;
        end
    end
end

if i > max23
    if ifzero3 == 0
        if m_save3(23,i,tests) <= 2e9
            count = count + 1;
            ifzero3 = 1;
        end
    end
end

if i > max24
    if ifzero4 == 0
        if m_save3(24,i,tests) <= 2e9
            count = count + 1;
            ifzero4 = 1;
        end
    end
end

if i > max25
    if ifzero5 == 0
        if m_save3(25,i,tests) <= 2e9
            count = count + 1;
            ifzero5 = 1;
        end
    end
end

if count == 6
    under_capacity(tests) = 1;
    datarate_undercap(dtcount) = sat_datarate(tests);
    dtcount = dtcount + 1;
else
    under_capacity(tests) = 0;
end

end  %% end of first test loop for data generation

if count == 6
    min_data_rate = min(datarate_undercap);
    fprintf(‘Minimum required data rate is %d Mbps\n’,min_data_rate/(10^6));
end

% Minimum for current test (55 nodes) was 3.542857e+01 Mbps
%% Plot Generation

if do_plots == 1
  set(gcf,'Units','Normalized','OuterPosition',[0,0.04,1,0.96]);
  for i = 1:25
    subplot(5,5,i)
    for tests = 1:num_tests
      plot(t_save,m_save3(i,:,tests))
    end
    title(['Node ',num2str(i),' Data Stored'])
    xlabel('Time [min]')
    ylabel('Bits in Storage')
  end
  figure
  hold on
  set(gcf,'Units','Normalized','OuterPosition',[0,0.04,1,0.96]);
  for i = 26:55
    subplot(6,5,i-25)
    for tests = 1:num_tests
      plot(t_save,m_save3(i,:,tests))
    end
    title(['Node ',num2str(i),' Data Stored'])
    xlabel('Time [min]')
    ylabel('Bits in Storage')
  end
  sgtitle('Totals')
end

if capacity_plot == 1
  count = 0;
  count1 = 0;
  p1 = 0;
  p2 = 0;
  figure
  set(gcf,'Units','Normalized','OuterPosition',[0,0.04,1,0.96]);
  for tests = 1:num_tests
    under_capacity(tests) == 1
      if count == 0
        p1 = plot(sat_datarate(tests)/10^6,under_capacity(tests),'b*','MarkerSize',25);
        count = 1;
        hold on
      else
        plot(sat_datarate(tests)/10^6,under_capacity(tests),'b*','MarkerSize',25)
      end
      if count1 == 0
        p2 = plot(sat_datarate(tests)/10^6,under_capacity(tests),'ro','MarkerSize',25);
        count1 = 1;
        hold on
      else
        plot(sat_datarate(tests)/10^6,under_capacity(tests),'ro','MarkerSize',25)
      end
    end
end
10.2 Constellation Design Figures

Figure 32: Image of the constellation design around the moon
Figure 33: Image of the groundtracks of the baseline constellation design as well as the distribution of the satellites

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(\degree)$</th>
<th>$\nu(\degree)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>120</td>
</tr>
</tbody>
</table>

Table 17: 6/6/4 satellite orbit characteristics

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(\degree)$</th>
<th>$\nu(\degree)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>120</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>180</td>
</tr>
<tr>
<td>8</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>9</td>
<td>120</td>
<td>300</td>
</tr>
<tr>
<td>10</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>60</td>
</tr>
<tr>
<td>12</td>
<td>300</td>
<td>300</td>
</tr>
</tbody>
</table>

Table 18: 12/6/4 satellite orbit characteristics
<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(°)$</th>
<th>$\nu(°)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>300</td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>4</td>
<td>90</td>
<td>180</td>
</tr>
<tr>
<td>5</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>6</td>
<td>150</td>
<td>60</td>
</tr>
<tr>
<td>7</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>210</td>
<td>300</td>
</tr>
<tr>
<td>9</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>10</td>
<td>270</td>
<td>180</td>
</tr>
<tr>
<td>11</td>
<td>300</td>
<td>120</td>
</tr>
<tr>
<td>12</td>
<td>330</td>
<td>60</td>
</tr>
</tbody>
</table>

Table 19: 12/12/10 satellite orbit characteristics

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(°)$</th>
<th>$\nu(°)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>96</td>
</tr>
<tr>
<td>3</td>
<td>48</td>
<td>192</td>
</tr>
<tr>
<td>4</td>
<td>72</td>
<td>288</td>
</tr>
<tr>
<td>5</td>
<td>96</td>
<td>24</td>
</tr>
<tr>
<td>6</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>7</td>
<td>144</td>
<td>216</td>
</tr>
<tr>
<td>8</td>
<td>168</td>
<td>312</td>
</tr>
<tr>
<td>9</td>
<td>192</td>
<td>48</td>
</tr>
<tr>
<td>10</td>
<td>216</td>
<td>144</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>12</td>
<td>264</td>
<td>336</td>
</tr>
<tr>
<td>13</td>
<td>288</td>
<td>72</td>
</tr>
<tr>
<td>14</td>
<td>312</td>
<td>168</td>
</tr>
<tr>
<td>15</td>
<td>336</td>
<td>264</td>
</tr>
</tbody>
</table>

Table 20: 15/15/6 satellite orbit characteristics
<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(\degree)$</th>
<th>$\nu(\degree)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>48</td>
</tr>
<tr>
<td>3</td>
<td>48</td>
<td>96</td>
</tr>
<tr>
<td>4</td>
<td>72</td>
<td>144</td>
</tr>
<tr>
<td>5</td>
<td>96</td>
<td>192</td>
</tr>
<tr>
<td>6</td>
<td>120</td>
<td>240</td>
</tr>
<tr>
<td>7</td>
<td>144</td>
<td>288</td>
</tr>
<tr>
<td>8</td>
<td>168</td>
<td>336</td>
</tr>
<tr>
<td>9</td>
<td>192</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>216</td>
<td>72</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>120</td>
</tr>
<tr>
<td>12</td>
<td>264</td>
<td>168</td>
</tr>
<tr>
<td>13</td>
<td>288</td>
<td>216</td>
</tr>
<tr>
<td>14</td>
<td>312</td>
<td>264</td>
</tr>
<tr>
<td>15</td>
<td>336</td>
<td>312</td>
</tr>
</tbody>
</table>

Table 21: 15/15/2 satellite orbit characteristics

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega(\degree)$</th>
<th>$\nu(\degree)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>120</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>120</td>
</tr>
<tr>
<td>8</td>
<td>60</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>120</td>
<td>240</td>
</tr>
<tr>
<td>10</td>
<td>180</td>
<td>120</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>300</td>
<td>240</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>240</td>
</tr>
<tr>
<td>14</td>
<td>60</td>
<td>120</td>
</tr>
<tr>
<td>15</td>
<td>120</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>180</td>
<td>240</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
<td>120</td>
</tr>
<tr>
<td>18</td>
<td>300</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 22: 18/6/4 satellite orbit characteristics
Table 23: 24/6/4 satellite orbit characteristics

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\Omega^\circ$</th>
<th>$\nu^\circ$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>240</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>240</td>
<td>240</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>120</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>90</td>
</tr>
<tr>
<td>8</td>
<td>60</td>
<td>330</td>
</tr>
<tr>
<td>9</td>
<td>120</td>
<td>210</td>
</tr>
<tr>
<td>10</td>
<td>180</td>
<td>90</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>330</td>
</tr>
<tr>
<td>12</td>
<td>300</td>
<td>210</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>180</td>
</tr>
<tr>
<td>14</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>15</td>
<td>120</td>
<td>300</td>
</tr>
<tr>
<td>16</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
<td>60</td>
</tr>
<tr>
<td>18</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>270</td>
</tr>
<tr>
<td>20</td>
<td>60</td>
<td>150</td>
</tr>
<tr>
<td>21</td>
<td>120</td>
<td>30</td>
</tr>
<tr>
<td>22</td>
<td>180</td>
<td>270</td>
</tr>
<tr>
<td>23</td>
<td>240</td>
<td>150</td>
</tr>
<tr>
<td>24</td>
<td>300</td>
<td>30</td>
</tr>
</tbody>
</table>

%This script calculates the surface coverage, including coverage maps and coverage multiple distribution, for the 6/6/4 constellation over one orbital period

%Inputs: Satellite ephemeris (STK)

%Outputs: Surface coverage(coverageGeo), coverage multiple percentages (a variables)

%Written by: Conner Lewis

clear all
close all
clc

%importing satellite ephemeris data from STK
Sat(:,1) = importdata(’Satellite1.e’, ’’,28).data;
Sat(:,2) = importdata(’Satellite2.e’, ’’,28).data;
Sat(:,3) = importdata(’Satellite3.e’, ’’,28).data;
Sat(:,4) = importdata(’Satellite4.e’, ’’,28).data;
Sat(:,5) = importdata(’Satellite5.e’, ’’,28).data;
Sat(:,6) = importdata(’Satellite6.e’, ’’,28).data;

surfaceGeoX = linspace(-180,180,360) + 180; %converting from geodetic to spherical
surfaceGeoY = linspace(-90,90,180) - 90;
cartX = 1737000*sind(surfaceGeoY).*cosd(surfaceGeoX); %converting from spherical to cartesian
cartY = 1737000*sind(surfaceGeoY).*sind(surfaceGeoX);
cartZ = 1737000*cosd(surfaceGeoY);

coverageGeo = zeros(length(surfaceGeoX),length(surfaceGeoY),length(Sat)); %initializing large matrices
coveragePercent = zeros(length(Sat),1);
sums = zeros(length(Sat),1);

for t = 1:length(Sat)
    for i = 1:length(surfaceGeoX)
        for j = 1:length(surfaceGeoY)
            for k = 1:length(Sat(:,1,:))
                d = sqrt((Sat(t,2,k) - cartX(j,i))^2 + (Sat(t,3,k) - cartY(j,i))^2 + (Sat(t,4,k) - cartZ(j))^2); %pythagorean theorem
                if d < 6885000
                    coverageGeo(i,j,t) = coverageGeo(i,j,t) + 1; %determining coverage
                end
            end
        end
    end

    cover = find(coverageGeo(:,:,t) == 0); %determining if a point has any coverage
    coveragePercent(t) = 100*(1 - (length(cover)/(length(surfaceGeoX)*length(surfaceGeoY)))); %determining coverage percent per time step
    sums(t) = sum(coverageGeo(:,:,t), 'all'); %determining the weighted coverage of each point in time
end

%displaying best and worst coverage maps
figure(1)
    h = heatmap(coverageGeo(:, :, find(sums == min(sums)))');
    h.GridVisible = 'off';
    xlabel('Longitude (degrees)');
    ylabel('Latitude (degrees)');

figure(2)
    h = heatmap(coverageGeo(:, :, find(sums == max(sums)))');
    h.GridVisible = 'off';
    xlabel('Longitude (degrees)');
    ylabel('Latitude (degrees)');

%determining coverage multiple percentages
a = numel(coverageGeo);
a0 = length(coverageGeo(coverageGeo == 0))*100/a;
a1 = length(coverageGeo(coverageGeo == 1))*100/a;
a2 = length(coverageGeo(coverageGeo == 2))*100/a;
a3 = length(coverageGeo(coverageGeo == 3))*100/a;
a4 = length(coverageGeo(coverageGeo == 4))*100/a;
a5 = length(coverageGeo(coverageGeo == 5))*100/a;
a6 = length(coverageGeo(coverageGeo == 6))*100/a;

%This script calculates the orbital coverage in the form of the coverage
%multiple distribution, for the 6/6/4 constellation over one orbital
%period
%Inputs: Satellite ephemeris (STK)
%Outputs: Surface coverage(coverageGeo), coverage multiple percentages(a variables)
%Written by: Conner Lewis

clear all
close all
clc

%importing satellite ephemeris data from STK
Sat(:,1,:) = importdata('Satellite1.e', ',',28).data;
Sat(:,2,:) = importdata('Satellite2.e', ',',28).data;
Sat(:,3,:) = importdata('Satellite3.e', ',',28).data;
Sat(:,4,:) = importdata('Satellite4.e', ',',28).data;
Sat(:,5,:) = importdata('Satellite5.e', ',',28).data;
Sat(:,6,:) = importdata('Satellite6.e', ',',28).data;

orbGeoTheta = linspace(-180,180,180) + 180; %converting from geodetic to spherical
orbGeoPhi = linspace(-90,90,90) - 90;
orbGeoR = linspace(1887000,1148700,50);
cartA = sind(orbGeoPhi') * cosd(orbGeoTheta); %converting from spherical to cartesian
cartB = sind(orbGeoPhi') * sind(orbGeoTheta);
cartX = zeros(length(orbGeoPhi),length(orbGeoTheta),length(orbGeoR));
cartY = zeros(length(orbGeoPhi),length(orbGeoTheta),length(orbGeoR));
for i = 1:length(orbGeoR)
cartX(:,:,i) = cartA*orbGeoR(i);
cartY(:,:,i) = cartB*orbGeoR(i);
end
cartZ = cosd(orbGeoPhi') * orbGeoR;
coverageGeo = zeros(length(orbGeoTheta),length(orbGeoPhi),length(orbGeoR),length(Sat),'int8'); %initializing (very) large matrices
coveragePercent = zeros(length(Sat),1);
sums = zeros(length(Sat),1);
for t = 1:length(Sat)
for i = 1:length(orbGeoTheta)
for j = 1:length(orbGeoPhi)
for k = 1:length(orbGeoR)
for l = 1:length(Sat(1,1,:))
d1 = sqrt((Sat(t,2,l) - cartX(j,i,k))^2 + (Sat(t,3,l) - cartY(j,i,k))^2 + (Sat(t,4,l) - cartZ(j,k))^2); %pythagorean theorem
d2 = sqrt((Sat(t,2,l) + cartX(j,i,k))^2 + (Sat(t,3,l) + cartY(j,i,k))^2 + (Sat(t,4,l) + cartZ(j,k))^2); %shadow satellite
if d1 > d2 %determining if the point in question is on the appropriate
    continue;
elseif d1 < 11024000
    coverageGeo(i,j,k,t) = coverageGeo(i,j,k,t) + 1; %determining coverage
end
end
end
end
cover = find(coverageGeo(:, :, :, t) == 0); %determining if a point has any coverage
coveragePercent(t) = 100*(1 - (length(cov) / (length(orbGeoTheta) * length(orbGeoPhi) * length(orbGeoR)))); %determining coverage percent per time step
sums(t) = sum(coverageGeo(:, :, :, t), 'all'); %determining the weighted coverage of each point in time
end
%determining coverage multiple percentages
a = numel(coverageGeo);
a0 = length(find(coverageGeo == 0)) * 100 / a;
a1 = length(find(coverageGeo == 1)) * 100 / a;
a2 = length(find(coverageGeo == 2)) * 100 / a;
a3 = length(find(coverageGeo == 3)) * 100 / a;
a4 = length(find(coverageGeo == 4)) * 100 / a;
a5 = length(find(coverageGeo == 5)) * 100 / a;
a6 = length(find(coverageGeo == 6)) * 100 / a;

%This script determines the connection windows between constellation
%satellites for the 6/6/4 constellation over one orbital period
%Inputs: Satellite ephemeris (STK)
%Outputs: Text files for each satellite describing whether or not they are
%able to connect to each other satellite at each point in time
%Written by: Conner Lewis
clear all
clc
importing satellite ephemeris data from STK
Sat(:, :, 1) = importdata('Satellite1.e',',',27).data;
Sat(:, :, 2) = importdata('Satellite2.e',',',27).data;
Sat(:, :, 3) = importdata('Satellite3.e',',',27).data;
Sat(:, :, 4) = importdata(’Satellite4.e’, ’ ’, 27).data;
Sat(:, :, 5) = importdata(’Satellite5.e’, ’ ’, 27).data;
Sat(:, :, 6) = importdata(’Satellite6.e’, ’ ’, 27).data;
t = Sat(:, 1, 1); % times are in seconds
windows = zeros(length(Sat), 7, 6); % initializing the windows matrix
count = 1;
for i = 1:6
    windows(:, 1, i) = t;
    for j = 1:6
        for k = 1:length(Sat)
            if i == j
                windows(k, j+1, i) = NaN; % condition for satellite connecting to itself
            else % connecting to other satellites
                dist = sqrt((Sat(k,2, i) - Sat(k,2, j))^2 + (Sat(k,3, i) - Sat(k,3, j))^2 + (Sat(k,4, i) - Sat(k,4, j))^2); % pythagorean theorem
                if dist < 14069000 % maximum connection distance from orbit geometry
                    windows(k, j+1, i) = 1; % can connect
                else
                    windows(k, j+1, i) = 0; % cannot connect
                end
            end
        end
    end
% outputting connection windows
fileID1 = fopen(’Sat1Con.txt’, ’w’);
fprintf(fileID1, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 1, 1))
fclose(fileID1);
fileID2 = fopen(’Sat2Con.txt’, ’w’);
fprintf(fileID2, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 2, 1))
fclose(fileID2);
fileID3 = fopen(’Sat3Con.txt’, ’w’);
fprintf(fileID3, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 3, 1))
fclose(fileID3);
fileID4 = fopen(’Sat4Con.txt’, ’w’);
fprintf(fileID4, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 4, 1))
fclose(fileID4);
fileID5 = fopen(’Sat5Con.txt’, ’w’);
fprintf(fileID5, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 5, 1))
fclose(fileID5);
fileID6 = fopen(’Sat6Con.txt’, ’w’);
fprintf(fileID6, ’%5d %4d %4d %4d %4d %4d %4d
’, windows(:, 6, 1))
fclose(fileID6);
figure(1)
stackedplot(t/3600, windows(:, 3:7, 1))
ttitle(’6/6/4 Satellite 1’)
xlabel(’time(hours)’)
10.3 Hardware Figures

Figure 34: Level 1 Hardware Design

Legend

Raspberry Pi node

Ground station to ground station link

Figure 35: Level 2 Hardware Design

Legend

Raspberry Pi node

Ground station to constellation satellite link

Constellation satellite to constellation satellite link
Legend

- Raspberry Pi node
- Ground station to constellation satellite link
- Constellation satellite to constellation satellite link
- Constellation satellite to ground vehicle link
- Constellation satellite to orbital vehicle link

Figure 36: Level 3 Hardware Design

Figure 37: Level 4 Hardware Design
Figure 38: Asynchronous Transmission Latency Calculations

Figure 39: Synchronous Transmission Latency Calculations
10.4 SysML Diagrams

Figure 40: Earth Ground Station IBD

Figure 41: Earth Ground Station State Machine
Figure 42: Possible Earth Satellite IBD

Figure 43: Possible Earth Satellite State Machine
Figure 44: Earth Ground Station IBD

Figure 45: Lunar Ground Station IBD
Figure 46: Lunar Satellite IBD

Figure 47: Lunar Satellite State Machine
Figure 48: Example Lunar Ground Vehicle IBD

Figure 49: Example Lunar Orbital Vehicle IBD
10.5 RF and Laser Link Budget Information and Equations

![Predicted BER Curves as a Function of $E_b/N_0$](image)

Figure 50: BER Curve: Found in Space Mission Engineering: the New SMAD
This is the range equation [2] used in this calculator:

$$P_r = P_t \left( \frac{n D_t^2}{\lambda} \right) \left( \frac{n D_t^2}{\lambda} \right) \left( 4\pi R \right) \left( \frac{\lambda^2}{4\pi R} \right) \left( e^{-\frac{4\pi R}{\lambda^2}} \right) \eta_t \eta_r \eta_p$$

Where atmospheric loss, $\eta_a$, is given by:

$$\eta_a = e^{-2a}$$

A. Transmitter Gain, $\frac{n D_t^2}{\lambda}$, with wavelength [E31], $\lambda$, and Transmit Diameter [E5], $D_t$.
B. Receiver Gain, $\frac{n D_t^2}{\lambda}$, with wavelength [E31], $\lambda$, and Receiver Aperture Diameter [E6], $D_r$.
C. Range Loss, $\frac{R^2}{4\pi R}$, with wavelength [E31], $\lambda$, and Range [B31], $R$.
D. Strehl Ratio Loss, $e^{-\frac{4\pi R}{\lambda^2}}$, with wavelength [E31], $\lambda$, and RMS Wavefront Error [B20], $\sigma$.
E. Transmitter Pointing Error Loss, $e^{-2(\theta_0/2)}$, with Laser Divergence [B11], $\theta_0$, and Transmitter Pointing Error [B14], $\theta_t$.
F. System Losses, $\eta_t\eta_r\eta_p$, with Transmitter Optical Efficiency [B13], $\eta_t$, Receiver Optical Efficiency [B19], $\eta_r$, and Receiver Pointing Loss [B21], $\eta_p$.

Calculates the diffraction limited gain, $G_r$, portion of the range equation using the following equation:

$$G_r = 10 \log_{10} \left( \frac{n D_t^2}{\lambda} \right)$$

with wavelength [E31], $\lambda$, and Transmit Diameter [E5], $D_t$. This assumes a diffraction limited laser beam emitting from the transmitter which is calculated in [B9] [2]

Calculates the divergence dependent gain, $G_{rd}$, portion of the range equation using the following equation:

$$G_{rd} = 10 \log_{10} \left( \frac{3\pi^2}{\sigma} \right)$$

With Laser Divergence [B11], $\theta_0$, [1]

Calculates the receiver gain, $G_r$, portion of the range equation using the following equation:

$$G_r = 10 \log_{10} \left( \frac{n D_t^2}{\lambda} \right)$$

With wavelength [E31], $\lambda$, and Receiver Aperture Diameter [E6], $D_r$, [2]

Calculates the Range Loss, $L_r$, portion of the range equation using the following equation:

$$L_r = 10 \log_{10} \left( \frac{\lambda^2}{4\pi R} \right)$$

with wavelength [E31], $\lambda$, and Range [B31], $R$, [2]

Figure 51: Optical Communications Power Received Equations 1

Figure 52: Optical Communications Power Received Equations 2
10.6 RF Link Model MATLAB Code

```matlab
% Power Received Equations
clc; clear all; close all;

%%% Input Variables

%% Constants

% Speed of Light m/s
% Planks Constant
Pt = 10*log10(5); % Transmit Power (dBW)

% Approximation (radians) Laser Divergence (updated 2/4)
theta = 1.24*lambda/Dt; % 1.24 approximates amplifier, magnitude, different parts of fiber laser,
% different parts of the optical path (from GA optical tool approx.)
```

Figure 53: Laser Link between Earth and Moon from Deep Space Optical Communications
\theta_t = 10 \times 10^{-6}; \text{ (radians) Transmitter Pointing Error} \rightarrow (Given \text{ by Loris})

\% System losses (Range from 50–80)

\begin{align*}
\eta_t &= 10 \log_{10}(0.7); \text{ % Transmit efficiency} \\
\eta_r &= 10 \log_{10}(0.7); \text{ % Receiver efficiency} \\
\eta_p &= 0; \text{ % Receiver Pointing Loss} \rightarrow \text{Can be ignored because in this case it is extremely minimal} \\
\eta_a &= -5; \text{ % Atmospheric Losses}
\end{align*}

\% Free Space Loss

\begin{align*}
L_s &= \left(\frac{\lambda}{4 \pi r}\right)^2; \\
L_s &= 10 \log_{10}(L_s); \text{ % } L_s \rightarrow \text{ dB}
\end{align*}

\% Transmitter and Receiver Gain

\begin{align*}
G_t &= 10 \log_{10}\left(\left(\frac{\pi D_t}{\lambda}\right)^2\right); \\
G_r &= 10 \log_{10}\left(\left(\frac{\pi D_r}{\lambda}\right)^2\right)
\end{align*}

\% Strehl Ratio Loss (found that -1.5 dB is a good estimate)

\begin{align*}
SR &= 10 \log_{10}\left(1 - 10^{0.5 R^2}\right) \cdot 1.5; \\
SR &= 10 \log_{10}(1 - 10^{0.5 \text{ RMS}^2})
\end{align*}

\% Transmitter Pointing Error Loss

\begin{align*}
\text{TPE} &= -2; \text{ % From Deep Space Optical Communications Estimations}
\end{align*}

\% Power Received

\begin{align*}
Pr &= Pt + G_t + G_r + L_s + TPE + SR + \eta_t + \eta_r + \eta_p + \eta_a;
\end{align*}

\% Power Required

\begin{align*}
Nb &= 70; \text{ % Receiver sensitivity (\# of photons per bit} \rightarrow \text{ corresponds to } 10^{-9} \text{ BER} \\
R &= 500 \times 10^{-6}; \text{ % 500 Mbps} \rightarrow \text{ bps} \\
P_{\text{req}} &= Nb \times R \times h \times c / \lambda; \\
P_{\text{req}} &= 10 \log_{10}(P_{\text{req}})
\end{align*}

\begin{align*}
\text{Link Margin} &= Pr - P_{\text{req}};
\end{align*}

\% Power Received Equations

\begin{align*}
\text{clc; clear all; close all;} \\
% \text{Input Variables}
\% \text{Constants}
\begin{align*}
c &= 3 \times 10^{-8}; \text{ % Speed of Light m/s} \\
h &= 6.62607004 \times 10^{-34}; \text{ % Planks Constant}
\end{align*}
\% Non–constant
\begin{align*}
Pt &= 10 \log_{10}(5); \text{ % Transmit Power (dBW)} \\
\lambda &= 1550 \times 10^{-9}; \text{ % 1550 nm} \rightarrow \text{ m; % wavelength (m)} \\
D_t &= 1; \text{ % Transmit Diameter (m)} \\
D_r &= 1; \text{ % Receiver Diameter (m)}
\end{align*}
\% Range
\begin{align*}
r &= 384000; \text{ %range (km)} \\
r &= r \times 10^{-3}; \text{ % range in m (km} \rightarrow \text{ m conversion)}
\end{align*}
\% RMS Wavefront Error \[ \text{sigma} \]
\begin{align*}
\text{RMS} &= 0.1; \text{ % RMS Wavefront Error [sigma]} \\
\theta &= 1.24 \times \lambda / D_t; \text{ % Approximation (radians) Laser Divergence (updated 2/4)} \\
\theta &= 1.24 \text{ approximates amplifier, magnitude, different parts of fiber laser,} \\
\theta &= \text{different parts of the optical path (from GA optical tool approx.)}
\end{align*}

\begin{align*}
\text{theta}_t &= 10 \times 10^{-6}; \text{ (radians) Transmitter Pointing Error} \rightarrow (Given \text{ by Loris})
\end{align*}
% System losses (% Range from 50−80)
h_t = 10*\log_{10}(.7); \% Transmit efficiency
h_r = 10*\log_{10}(.7); \% Receiver efficiency
h_p = 0; \% Receiver Pointing Loss — Can be ignored because in this case it is extremely minimal
h_a = -5; \% Atmospheric Losses

% Free Space Loss
Ls = \left(\frac{\lambda}{4 \pi r}\right)^2;
Ls = 10*\log_{10}(Ls); \% Ls — dB

% Transmitter and Receiver Gain
Gt = 10*\log_{10}(\frac{(\pi \times Dt)}{\lambda})^2;
Gr = 10*\log_{10}(\frac{(\pi \times Dr)}{\lambda})^2;

% Strehl Ratio Loss (found that -1.5 dB is a good estimate)
SR = 10*\log_{10}(1-10^{\times\text{RMS}^1.5});

% Transmitter Pointing Error Loss
TPE = -2; \% From Deep Space Optical Communications Estimations

%% Power Received
Pr = Pt + Gt + Gr + Ls + TPE + SR + h_t + h_r + h_p + h_a;

%% Power Required
Nb = 70; \% Receiver sensitivity (# of photons per bit —> corresponds to 10^{-9} BER)
R = 500*10^{-6}; \% 500 Mbps —> bps
P_req = Nb*R*h*c/\lambda;
P_req = 10*\log_{10}(P\_req);
LinkMargin = Pr - P\_req;

clc; clear all; close all;

% Input Variables

% Constants
c = 3*10^{-8}; \% Speed of Light m/s
h = 6.62607004*10^{-34}; \% Planks Constant

% Non-constant
Pt = 10*\log_{10}(5); \% Transmit Power (dBW)
lambda = 1550*10^{-9}; \% 1550 nm —> m; \% wavelength (m)
Dt = 1; \% Transmit Diameter (m)
Dr = 1; \% Receiver Diameter (m)
r = 384000; \%range (km)
r = r*10^{-3}; \% range in m (km —> m conversion)

RMS = 0.1; \% RMS Wavefront Error \[\sigma\]
theta = 1.24*lambda/Dt; \% Approximation (radians) Laser Divergence (updated 2/4)
\% 1.24 approximates amplifier, magnitude, different parts of fiber laser,
\% different parts of the optical path (from GA optical tool approx.)
theta_t = 10*10^{-6}; \% (radians) Transmitter Pointing Error —> (Given by Loris)

% System losses (% Range from 50−80)
\[ h_t = 10 \log_{10}(.7); \text{ % Transmit efficiency} \]
\[ h_r = 10 \log_{10}(.7); \text{ % Receiver efficiency} \]
\[ h_p = 0; \text{ % Receiver Pointing Loss} \to \text{ Can be ignored because in this case it is extremely minimal} \]
\[ h_a = -5; \text{ % Atmospheric Losses} \]
\[ L_s = \left( \frac{\lambda}{4 \pi r} \right)^2; \]
\[ L_s = 10 \log_{10}(L_s); \text{ % } L_s \to \text{ dB} \]
\[ G_t = 10 \log_{10}(\left( \frac{\pi Dt}{\lambda} \right)^2); \]
\[ G_r = 10 \log_{10}(\left( \frac{\pi Dr}{\lambda} \right)^2); \]
\[ SR = 10 \log_{10}(1 - 10^{1.5 \times \text{RMS}}); \]
\[ TPE = -2; \text{ % From Deep Space Optical Communications Estimations} \]

% Power Received
\[ Pr = Pt + G_t + G_r + L_s + TPE + SR + h_t + h_r + h_p + h_a; \]

% Power Required
\[ Nb = 70; \text{ % Receiver sensitivity (# of photons per bit} \to \text{ corresponds to } 10^{-9} \text{ BER} \]
\[ R = 500 \times 10^{-6}; \text{ % 500 Mbps} \to \text{ bps} \]
\[ P_{\text{req}} = Nb \times R \times h \times c / \lambda; \]
\[ LinkMargin = Pr - P_{\text{req}}; \]

% Input Variables
\[ c = 3 \times 10^{-8}; \text{ % Speed of Light m/s} \]
\[ h = 6.62607004 \times 10^{-34}; \text{ % Planks Constant} \]
\[ \text{ % Non-constant} \]
\[ \lambda = 1550 \times 10^{-9}; \text{ % 1550 nm} \to \text{ m; % wavelength (m)} \]
\[ Dt = 1; \text{ % Transmit Diameter (m)} \]
\[ Dr = 1; \text{ % Receiver Diameter (m)} \]
\[ r = 384000; \text{ %range (km)} \]
\[ r = r \times 10^{-3}; \text{ % range in m (km} \to \text{ m conversion)} \]
\[ \text{ % RMS Wavefront Error [sigma]} \]
\[ \theta = 1.24 \times \lambda / Dt; \text{ % Approximation (radians) Laser Divergence (updated 2/4)} \]
\[ \theta_t = 10 \times 10^{-6}; \text{ % (radians) Transmitter Pointing Error} \to \text{ (Given by Loris)} \]
\[ \text{ % System losses (Range from 50–80)} \]
\[ h_t = 10 \log_{10}(.7); \text{ % Transmit efficiency} \]
\[ h_r = 10 \log_{10}(.7); \text{ % Receiver efficiency} \]
h_p = 0; % Receiver Pointing Loss —–> Can be ignored because in this case it is extremely minimal
h_a = -5; % Atmospheric Losses

% Free Space Loss
Ls = (lambda/(4*pi*r))^2;
Ls = 10*log10(Ls); % Ls —> dB

% Transmitter and Receiver Gain
Gt = 10*log10((pi*Dt)/lambda)^2);
Gr = 10*log10((pi*Dr)/lambda)^2);

% Strehl Ratio Loss (found that -1.5 dB is a good estimate)
SR = 10*log10(1-10*(RMS)^1.5);

% Transmitter Pointing Error Loss
TPE = -2; % From Deep Space Optical Communications Estimations

% Power Received
Pr = Pt + Gt + Gr + Ls + TPE + SR + h_t + h_r + h_p + h_a;

% Power Required
Nb = 70; % Receiver sensitivity (# of photons per bit —> corresponds to 10^-9 BER)
R = 500*10^6; % 500 Mbps —> bps
P_req = Nb*R*h*c/lambda;
P_req = 10*log10(P_req);
LinkMargin = Pr - P_req;

10.7 Laser Link MATLAB Code

% Power Received Equations
clc; clear all; close all;

%%% Input Variables

% Constants
c = 3*10^-8; % Speed of Light m/s
h = 6.62607004*10^-34; % Planks Constant

% Non-constant
Pt = 10*log10(5); % Transmit Power (dBW)
lambda = 1550*10^-9; % 1550 nm —> m; % wavelength (m)
Dt = 1; % Transmit Diameter (m)
Dr = 1; % Receiver Diameter (m)

r = 384000; % range (km)
r = r*10^-3; % range in m (km —> m conversion)

RMS = 0.1; % RMS Wavefront Error [sigma]
theta = 1.24*lambda/Dt; % Approximation (radians) Laser Divergence (updated 2/4)
% 1.24 approximates amplifier, magnitude, different parts of fiber laser,
% different parts of the optical path (from GA optical tool approx.)
theta_t = 10*log10(5); % (radians) Transmitter Pointing Error —> (Given by Loris)

% System losses (Range from 50–80)
h_t = 10*log10(.7); % Transmit efficiency
h_r = 10*log10(.7); % Receiver efficiency
h_p = 0; % Receiver Pointing Loss —> Can be ignored because in this case it is extremely minimal
h_a = -5; % Atmospheric Losses

% Free Space Loss
Ls = (lambda/(4*pi*r))^2;
Ls = 10*log10(Ls); % Ls —> dB

% Transmitter and Receiver Gain
Gt = 10*log10( ((pi*D_t)/lambda)^2);
Gr = 10*log10( ((pi*D_r)/lambda)^2);

% Strehl Ratio Loss (found that -1.5 dB is a good estimate)
SR = 10*log10(1 - 10*(RMS)^1.5);

% Transmitter Pointing Error Loss
TPE = -2; % From Deep Space Optical Communications Estimations

%% Power Received
Pr = Pt + Gt + Gr + Ls + TPE + SR + h_t + h_r + h_p + h_a;

%% Power Required
Nb = 70; % Receiver sensitivity (# of photons per bit —> corresponds to 10^-9 BER)
R = 500*10^6; % 500 Mbps —> bps
P_req = Nb*R*h*c/lambda;
P_req = 10*log10(P_req);

LinkMargin = Pr - P_req;
10.8 System Architecture Trade Study

![Illustration of each orbital height, high, medium, and low, over the moon along with their coverage percentages]

**High-Altitude Constellation**

The high-altitude orbital design considers a Lunar analog of geostationary orbit (GEO), in which the percentage of planetary surface area covered by each satellite is the same as that of a satellite in GEO. A satellite in GEO with an altitude of 35,786 km will cover approximately 41% of the Earth [7]. Scaling this down results in an altitude of 9,757 km to cover the same percentage of the Moon.

Singh et al.[8] identifies a mission design that provides near-total coverage of the Earth’s surface using four satellites in GEO. Thus, it is achievable from an engineering perspective to design a constellation with five satellites that will achieve the desired coverage at this altitude. This makes such a constellation a viable design option for this project. Similar constellations have been demonstrated for full-coverage use in Earth orbit, and the high coverage per satellite makes this solution very cost-effective. However, this solution does have drawbacks, as transmitters at higher altitudes will require more power and higher pointing accuracy compared to lower altitudes in order to ensure signals reach their destination on the ground.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low complexity</td>
<td>High transmission power required</td>
</tr>
<tr>
<td>Low cost</td>
<td>High pointing accuracy required</td>
</tr>
<tr>
<td>High coverage</td>
<td>High vulnerability to failure</td>
</tr>
</tbody>
</table>

Table 24: High-Altitude Constellation Pros and Cons
Low-Altitude Constellation

The low-altitude orbital design considers a Lunar analog of low-Earth orbit (LEO), in which the percentage of planetary surface area covered by each satellite is the same as that of a satellite in LEO. A satellite in LEO with an altitude of 1,000 km will cover approximately 6.7% of the Earth. Scaling this down results in an altitude of 273 km to cover the same percentage of the Moon.

Using the previously described high-altitude constellation as a baseline, scaling by the area-coverage ratio suggests this low-altitude constellation will require at least 24 satellites. This solution benefits from having less strict requirements for transmission power and pointing accuracy compared to the high-altitude solution, as well as being much more accommodating of individual satellite failure. However, this solution suffers from the perspective of coverage and therefore cost. The prices of launch and orbital insertion represent a very significant portion of the cost of such missions, and the large size of this constellation indicates it would cost roughly five times more than the high-altitude solution.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low transmission power required</td>
<td>High complexity</td>
</tr>
<tr>
<td>Low pointing accuracy required</td>
<td>Very high cost</td>
</tr>
<tr>
<td>Resilient to failure</td>
<td>Low coverage</td>
</tr>
</tbody>
</table>

Table 25: Low-Altitude Constellation Pros and Cons

Medium-Altitude Constellation

This medium-altitude orbital design considers a Lunar analog of medium-Earth orbit (MEO), in which the percentage of planetary surface area covered by each satellite is the same as that of a satellite in MEO. A satellite in MEO with an altitude of 20,200 km will cover approximately 38% of the Earth. Scaling this down results in an altitude of 5,509 km to cover the same percentage of the Moon.

Walker [9] describes a constellation pattern consisting of six satellites that provides full coverage of the surface of a body, and is very similar to the pattern used for GPS satellites, which also operate in MEO. This solution is very coverage-effective, requiring just one more satellite than the high-altitude solution. Meanwhile, at just over half the altitude, it is less demanding in terms of transmission power required and pointing accuracy. However, with individual satellites still covering a very large swath, it is also vulnerable to failure.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low complexity</td>
<td>Moderately high transmission power required</td>
</tr>
<tr>
<td>Low cost</td>
<td>Moderately high pointing accuracy required</td>
</tr>
<tr>
<td>High coverage</td>
<td>High vulnerability to failure</td>
</tr>
</tbody>
</table>

Table 26: Medium-Altitude Constellation Pros and Cons

10.8.1 Trade Study

Essential to the function of the network is the anatomy of the satellite constellation. The satellites in the constellation will distribute the communication hardware around the body in question, and make the goal of >99% coverage feasible. This trade study was conducted to evaluate the trade-offs of full-coverage constellations at different altitudes, and subsequently the number of satellites that they contain. The criteria for the trade study were selected as follows.

Coverage

Coverage was selected as the primary criteria because it has a direct effect on the overall cost of the system, with higher coverage translating to fewer satellites and lower cost which, along with engineering achievability, serves as a primary determinant of how feasible a project is. Though constructing the entire system is outside the scope of this project, cost will ultimately always be a limiting factor in engineering designs and decisions.
Coverage

<table>
<thead>
<tr>
<th>Score</th>
<th>Poor</th>
<th>Moderate</th>
<th>Excellent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 27: Value assignments for Coverage

Link Budget Requirements

The link budget requirements serve to represent the communications challenges associated with each design. From a systems engineering perspective, it is important to identify solutions that not only confront a specific engineering challenge, but are also able to accommodate all components of the mission. This assessment primarily considered space loss, which is the most significant source of loss in an RF link budget.

<table>
<thead>
<tr>
<th>Score</th>
<th>Impossible</th>
<th>Challenging</th>
<th>Trivial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 28: Value assignments for Link Budget Requirements

Failure Vulnerability

Failure vulnerability represents the impact to system functionality that would result from the loss of a single satellite. This is an important consideration with regard to the system’s implementation and functionality in the real world. Particularly in the harsh environment of space, hazardous and damaging events can be unpredictable, and a system’s resilience to such events is an important consideration in the ultimate feasibility of the design.

<table>
<thead>
<tr>
<th>Score</th>
<th>Catastrophic</th>
<th>Debilitating</th>
<th>Negligible</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 29: Value assignments for Failure Vulnerability

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>Evaluation</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>Coverage</td>
<td>5</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>Link Budget Requirements</td>
<td>3</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Failure Vulnerability</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Total Score</td>
<td>63</td>
<td>66</td>
<td>51</td>
</tr>
</tbody>
</table>

Table 30: Satellite Constellation Design Trade Study

The above trade study led to the selection of a medium-altitude constellation, consisting of 6 satellites at an altitude analogous to that of GPS satellites around Earth, scaled down to the relative size of the body.

10.9 Network Protocol

10.9.1 Conceptual Designs Considered

A network’s protocol determines how data will be sent and received within nodes in the same network. The network protocol of project DOTCOM’s communication architecture will allow the various nodes of the network to transfer data between themselves. These nodes could be manned spacecraft, ground stations, satellite constellations, or mission vehicles. The selected protocol scheme must be able overcome or work within the limitations imposed on communications by the space environment. The space communications environment is characterized by long distances, asymmetric data rates, high bit error rates, and frequent interruptions in end to end connections. There are two ‘core’ protocol strategies that could be applied to the DOTCOM network, Internet Protocol (IP) and Delay Tolerant Networking, also known as bundle protocol.
Transmission Control Protocol / Internet Protocol (TCP/IP)

TCP/IP (or simply TCP) is the data protocol used in terrestrial internet. It is based on four core assumptions. TCP/IP assumes that network nodes are continuously connected, networks have short signal propagation delays, data links are symmetric and bi-directional, and bit error rates are low. IP protocol functions can be organized graphically into stacks, where the top layer function relies on services of the layers below it. Figure 55 shows the basic IP stack. In an IP based internet, the start and end points of a transmission must be able to run all of the stack layers, but the intermediary nodes don’t need to run the transport or application layers. This helps reduce latency through the network. A basic IP network is visualized in figure 56.

In terrestrial internet applications, round trip signals are fast and cheap. This implies that bidirectional network connections established between two communicating parties is straightforward. TCP/IP supports reliable communications through the use of acknowledgements and re-transmissions, where receivers acknowledge all data received. If a download fails or packet loss happens, one endpoint can quickly communicate with the other to ask for a data resend. Nodes and endpoints can ask for any information they need from a host server. TCP/IP is also embedded with efficient congestion control algorithms, where TCP/IP senders will detect significant packet losses in the network and respond by reducing transmission rates. While this is an effective strategy when the network is congested, the protocol assumes that losses indicate network congestion only and may try to compensate when the network is not congested, such as when transmission disruption causes packet loss for other reasons. However, TCP/IP protocols are sensitive to delays, and the

---

**Figure 55: Basic IP Protocol Stack [1]**

**Figure 56: Basic IP Network [1]**
propagation delays arising from internet protocol exchange between satellite paths must be considered when attempting to utilize this protocol in space. This propagation error can be attributed to signal shadowing and blockage effects, which can both influence the performance of the protocol by making transmission windows temporarily unavailable [2].

For space applications, TCP/IP is utilized primarily in LEO networks where they experience lower latencies than in GEO configurations. When used in GEO, TCP/IP is meant to provide high availability high data rate links with large total capacities [2]. Traditional TCP/IP protocol does not translate directly to the space environment, and for this reason many alternatives that act to enhance certain aspects of the network exist. These alternative protocols build upon fundamental TCP/IP protocol by introducing supplemental network assistance across the transmission path.

The TCP Tranquility network is one variant that was created to enhance the performance of standard TCP protocol. It places performance enhancing proxies (PEPs) in the middle of the network that helps to reduce network congestion and optimize transmission window availability [3]. The TCP Tranquility is also referred to as Space Communications Protocol Standard - Transport Protocol (SCPS-TP), and is considered the space communication extension to TCP/IP.

There also exist TCP variants that enhance the performance of the end systems. In particular, TCP Westwood can accelerate the transfer rate of standard TCP by optimizing available bandwidth. It does this by taking note of acknowledgement spacing between endpoints and infers the available connection bandwidths. In addition, TCP Peach seeks to further identify the causes of packet loss, which assists the network’s reorientation response to loss. This is done through the use of ‘dummy’ packets, which register themselves low on the priority transmission stack. Based on the behavior of the packet loss, the protocol is then able to identify the source of loss. Dummy and real packets being dropped will indicate a loss due to network congestion. If the protocol detects that few dummy packets were lost, it determines the cause of loss to be corruption due to bit errors. The network will subsequently respond by not cutting transmission rates [4]. Additional methods such as increasing satellite diversity in the network, header compression, and

The variants described above supplement ‘stock’ TCP through tuning for specific connection scenarios. They can dramatically bolster the performance of stock TCP protocols through increasing the transmission window size. In practice this is difficult, seeing as it is very intensive and round trip times (RTT) can be highly variable. To fully optimize TCP, the protocol needs to be adjusted depending on the network’s number of shared connections. A problem of implementing TCP variants on top of stock TCP can also lead to unfairness between the networks. Because network resources are get shared between the two protocols, it is possible that the TCP variant protocol can starve out the resources from the stock TCP [4].

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>High coverage achievable</td>
<td>Environmental noise leads to high BER</td>
</tr>
<tr>
<td>Effective congestion control algorithms</td>
<td>Sensitive to transmission availability</td>
</tr>
<tr>
<td>Multiple variants exist to overcome specific challenges</td>
<td>Variants are widely incompatible on top of stock TCP</td>
</tr>
<tr>
<td>Reliable transmission via acknowledge/retransmit</td>
<td>Asymmetries exist</td>
</tr>
</tbody>
</table>

Table 31: TCP/IP Pros and Cons

Delay/Disruption Tolerant Network (DTN) Protocol

Delay tolerant networking operates on a store and forward principle. Data to be sent over the network is first broken into bundles and stored locally on the sending node. The sender node then transmits the bundles to the next node in the network once a connection becomes available. The receiving node then stores the bundles in local memory, and becomes the new ‘sender’ node. This data handling strategy helps overcome variable or interrupted communication routes, long round trip communication latency, and asymmetric link data rates. The basic DTN protocol stack is shown in figure 57.
Because of the added stack layers and store and forward requirements of a DTN, intermediary nodes have to run more stack layers than IP. The intermediary nodes operate all the way up to the bundle protocol layer. In the space environment, a satellite, rover, or ground station could constitute a node in the network. Each of these nodes is transient. As a planet spins or a satellite moves in orbit, the network topology will change. Node connections will be continuously shifting. This necessitates that each node must have a basic understanding of the local network architecture (when local nodes will be available for data transmission). It is important to note that DTN protocols function in a similar manner to IP on networks with continuous connections. Additionally, a conversion layer can be added to the DTN stack in order to allow DTN to function over network sections that are only capable of running TCP/IP. A representation of a DTN network with mixed protocols is shown in figure 58. A compiled list of pros and cons for DTN protocol is found in table 32.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reliable data transmission in long delay scenarios</td>
<td>Nodes must have data storage capability</td>
</tr>
<tr>
<td>Less latency over variable connection network scenarios</td>
<td>Intermediary nodes must be 'smart' (operate on a higher stack level)</td>
</tr>
<tr>
<td>Faster corrections for high bit error rates or lost packets</td>
<td>Nodes must have a basic understanding of local network architecture</td>
</tr>
<tr>
<td>Can handle large asymmetric data rates</td>
<td>-</td>
</tr>
<tr>
<td>Can run over an existing IP stack or as an independent stack</td>
<td>-</td>
</tr>
<tr>
<td>Encryption and security without adding a stack layer</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 32: DTN Protocol Pros and Cons

NASA identified DTN as a potential upgrade to IP for interplanetary networks in 1998. This kicked off research and development that has culminated in the Interplanetary Overlay Network (ION). ION is
NASA’s implementation of the DTN protocols for use in interplanetary space networks. The current ION code has been developed in a LINUX environment for both 32 and 64 bit processors. The ION protocol code is fully developed, and implements custody transfer, status reports, delivery options, priority, reassembly from fragments, a TCP convergence layer adapter, flow control, congestion control, and security encryption options [5]. NASA also released several YouTube lab and lecture courses detailing how to implement ION for space communications in July 2020 [1]. ION is open source and available through source forge [6].

10.9.2 Trade Study

The network protocol is a designated transmission package that writes the rules for how data is sent between nodes in a network. For the purposes of project DOTCOM, this protocol must demonstrate high efficiency and reliability in the space environment. Other key considerations are how the protocol handles network congestion, reduces latency, interfaces with other systems, and scales with network size. Protocol complexity should also be considered. These ideas are used as trade metrics to contrast TCP/IP with DTN protocol for use in the DOTCOM network.

Robustness

Because the DOTCOM network will involve a large number of interconnected nodes, it will require sufficient internal data loss resolution to account for transmission losses in the network. With a high-coverage network, there will be transmission congestion and loss of data due to the high activity. Robustness seeks to define how disruptions to the network in regards to bit errors, packet transmission failure, and loss of nodes are to the overall performance of the network, and the degree to which a network is robust. Scores will be assigned for this metric depending on a protocol performance’s sensitivity to these kinds of losses. A score of 10 indicates perfect instantaneous corrections of corrupted bits, node outages, and other data transfer issues. A score of 1 indicates that human intervention, hard coding, manual data resend, and other labor intensive tasks are required to correct any failures. This metric was weighted 2.5 because a thorough network response to data disruption is needed to justify an efficient network protocol with fast data transfer.

<table>
<thead>
<tr>
<th>Robustness</th>
<th>Very robust</th>
<th>Somewhat robust</th>
<th>Not robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>5.5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 33: Value assignments for degree of robustness

Latency

There will be a number of transmission latencies that the network will have to account for in the space environment. Signals in the DOTCOM network will be exchanged via ground stations, satellites, and various mission vehicles over long distances. As a result, minimizing the transmission latencies is essential to providing a fast access network. The latency metric will compare the methods used by each protocol to minimize latencies when transmitting across the space environment. This metric was given a weight of 2.5 because minimizing the environmental and internal effects on latencies will ensure fast and reliable communications throughout the network.

<table>
<thead>
<tr>
<th>Latency</th>
<th>Light delay latency only</th>
<th>Extended time between transmission windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 34: Value assignments for latency resolution

Network Congestion

This network will essentially act to support all communications originating from and sending between the Moon and Earth. As such, the high transfer of data will result in congestion as the network configures node availability and data routing. This metric will trade the network protocols based on how they react to network congestion as they pertain to routing significantly large amounts of data over varied transmission window availability. It is necessary for the selected network protocol to effectively and efficiently identify available nodes, optimize transmission windows, and direct routing in order to minimize network congestion. This
design aspect is key to an efficient network protocol that can satisfy signal transmission and communication coverage requirements, and was given a weight of 2 as a result.

<table>
<thead>
<tr>
<th>Optimal performance under:</th>
<th>Highly congested network</th>
<th>High variability in window availability</th>
<th>Low variability in window availability</th>
<th>Fully open transmission windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>7</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 35: Value assignments for network congestion

### Complexity

Network protocols are complex on a high level. Numerous protocol stacks defined for different purposes, and require an assortment of physical hardware connections and software protocol initiatives used to dictate the transmission methods. The complexity metric will compare number the hardware and software connections required for sufficient implementation of the protocol to the overall network.

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Simple hardware and software</th>
<th>Simple hardware, complex software</th>
<th>Simple software, complex hardware</th>
<th>Complex software and hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>7</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 36: Value assignments for network complexity

### Scalability

Project DOTCOM will be illustrated using a hardware representation as well as a representative simulation of network procedures that uses a model-based systems engineering approach. The evaluation of the network’s communication and relay methodology must touch on how the network can be scaled in real life to support an interplanetary communication. This is very important to General Atomics because the DOTCOM team must illustrate the viability of scaling up the theoretical architecture of the network. The scores for this metric will be based on the feasibility associated with up-scaling adding new nodes to the network. A score of 10 means that existing nodes can identify nodes that are added without the need for the network as a whole to be aware of the addition. A score of 1 means that nodes of the entire network will have to be re-oriented to acknowledge additional nodes. For team purposes, the scalability of a network protocol will only apply when simulating the network as a whole. As a result, the degree to which a network protocol is scalable in a simulation is not significant. This metric was given a weight of 1 because high scalability must be justified in theory for real-world applications, but the simulated network representation does not depend on protocol scalability.

<table>
<thead>
<tr>
<th>Scalability</th>
<th>Autonomous connectivity</th>
<th>Manual rework of each node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 37: Value assignments for scalability

### Compatibility

General Atomics works with NASA and other partners to create satellite and communications solutions in the space environment. In accordance with FR 3, it is fundamental for this network to be able to interface with and support existing and future communications infrastructure. This applies to existing communications infrastructure used by NASA, future NASA missions such as the Lunar Gateway, and other government, military, and commercial space communications infrastructures both existing and to come. The degree to which a network protocol is compatible with current and future communications infrastructures will depend on its ability to cooperate with the protocols used. In essence, the network protocol selected will have to be universally compatible with any network protocol (DTN, TCP/IP with performance enhancing proxies, and
others). This metric is weighted at 1.5 because it is needed to meet all design requirements encapsulated by FR 3.

<table>
<thead>
<tr>
<th>Network protocol compatibility</th>
<th>Universal compatibility</th>
<th>Terrestrial/space/military</th>
<th>Terrestrial/space</th>
<th>Terrestrial only</th>
<th>Insufficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>10</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 38: Value assignments for protocol compatibility

**Trade Study Evaluation**

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>IP</th>
<th>DTN</th>
<th>IP</th>
<th>DTN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robustness</td>
<td>2.5</td>
<td>4</td>
<td>8</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>Latency</td>
<td>2.5</td>
<td>4</td>
<td>6</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>Network Congestion</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>Complexity</td>
<td>.5</td>
<td>7</td>
<td>3</td>
<td>3.5</td>
<td>1.5</td>
</tr>
<tr>
<td>Scalability</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>Compatibility</td>
<td>1.5</td>
<td>4</td>
<td>9</td>
<td>6</td>
<td>13.5</td>
</tr>
<tr>
<td><strong>Total Score</strong></td>
<td>41.5</td>
<td>71</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Robustness:**

IP was given a score of 4 in robustness because the network assumptions that IP relies on begin to break down in the space environment. If a bit gets corrupted, or a parity check fails in IP, the receiving node sends a signal back to the sending node to initiate a resend of the missing data. This solution relies on low ping and short round trip communication times. It works great for terrestrial internet, but breaks down under interplanetary distances. The average round trip light time of an Earth-Moon signal is 2.46 seconds based on light-speed propagation. A typical round trip time in the terrestrial internet is 200ms, or 0.2 seconds [1]. DTN was given a score of 8 because of the store and forward assumptions it was built on. If a bundle fails to arrive at the end node in a DTN network, the previous node can resend the data. This eliminates the need for a round trip query to travel the full distance between sender and receiver. We also considered that IP requires an unbroken end to end connection to transmit data. A DTN does not. In a DTN network, data is forwarded as soon as the next node in the network becomes available.

**Latency:**

DTN has the edge over IP for latency in the space environment. IP received a 4 because of the effect of limited transmission windows on data transfer. DTN received a 6 because it doesn’t have to wait for a complete end to end connection to send data. Figure 59 is an illustration of data transmission over an IP network vs a DTN network. Because a TCP/IP network requires a complete end to end path, data must wait to be sent until the full path becomes available. In contrast, the DTN network can start sending data to the next node as soon as it becomes available. It doesn’t need to wait for a complete end to end path. In a space network characterized by long signal delays and intermittent connectivity, the DTN network outperforms the TCP/IP network in latency and throughput.
Network Congestion:

IP received a 4 in network congestion because of the lack of end to end communication windows in a space environment. This will limit the available transmission time for data, and result in high network congestion. DTN received a 7 because transmission windows are more frequent and results in a larger throughput. DTN also has the ability to manually set priority destinations or transmissions by the user to assist with network traffic control in a congested state.

Complexity:

IP received a 7 for complexity. The hardware required is simple, and each intermediary node only needs to run up to the network layer in the IP stack. DTN received a 3 because each node needs to run more protocol layers, and must also have local storage. The addition of a radiation hardened solid state memory device enhances the hardware complexity for each node of a DTN network.

Scalability:

IP received a 4 in scalability because in standard IP, nodes are assumed to be stationary. This poses a challenge when trying to integrate a new satellite into the existing network. In the absence of a DTN network, data transmissions to any new satellites would need to be manually scheduled to avoid congestion and ensure end to end connectivity. This becomes labor prohibitive as new nodes are added to the network. DTN received a 7 in scalability. When adding a new node to a DTN network, the local nodes will need to be manually patched so that they know a new node exists. Once that is complete, the new node is fully integrated into the network.

Compatibility:

IP received a 4 in compatibility because it only integrates well with the terrestrial internet. Current space networks that run IP are closed systems that must be manually reworked to receive new nodes. DTN received a score of 9 because of the versatility that the conversion layer brings to the system. The DTN conversion layer allows DTN transmissions to adjust to multiple transport protocols through any portion of the network. Essentially, DTN protocol can run on TCP/IP or as a stand alone. In addition, there is built in security and encryption options for interfacing with military communications.
Total Score:
IP received a total score of 41.5 out of 100. It is clear that TCP/IP protocol will be insufficient for a deep space network. The core assumptions that TCP/IP was designed around do not hold true for the space environment. DTN received a total score of 71 out of 100. DTN, and ION by extension, has been designed to run on an interplanetary network. DTN protocol is the best choice to meet the relevant functional and design requirements for this project.

10.10 Extraterrestrial Relay Station

10.10.1 Conceptual Designs Considered
The extraterrestrial relay station is a key part of the network system’s high level hardware design. The relay system is necessary to transport large amounts of data between Earth and other far-separated bodies such as the Moon. It’s importance is based on significantly cutting down the cost of the constellation by allowing it’s satellites to be designed exclusively for short range communication, where a few independent stations will be designed to transport all the data to and from Earth and the respective constellation. A relay station on the Moon with direct line-of-sight to Earth will be sent all data from the surrounding constellation using short range communications. From there, it will send that data to an Earth ground station using long range communications in order to ensure all data is accurately relayed. The same process is reversed for Earth-to-Moon data transfer. For the purpose of DOTCOM, two design options were considered for long distance data transfer hubs: satellite relay and ground station relay.

Support Satellite Relay

Figure 60: Illustration of Satellite Relay Data Transfer Path

Satellite relay consists of a small constellation of satellites orbiting the larger communications constellation that are built specifically for long range data transfer. They are placed in strategic orbits such that at least one always has direct line of sight of Earth so that it can always send/receive information between there and it’s respective communications constellation. Satellite relay stations has only recently become more popular
as demand has begun to increase. They allow for data transmission to specific ground based locations from satellites that would typically be out of range because they aren’t positioned overhead. However, satellite relay stations are significantly more expensive and difficult to maintain than ground stations, and thus are only a viable option with a large amount of customers that significantly benefit from the highlighted advantages.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>No location restrictions</td>
<td>Lower bandwidth</td>
</tr>
<tr>
<td>Less infrastructure required</td>
<td>High development costs</td>
</tr>
<tr>
<td></td>
<td>High maintenance costs</td>
</tr>
<tr>
<td></td>
<td>High maintenance time/difficulty</td>
</tr>
<tr>
<td></td>
<td>Scalability issues</td>
</tr>
<tr>
<td></td>
<td>Increased cost and complexity of communication</td>
</tr>
<tr>
<td></td>
<td>constellation satellites</td>
</tr>
</tbody>
</table>

Table 39: Support Satellite Relay Pros and Cons

Ground Station Relay

Ground station relay consists of multiple ground stations positioned strategically around a moon or planet such that at least one will always have direct line of sight with Earth. They are built specifically to handle long range and short range communications in order to send/receive data between their orbiting constellation and Earth, respectively. Ground stations have always been the primary solution for data transmission between Earth and satellites, but have never been used on other cosmic bodies since there have been very few long term missions to such. They allow for much lower costs and better specifications (with thin to no atmosphere), but are limited by geographical and land-ownership placement restrictions.
Advantages | Disadvantages
---|---
Higher bandwidth | Location restrictions
Low development costs | More total infrastructure
Low maintenance costs | 
Low maintenance time/difficulty |

Table 40: Ground Station Relay Pros and Cons

10.10.2 Trade Study

A trade study on the extraterrestrial relay station was done to determine the most effective way to transfer data to and from the Earth and Moon. As this is a key part of the project with multiple established ways of going about it that are significantly different, it was important to break down the strengths and weaknesses of each concept to determine which would be most applicable for DOTCOM.

Accessibility

Accessibility was 50% of the criteria because of its significant importance since the system will be based so far from Earth. Tasks such as maintenance or upgrading hardware are extremely difficult off planet but crucial to keeping a system working correctly and efficiently, so it was determined that ease of performing such tasks was a large factor in the decision.

<table>
<thead>
<tr>
<th>Accessibility</th>
<th>Impossible</th>
<th>Challenging</th>
<th>Easy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 41: Value assignments for Accessibility

Cost

Cost was included as 30% of the criteria because of its importance in determining a project’s profitability (and thus, success). This includes all life cycle financial requirements such as development, deployment, maintenance, and any other associated costs.

<table>
<thead>
<tr>
<th>Cost</th>
<th>Low</th>
<th>Moderate</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 42: Value assignments for Cost

Data Specs

Data specs was included as 20% of the criteria because of the high goals set by the project. Although the bandwidth, latency, etc. won’t significantly effect the project’s success over a certain threshold, it could still benefit the project’s value proposition long term and increase the lifespan of certain components, thus increasing profits through reduced cost.

<table>
<thead>
<tr>
<th>Data Specs</th>
<th>Bad</th>
<th>Moderate</th>
<th>Good</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 43: Value assignments for Data Specs

Trade Study Evaluation

This trade study focused on the differences between satellite relay and ground station (GS) relay. In the study, it was found that satellite relay is an emerging technology used to increase the window of time or frequency a satellite can transmit data to a specific ground station. However, this form of data relay has many significant disadvantages to standard ground relay stations such as higher cost, lower bandwidth, and more difficult maintenance.
### Table 44: Extraterrestrial Relay Station Trade Study

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>Evaluation</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Satellite Relay</td>
<td>GS Relay</td>
</tr>
<tr>
<td>Accessibility</td>
<td>5</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>Cost</td>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>Data Specs</td>
<td>2</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total Score</td>
<td></td>
</tr>
</tbody>
</table>

GS relay received a high score for accessibility due to the fact that it would be stationed on the ground, likely near some form of base camp. Satellite relay received a very low score due to the difficulty of repairing objects in space, paired with the lack of resources Moon would have to perform such tasks compared to Earth for the foreseeable future. GS relay also received a high score for cost, as ground stations are relatively simple and inexpensive to manufacture. Satellite relay, again, received a very low score due to the high cost associated with designing and manufacturing multiple satellites. Launch costs were not taken into account for either system, as it was estimated they would be very similar for both. Last, data specs were rated high for both, though multiple reports suggest bandwidth is higher in ground stations due to typical increased power accessibility.

### 10.11 Wireless Communication Spectrum

#### 10.11.1 Conceptual Designs Considered

The two communication frequency spectrum considered for data links in the network are RF and laser communications. It was determined that both RF and laser must be considered because there are advantages and disadvantages to using RF and laser for short and long distance communications. The pros and cons of the two different types of communications and the trade studies done are discussed in this section.

**RF Communications**

For the DOTCOM network, RF is a safe method of communication regardless of operating conditions. The majority of RF frequencies are unaffected by rain fade (the main cause of atmospheric interference) which is very beneficial for communication between ground stations and satellites. RF transmissions are not extremely sensitive to pointing accuracy due to the width of the signal and size of receiving dishes. Additionally, the data rate thresholds can be met utilizing strictly RF [10]. However there are downsides to RF. The objective data rate for the design is significantly higher, where 5 Gbps is required between the Earth and Moon. This is outside of the capabilities of RF for the foreseeable future. Moreover, for high frequency (Ka band) RF communications outside of the atmosphere, transceivers can be power hungry and expensive and may exceed operational capabilities for many spacecraft.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>High stability and range</td>
<td>High frequency transmission equipment is power hungry and expensive</td>
</tr>
<tr>
<td>Minimal atmospheric attenuation</td>
<td>Cannot meet objective requirements utilizing strictly RF <em>(DR 1.6, DR 1.7)</em></td>
</tr>
<tr>
<td>Meets threshold data rate requirements <em>(DR 1.6, DR 1.7)</em></td>
<td></td>
</tr>
<tr>
<td>Less strict pointing accuracy requirements</td>
<td></td>
</tr>
</tbody>
</table>

**Table 45: RF Communications Pros and Cons**

**Laser Communications**

Laser communication is becoming the future of space communication. As technology advances, space agencies around the globe are investing into projects to research and develop complex laser communication systems for deep space applications. One of the biggest advantages of using laser communications in a deep space communication network is the ability to achieve high data-transfer rates. The laser data-transfer rates have
been estimated to be about 2 to 100 times greater than the data-transfer rates achieved with RF. The higher bandwidth associated with laser communication, compared to RF, allows for this higher data-transfer rate. This means that satellites can send much more information in a shorter period of time compared to RF communication. NASA was able to achieve data transfer at the rate of 622 Mbps from an Earth ground station to Lunar Orbit during the Lunar Laser Communications Demonstration (LLCD) [17]. In the next few years NASA estimates it will be able to achieve bi-directional communications between ground stations and satellites at a data rate of 1.244 Gbps [16]. Another benefit of laser communication is the optical band is also unlicensed and unregulated by the Federal Communications Commission unlike the electromagnetic spectrum which is regulated and has limited unlicensed space.

There are also some disadvantages to using laser communications for Project DOTCOM’s communication network. The two biggest downfalls of laser communication are that line of sight is required to relay information and Earth’s atmosphere has a detrimental effect on the laser connection. First, line of sight is required for laser communications which makes the overall network architecture more complex. In other words, the number of satellites and ground stations in the system must be greater to ensure total coverage over a certain celestial body, compared to the total number of satellites and ground stations in a network that uses RF communication that ensures the same coverage. Second, atmospheric turbulence can lead to a degradation in the bit error rate (BER) and even a loss of the optical connection. The laser connection is also affected by absorption, scattering, and severe weather conditions. Lasers do not travel well through Earth’s atmosphere and if there is any type of adverse weather in the atmosphere, there is a chance that the data will not be able to be transferred between ground stations. Finally, with laser communication there is a need for a very narrow beam width to ensure high reliability of reception, which is another limitation of laser communication.

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meets threshold and has potential to meet objective requirements (DR 1.6, DR 1.7)</td>
<td>Line of sight required</td>
</tr>
<tr>
<td>Low power requirements</td>
<td>Costly tracking equipment</td>
</tr>
<tr>
<td>Optical spectrum unlicensed and unregulated</td>
<td>High atmospheric losses</td>
</tr>
</tbody>
</table>

Table 46: Laser Communications Pros and Cons

10.11.2 Trade Study

There were two different trade studies done to determine which type of communication would be best for interplanetary links and surface to constellation links. The trade studies conducted considered different aspects of the two communication types including Bandwidth and Frequency, Power Requirements, as well as Range and Coverage.

Bandwidth and Frequency

As data rates are of utmost importance in DOTCOM, bandwidth availability and frequency was a crucial criteria. With an increase in frequency, more information can be packed into a single transmission and larger bandwidths can support multiple signals to boost data rates even further. Additionally, the threshold requirement is 500 Mbps Lunar to Earth communication. As these are driving requirements, so significant weight must be associated with data rates. Therefore, the bandwidth and frequency was weighted the highest of the three criteria, with a weight of 5.

<table>
<thead>
<tr>
<th>Bandwidth and Frequency</th>
<th>Met data rate requirement</th>
<th>Met threshold requirements (DR 1.6)</th>
<th>Met objective requirements (DR 1.7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 47: Value assignments for Bandwidth and Frequency
Power Requirements

On any spacecraft, available power is limited and communications systems can be greedy when transmitting high energy signals. To achieve high frequency and wide bandwidth communications, power consumption needs to be monitored and the on board battery and solar power generation must be capable of supporting these needs. As DOTCOM is not limited to small satellites, power consumption can be handled at the expense of higher cost satellites with more power generation capability. Due to this, the weighting of the power requirements for RF and Laser is the lowest, with a weight of 2.

<table>
<thead>
<tr>
<th>Power Requirements</th>
<th>High</th>
<th>Medium</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 48: Value assignments for Power Requirements

Range and Coverage

DOTCOM focuses on both ground to satellite communications as well as interplanetary communications. With such long distances and the need to transmit through atmospheres, the range of the communication method is key. During ground to satellite communications interference is common, however the signal will still need to be intact at the receiving end in order to successfully communicate. Additionally, for interplanetary communications the transmitted signal must maintain its integrity across vast distances. Not meeting this requirement would invalidate the rest of the design regarding communications methods as it would be impossible for communication to occur. Due to this necessity, range and coverage was weighted at 4, just under the bandwidth and frequency criteria.

<table>
<thead>
<tr>
<th>Range and Coverage</th>
<th>Short Range (LEO)</th>
<th>Earth-to-space Communication</th>
<th>Earth-to-Moon Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td>1</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 49: Value assignments for Range and Coverage

Trade Study Evaluation

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>Evaluation</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>RF Coms</td>
<td>Laser Coms</td>
</tr>
<tr>
<td>Power Requirements</td>
<td>1</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>Range and Coverage</td>
<td>4</td>
<td>10</td>
<td>4</td>
</tr>
<tr>
<td>Bandwidth and Frequency</td>
<td>5</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Total Score</td>
<td></td>
<td>94</td>
<td>75</td>
</tr>
</tbody>
</table>

Table 50: Communications Method Trade Study Surface To Constellation

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
<th>Evaluation</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>RF Coms</td>
<td>Laser Coms</td>
</tr>
<tr>
<td>Power Requirements</td>
<td>1</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Range and Coverage</td>
<td>4</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>Bandwidth and Frequency</td>
<td>5</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>Total Score</td>
<td></td>
<td>53</td>
<td>82</td>
</tr>
</tbody>
</table>

Table 51: Communications Method Trade Study Interplanetary
The RF and laser communication methods were evaluated based on the pros and cons of each method. The evaluation values are out of 10, where the higher the score means that this criteria is more advantageous for that communication method. The score for Range and Coverage for RF communication is much higher than the laser communication score because laser communications requires direct line of sight and has high atmospheric attenuation. For the surface to constellation trade study the range of both RF and laser communications are essentially the same, however achieving total coverage with a laser communication system would require many more ground stations, more satellites in the constellation, and would cost much more that using RF communication, which is the RF evaluation is much higher. For the interplanetary trade study, the coverage aspect does not influence the evaluation of laser communication as much because the communication is only between two planets not between the constellation and the ground. Therefore, the Range and Coverage evaluation for laser communication is higher for the interplanetary trade study compared to the surface to constellation trade study. Based on these weights and evaluations, RF communication for shorter distances and laser communication for long distance data-relay will be most beneficial for Project DOTCOM’s communication network.