CLOSeSat: Perigee-Lowering Techniques and Preliminary Design for a Small Optical Imaging Satellite Operating in Very Low Earth Orbit

by

Jared K. Krueger
B.S., United States Air Force Academy (2008)

Submitted to the Department of Aeronautics and Astronautics in partial fulfillment of the requirements for the degree of Master of Science at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY May 2010

© Massachusetts Institute of Technology 2010. All rights reserved.

Author .............................................
Department of Aeronautics and Astronautics
May 21, 2010

Certified by .................................
Col (Ret.) John Keesee, USAF
Senior Lecturer, Department of Aeronautics and Astronautics
Thesis Supervisor

Accepted by .................................
Eytan H. Modiano, Ph.D.
Associate Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students
CLOSeSat: Perigee-Lowering Techniques and Preliminary Design for a Small Optical Imaging Satellite Operating in Very Low Earth Orbit

by

Jared K. Krueger

Submitted to the Department of Aeronautics and Astronautics on May 21, 2010, in partial fulfillment of the requirements for the degree of Master of Science

Abstract

The ever-increasing role of intelligence, surveillance, and reconnaissance (ISR) assets in combat may require relatively large numbers of earth observation spacecraft to maintain situational awareness. One way to reduce the cost of such systems is to operate at very low altitudes, thereby minimizing optics size and cost for a given ground resolution. This outside-the-box idea attempts to bridge the gap between high-altitude aerial reconnaissance platforms and traditional LEO satellites. Possible benefits from such a design include enabling a series of cheap, small satellites with improved optical resolution, greater resistance to adversary tracking, and 'quick strike' capability. In this thesis satellite systems design processes and tools are utilized to analyze advanced concepts of low perigee systems and reduce the useful perigee boundary of satellite orbits. The feasibility and utility of such designs are evaluated through the use of the Satellite System Design Tool (SSDT), an integrated approach using models and simulations in MATLAB and Satellite Tool Kit (STK). Finally a potential system design is suggested for a conceptual Continuous Low Orbit Surveillance Satellite (CLOSeSat). The proposed CLOSeSat design utilizes an advanced propulsion system and swooping maneuvers to improve survivability and extend lifetime at operational perigees as low as 160 kilometers, with sustained circular orbits at 240 kilometers.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

Thesis Supervisor: Col (Ret.) John Keesee, USAF
Title: Senior Lecturer, Department of Aeronautics and Astronautics
Acknowledgments

First things first: I extend my utmost gratitude to my research advisor, John Keesee, for guiding me through this lengthy process. I felt completely over my head when I first arrived at MIT and Lincoln Lab, but Colonel Keesee exhibited incredible patience and wisdom in helping me find my way. This may be the last thesis that he approves at MIT (he’s taking a new job next week), and I hope that he looks back on this experience as fondly as I do. On behalf of all of the young Air Force officers who have benefited from your professional example and guidance over the past several years - thank you!

Thanks are also due to the following, in no particular order:

- The incredibly intelligent and helpful staff at Lincoln Lab, and Group 95 in particular. John Kuconis, Jack Fleischman, Sue Andrews, Marshall Brenizer, Tony Hotz, Gene Rork, Jon Schoenberg, Ken MacCabe, Kat Canney, Robert Legge, and Sabino Pietrangelo; I was humbled by your expertise, and I am ever grateful for your assistance and support. Dave Gonzales deserves special recognition for his assistance as an unofficial advisor in the field of satellite drag modeling. Hopefully I will return to Lincoln Lab for a conference sometime soon!

- The Astronautical Engineering Department and the faculty at the U. S. Air Force Academy, for steering me toward MIT and preparing me well. Col. Martin France, Lt. Col. Mike Chapa, Lt. Col. Mike Bettner, Maj. Derek Varble, Capt. Luke Sauter, Dr. Rob Brown, Dr. Paul Vergez, Dr. Scott Dahlke, and all the rest; I cannot think of a better place to have spent my undergraduate career, and I cannot think of a finer group of instructors and role models!

- All of my classmates and instructors at MIT, but especially Matt Smith and Daniel Selva for their work on the CRISIS project, and all the lieutenants who have bonded together to “collaborate and graduate.” This of course includes my gracious roommates over the past two years: Kyle Volpe, Geoff Carrigan,
Chris Oravetz, Todd Small, Corey Crowell, and Will Deike; thanks for putting up with excessive Dave Matthews Band songs, and for teaching me how to cook. I feel so lucky to have had such great roommates and friends. It’s a small Air Force, so I’m sure I’ll see you all around!

- All the other organizations and activities that kept me sane in Boston. Maestro Adam Boyles and the MIT Symphony Orchestra, Tim Hawkins and my church community at Reunion/Sojourn, MIT SEED Academy, and the Boston Red Sox; there really is no excuse for being bored in this city.

- Finally, I would be lost were it not for my lovely family. Mom, Dad, Gammy, Bumpa, Tawnya and Jason, Justin and Ann-Morgan, and all the blond kids: thanks for keeping me grounded and loved! From the bottom of my heart, thanks for always praying for me and supporting me. You are the best family a man could have.

I must also thank my Lord and Savior, Jesus Christ, for creating this amazing universe and granting me the intelligence and opportunity to understand it just a little more clearly. He has brought me this far, and all the glory belongs to Him. (Psalm 8)
# Contents

1 Introduction .................................................. 17
   1.1 Motivation ............................................... 17
   1.2 Mission Statement ...................................... 21
   1.3 Overview ............................................... 21

2 Background Information and Literature Review .................. 23
   2.1 Potential Benefits of Operating in Very Low Earth Orbit (VLEO) .......... 23
      2.1.1 Improved Optical Resolution .......................... 23
      2.1.2 Improving Ratio of Received Energy-Per- Bit to Noise-Density $\left( \frac{E_\text{r}}{N_0} \right)$ .................................. 27
      2.1.3 Difficult to Track and Predict (Red Team) ................. 27
      2.1.4 Global Airspace Availability .......................... 28
   2.2 Review of Current Literature and Research ...................... 28
   2.3 Challenges of Operating in the Near Earth Environment .............. 30
      2.3.1 The Lower Thermospheric Environment .................. 30
      2.3.2 A Closer Look at Drag ................................ 32
      2.3.3 Effects on Orbits, Lifetime, and Coverage ................. 38
   2.4 Enabling Technologies .................................... 42
      2.4.1 Skipping Trajectories ................................ 42
      2.4.2 Variable Aerodynamic Properties ($\beta$) ................. 43
      2.4.3 Eccentric Orbits ..................................... 43
      2.4.4 Advanced Propulsion Techniques ....................... 43
      2.4.5 Precedents .......................................... 45
3 Analysis

3.1 Requirements ................................................. 47
  3.1.1 Concept of Operations ................................. 47
  3.1.2 System-Level Requirements ......................... 49

3.2 Development and Implementation of SSDT ............... 51
  3.2.1 Overall Framework of SSDT ......................... 51
  3.2.2 Subsystem Descriptions ................................ 54
  3.2.3 STK/MATLAB Interface ................................. 63

3.3 SSDT Modification for CLOSeSat Analysis ................. 68
  3.3.1 Inclusion of Swooping Maneuver Modeling .......... 69
  3.3.2 Inclusion of Drag Model ............................. 70

4 Results and Discussion ......................................... 73

4.1 Summary of Simulations ..................................... 73

4.2 Determining Minimum CLOSeSat Altitude for a Continuous Circular Orbit .................................................. 74
  4.2.1 Simulation A: Varying Altitude and Specific Impulse ... 74
  4.2.2 Simulation B: Varying Altitude and Specific Impulse in the Lower Thermosphere ................................. 76
  4.2.3 Simulation C: Finding the Minimum Sustained CLOSeSat Altitude .................................................. 78
  4.2.4 Simulation D: Varying Altitude and the Ground Resolution Requirement ........................................ 80
  4.2.5 Simulation E: Testing Optical Systems with Varying Orbits, Propulsion Systems, and Ground Resolution Requirements ........................................ 81
  4.2.6 Simulation F: Varying Altitude, Pointing Angle, Specific Impulse, and Number of Pixels ..................... 82
  4.2.7 Simulation G: Analyzing Constellations for Various Orbits .................................................. 84

4.3 Evaluation of Swooping Orbits ................................ 86
  4.3.1 Simulation H: Swooping Maneuver Capabilities at Various Perigees 86
4.4 Optimal CLOSeSat System Design ........................................ 88
  4.4.1 Comparison with Existing Technologies ......................... 90

5 Conclusions and Recommendations ......................................... 91
  5.1 Conclusions ........................................................... 91
  5.2 Summary of Contributions ........................................... 91
  5.3 Future Work .......................................................... 92
    5.3.1 Single Satellite Design → Constellation Design .............. 92
    5.3.2 Inclusion of Lifetime Prediction Function .................... 93
    5.3.3 Modeling of Skipping Orbits and Variable Drag Designs .... 93

A Additional Simulations ..................................................... 95
  A.1 Simulation i: Varying Altitude, Ground Resolution, and Specific Impulse 95
  A.2 Simulation ii: Testing Elliptic Orbits with Varying Altitude, Ground Resolution, and Specific Impulse 96
  A.3 Simulation iii: Swooping Maneuvers and Propulsion Systems from Various Parking Orbits ......................... 98
  A.4 Simulation iv: Incorporating the Drag Penalty ................. 100

B MATLAB Code ................................................................. 101
  B.1 SSDT Overview ......................................................... 101
    B.1.1 ‘sat_system.m’ .................................................. 101
    B.1.2 ‘sat_initialize.m’ .............................................. 104
    B.1.3 ‘sat_stk.m’ ...................................................... 112
    B.1.4 ‘sat_prop.m’ ..................................................... 121
    B.1.5 ‘sat_lifetime.m’ ............................................... 123
List of Figures

1-1 ATK/Goodrich ORS-1 Concept .................................................. 19

2-1 Composite image of Oakland, California, taken on August 23, 1994, with a spatial resolution of 3.0 meters per pixel ............................. 26

2-2 Composite image of Hartford, Connecticut, taken on May 1, 1999, with a spatial resolution of 0.5 meters per pixel ............................. 26

2-3 Visual Diagram of NASA’s TIMED mission studying the MLTI region and the overlapping CLOSeSat altitudes [2] ............................. 31

2-4 Atmospheric Densities from MSIS Thermospheric Model ............... 34

2-5 Best and Worst Case Drag Forces Encountered by Representative Small Satellites [20] ................................................................. 36

2-6 Estimated Orbit Lifetimes from 100 km to 500 km .......................... 39

2-7 Annual ΔV Required to Maintain Altitude from 100 km to 500 km .... 40

2-8 Required Percentage of Propellant Mass (Isp = 1500 s) to Maintain Altitude from 100 km to 500 km for One Year ........................... 40

2-9 GOCE’s Elegant Aerodynamic Design Minimizes Drag and Maximizes Solar Exposure at 250 km Altitude ............................................. 45

2-10 An Early Prototype of the NanoEye Camera .................................... 46

3-1 Overview of the Integrated CLOSeSat Model .................................. 52

3-2 Geometry of an Earth-Pointing Space-Based Imaging System .......... 57

3-3 Summary of System-Level Slewing Requirements and Constraints ...... 62

3-4 Typical Orbital Power Profile for Realtime Mode ............................ 63
<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-5</td>
<td>STK representation of Kabul mobile ground station and nominal CLOS-eSat orbit, as defined in ‘sat_mini_stk.m’</td>
</tr>
<tr>
<td>3-6</td>
<td>STK 3D visualization of sample CLOSeSat orbit with polar ground station and coverage grid, as defined in ‘sat_stk.m’</td>
</tr>
<tr>
<td>3-7</td>
<td>STK 3D visualization of CLOSeSat constellation in low orbits, as defined in ‘sat_stk.m’</td>
</tr>
<tr>
<td>3-8</td>
<td>STK ground track representation of CLOSeSat constellation, as defined in ‘sat_stk.m’</td>
</tr>
<tr>
<td>3-9</td>
<td>N² Diagram of SSDT interface displaying variable relationships throughout subfunctions</td>
</tr>
<tr>
<td>3-10</td>
<td>Atmospheric Densities with Power Function Curve Fit from 100 km to 1,000 km</td>
</tr>
<tr>
<td>4-1</td>
<td>Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation A</td>
</tr>
<tr>
<td>4-2</td>
<td>Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation B</td>
</tr>
<tr>
<td>4-3</td>
<td>Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation C</td>
</tr>
<tr>
<td>4-4</td>
<td>Aperture Diameter vs. Altitude for Simulation D</td>
</tr>
<tr>
<td>4-5</td>
<td>Mean Response Time vs. Altitude for Simulation E</td>
</tr>
<tr>
<td>4-6</td>
<td>Optical System Mass vs. Cost Per Image for Simulation F</td>
</tr>
<tr>
<td>4-7</td>
<td>Mean Response Time vs. Number of Images Per Target Per Day for Simulation G</td>
</tr>
<tr>
<td>4-8</td>
<td>Mean Response Time vs. Cost Per Image for Simulation G</td>
</tr>
<tr>
<td>4-9</td>
<td>Total Propulsion System Mass vs. Swoop Altitude for Simulation H</td>
</tr>
<tr>
<td>4-10</td>
<td>Total Propulsion System Mass vs. Swoop Altitude for Simulation H</td>
</tr>
<tr>
<td>A-1</td>
<td>Cost Per Image vs. Altitude for Simulation i</td>
</tr>
<tr>
<td>A-2</td>
<td>Mean Response Time vs. Altitude for Simulation i</td>
</tr>
<tr>
<td>A-3</td>
<td>Total Propulsion System Mass vs. Cost Per Image for Simulation ii</td>
</tr>
</tbody>
</table>
## List of Tables

1.1 Altitude Comparison of Standard and Conceptual Aerial and Space-Based Reconnaissance Platforms ........................................... 19

2.1 Diffraction-Limited Ground Resolution ............................................ 24

2.2 Relative Percentage Density Variations from Various Disturbance Effects 33

2.3 Atmospheric Densities (in kg/m$^3$) from MSIS Thermospheric Model . 33

2.4 Values of $C_d$ for a Sphere in Hyperthermal Flow with Diffuse Re-emission and Constant $\alpha$ .................................................. 35

2.5 Representative Performance and Operating Characteristics of Spacecraft Propulsion Systems .................................................. 37

2.6 Estimated Orbit Lifetime for Circular Orbits (In Days) ..................... 38

3.1 Summary of System-Level Requirements ....................................... 49

3.2 STK Constellation Analysis Results ................................................ 56

3.3 Comparison of Different Communications Architectures .................... 59

3.4 Summary of Slewing Requirements ................................................ 61

4.1 Summary of Parameters Varied in Each Simulation ............................ 73

4.2 Setup of Simulation A: Validating the SSDT by Varying Altitude and Specific Impulse ...................................................... 75

4.3 Setup of Simulation B: Varying Altitude and Specific Impulse in the Lower Thermosphere ...................................................... 76

4.4 Setup of Simulation C: Varying Altitude, Specific Impulse, and Design Lifetime ................................................................. 78
4.5 Setup of Simulation D: Varying Altitude and the Ground Resolution Requirement .................................................. 80
4.6 Setup of Simulation E: Varying Altitude, Ground Resolution, Specific Impulse, CCD Size, and Eccentricity ................................................................. 81
4.7 Setup of Simulation F: Varying Altitude, Pointing Angle, Specific Impulse, and Number of Pixels ................................................................. 83
4.8 Setup of Simulation G: Varying Altitude, Eccentricity, and Constellation Size ................................................................. 84
4.9 Setup of Simulation H: Varying Swoop Altitude (Perigee), Lifetime, and Frequency ................................................................. 86
4.10 Design Vector for Optimal CLOSeSat Swooping Configuration ................................................................. 88
4.11 Performance and Cost for Optimal CLOSeSat Swooping Configuration ................................................................. 89

A.1 Setup of Simulation i: Varying Altitude, Ground Resolution, and Specific Impulse ................................................................. 95
A.2 Setup of Simulation ii: Varying Altitude, Ground Resolution, Specific Impulse, and Eccentricity ................................................................. 97
A.3 Setup of Simulation iii: Varying Altitude, Specific Impulse, and Eccentricity ................................................................. 98
A.4 Setup of Simulation iv: Varying Altitude, Ballistic Coefficient, and Lifetime ................................................................. 100
Chapter 1

Introduction

1.1 Motivation

The concept of operationally responsive space (ORS) has recently become a point of emphasis in the intelligence and space communities. The ORS concept has been defined as "an affordable capability to promptly, accurately, and decisively position and operate national and military assets in and through space and near space... [whose] vision is to provide rapid, tailorable space power focused at the operational and tactical level of war." In April 2007 the National Space Security Office published its Plan for Operationally Responsive Space which sought to "create a joint ORS Office to provide assured space power focused on timely satisfaction of Joint Force Commanders' needs and the needs of other users"[14]. This concerted effort has resulted in an influx of proposed intelligence, surveillance, and reconnaissance (ISR) assets to the mission. Each of these assets strives to provide unique functionality and relevance to the user.

The National Security Space Office lists three tiers of classification for developing and operating technologies:

- Tier-1: On-orbit (current assets leveraged and prioritized for war fighter)
- Tier-2: ORS assets in ready reserve; ready for launch and deployment
- Tier-3: New assets rapidly acquired to meet specific COCOM/User needs
This study seeks to align itself within these tiers of activity. By stressing simplicity and flexibility, such a system would start at Tier-3 as a potential transformational technology in development. One of the primary advantages of small, cheap satellites is the accelerated timetable in which they can be designed and produced. The satellite design, build, and test process could ideally be condensed to a matter of months, moving these new satellites into Tier-2 as ready-made assets awaiting launch. The launch schedule is often the limiting factor in moving from Tier-2 to Tier-1 (on-orbit assets), but flexible launch platforms such as SpaceX's Falcon1 and Orbital Science's Pegasus XL also show promise in accelerating this phase. The goal for this project is to focus on Tier-3, developing an exciting new concept, with the eventual target of moving to Tier-1 for immediate assistance to the war fighter.

At a recent Southwest Asia Theater Space Conference, Army Major Todd Fenner commented that the goal of ORS is to “provide a 3-D visualization of the battle space to our coalition partners... Space is all about completing that visual picture, whether it’s the enemy, the terrain, the time or the target, the more complete the visualization we can provide for our coalition forces, the more effective they can be” [21]. Such a goal can be realized by a combination or air, space, and ground assets, the composition of which is determined by the required response time, hazards of the terrain and airspace, and availability of nearby ISR platforms. This research does not aim to analyze the greater ORS-ISR network as a whole, but rather to focus on one possible element of the bigger picture by means of a previously unconsidered design: very low-orbiting optical satellites.

Table 1.1 illustrates how such a design (tentatively referred to here as “Continuous Low Orbit Surveillance Satellite,” or CLOSeSat) might “fill the gap” between the flight regimes of current aerial and space-based reconnaissance platforms and future concepts.

The Operationally Responsive Space Office is currently developing an operational proof of concept for a rapidly deployable reconnaissance satellite called simply ORS-1. Tasked primarily with providing time-critical multi-spectral electro-optical and infrared images to U.S. Central Command (CENTCOM) leaders, ORS-1’s distin-
Table 1.1: Altitude Comparison of Standard and Conceptual Aerial and Space-Based Reconnaissance Platforms

<table>
<thead>
<tr>
<th>Reconnaissance Platform</th>
<th>Operating Regime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predator/UAV</td>
<td>18,000–29,000 ft</td>
</tr>
<tr>
<td>Global Hawk/LM High-Altitude Airship</td>
<td>65,000 ft</td>
</tr>
<tr>
<td>APL HARVe Concept</td>
<td>65,000–100,000 ft</td>
</tr>
<tr>
<td>U-2/SR-71</td>
<td>70,000–85,000 ft</td>
</tr>
<tr>
<td>High-Altitude Balloons</td>
<td>up to 164,000 ft</td>
</tr>
<tr>
<td>CLOSeSat</td>
<td><strong>328,000–820,000 ft</strong></td>
</tr>
<tr>
<td>Traditional LEO Sat</td>
<td>≥ 984,000 ft</td>
</tr>
</tbody>
</table>

The distinguishing characteristic is its responsiveness; the goal is to launch within 24 months of approval to start development. This lofty vision is enabled by a streamlined acquisitions process, utilization of commercial off-the-shelf (COTS) hardware (such as a modified version of the U-2 aircraft’s electro-optical/infrared sensor payload), and high prioritization within the Pentagon’s budget.

Figure 1-1: ATK/Goodrich ORS-1 Concept

Assuming ORS-1 remains within its timeline and mass and cost budgets, it will “contribute to the growing echelon of systems collecting information in CENTCOM” by providing tactical images directly to theater commanders multiple times per day. If the future of space is faster, cheaper, smaller, and more responsive, ORS-1 is a step in the right direction, but there is still much progress to be made. ORS-1 weighs 450
kilograms and has a pricetag of $215 million, while taking two years to go from the
drawing board to orbit. CLOSeSat seeks to accelerate this progress with ambitious
goals of reducing each of these metrics by 50 percent (i.e., similar performance at half
the cost, half the mass, and half the development time).

The operational need for more responsive spacecraft like ORS-1 is demonstrated
by the increasing requests of in-theater ISR “on demand.” The U.S. Air Force is
struggling to keep up with the growing demand for ISR support [7]. Unmanned aerial
vehicles, for example, have flown over 600,000 hours over Iraq and Afghanistan, and
the UAV fleet has grown by 400 percent since 2007 [17]. Asymmetric warfare in a
heavily populated urban environment creates an insatiable demand for updated in-
telligence. General James Conway, Commandant of the United States Marine Corps,
recently observed that the military’s demand for “intelligence, surveillance and re-
connaissance (ISR) assets... has become a sickness, you can’t get enough of it, the
appetite is unquenchable and it is very, very expensive... I’m not sure we’re using
it the right way entirely” [15]. For reasons to be developed in later chapters, one
potential solution involves operating cheap, small satellites at altitudes lower than
those traditionally considered for LEO orbits.

This study will attempt to answer a number of questions, such as:

- What is the lowest viable altitude for an optical imaging micro- or minisatellite
to survive and produce useful information?

- What might be a preliminary system-level design of such a satellite?

- Is such a design feasible in the near future given the inherent challenges of
operating at such an altitude?

- Which emerging technologies might enable such a design to become a reality in
the near future?
1.2 Mission Statement

The purpose of this study/thesis is to analyze advanced concepts of satellite systems design and operation that push the useful perigee boundary of satellite orbits. The continuous need for prompt intelligence and reconnaissance information in combat, and the corresponding high cost and lengthy development track of current systems, compels forward-thinking ideas to support the war fighter. One such outside-the-box idea is the use of optical satellites in Very Low Earth Orbits (VLEO), bridging the gap between high-altitude aerial reconnaissance platforms and traditional LEO satellites. For a given ground resolution requirement, a lower perigee and operational altitude shrinks the necessary optical payload and satellite bus, thus enabling a series of cheaper, smaller satellites. For a given optical payload, lowering perigee and the distance to its ground target results in improved optical resolution and an enhanced image. Additionally, a constellation of such spacecraft at very low altitudes would provide global coverage with fast response times and ‘quick strike’ capability for any conflict.

1.3 Overview

The remainder of this thesis is divided into four chapters, followed by appendices and references. Chapter 2 discusses the background information pertinent to the thesis. This includes a closer look at the lower thermospheric environment (100–220 km) and the corresponding effects, both beneficial and challenging, of operating at such low altitudes. Chapter 2 also briefly introduces each of the enabling technologies to be considered in the study. Chapter 3 lists the driving requirements, constraints, and concept of operations governing the design problem. It goes on to outline the development and implementation of the Satellite System Design Tool (SSDT), which is the primary analysis tool used. Chapter 4 presents eight SSDT simulations and compares these results with existing technologies and capabilities. Also included in this chapter is a preliminary conceptual design. Chapter 5 elaborates on the conclusions
developed from the results section, summarizes the study’s overall contributions, and suggests possible future work to be done in the field of small VLEO satellites, such as constellation design.
Chapter 2

Background Information and Literature Review

2.1 Potential Benefits of Operating in Very Low Earth Orbit (VLEO)

2.1.1 Improved Optical Resolution

Because space objects orbit over all portions of the earth without regard for political boundaries, earth-orbiting satellites have been used to observe situations on the ground for the last half century. One benefit of operating earth-imaging satellites at very low altitudes is the resulting improved optical resolution for a given payload, or smaller optic system dimensions for equivalent resolution. This makes intuitive sense as the spacecraft will be closer to its intended target, but it is also supported by physics. The Rayleigh diffraction-limited ground resolution of an optical imaging satellite is determined by the following equation:

\[ X' = \frac{2.44 \lambda h}{D} \]  

(2.1)

where \( X' \) is the ground resolution, \( h \) is the altitude above the Earth, \( \lambda \) is the wavelength being viewed, and \( D \) is the aperture diameter of the optical instrument. This
equation demonstrates that *ground resolution is directly proportional to altitude*, and therefore a reduction in perigee would improve resolution by the same degree of change. For example, based on the above equation a simple optical system with an aperture diameter of one meter, observing visible light at a wavelength of 500 nm and an altitude of 900 km produces an image with a diffraction-limited ground resolution of 1.098 m. The same payload, when lowered to an altitude of 200 km, improves its resolution to 0.244 m; a 75% improvement! Table 2.1, adapted from SMAD (Table 9-9), further summarizes this concept of diffraction-limited resolution for typical ISR systems:

<table>
<thead>
<tr>
<th>Aperture Size, D</th>
<th>Visible [λ = 0.5μm]</th>
<th>IR [λ = 3μm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>From an orbiting spacecraft at h = 900 km</td>
<td>1 m</td>
<td>1.1 m</td>
</tr>
<tr>
<td>From an orbiting spacecraft at h = 200 km</td>
<td>1 m</td>
<td>0.244 m</td>
</tr>
<tr>
<td>From a synchronous spacecraft (GEO) (h = 35,800 km)</td>
<td>1 m</td>
<td>0.0813 m</td>
</tr>
<tr>
<td>From SR-71 at h = 20 km (70,000 ft)</td>
<td>0.3 m</td>
<td>0.081 m</td>
</tr>
</tbody>
</table>

This table describes the obvious but intriguing allure of placing optical imaging satellites in very low orbits: a satellite with a 3-meter aperture, orbiting at an altitude of 200 km is theoretically capable of producing identical images as those taken by a SR-71 aircraft at 70,000 feet. Although such a huge telescope would not meet the ORS and CLOSeSat goals of ‘smaller’ and ‘cheaper’ - the primary mirror of the Hubble Space Telescope is only 2.4 meters in diameter - it exemplifies the potential for increased resolution at decreased altitude. The tangible advantages of improved resolution are of immediate interest to military commanders. Whereas an image with three-meter resolution may be sufficient for locating specific buildings or aircraft, images with better than one-meter resolution improve situational awareness by
pinpointing specific vehicles. Figures 2-1 and 2-2, obtained from the SUNY College at Oneonta high resolution imagery project, highlight this improvement in resolution. Whereas the first image shows a baseball stadium and airplanes on the tarmac, the second image details individual vehicles, demonstrating why a 1.0 meter ground resolution is a good starting assumption for a baseline requirement (see Table 3.1).

Alternatively, historical evidence suggests that the cost of space-borne imaging systems is proportional to the aperture diameter squared (\(\text{Cost} \propto D^2\)) [19]. This relationship further exacerbates the cost savings available by utilizing smaller mirrors at lower altitudes to generate the same resolution. For example, using the example from Table 2.1 and a 1.0 meter resolution requirement, moving from a 900 km orbit to a 200 km orbit would result in an optical system costing just 5% of its original cost \([(200/900)^2]\). Together, the potential for improved resolution and/or optical system cost savings provide a compelling incentive for low altitude observation.
Figure 2-1: Composite image of Oakland, California, taken on August 23, 1994, with a spatial resolution of 3.0 meters per pixel

Figure 2-2: Composite image of Hartford, Connecticut, taken on May 1, 1999, with a spatial resolution of 0.5 meters per pixel
2.1.2 Improving Ratio of Received Energy-Per-Bit to Noise-Density ($\frac{E_b}{N_0}$)

A secondary benefit of operating at lower altitudes is the subsequent improvement in the ratio of received energy-per-bit to noise-density, or $\frac{E_b}{N_0}$. This term is useful for calculating the strength of a digital data link, and is related to the signal-to-noise ratio (SNR) often used in determining analog communications. The basic link equation used for sizing a digital data signal is:

$$\frac{E_b}{N_0} = \frac{P L_t G_t L_s L_a G_r}{k T_s R},$$  \hspace{1cm} (2.2)

where $P$ is the transmitter power, $L_t$ is the transmission-to-antenna line loss, $G_t$ is the transmit antenna gain, $L_s$ is the space loss, $L_a$ is transmission path loss, $G_r$ is the receive antenna gain, $k$ is Boltzmann’s constant, $T_s$ is the system noise temperature, and $R$ is the data rate. Generally a $\frac{E_b}{N_0}$ ratio of between 5 and 10 is sufficient for sending and receiving digital data with a low probability of error with some forward correction, as shown in SMAD Figure 13-9 [20, p. 561]. Both the transmitter power and antenna size depend on the range from the satellite to the ground station. Therefore, closing this gap by lowering altitude will result in direct savings in the spacecraft’s power consumption, mass budget, and sizing demands.

2.1.3 Difficult to Track and Predict (Red Team)

While conducting ISR missions from the space domain trumps some ground-based electronic warfare attempts, satellites face their own suite of threats: an evolving adversary may have access to anti-satellite weapons (ASAT) and in-orbit signal jammers/interceptors. International controversy over the weaponization of space aside, new technologies should be prepared to mitigate and defend against such attacks. Here the usefulness of a very low, very fast orbit is seen through the lens of a possible real world threat. The CLOSeSat orbit would be especially difficult to determine and predict due to the fluctuating atmospheric conditions at such low altitudes. This
makes CLOSeSat a formidable target for enemy eyes hoping to interfere with its mission.

2.1.4 Global Airspace Availability

One hazard of using airborne assets to fly missions over contested airspace is the risk of surface-to-air missile intercept. In contrast, space-based assets do not incur such a risk because their orbits are at high enough altitudes to avoid conventional surface-to-air missiles (SAMs) or any applicable “no fly zones.” Similarly, there is no such international law or treaty governing the flight of adversary assets above the Kármán Line (the unofficial line at about 100 km altitude demarcating the edge of space and the Earth’s atmosphere). As first realized in the flight of Sputnik 1, “while airplane overflight was clearly considered an intrusion on a nation’s sovereignty, spaceflight was not so clearly defined” [9, p. 35]. Thus one of the critical advantages of utilizing space-based assets, such as CLOSeSat, over a traditional airborne reconnaissance system is political in nature. Due in part to the legal precedent set by Sputnik 1, there are no international regulations restricting LEO access to airspace over specific areas of interest. (Such guidelines do exist at GEO, where space is at a premium and spacecraft are allocated specific “parking spots” over one spot on the globe.)

2.2 Review of Current Literature and Research

A recent preliminary trade study comparing low-flying circular and elliptical orbits was conducted by the French Centre National d’Etudes Spatiales and published at an IEEE Geospace and Remote Sensing Symposium. This study serves as a welcome introductory discussion to the topic of systems-level trade studies in optical imaging satellites at LEO. Ultimately the French scientists conclude:

Low flying enables higher resolution imaging capabilities within a given instrument and satellite format, but at a price of higher atmospheric drag and propulsion constraints if we keep with a circular orbit... Unlike circu-
lar orbits, elliptic orbits allow for flights at low altitude without too much erosion due to atmospheric drag...[especially in the] emerging needs of ‘theater observation’ (high resolution and revisit on a limited region) [1].

Taking this idea one step further, the researchers suggest that a dynamic orbit could be used. Rather than remain in a constant, degrading circular or elliptical orbit, designers could exploit the advantages of both geometries.

“Another approach is to use a parking or routine orbit at higher altitude (and low erosion) and to ‘plunge’ to the theater on request” [1].

This “swooping maneuver” will be analyzed in more detail in a later chapter as a primary candidate for enabling very low perigees. The French researchers provide one suggested solution:

“With the ‘wait and plunge’ approach we can have:

- a global sun-synchronous routine on 260-by-8500 km orbit with low ΔV cost and better resolution than 400 km circular orbit but with half the revisit or half coverage capabilities

- at least 3 plunges (requiring a ΔV of 60 m/s each) to the theater orbit 200-by-8500 km over the satellite lifetime (assuming a baseline chemical propulsion system with a ΔV of 200 m/s)” [1].

Ultimately it is concluded that “highly elliptic orbits offer clear advantages with respect to circular only for observation of theaters (within a predefined 30° latitude window):

- Enables a 300% gain in [optical] resolution at equal revisit time (one per day per spacecraft) on theater

- Enables orbit changes for theater displacements with smaller ΔV and shorter delays

- Provides a minimum demand on chemical propulsion
• Avoids permanent V-plane configuration and oversizing of solar arrays” [since arrays may be sun-tracking at higher altitudes with no drag penalty] [1].

This idea sounds promising from the perspective of a basic system-level trade study, but there is a reason such designs have not been implemented frequently throughout history: the high-drag environment of the intended perigee altitude poses a daunting challenge for small spacecraft.

2.3 Challenges of Operating in the Near Earth Environment

2.3.1 The Lower Thermospheric Environment

The target altitudes for this project fall on the fringe of the Earth’s atmosphere, also known as the lower thermosphere (defined here as the 100–300 km range). Sometimes more broadly referred to as the mesosphere and lower thermosphere/ionosphere (MLTI), this region is a dynamic transitional environment that is notoriously difficult to predict and navigate. Atmospheric science experts admit that “data from the lower thermosphere are sparse, since neither balloons nor most rockets reach lower thermospheric altitudes, and the high atmospheric density at these heights imposes short lifetimes on satellites” [4]. In addition to the lack of data, the environment is highly variable, being strongly influenced by a number of external sources. These sources include “forcing by wave activity penetrating upward from the lower atmosphere, solar EUV and UV radiation, auroral and energetic particle precipitation, and magnetospheric plasma convection,” and the resulting coupled effects of heating, dissociation, and ionization [8]. In addition to such natural processes, anthropogenic effects from increased emissions of chemicals like CO₂ and water vapor from launch vehicles may be responsible for the increasing number of noctilucent clouds at the summer polar mesopause - the coldest spot in the atmosphere. [8]

In an attempt to better understand the MLTI region, NASA launched its Thermosphere Ionosphere Mesosphere Energetics and Dynamics (TIMED) mission on De-
The TIMED mission includes an orbiting spacecraft at 625 km altitude that directs its suite of remote-sensing instruments at the MLTI region, focusing especially on the atmosphere from 60–180 km altitude. Figure 2-3 depicts the TIMED mission with the overlapping CLOSeSat target altitudes shown in red. TIMED data have resulted in the publication of over 500 scientific papers and a more complete understanding of the solar, geomagnetic, and atmospheric forces that affect satellites in the lower thermosphere. For example, using the Solar EUV Experiment (SEE) aboard TIMED, NASA scientists have measured solar irradiance values in the MLTI region throughout the solar cycle. Additionally, SEE has observed over 200 solar flares and measured the resulting irradiance variability over the full spectrum of EUV and XUV ranges [2]. Such solar activity represents just one of many factors that determine the drag environment, further discussed in Subsection 2.3.2. TIMED has produced a wealth of valuable information that enhances the atmospheric models in the lower thermospheric region, a critical area at the lower bound of the CLOSeSat orbit.

![Visual Diagram of NASA’s TIMED mission studying the MLTI region and the overlapping CLOSeSat altitudes](image-url)
2.3.2 A Closer Look at Drag

Atmospheric drag is by far the largest environmental influence on small satellites in very low orbits. The drag force \( D \) acting on an object is defined by the following classical equation:

\[
D = -\frac{1}{2} \rho V^2 SC_D,
\]  

(2.3)

where \( \rho \) is the atmospheric density, \( V \) is the relative velocity of the spacecraft, \( S \) is the cross-sectional area of the spacecraft perpendicular to the direction of motion, and \( C_d \) is the non-dimensional drag coefficient. Dividing by the spacecraft mass, \( m \), gives us the equation for acceleration due to drag:

\[
a_D = -\frac{1}{2} \rho V^2 \frac{SC_D}{m}.
\]  

(2.4)

In this equation the first variable, \( \rho \), is notoriously difficult to predict, making it “the dominant uncertainty in determining drag acceleration” [13]. Numerous attempts have been made to accurately model atmospheric density, with two empirical models gaining prominence: first the Jacchia (now Jacchia-Bowman) model in 1964, followed by the Mass Spectrometer Incoherent Scatter (MSIS) model in 1977. The most recent iterations of both models are still used operationally today, but are often found to have persisting statistical errors of about 15% [13]. The primary source of these errors is the high variability of the atmospheric structure in low Earth orbit, due to the dynamic environment previously described. Table 2.2, from Table 1 in the Air Force Research Lab publication “Towards a Golden Age of Satellite Drag” [13, p. 2], summarizes the major disturbances that drive atmospheric density variations at representative altitudes of 200 and 400 km.

Table 2.2 describes the formidable challenge of modeling atmospheric density and accurately predicting satellite drag, but it also provides a source of hope for this study: density variations appear to be less drastic at lower altitudes. Indeed at 150 km “density is not strongly affected by solar activity,” while at higher, more typical LEO satellite altitudes of 500 to 800 km, “the density variations between solar maximum
Table 2.2: Relative Percentage Density Variations from Various Disturbance Effects

<table>
<thead>
<tr>
<th>Effect</th>
<th>200 km</th>
<th>400 km</th>
<th>Time Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flux (Solar Cycle)</td>
<td>110</td>
<td>1165</td>
<td>Years</td>
</tr>
<tr>
<td>Flux (Daily)</td>
<td>1</td>
<td>5</td>
<td>Day</td>
</tr>
<tr>
<td>Magnetic Activity</td>
<td>35</td>
<td>60</td>
<td>Hours</td>
</tr>
<tr>
<td>Local Time</td>
<td>25</td>
<td>115</td>
<td>Day</td>
</tr>
<tr>
<td>Semiannual</td>
<td>15</td>
<td>50</td>
<td>Months</td>
</tr>
<tr>
<td>Latitude</td>
<td>15</td>
<td>60</td>
<td>Months</td>
</tr>
<tr>
<td>Longitude</td>
<td>2</td>
<td>5</td>
<td>Day</td>
</tr>
</tbody>
</table>

and solar minimum are approximately two orders of magnitude” [20, p. 208]. The variation of density with altitude is tabulated in Table 2.3 and shown graphically in Figure 2-4, where the gap between minimum and maximum density variations peaks around 600 km. While these conclusions do little to solve the satellite drag problem at very low altitudes, they do imply that variations in density are somewhat less of a concern.

Table 2.3: Atmospheric Densities (in kg/m³) from MSIS Thermospheric Model

<table>
<thead>
<tr>
<th>Altitude (km)</th>
<th>Minimum</th>
<th>Mean</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>100</td>
<td>4.61E-07</td>
<td>4.79E-07</td>
<td>5.10E-07</td>
</tr>
<tr>
<td>150</td>
<td>1.65E-09</td>
<td>1.81E-09</td>
<td>2.04E-09</td>
</tr>
<tr>
<td>200</td>
<td>1.78E-10</td>
<td>2.53E-10</td>
<td>3.52E-10</td>
</tr>
<tr>
<td>250</td>
<td>3.35E-11</td>
<td>6.24E-11</td>
<td>1.06E-10</td>
</tr>
<tr>
<td>300</td>
<td>8.19E-12</td>
<td>1.95E-11</td>
<td>3.96E-11</td>
</tr>
<tr>
<td>350</td>
<td>2.34E-12</td>
<td>6.98E-12</td>
<td>1.06E-10</td>
</tr>
<tr>
<td>400</td>
<td>7.32E-13</td>
<td>2.72E-12</td>
<td>7.55E-12</td>
</tr>
<tr>
<td>450</td>
<td>2.47E-13</td>
<td>1.13E-12</td>
<td>3.61E-12</td>
</tr>
<tr>
<td>500</td>
<td>8.98E-14</td>
<td>4.89E-13</td>
<td>1.80E-12</td>
</tr>
<tr>
<td>550</td>
<td>3.63E-14</td>
<td>2.21E-13</td>
<td>9.25E-13</td>
</tr>
<tr>
<td>600</td>
<td>1.68E-14</td>
<td>1.04E-13</td>
<td>4.89E-13</td>
</tr>
<tr>
<td>650</td>
<td>9.14E-15</td>
<td>5.15E-14</td>
<td>2.64E-13</td>
</tr>
<tr>
<td>700</td>
<td>5.74E-15</td>
<td>2.72E-14</td>
<td>1.47E-13</td>
</tr>
<tr>
<td>750</td>
<td>3.99E-15</td>
<td>1.55E-14</td>
<td>8.37E-14</td>
</tr>
<tr>
<td>800</td>
<td>2.96E-15</td>
<td>9.63E-15</td>
<td>4.39E-14</td>
</tr>
<tr>
<td>850</td>
<td>2.28E-15</td>
<td>6.47E-15</td>
<td>3.00E-14</td>
</tr>
<tr>
<td>900</td>
<td>1.80E-15</td>
<td>4.66E-15</td>
<td>1.91E-14</td>
</tr>
<tr>
<td>950</td>
<td>1.44E-15</td>
<td>3.54E-15</td>
<td>1.27E-14</td>
</tr>
<tr>
<td>1000</td>
<td>1.17E-15</td>
<td>2.79E-15</td>
<td>8.84E-15</td>
</tr>
</tbody>
</table>
Another term in Equation 2.4 is the spacecraft’s ballistic coefficient, $\beta$, which measures its ability to overcome air resistance. It is defined by the equation:

$$\beta = \frac{m}{C_D S}.$$  \hspace{1cm} (2.5)

The process for calculation of $C_d$ for a given satellite involves the satellite’s shape, material, orientation, altitude, and atmospheric composition. In hyperthermal flow, when the spacecraft velocity greatly exceeds that of the free molecules, the general equation for the drag coefficient of a spacecraft with its cross-sectional area perpendicular to the direction of motion may be written as:

$$C_d = 2 + W,$$  \hspace{1cm} (2.6)

where $W$ is dependent on the satellite’s shape, its accommodation coefficient, $\alpha$ (which describes the re-emission of energy associated with molecular collisions at the satellite’s surface), and the mode of reflection [11]. Table 2.4 lists drag coefficients for a sphere in hyperthermal flow (when the satellite speed greatly exceeds the mean molecular speed), with diffuse re-emission and constant accommodation coefficient.
over its surface:

Table 2.4: Values of $C_d$ for a Sphere in Hyperthermal Flow with Diffuse Re-emission and Constant $\alpha$

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>0.6</th>
<th>0.8</th>
<th>0.9</th>
<th>0.95</th>
<th>0.98</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_d$</td>
<td>2.56</td>
<td>2.40</td>
<td>2.28</td>
<td>2.21</td>
<td>2.14</td>
<td>2.07</td>
</tr>
</tbody>
</table>

It is generally accepted to use an average value of $C_d = 2.2$ to represent the drag coefficient of small spherical satellites and those with rotating cylindrical or convex bodies [11]. However, numerical Monte Carlo algorithms enable the user to better model both the satellite’s drag coefficient as well as its exposed surface area at any given time. Such modeling efforts lead to a more accurate estimate of the satellite’s drag perturbation force, and therefore a more accurate orbit propagation model. One additional complication to the theory actually improves the satellite’s lifetime at very low altitudes. The majority of an average LEO satellite’s lifetime is spent in “freemolecule flow,” which is a simplified approximation that ignores all molecule-molecule collisions occurring within one mean free path of the satellite. This includes collisions between incident and re-emitted molecules. Free-molecule flow is a good assumption when the mean free path of the molecules, $\lambda$, is much greater than the characteristic length of the satellite body, $l$. This ratio, referred to as the Knudsen number, $K$, is represented by the equation:

$$K = \frac{\lambda}{l}.$$  \hspace{1cm} (2.7)

Free-molecule flow typically applies when the value of $K$ is greater than about 10.

As the satellite descends into the “transition region” where molecule-molecule collisions are important (below about 200 km), the assumptions of free-molecule flow no longer apply. Thus values for $C_d$ are typically smaller in the transition region, compared to free-molecule values. King-Hele concludes “that most satellites during their last few revolutions in orbit, or the last few days for some dense or eccentric satellites, experience a considerable decrease in drag coefficient. This lengthens the lifetime for most satellites by one or two revolutions beyond the date given by the

\[\text{This is not the same } \lambda \text{ discussed previously in this chapter, which represented the wavelength detected by an optical instrument.}\]
theory, which assumes a constant drag coefficient” [11].

One possible way to overcome drag in low-Earth orbit is the use of a propulsion system to periodically or continually restore the energy lost to drag and boost the satellite back into its original orbit, thereby achieving a longer working lifetime [20]. The primary parameters for characterizing a propulsion system are specific impulse ($I_{sp}$) and thrust ($T$), discussed in greater detail in Subsection 2.4.4. Figure 2-5 shows typical drag forces encountered by small satellites at various altitudes, while Table 2.5 lists specific impulses and thrust outputs of common propulsion systems used for altitude maintenance. The Viking satellite is a small, octagonal, Swedish plasma-measuring satellite (277 kg, maximum $\beta = 128$ kg/m², minimum $\beta = 30.8$ kg/m²), while the seventh Orbiting Solar Observatory (OSO-7) is a 9-sided solar physics satellite (634 kg, maximum $\beta = 437$ kg/m², minimum $\beta = 165$ kg/m²) [20]. The “best case” data use maximum ballistic coefficient (minimum drag coefficient) and minimum atmospheric density, while the “worst case” lines are derived from minimum ballistic coefficient (maximum drag coefficient) and maximum atmospheric density.

![Figure 2-5: Best and Worst Case Drag Forces Encountered by Representative Small Satellites [20]](image)

Based on these worst-case results, continuous operation at 200 km altitude requires
Table 2.5: Representative Performance and Operating Characteristics of Spacecraft Propulsion Systems

<table>
<thead>
<tr>
<th>Engine</th>
<th>Company</th>
<th>Type</th>
<th>Thrust (N)</th>
<th>$I_{sp}$ (s)</th>
<th>Mass (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Star 13B</td>
<td>Thiokol</td>
<td>Solid</td>
<td>7,015</td>
<td>285.7</td>
<td>47</td>
</tr>
<tr>
<td>XLR-132</td>
<td>Rocketdyne</td>
<td>Liquid</td>
<td>16,700</td>
<td>340</td>
<td>51.26</td>
</tr>
<tr>
<td>R-40A</td>
<td>Marquadt</td>
<td>Liquid</td>
<td>4,000</td>
<td>309</td>
<td>7.26</td>
</tr>
<tr>
<td>DM/LAE</td>
<td>TRW</td>
<td>Liquid</td>
<td>445</td>
<td>315</td>
<td>4.54</td>
</tr>
<tr>
<td>MR-501B</td>
<td>Aerojet</td>
<td>Resistojet</td>
<td>0.37</td>
<td>303</td>
<td>0.89</td>
</tr>
<tr>
<td>MR-510</td>
<td>Aerojet</td>
<td>Arcjet</td>
<td>0.25</td>
<td>600</td>
<td>1.58</td>
</tr>
<tr>
<td>PRS-101</td>
<td>Aerojet</td>
<td>Pulsed Plasma</td>
<td>0.0012</td>
<td>1,350</td>
<td>4.74</td>
</tr>
<tr>
<td>HPHS SPT-140</td>
<td>Atlantic/Fakel</td>
<td>Hall Effect</td>
<td>0.29</td>
<td>1,770</td>
<td>6.8</td>
</tr>
<tr>
<td>NSTAR</td>
<td>Hughes</td>
<td>Ion</td>
<td>0.092</td>
<td>3,100</td>
<td>8</td>
</tr>
</tbody>
</table>

Continuous thrust of 0.1 N to counteract the drag force. Using equation 2.8, chemical propulsion systems ($I_{sp} \approx 300$ s) would burn fuel at a rate of 2.94 kg per day per kg of spacecraft weight to maintain that level of thrust. This means a 100 kg-class satellite would require over 2,000 kg of propellant just to extend its lifetime one week at this orbit! In the same scenario electric propulsion systems ($I_{sp} \approx 1,500$ s) would require a mass flow rate of only 0.59 kg per day per kg of spacecraft weight, or 411 kg of propellant per week for a 100 kg spacecraft. Such large propellant masses quickly limit the value gained by operating in VLEO. Alternatively, as we will see, the orbit maintenance demands on the propulsion system are significantly less if only a portion of the orbit takes place at such low altitudes.
2.3.3 Effects on Orbits, Lifetime, and Coverage

Lowering the perigee has a significant effect on numerous characteristics of a satellite’s orbit. Predicted satellite lifetime in particular decreases exponentially in relation to altitude. For example, given a constant ballistic coefficient of 50 kg/m\(^2\), a satellite will operate for 10.06 days during solar minimum at 250 km, but only 1.65 days at 200 km [20]. Table 2.6, taken from data published in SMAD and produced using the software package SatLife, provides a more complete picture of this problem. SatLife is an orbit lifetime prediction tool developed by Microcosm, Inc., that uses projected solar cycle data and an Euler integrator with variable step sizes to project the orbit ephemerids until reentry. Data apply to the extremes of the solar cycle and include two different ballistic coefficients.

Table 2.6: Estimated Orbit Lifetime for Circular Orbits (In Days)

<table>
<thead>
<tr>
<th>Altitude (km)</th>
<th>Solar Min $\beta=50$ kg/m(^2)</th>
<th>Solar Max $\beta=50$ kg/m(^2)</th>
<th>Solar Min $\beta=200$ kg/m(^2)</th>
<th>Solar Max $\beta=200$ kg/m(^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.06</td>
<td>0.06</td>
<td>0.06</td>
<td>0.06</td>
</tr>
<tr>
<td>150</td>
<td>0.24</td>
<td>0.18</td>
<td>0.54</td>
<td>0.48</td>
</tr>
<tr>
<td>200</td>
<td>1.65</td>
<td>1.03</td>
<td>5.99</td>
<td>3.6</td>
</tr>
<tr>
<td>300</td>
<td>49.9</td>
<td>11</td>
<td>196.7</td>
<td>49.2</td>
</tr>
<tr>
<td>350</td>
<td>195.6</td>
<td>30.9</td>
<td>615.9</td>
<td>140.3</td>
</tr>
<tr>
<td>400</td>
<td>552.2</td>
<td>77.4</td>
<td>1024.5</td>
<td>346.9</td>
</tr>
<tr>
<td>450</td>
<td>872</td>
<td>181</td>
<td>1497</td>
<td>724</td>
</tr>
<tr>
<td>500</td>
<td>1205</td>
<td>393</td>
<td>2377</td>
<td>3310</td>
</tr>
<tr>
<td>550</td>
<td>1638</td>
<td>801</td>
<td>5470</td>
<td>4775</td>
</tr>
<tr>
<td>600</td>
<td>2580</td>
<td>3430</td>
<td>14100</td>
<td>13400</td>
</tr>
<tr>
<td>650</td>
<td>5560</td>
<td>4550</td>
<td>28500</td>
<td>27900</td>
</tr>
<tr>
<td>700</td>
<td>13400</td>
<td>12600</td>
<td>53400</td>
<td>52700</td>
</tr>
<tr>
<td>750</td>
<td>24400</td>
<td>24300</td>
<td>98500</td>
<td>97700</td>
</tr>
<tr>
<td>800</td>
<td>42000</td>
<td>41000</td>
<td>175200</td>
<td>174200</td>
</tr>
<tr>
<td>850</td>
<td>76600</td>
<td>76200</td>
<td>307400</td>
<td>306700</td>
</tr>
<tr>
<td>900</td>
<td>127000</td>
<td>128000</td>
<td>521000</td>
<td>520000</td>
</tr>
<tr>
<td>950</td>
<td>211000</td>
<td>210000</td>
<td>853000</td>
<td>852000</td>
</tr>
<tr>
<td>1000</td>
<td>341000</td>
<td>340000</td>
<td>1361000</td>
<td>1362000</td>
</tr>
</tbody>
</table>

Figure 2-6 depicts these values up to 500 kilometers, clearly demonstrating the exponential decrease in expected lifetime for satellites at lower altitudes.

Similarly, the required $\Delta V$ to maintain altitude and extend lifetime at very low
altitudes is exponentially greater as the spacecraft descends closer to the Earth's atmosphere. Figure 2-7 shows this relationship for the same four combinations of solar cycle and ballistic coefficient. This relates to the massive increase in propellant required, as discussed in Subsection 2.3.2 and displayed in Figure 2-8 for an electric propulsion system with a specific impulse of 1,500 seconds. Together these data and figures lead to a decisive conclusion: unaided by propulsion or an extremely high ballistic coefficient, it is impossible to operate a satellite for a year below approximately 400 km altitude, depending on the solar cycle. Either the designed satellite mission lifetime must be reduced, or lifetime-extending technologies, such as those discussed in Section 2.4, must be employed.
Figure 2-7: Annual ΔV Required to Maintain Altitude from 100 km to 500 km

Figure 2-8: Required Percentage of Propellant Mass ($I_{sp} = 1500$ s) to Maintain Altitude from 100 km to 500 km for One Year
In summary, the primary benefits and challenges of designing and operating optical imaging spacecraft in very low orbits are listed below.

Advantages:

- Improved optical resolution
- Reduced communication system size and power consumption
- Increased difficulty of adversary tracking and targeting
- Decreased payload mass and cost for fixed performance → reduced launch costs

Disadvantages:

- Increased drag → increased fuel required to maintain lifetime (or a shortened lifetime)
- Environment is highly uncertain, requiring large margins in propulsion system design
2.4 Enabling Technologies

2.4.1 Skipping Trajectories

Perhaps the most promising and most challenging exotic technique for flying spacecraft at low altitudes involves “skipping” the spacecraft on the fringe of the Earth’s atmosphere, like skipping a stone across a pond. Rather than fighting the effects of drag, this technique harnesses the potential of the increased air density at such low altitudes to provide lift and extend mission lifetime. Such skipping trajectories have primarily been limited for use in reentry systems for ballistic missiles and other long-range aerospace concepts [18]. For example, engineers at the Air Force Research Laboratory at Edwards Air Force Base have calculated potential skipping trajectories for a conceptual Reusable Aero Space Vehicle (RASV) under various atmospheric conditions [16]. The team ran “a sequence of 80 Monte Carlo, geographically varying atmospheres, together with flight parameters obtained from a baseline trajectory for a conceptual Boeing RASV spaceplane,” and determined that around-the-world polar, skipping trajectories could be used to boost large payloads (up to 11,340 kg) into “near-orbit conditions” (a maximum of 180 km).

This research bears relevance to the CLOSeSat mission in a number of areas. First, this study validates the potential for aerodynamic vehicles utilizing skipping trajectories along the lower thermosphere to circumnavigate the globe. Second, it accounts for the variance in altitudes attained and heating rates encountered at various times of year and day, depending on solar activity and the geomagnetic index. Finally, a skipping RASV serves as an interesting potential launch vehicle for CLOSeSat due to its responsiveness and flexibility. As a small satellite being inserted at a very low altitude, CLOSeSat would be hindered by the lengthy cycle and high launch costs of a more traditional rocket - such as relying on the ESPA ring of an Atlas V. The ideal launch solution for CLOSeSat would entail a simple, dedicated launch vehicle ready to launch at a moment’s notice. In this case, the conceptual RASV provides just such an opportunity, while additionally inserting the satellite into its very low, polar, skipping orbit.
2.4.2 Variable Aerodynamic Properties ($\beta$)

An additional method of minimizing the effect of drag on a satellite’s orbit includes increasing the spacecraft’s ballistic coefficient by reducing its coefficient of drag and/or cross-sectional area. For satellites that operate in lower altitudes with greater atmospheric density, the resulting drag force can be reduced by the use of a more aerodynamic design. This may be accomplished in a number of ways. First, the design could utilize higher density materials, thus increasing the ballistic coefficient of the spacecraft but also increasing the launch cost. Though satellites are often manufactured with the lightest materials available in an effort to meet the design’s mass budget (optical system mirrors are a good example of this trend), a denser, heavier spacecraft may also extend the expected lifetime. Another more promising potential design would use streamlined solar panels that fold in at perigee, decreasing the cross-sectional area during the period of highest drag. Such solar panels might then expand for the remainder of the orbit, storing enough power for the satellite to operate throughout its dynamic orbit.

2.4.3 Eccentric Orbits

Another possible solution for operating in very low orbits is to minimize the time spent at low altitudes by utilizing high eccentricities. Such orbits could greatly extend the lifetime of the spacecraft while still allowing for global coverage and fast response times. CLOSeSat would use its eccentric orbit with its primary operational mode at perigee, taking advantage of this fast pass at low altitude while recovering for the remainder of the orbit. Such “swooping maneuvers” are analyzed in greater detail in the following chapters.

2.4.4 Advanced Propulsion Techniques

Advances in cutting-edge space propulsion techniques show promising potential gains in future satellite propulsion systems. This is an area of emphasis for this study because a robust propulsion system would be required on any CLOSeSat spacecraft
expecting to stay in orbit longer than a couple days (see Table 2.6). The two primary indicators of a propulsion system’s performance capabilities are thrust, \( T \), and specific impulse, \( I_{sp} \), as listed in Table 2.5. While thrust refers to the force applied to the spacecraft by the propulsion system, specific impulse is the ratio of thrust to weight flow rate and is a measure of propellant efficiency:

\[
I_{sp} = \frac{T}{mg},
\]  

(2.8)

where \( \dot{m} \) is the mass flow rate and \( g \) is the acceleration due to gravity at sea level. A further derivation leads to Konstantin Tsiolkovsky’s ideal rocket equation, which relates \( I_{sp} \) and propellant mass to change in velocity, \( \Delta V \), the ultimate measure of propulsion capability:

\[
\Delta V = gI_{sp} \ln\left(\frac{m_0}{m_f}\right),
\]  

(2.9)

where \( m_0 \) is the initial spacecraft mass and \( m_f \) is the final spacecraft mass, the difference between the two terms being the propellant mass, \( m_p \).

There is a large selection of commercial-off-the-shelf (COTS) propulsion systems from which to choose, encompassing a wide range of performance characteristics. As Table 2.5 implies, propulsion technologies suggest an inverse trend between current thrust and specific impulse capabilities; as \( I_{sp} \) increases in the table, the available thrust becomes much smaller. Though this table only lists a handful of possible engines, its representative systems provide a relative distribution of propulsion system candidates. CLOSeSat will be a small satellite that requires a number of thrusting maneuvers, both for orbit altitude maintenance and possible simple burns into elliptic “swooping maneuvers” from a higher parking orbit. Therefore, the ideal propulsion system for CLOSeSat will likely be a throttleable, high-\( I_{sp} \), low-thrust, and low-mass design.
2.4.5 Precedents

There have only been a few documented attempts to continuously operate a satellite in VLEO, as the challenges have consistently proven too great for the existing technology. Out of the 919 current unclassified satellites listed in the most recent Satellite Database published by the Union of Concerned Scientists, only seven have a perigee of lower than 300 km. Six of those seven utilize moderately to highly eccentric orbits with apogees of at least 1000 km, thus limiting the time they spend in the turbulent lower thermosphere range. The lone exception to this trend is the European Space Agency’s Gravity field and steady-state Ocean Circulation Explorer (GOCE) satellite, launched on 17 March 2009. GOCE’s unique gravity field-mapping mission requires it to fly at the lowest possible altitude. Its sleek design visible in Figure 2-9 (it weighs over 1000 kg), and advanced electric ion propulsion system allow it to orbit within the MLTI region (see Subsection 2.3.1) at about 250 kilometers above the Earth’s surface.

Figure 2-9: GOCE’s Elegant Aerodynamic Design Minimizes Drag and Maximizes Solar Exposure at 250 km Altitude

A promising future concept of a small, low-orbit, low drag, high ballistic coeffi-
cient space system is Microcosm Inc’s NanoEye prototype. Space News reports that NanoEye utilizes a 0.25 meter diameter aperture telescope to produce imagery with ground resolution of 0.5 to 0.7 meters. It accomplishes this by flying in very low orbits (typically 200 to 300 kilometers, and as low as 160 kilometers) for a limited lifetime duration of six months to one year. This design was conceived to “provide rapid access to imagery over a specific location and launch within hours of call-up,” at a projected cost of just $1 million per satellite [3]. The NanoEye concept is a brand new development (this article was published in March 2010) and it further validates the potential capability of exploiting small satellites in very low orbits. Though NanoEye is still in the early stages of development, its concept of operations is closely aligned with that of CLOSeSat, and it seems feasible (though ambitious) based on the CLOSeSat analysis described in the following chapters. Thus it serves as a perfect lead-in to the remainder of this study.

Figure 2-10: An Early Prototype of the NanoEye Camera
Chapter 3

Analysis

3.1 Requirements

In order to develop a conceptual example of a possible CLOSeSat prototype, first a set of system-level requirements must be determined. Since these requirements were not given explicitly by a customer such as the Department of Defense, they are instead derived from a concept of operations and a projected scenario in the field. This section describes the concept of operations and the following requirements, which serve as the basis for the CLOSeSat system design.

3.1.1 Concept of Operations

The concept of operations is based on the definition of two different scenarios. The store and download scenario is the nominal scenario in which the spacecraft will operate most of the time. The realtime imaging in-theater scenario is a special scenario using a mobile ground station (MGS) that has been analyzed in this study.

Store and Download

In the store and download scenario, the ground segment sends a list of targets to observe to each satellite. At the next opportunity, a given satellite will sequentially slew to point at each target and take an image of it when it is in view (below a
maximum off-nadir angle which is a design parameter of the system). Images are stored on the spacecraft until they can be downloaded during communication accesses to the ground stations.

The maximum number of targets in the store and download mode is theoretically limited by:

- **On-Board Data Handling (OBDH):** the data storage capacity in the satellite (depending on the size of the image which is fixed by the size of the optical array)

- **Communications:** the downlink data rate (determined by the satellite’s effective isotropic radiative power (EIRP), the ground station’s gain-per-transmission $G/T$ factor, and the range between the ground station and the satellite)

- **Attitude Determination and Control System (ADCS):** the maximum slewing rate (given by the reaction wheels’s size and maximum rotation speed).

- **Electrical Power System (EPS):** the power generation and storage capacity

It is shown in the respective subsystem design sections that five targets per satellite per orbit is an achievable compromise.

In the *Store and Download* scenario, the ground stations are assumed to be fixed near the poles in order to increase access time for polar orbits. The McMurdo ground station (MCM) on Ross Island, Antarctica, and the Svalbard Ground Station (SGS) in Norway were selected because of their convenient locations and widespread use for data acquisition from polar remote sensing satellites.

**Realtime Imaging (In-Theater)**

In the *realtime imaging* scenario, a mobile ground station (MGS) is located in-theater, i.e. near the target location. When a satellite comes into view of the MGS, a command is sent to switch the satellite into the realtime mode. At reception of this command, the satellite immediately starts slewing to point the telescope at the target (slewing occurs only in the cross-track direction - yaw angle). The picture is taken at the point
where range is minimum (zero pitch angle) approximately at half access duration and then, instead of being stored, it is immediately downloaded in the remaining half access duration.

This scenario is more constraining than the store and download scenario in terms of ADCS and communications requirements because:

- **ADCS:** Slewing requirements are very high; the satellite must be able to slew twice the maximum off-nadir angle in half an access duration for the worst case scenario.

- **Communications:** Requirements in terms of downlink data rate will also be important because one image is completely downloaded during only half an access duration.

The subsystem design section shows how the requirements in this scenario actually drive the design of the ADCS and communications subsystems.

### 3.1.2 System-Level Requirements

System-level requirements and constraints are summarized in Table 3.1.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>Functional</td>
<td>Ground Resolution ≤ 1.0 m</td>
</tr>
<tr>
<td>Responsiveness</td>
<td>Functional</td>
<td>Mean Response Time ≤ 12h</td>
</tr>
<tr>
<td>Frequency Content</td>
<td>Functional</td>
<td>4 bands: RGB + NIR</td>
</tr>
<tr>
<td>Image Size</td>
<td>Functional</td>
<td>5x5 km²</td>
</tr>
<tr>
<td>Coverage</td>
<td>Functional</td>
<td>Global in ~ 12h</td>
</tr>
<tr>
<td>Duration</td>
<td>Operational</td>
<td>Lifetime ~ 0.5 year</td>
</tr>
<tr>
<td>Targets</td>
<td>Operational</td>
<td>$N_{target} = 5$ (S&amp;D) or 1 (RT)</td>
</tr>
<tr>
<td>Cost</td>
<td>Constraint</td>
<td>Lifecycle System Cost ~ $110M</td>
</tr>
<tr>
<td>Mass</td>
<td>Constraint</td>
<td>$m \leq 240$ kg</td>
</tr>
<tr>
<td>Primary Mirror Diameter</td>
<td>Constraint</td>
<td>≤ 1.0 m (Pegasus)</td>
</tr>
<tr>
<td>Regulations</td>
<td>Constraint</td>
<td>DoD mission</td>
</tr>
</tbody>
</table>

The 1.0 m ground resolution was selected as representative of commercial imaging systems. A mean response time of twelve hours allows the use of CLOSeSat image
to support daily operations in fighting impending combat situations. The spectral content of the image contains the three classical optical bands (Red, Green and Blue) which are necessary to produce images in real color, plus a Near Infrared band useful to see through the clouds or to operate during the night. The image size of $5 \times 5$ km$^2$ is big enough to provide situational awareness in a typical area such as a city (e.g. downtown Boston is $1 \times 1$ km$^2$). The value of twelve hours to global coverage was chosen to be consistent with the mean response time requirement. Concerning the mission duration, a lifetime on the order of six months was considered as a first input but extensions to one year or more could be possible. This constraint primarily limits the required propulsion system and maximum number of orbital burns. The choice of the number of simultaneous targets per satellite is discussed later in the paper. The constraints in mass and mirror diameter are set by the requirements of small, responsive, inexpensive launch vehicles such as Pegasus. Pegasus has a fairing diameter of 1.12 m and is capable of launching a 240 kg payload into a 600 km orbit. The maximum mirror diameter of 1.0 m was derived from the Pegasus fairing diameter, with a 12 cm margin for surrounding spacecraft structure. Finally, the target in total lifecycle system cost was set to $110M, which corresponds to roughly half the $215M pricetag of ORS-1 discussed in the Motivation section. The schedule for the mission would flow from the total cost and the assigned yearly budget.

As it will be discussed during the remainder of the paper, the three driving requirements of the mission are the high resolution, high responsiveness, and sufficient operational lifetime. Note that these requirements push the design in opposite directions. For instance, resolution is higher at lower altitudes but high responsiveness and longer lifetime demand higher altitudes for better coverage and less drag.
3.2 Development and Implementation of SSDT

This section describes the development of the Satellite System Design Tool (SSDT\textsuperscript{1}), the modeling of each of the pertinent subsystems, and the necessary modifications for analysis of CLOSeSat.

3.2.1 Overall Framework of SSDT

Given the objectives, requirements, and constraints outlined above, the final system design could take a wide variety of forms. There are numerous trades that are best answered considering all levels of the CLOSeSat design—subsystems, individual spacecraft, and the entire constellation.

The challenge of specifying an optimal system \textit{a priori} is therefore intractable using a top-down, linear design approach. Selecting certain architectures in isolation without consideration for the impact on other subsystems would likely eliminate prematurely designs that may, when considered in an integrated sense, might produce better performance.

In light of this, an integrated model of CLOSeSat has been developed. It is a parameterized representation of the spacecraft and ground stations that can be used to evaluate competing system configurations. The model allows for the comparison of various designs in terms of metrics such as mass, cost, coverage, response time, and cost per image.

Figure 3-1 below presents a schematic view of the parameterized CLOSeSat model. MATLAB was chosen for the model implementation given its widespread use, the relative ease with which certain constructions are implemented (e.g. for loops and data structures), and the availability to interface with other programs such as STK and Simulink.

The fundamental data representing a single realization of the CLOSeSat system

\textsuperscript{1}The SSDT was conceived and written in partnership with two fellow graduate students in MIT's Department of Aeronautics and Astronautics, Matthew Smith (optics) and Daniel Selva (communications and ADCS), to whom the author is extremely grateful. The respective sections of this thesis were adapted from their original work. See “Spacecraft and Constellation Design for a Continuous Responsive Imaging System in Space,” AIAA 2009-6773 [12].
Simulation setup and initialization
(choose parameters of interest to vary)

Satellite subsystem modules

Satellite system struct

\texttt{sat(i)}

- Altitude = 567 km
- Inclination = 98 deg
- Resolution = 1 m
- GSD = 0.5 m
- Ixx = 60 kg m^2
- Range = []
- Aperture = []
- SwathWidth = []
- Mass = []
- Cost = []
- ...

Initial system

\texttt{sat(i)}

(no outputs computed)

Final system

\texttt{sat(i)}

(outputs computed)

Visualization & analysis

System design decision

MATLAB

-orbit.m
-optics.m
-STK1.m
-ADCS.m
-ORDH.m
-comm.m
-STK2.m
-prop.m
-power.m
-mass.m
-cost.m

Realtime scenario
- comm pass duration
- slew rate

(constellation)

Global figures of merit
- response time
- coverage
- time to 100% coverage

Figure 3-1: Overview of the Integrated CLOSeSat Model
is a MATLAB structure containing all of the system parameters. This structure is initialized with inputs of interest to the user, then is passed through the integrated model, which populates the output fields sequentially. Entire families of CLOSeSat configurations (represented as an array of structures) is passed through the model, resulting in figure of merit calculations across the specified trade space.

Each spacecraft subsystem is modeled by a single MATLAB m-file that computes elements of the design structure specific to that subsystem. Two of the modules call Satellite Toolkit (STK) in order to obtain high fidelity data on orbit- and constellation-dependent figures of merit (e.g. access duration, revisit time, coverage, communications link geometry, etc).

Once a family of architectures is run through the model, the resulting trade space can be visualized using interactive plotting tools developed for the CLOSeSat model. The effect of constraining certain design parameters can be evaluated, and Pareto-optimal architectures can be easily compared in an effort to arrive at a solution. This is accomplished through the use of ‘sat_plot.m,’ which enables the user to plot any two variables for the entire design tradespace. A “point-and-click” functionality allows the user to select any point from the trade space, printing the critical design parameters and results to the MATLAB command window.

To summarize, the SSDT modeling approach has several advantages.

- It is parametric, allowing the user to evaluate a range of possible CLOSeSat architectures that span many different types of input variables.

- It is modular, allowing for individual subsystem modules of arbitrary complexity. For CLOSeSat, for example, there is considerable emphasis on the optical payload, propulsion subsystem, and orbital dynamics, hence those modules are relatively more complex. Different missions may require that model complexity be shifted to other subsystems that are more important in that context. Modularity also allows the user to add as many or as few subsystems as desired.

- The flow of the model allows the user to enter simulations “downstream” and avoid computational expense if modifications take place in later modules (e.g.
cost, mass, and power).

3.2.2 Subsystem Descriptions

Orbit Selection

Based on the mission objectives and system-level objectives, the following requirements were derived for the orbital subsystem:

1. 100% global coverage within twelve hours. This requirement denotes the time required to complete 100% accumulated global coverage. It also relates to the constellation’s ability to image any point on the globe at least once in the allotted time.

2. Image updated at least every twelve hours. This quantifies revisit time requirement for a given point and relates to the constellation’s ability to provide rapidly updating imagery with small coverage gaps.

3. One meter ground resolution or better. Ground resolution is a fundamental driving requirement for the optical payload, but it is also related to the orbital altitude. Therefore, an altitude must be used that does not create an undue demand on the optical subsystem. Such an altitude would allow the ground resolution requirement to be met even in worst case pass scenarios.

The first two subsystem requirements were developed as a means of focusing the design and presenting goals related to orbital coverage and revisit time. The resolution requirement is also interrelated with the optical subsystem. Because it is directly related to altitude and orbital mechanics, it must also be included in this section as a driving requirement for the orbit design.

The orbital MATLAB module (sat_orbit.m) takes inputs from the initialized satellite structure and applies general orbital dynamics equations to obtain outputs that are used in later subsystem models. Inputs include altitude, classical orbital elements, constellation definition variables, and a given maximum pointing angle ($\eta$). Outputs
include inclination, velocity, range, atmospheric density, earth-based reference angles, and coverage width. For example, the inclination of a sun-synchronous orbit at a given altitude can be calculated by setting the nodal precession rate caused by \( J_2 \) equal to 0.9856 deg/day, the Earth’s average rotation rate around the Sun.

\[
\dot{\Omega}_{J_2} \approx -2.06474 \times 10^{14} a^{-7/2} (\cos i) (1 - e^2)^{-2}
\]  

(3.1)

In addition to the rudimentary MATLAB module, Satellite Tool Kit (STK) is a powerful tool for depicting and analyzing orbits. STK is therefore used as the primary orbital analysis tool, generating access and coverage reports for various constellations. Several constellation simulations were run in STK for the first design spiral. All simulations occurred for the duration of one day. Coverage statistics were calculated using the “Coverage” tool in STK, while access statistics were calculated to a single ground facility located at 40 degrees North latitude (MIT). All orbits were sun-synchronous at an altitude of 567 kilometers. This altitude was selected to ensure a “zero drift orbit” – representing the CLOSeSat parking orbit for swooping maneuvers – as the orbital period equals exactly \( 1/15 \)th of a day, so each satellite has a unique ground track that is repeated daily [5]. Satellites are assumed to be evenly distributed throughout orbital planes with equivalent spacing. This comparison of constellations is also valuable at lower altitudes, such as those of interest to CLOSeSat, as the coverage statistics change proportionally with respect to altitude. A similar comparison for VLEO altitudes would also be a logical next step in CLOSeSat design, but it is beyond the scope of this study (see Future Work, Section 5.3).

Key parameters include 100% Coverage Time (accumulated, how long it takes the constellation to cover the entire globe); Median Coverage Gap (describes the statistical spread of coverage gaps); and Max Revisit Time (between passes to the target).

Based on these parameters an initial design included eight satellites distributed in four orbital planes. Though this results in a greater median coverage gap, the maximum revisit time is much smaller (on the order of 1/4). Also it only takes the
constellation about an hour to cover the entire globe. This design must be iterated however to include cost models, which are affected by the number of satellites and planes in the constellation. This feat is accomplished by integrating STK with MATLAB, allowing these figures of merit to be calculated for a number of architectures in one simulation. The optimum design orbit and constellation are thus determined by the final multi-axis trade, discussed at length later. Some conclusions and assumptions factor into the general orbit design. First a circular, sun-synchronous orbit is used as the baseline for this and most similar earth-observing missions. Also an off-line analysis revealed that satellites should be evenly distributed among orbital planes, and evenly spaced within the plane to ensure maximum coverage.

### Optical Payload

Geometry of the optical system will play a dominant role in the ability of CLOSeSat to fulfill the 1m ground resolution functional requirement, hence the initial optical model implements a fairly simple approach that treats the system as diffraction-limited and quasi-static. This is of course a gross oversimplification of actual imaging systems, however the aim is to capture first-order effects that will dominate the resolving capability. Figure 3-2 below shows the imaging geometry.

In addition to the 1m ground resolution requirement, a ground sample distance (GSD) requirement of 0.5 m per pixel is imposed. This allows the system to achieve Nyquist sampling at the desired resolution, ensuring that no information is lost to undersampling and that no aliasing occurs. As was mentioned in the discussion of

---

**Table 3.2: STK Constellation Analysis Results**

<table>
<thead>
<tr>
<th>Number of Planes</th>
<th>Number of Satellites</th>
<th>100% Coverage Time (min)</th>
<th>Mean Instant % Coverage</th>
<th>Median Coverage Gap (min)</th>
<th>Max Revisit Time (min)</th>
<th>Mean Revisit Time (min)</th>
<th>Mean Duration (sec)</th>
<th>Number of Passes Per Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8</td>
<td>189</td>
<td>28.15</td>
<td>12.66</td>
<td>168.39</td>
<td>22.64</td>
<td>574.274</td>
<td>44</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>63</td>
<td>27.52</td>
<td>31.26</td>
<td>42.72</td>
<td>24.81</td>
<td>574.209</td>
<td>44</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>229</td>
<td>14.08</td>
<td>36.55</td>
<td>211.21</td>
<td>53.48</td>
<td>572.851</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>139</td>
<td>13.78</td>
<td>63.49</td>
<td>135</td>
<td>62.75</td>
<td>585.44</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>570</td>
<td>7.11</td>
<td>37.11</td>
<td>450</td>
<td>111.28</td>
<td>570.037</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>471</td>
<td>7.05</td>
<td>84.87</td>
<td>303.63</td>
<td>122.2</td>
<td>574.762</td>
<td>10</td>
</tr>
</tbody>
</table>

---

56
requirements in Section 3.1, the optical system must image in four spectral bands, which are fixed to be red (\(\sim\)700 nm), green (\(\sim\)530 nm), blue (\(\sim\)460 nm), and near infrared (1-3 \(\mu\)m).

Inputs to the optical module include required ground resolution (nominally 1.0 m), ground sample distance (0.5 m) observing wavelength (500 nm), dynamic range of the detector (in bits per pixel, nominally 12), and primary mirror areal density. Note that we have assumed a pushbroom-style imaging system due to its simplicity and low pixel count relative to other schemes (i.e. matrix imaging and whiskbroom scanning). Outputs include required aperture diameter, focal length, instrument field of view, swath width at nadir and at the maximum off-axis angle \(\eta_{\text{max}}\), image data size (in bits), and mass of the optical system (as approximated by the mass of the primary mirror mass).

Thus the optical model treats the diameter and focal length as driven quantities. Based on the optical geometry in Figure 3-2, the governing relationships are:

\[
D = \frac{2.44h\lambda}{X \cos^2 \eta} \quad (3.2)
\]

and

\[
f = \frac{Ph}{\text{GSD} \cos^2 \eta}. \quad (3.3)
\]

These equations are used in the 'sat_optics.m' module to size the optical system for a given set of requirements. The primary relationship of interest in this thesis is
the relationship between altitude and aperture diameter/focal length, as discussed in Subsection 2.1.1. Note that both slant range and projection effects degrade image quality at off-nadir angles (i.e. the distance to the target is greater and the scene is at an angle to surface-normal viewing). This is accounted for by the \(1/\cos^2 \eta\) term.

**Communications**

The communications model is based on the RF link equation:

\[
\frac{E_b}{N_0} = 10 \log \left( \frac{P_T G_T G_R L \lambda^2}{kT_R R_b (4\pi S)^2} \right)
\]

(3.4)

In addition, a margin is defined as a function of \(\frac{E_b}{N_0}|_{req}\) for a given modulation:

\[
\frac{E_b}{N_0} = \frac{E_b}{N_0}|_{req} + M
\]

(3.5)

These two equations were applied to the uplink in order to find the diameter or gain of the antenna in the satellite and to the downlink in order to calculate the power that the satellite needs to radiate to achieve the required data rate.

The primary requirements for the communications subsystem are the following:

**Uplink:**

- \(\frac{E_b}{N_0} \geq 11.3dB\) (BPSK for a BER of \(10^{-7}\))
- \(R_b = 9.6\) kbps

**Downlink:**

- \(\frac{E_b}{N_0} \geq 9.6dB\) (BPSK for a BER of \(10^{-5}\))
- \(R_b\) enough to download one image in half of the access duration (real-time scenario) or to download \(N_{target}\) images in two accesses. Note that although the second requirement can seem more constraining, it depends on the characteristics of the ground stations. In fact, the MGS’s antenna being smaller than the fix ground stations’ antennas, the real-time scenario turns out to be the driving factor.
Note that the data rate and thus the radiated power depends on the image size—mainly driven by the array size—and on the access duration, fixed by the orbital characteristics (mostly altitude) and the position of the ground station.

Four different communications architectures were considered: first, two fixed ground stations near the poles to maximize coverage; second, one fixed ground station and cross-links between satellites; third, one fixed ground station and the Tracking and Data Relay Satellite System (TDRSS); and fourth, one mobile ground station (MGS) in theater (realtime scenario). For all these architectures, the access duration was computed as well as the downlink data rates to give the gain and the power on the satellite. The results are summarized in Table 3.3.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>$D_{GS}$ (m)</th>
<th>$T_{GS}$</th>
<th>$D_{SAT}$</th>
<th>$P_{SAT}$</th>
<th>$R_{b,DL}$</th>
<th>$\Delta t_{DL, image}$</th>
<th>$N_{Acc}$</th>
<th>$\Delta t_{Access}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 fix polar GS</td>
<td>2.4 (SGS)</td>
<td>150K omni</td>
<td>1W</td>
<td>9.6Mbps</td>
<td>7min</td>
<td>1/orbit</td>
<td>11min</td>
<td></td>
</tr>
<tr>
<td>2 fix polar GS</td>
<td>10 (MGS)</td>
<td>150K omni</td>
<td>1W</td>
<td>85Mbps</td>
<td>47s</td>
<td>1/orbit</td>
<td>11min</td>
<td></td>
</tr>
<tr>
<td>1 fix GS + crosslinks</td>
<td>10m (SGS)</td>
<td>150K omni</td>
<td>1W</td>
<td>9.6Mbps</td>
<td>7min</td>
<td>2/day</td>
<td>10min</td>
<td></td>
</tr>
<tr>
<td>1 fix GS + crosslinks</td>
<td>0.38m (SGS)</td>
<td>290K 0.45m</td>
<td>2W</td>
<td>4.0Mbps</td>
<td>16.6min</td>
<td>2/orbit</td>
<td>18min</td>
<td></td>
</tr>
<tr>
<td>1 fix GS + TDRSS</td>
<td>4.9m (SGS)</td>
<td>290K 1.3m</td>
<td>1W</td>
<td>6Mbps</td>
<td>11min</td>
<td>20/day</td>
<td>16min</td>
<td></td>
</tr>
<tr>
<td>1 MGS in-theater</td>
<td>2m (SGS)</td>
<td>150K omni</td>
<td>3W</td>
<td>16Mbps</td>
<td>4.15min</td>
<td>2/day</td>
<td>10min</td>
<td></td>
</tr>
</tbody>
</table>

The two rows for the fixed polar GS configuration correspond to each ground station (one near the North Pole, one near the South Pole). The two rows for the crosslinks architecture correspond to the fix ground station and to the receiving satellite respectively. The row for TDRSS corresponds to a TDRSS satellite.

For this study, a conservative value of 5.4 dB margin was used ($E_b/N_0 = 15$dB). A value of 4GB was chosen for the image size in this analysis, corresponding to a 4-band image of 12.5x12.5km with a 1.0m resolution.

The results show that the configuration based on TDRSS requires too big of an antenna (1.3m) to achieve the 6Mbps necessary to download an image of 4GB within the access duration (assuming a conservative compression factor of 8). The configuration based on crosslinks is not optimal because at low altitudes each satellite in the constellation only sees one satellite (always the same) twice per orbit (near the poles). On the other hand, the simple configuration with two fixed ground stations near the
poles is feasible with an omnidirectional antenna and provides data rates which meet the requirements. In addition, a 2m antenna on the MGS allows CLOSeSat to have an omnidirectional antenna to meet the severe requirements of the realtime scenario.

Therefore, the selected configuration for the communications subsystem is to have an omnidirectional antenna in the satellite, two fixed polar ground stations and one MGS with a 2m antenna for the real time scenario.

Note that the selected architecture is compatible with a value of $N_{target} = 5$, using the final size of the array. Indeed, assuming a ground station diameter of 2 meters (MGS), five images can be downloaded in 16 minutes with an omnidirectional antenna radiating 3.67W (final design parameters) and a compression factor of 10. A greater number of targets can be handled by larger ground station antennas (such as 10.0 m for SGS) but since there might be limitations in the data rate assigned to a client by a ground station, the value of five targets seems appropriate as far as the communications system is concerned.

**Attitude Determination and Control System (ADCS)**

A zero-momentum 3-axis stabilized attitude control scheme was selected because:

- passive techniques do not provide sufficient control accuracy for a big optical system.

- momentum bias configurations hinder the ability of the spacecraft to slew because of gyroscopic rigidity, and as it has been mentioned before, the CLOSeSat spacecraft needs to slew rapidly to new targets, especially in the realtime scenario. In particular, a quick calculation shows that a torque of several Nm would be needed in the worst case slewing maneuver (rate of twice the maximum off-nadir angle in half a typical access duration).

In order to size the actuators, two inputs are required: disturbance torques and slewing requirements. Disturbance torques were modeled using the simple equations provided in SMAD [20]. Four types of disturbances are taken into account by the model:
• disturbance torque due to aerodynamic friction: $T_{aero}$ is proportional to $\rho V^2$ which depends only on the altitude. A classical exponential approximation was used to estimate atmospheric density as a function of the altitude.

• disturbance torque due to the Earth’s gravity field: $T_{grav}$ decreases with altitude as $1/R^3$ and is proportional to the sine of twice the off-nadir angle. The model assumes the worst case in which the off-nadir angle is the maximum off-nadir angle (design parameter).

• disturbance torque due to the Earth’s magnetic field: $T_{magn}$ is proportional to both the residual dipole in the spacecraft and the magnitude of the Earth’s magnetic field. A classical far-field approximation was used to estimate the magnitude of the Earth’s magnetic field, yielding an expression proportional to $1/R^3$.

• disturbance torque due to the solar wind: $T_{solar}$ was estimated assuming a constant solar flux of 1367 $W/m^2$, a conservative value for the surface area of 3$m^2$, a null Sun angle and a reflectance factor of 0.6. Notice that unlike the three other disturbances, solar pressure does not depend on altitude.

The exact expressions of these disturbance torques can be found in SMAD [20]. Note that for a given off-nadir angle, altitude is the driving factor for almost all the disturbances. Low altitudes yield higher disturbance torques. Altitudes lower than 500km are dominated by atmospheric drag which increases steeply as altitude decreases. At higher altitudes, the Earth’s magnetic field is the main disturbance.

The second input for the actuators sizing model is the summary of slewing requirements in Table 3.4.

<table>
<thead>
<tr>
<th>Slewing Requirements</th>
<th>S&amp;D scenario</th>
<th>RT scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td># of Maneuvers per Orbit $N_{man}$</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Slewing angle</td>
<td>$2\eta$</td>
<td>$2\eta$</td>
</tr>
<tr>
<td>Slewing time</td>
<td>$T-3M_{acc}$</td>
<td>$M_{acc}$</td>
</tr>
</tbody>
</table>
In Table 3.4, $\eta$ is the off-nadir angle; $\Delta t_{\text{acc}}$ is the access duration; $P$ is the orbital Period; and $N_{\text{man}}$ is the number of slewing maneuvers per orbit. Each slewing maneuver was assumed to have a slew rate profile as shown in Figure 3-3. The available slewing time in the S&D mode was estimated as the orbital period minus the time during which the satellite is communicating with the two polar ground stations and the time in which the satellite is in the RT mode. A further discussion of the SSDT ADCS module is available in “Spacecraft and Constellation Design for a Continuous Responsive Imaging System in Space” by Krueger, et al [12].

![Figure 3-3: Summary of System-Level Slewing Requirements and Constraints](image)

**Electrical Power System (EPS)**

The electrical power system was modeled and analyzed as a secondary subsystem. Though vitally important to the success of a satellite, power generation, storage, distribution, and control did not present interesting trades and possibilities in the CLOSeSat analysis. Instead, the power subsystem was designed using the conventional preliminary design approach outlined in SMAD Table 11-31. Thus the power MATLAB module ('sat_power.m') takes inputs of average power requirements from the other subsystems, determines the necessary size of the solar arrays and batteries to accommodate such a power profile, and outputs the total mass and power of the
subsystem. Multijunction Gallium Arsenide (GaAs) solar cells are used to generate the necessary power required for the spacecraft while in sunlight. These represent the industry standard in solar panels. Despite their increased cost, multijunction cells also exhibit improved efficiency and decreased performance degradation over time [6]. The solar panels are sized according to the end-of-life power requirement (P_{EOL}), resulting in a required surface area of 1.77 m^2. These solar panels will be body-mounted to eliminate the cost and complexity of deployable, sun-tracking structures or control systems, and to result in a more streamlined, aerodynamic shape (less cross-sectional surface area). Secondary batteries are required to power the satellite through eclipse, as well as during times of high power demand. Each CLOSeSat satellite will utilize three Lithium-ion batteries to meet energy storage requirements. These batteries were chosen based on their superior energy density characteristics (100 W-hr/kg). Figure 3-4 depicts the satellite's power consumption for one orbit in the realtime mode.

![Figure 3-4: Typical Orbital Power Profile for Realtime Mode](image)

### 3.2.3 STK/MATLAB Interface

One of the challenges in designing CLOSeSat is specifying the optimal constellation design. Variables of interest include orbit altitudes, the number of orbital planes, and
the number of individual spacecraft per plane. These details were briefly discussed in the Orbit Selection subsection, but integrating the model with Satellite Tool Kit (STK) was critical in properly exploring this aspect of the trade space.

STK offers a wide range of capabilities that are of interest to CLOSeSat. Primarily, it provides the figures of merit by which the system performance is judged. These are calculated in two phases, first for a single satellite operating in the realtime (in-theater) mode, and second for an entire constellation of satellites. The first instance is relevant to the communications and attitude determination & control (ADCS) subsystems, which are sized using the stringent requirements of the realtime mode. The second instance is relevant to global figures of merit and hence to constellation design.

Figures 3-5 through 3-8 provide a glimpse into the development of CLOSeSat within STK. These screenshots depict the SSDT in various stages of analysis, from the placement of a mobile ground station in Afghanistan, to the enumeration of a sample CLOSeSat constellation and corresponding ground tracks. Figure 3-5 portrays a nominal, sun-synchronous CLOSeSat orbit over the mobile ground station in Kabul, Afghanistan, while the fields of view of the fixed polar ground stations are also visible. Figure 3-6 focuses on an example of the polar ground station and the coverage grid used by STK. Figures 3-7 and 3-8 show the 3D view and corresponding ground tracks, respectively, of the working CLOSeSat constellation while STK calculates response and coverage statistics. Together these images give an overview of the STK interface used to model orbital statistics for CLOSeSat.
Figure 3-5: STK representation of Kabul mobile ground station and nominal CLOSeSat orbit, as defined in 'sat_mini_stk.m'
Figure 3-6: STK 3D visualization of sample CLOSeSat orbit with polar ground station and coverage grid, as defined in ‘sat_stk.m’
Figure 3-7: STK 3D visualization of CLOSeSat constellation in low orbits, as defined in 'sat_stk.m'

Figure 3-8: STK ground track representation of CLOSeSat constellation, as defined in 'sat_stk.m'
The N-squared diagram depicted in Figure 3-9 describes each MATLAB function and subfunction, and shows the relationships between variables. MATLAB modules are represented in color along the diagonal axis, starting with ‘SAT_INIT’ and ending with ‘SAT_COST.’ The inputs to each function are listed along the associated vertical axis, while its outputs are listed along the horizontal axis. Of particular interest in this diagram are the feedback loops represented by variables listed below the diagonal. While variables listed above a given function have already been calculated by a previous function, those listed below are determined by a later function. The SSDT architecture attempts to minimize the number of feedback loops by assigning nominal guesses in the ‘sat_initialize.m’ function that could be overwritten as the program converges on a solution. This technique is acceptable for such a broad systems analysis that is more interested in the overall system solution than the specific solutions of individual spacecraft subsystems. By minimizing the number of feedback loops, the SSDT program is more streamlined and requires less computing time and power. The final SSDT runs one complete satellite architecture approximately every 30 seconds, which means hundreds of potential designs can be modeled in just a matter of hours. Such simplicity and responsiveness is not only helpful in determining a suitable CLOSeSat design, but is representative of the growing Operationally Responsive Space paradigm.

3.3 SSDT Modification for CLOSeSat Analysis

SSDT was developed as a general purpose satellite design tool. A few key enhancements were added to enable specific modeling of CLOSeSat. This section describes those modifications, which include a swooping maneuver model in the propulsion module and a new lifetime module including a drag model. These modifications improve the SSDT’s ability to incorporate very low-flying satellite designs with the enabling technologies previously discussed in Section 2.4.
3.3.1 Inclusion of Swooping Maneuver Modeling

Whereas the original SSDT only modeled a nominal solid rocket motor for a deorbit burn, CLOSeSat requires a more complicated propulsion module. In order to incorporate planned swooping maneuvers to lower perigees, the 'sat_prop.m' file was modified to include a number of Hohmann Transfers\(^2\) and the resulting $\Delta V$ required for each burn. The module then calculates the required propellant mass for each burn, which is the limiting factor in such maneuvers. The swooping model also evaluates the required propellant mass and $\Delta V$ to overcome energy lost due to drag throughout swooping maneuvers at low altitudes. Appendix B.1.4 lists the actual MATLAB code used for this analysis.

\(^2\)Note that the use of Hohmann Transfers precludes this model from utilizing electric propulsion techniques for swooping, which cannot be assumed to make impulsive burns.
3.3.2 Inclusion of Drag Model

Since the original SSDT did not model drag—a primary concern of low-altitude operation—a drag model was inserted in the form of the ‘sat.lifetime.m’ module (see Appendix B.1.5). This program uses a power function to model the atmospheric density, based on the mean values shown in Figure 2-4. The resulting curve fit equation is:

\[
\rho = (4 \times 10^8)h^{-7.765},
\]  

where \(\rho\) is the atmospheric density in kg/m\(^3\) and \(h\) is the altitude in kilometers. The accuracy of this approximation is visible in Figure 3-10. Using these values for \(\rho\), the propellant mass required for orbit maintenance is calculated for a given specific impulse and lifetime, as demonstrated in Subsection 2.3.2 using Equation 2.8. Alternatively, the results can be used to predict a maximum expected lifetime based on a limited propellant mass value.

![Figure 3-10: Atmospheric Densities with Power Function Curve Fit from 100 km to 1,000 km](image)

In conclusion, the Satellite System Design Tool provides a unique tool for analyzing the system-level design of a spacecraft. By integrating the subsystem modules
described here with the orbital modeling of STK, the SSDT enables its user to evaluate a number of potential design solutions through key design trades. In the case of CLOSeSat, two additional modifications were made to the SSDT to model swooping maneuvers and the drag environment’s impact on satellite lifetime and orbit maintenance requirements. A discussion of results from a number of SSDT simulations is now presented in Chapter 4.
Chapter 4

Results and Discussion

4.1 Summary of Simulations

A total of eight complete SSDT simulations\(^1\) were run, each testing the key parameter tradeoffs for various numbers of potential CLOSeSat architectures. These simulations were run to determine possible subsystem configurations that fit the CONOPS as described in Subsection 3.1.1 and the requirements listed in Table 3.1. Table 4.1 summarizes these eight simulations by indicating the key parameters tested in each one. Simulations A through C utilize the SSDT drag model to determine the lowest sustainability altitude for a circular CLOSeSat orbit. Simulations D through G extend this design to include additional characteristics such as optics system sizing and preliminary constellation design. Simulation H applies the swooping maneuver model to establish the potential of such techniques in extending lifetime at VLEO.

<table>
<thead>
<tr>
<th>Simulation</th>
<th>Altitude</th>
<th>Specific Impulse</th>
<th>Off-Nadir Resol.</th>
<th>Eccentricity</th>
<th># Pixels</th>
<th>Off-Axis Pointing</th>
<th># Sats</th>
<th># Planes</th>
<th>Swoop Lifetime</th>
<th>Swoop Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>V</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>V</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>V</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>V</td>
<td></td>
<td>V</td>
<td></td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>V</td>
<td>V</td>
<td>V</td>
<td>V</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>V</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(^1\)Four additional early simulations of the model are presented in Appendix A.
The key figures of merit discussed in this chapter include: cost per image, propellant mass required, and mean response time. Cost per image is defined as the total system lifetime cost divided by the number of images produced over the lifetime of the system. This is a more helpful metric than simple cost because it describes the cost effectiveness of the system (so long as it meets the overall cost requirement). Cost estimations were evaluated using standard parametric cost estimating relationships (CER) available in SMAD [20]. The overall lifecycle cost\(^2\) is primarily a function of the size of the optical mirror (as referenced in Subsection 2.1.1), the overall mass of the system, the total number of satellites produced, and the design lifetime. Each of the plotted data points often indicates a cluster of possible designs, where one parameter dominates the metrics of interest and others have little effect. For example, specific impulse has no explicit effect on mean response time, so each design point in Figure 4-5 actually represents a cluster of designs for the range of specific impulses simulated.

### 4.2 Determining Minimum CLOSeSat Altitude for a Continuous Circular Orbit

#### 4.2.1 Simulation A: Varying Altitude and Specific Impulse

The first simulation was simple and straightforward, and was intended as a functionality check for the system. By comparing changes in altitude and specific impulse as shown in Table 4.2, Simulation A verified the modified propulsion and lifetime modules (through a number of typical chemical specific impulse values) and the SSDT at VLEO (through a range of low altitudes). This simulation resulted in 20 complete potential CLOSeSat architectures. Simulation A assumed a lifetime of six months and a ballistic coefficient of 200 kg/m\(^2\).

Figure 4-1 verifies the system-level CLOSeSat assumptions regarding specific impulse: higher specific impulses result in lower required propellant masses for orbital

\(^2\)More information on the SSDT cost model is available in “CRISIS” [12].
Table 4.2: Setup of Simulation A: Validating the SSDT by Varying Altitude and Specific Impulse

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>200–300 km</td>
<td>25 km</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>100–400 s</td>
<td>100 s</td>
</tr>
</tbody>
</table>

Figure 4-1: Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation A

maintenance at a given altitude, and therefore a lighter propulsion system. This conclusion confirms the expectation that a high-$I_{sp}$ propulsion system is ideal for CLOSeSat orbital maintenance, as it will help keep the total mass down and within the design envelope. These results also verify the overall SSDT functionality at VLEO altitudes, for which the program was not initially developed. There was some concern that certain aspects of the SSDT, especially the communications module and STK interface, may fail at such low altitudes, but these concerns were abated by an error-free run of Simulation A.
4.2.2 Simulation B: Varying Altitude and Specific Impulse in the Lower Thermosphere

Similarly, Simulation B only varied the same two parameters but drove the baseline CLOSeSat design closer to Earth, below 200 km altitude. Simulation B also considered a much wider range of specific impulses, accounting for electric propulsion techniques in addition to more conventional chemical systems. Table 4.3 summarizes the inputs for this simulation. Figure 4-2 plots the propulsion system mass over various altitudes as determined by Simulation B. Simulation B assumed a six month lifetime, and a ballistic coefficient was calculated at each altitude using the required mirror diameter to achieve one-meter resolution. Simulation B expands on Simulation A to account for a wider range of potential altitudes and propulsion systems. Again the SSDT performs flawlessly at VLEO altitudes down to the Kármán Line.

Table 4.3: Setup of Simulation B: Varying Altitude and Specific Impulse in the Lower Thermosphere

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>100–300 km</td>
<td>20 km</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>100–2,000 s</td>
<td>100 s</td>
</tr>
</tbody>
</table>
Figure 4-2: Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation B
4.2.3 Simulation C: Finding the Minimum Sustained CLOSeSat Altitude

Finally, Simulation C tests a combination of the parameters influencing satellite drag and lifetime to determine the lowest sustained circular orbit for a given CLOSeSat design lifetime. Table 4.4 defines the varied parameters for this simulation. Using the CLOSeSat mission requirements outline in Table 3.1, a minimum sustained altitude can be derived from Figure 4-3. CLOSeSat must survive for at least half a year, notated by the cyan markers on the plot. Assuming an initial propellant mass fraction of about 50%, the maximum propellant mass for CLOSeSat is about 120 kg. This point corresponds to an altitude of 240 km, the minimum sustained circular orbit for a six month CLOSeSat lifetime under the prescribed mission requirements and constraints.

Table 4.4: Setup of Simulation C: Varying Altitude, Specific Impulse, and Design Lifetime

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>100–300 km</td>
<td>20 km</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>100–2,000 s</td>
<td>100 s</td>
</tr>
<tr>
<td>Lifetime</td>
<td>0.1–1.0 yr</td>
<td>0.1 yr</td>
</tr>
</tbody>
</table>

---

3This assumption is based on the propellant mass percentages in Figure 2-8.
Figure 4-3: Propellant Mass Required for Orbit Maintenance vs. Altitude for Simulation C
4.2.4 Simulation D: Varying Altitude and the Ground Resolution Requirement

The fourth trial analyzed the correlation between low altitude and another key design parameter: off-nadir ground resolution, as shown in Table 4.5. The combination of these two design parameters is integral to the design of the optical imaging subsystem, and this simulation tests the perceived resolution and cost benefits of operating CLOSeSat at very low altitudes.

Table 4.5: Setup of Simulation D: Varying Altitude and the Ground Resolution Requirement

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150-250 km</td>
<td>25 km</td>
</tr>
<tr>
<td>Off-Nadir Ground Resolution</td>
<td>0.8-1.5 m</td>
<td>0.1 m</td>
</tr>
</tbody>
</table>

Figure 4-4: Aperture Diameter vs. Altitude for Simulation D

As expected based on the Rayleigh diffraction-limited ground resolution equation (Equation 2.1) discussed in Subsection 2.1.1, there is a linear proportionality between altitude and the aperture diameter required to obtain the specified resolution. This
simulation further verifies the SSDT at low altitudes with a simple two-parameter trade, and demonstrates the optical system cost savings by operating at lower altitudes. The local extrema were selected for comparison, as indicated in Figure 4-4. As expected, the architecture with the smallest (and therefore cheapest) aperture, occurs at the lowest altitude (150 km) and has the most generous off-nadir resolution requirement (1.5 m) of the simulation. Meanwhile the largest aperture occurs at the highest altitude (250 km) and strictest resolution (0.8 m). While this figure portrays the design decision to be rather straightforward, it does not tell the whole story. A closer inspection of the selected architectures reveals that the lower, smaller design is also much more expensive - $4,748 per image, compared to just $358 for the higher, larger system. This fact confirms that though optical system costs may be lower at low altitudes, increased operating costs from other subsystems (especially propulsion, communications, and ADCS) will skyrocket. A more thorough investigation is required.

4.2.5 Simulation E: Testing Optical Systems with Varying Orbits, Propulsion Systems, and Ground Resolution Requirements

An additional simulation was run to account for optical payloads of various sizes (CCDs with varying number of pixels) and orbits of varying eccentricity, as summarized in Table 4.6. Figure 4-5 plots the corresponding altitudes and mean response times, based on the CCD size. Here the optimum altitude in terms of mean response time occurs at about 210 km.

Table 4.6: Setup of Simulation E: Varying Altitude, Ground Resolution, Specific Impulse, CCD Size, and Eccentricity

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–250 km</td>
<td>20 km</td>
</tr>
<tr>
<td>Off-Nadir Ground Resolution</td>
<td>0.8–1.5 m</td>
<td>0.1 m</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>150–450 s</td>
<td>100 s</td>
</tr>
<tr>
<td>Number of Pixels</td>
<td>5k–30k</td>
<td>5k</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0.0–0.5</td>
<td>0.1</td>
</tr>
</tbody>
</table>
4.2.6 Simulation F: Varying Altitude, Pointing Angle, Specific Impulse, and Number of Pixels

A related critical parameter is the off-axis pointing angle, $\eta$, which describes the spacecraft’s ability to slew to a target that does not fall directly on its ground track. Higher pointing angles allow the spacecraft to cover more of the Earth’s surface on each pass, thus decreasing response times, which is especially important for satellites at very low altitudes such as CLOSeSat. However, an increase in pointing angle capability also results in an increased strain on the attitude control system and increased cost and complexity for the overall spacecraft. Simulation F tests the parameters described in Table 4.7 and plots two key metrics in Figure 4-6: mass of the optical system and cost per image.

An ideal design for minimizing the size and cost of the optical system was selected along the Pareto optimal front in Figure 4-6. The design has an altitude of 210 km, a maximum $\eta$ of 35 degrees, a specific impulse of 450 seconds, and a CCD with 20k
### Table 4.7: Setup of Simulation F: Varying Altitude, Pointing Angle, Specific Impulse, and Number of Pixels

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–250 km</td>
<td>20 km</td>
</tr>
<tr>
<td>Off-Axis Pointing Angle</td>
<td>20–40°</td>
<td>5°</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>150–450 s</td>
<td>100 s</td>
</tr>
<tr>
<td>Number of Pixels</td>
<td>5k–30k</td>
<td>5k</td>
</tr>
</tbody>
</table>

![Optical System Mass vs. Cost Per Image for Simulation F](chart.png)

Figure 4-6: Optical System Mass vs. Cost Per Image for Simulation F

pixels in the cross-track dimension. According to the SSDT this design results in a cost per image of just $8.54 with an optics mass of 1.94 kg. This point represents an ideal solution for the bounded simulation in terms of mass and cost. The low optics mass demonstrates the value of operating at very low altitude, while the cheap imaging cost results from the high $\eta$. A high $\eta$ allows the spacecraft to image much more frequently over its lifetime, and when combined with the resolution capabilities at VLEO, provides a promising combination.
4.2.7 Simulation G: Analyzing Constellations for Various Orbits

So far CLOSeSat has been constrained to the default constellation, consisting of four evenly spaced planes with two evenly spaced satellites in each plane. Reconfiguring the CLOSeSat orbit and constellation as shown in Table 4.8 yields some interesting potential advances in responsiveness and imaging frequency. Figures 4-7 and 4-8 demonstrate these trends over the simulated range of parameters.

Table 4.8: Setup of Simulation G: Varying Altitude, Eccentricity, and Constellation Size

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–450 km</td>
<td>100 km</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0.0–0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>Number of Planes</td>
<td>2–4</td>
<td>1</td>
</tr>
<tr>
<td>Number of Satellites (per plane)</td>
<td>2–4</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 4-7: Mean Response Time vs. Number of Images Per Target Per Day for Simulation G
In both figures the optimal design occurs with the following architecture: four planes and four satellites per plane, with an altitude of 350 km and a slightly elliptic orbit (eccentricity = 0.2). The orbital parameters represent the middle of the range for this simulation, while the constellation size shows a preference for more satellites. Though a larger constellation will result in increased manufacturing, launch, and operating costs, these costs will be defrayed over time as the CLOSeSat assembly line becomes more efficient. Furthermore, research and development costs are spread out over a larger number of operational assets. In short, the early success of CLOSeSat could result in an order for more satellites, leading to a more robust constellation with better coverage and response times.
4.3 Evaluation of Swooping Orbits

4.3.1 Simulation H: Swooping Maneuver Capabilities at Various Perigees

Simulation H analyzed the swooping concept utilizing the swooping maneuver model developed for CLOSeSat analysis. This simulation varied the swoop altitude for a given parking orbit of 567 km. Table 4.9 summarizes the parameters of interest in this simulation, while Figure 4-9 plots the corresponding propulsion system mass and swoop altitude.

Table 4.9: Setup of Simulation H: Varying Swoop Altitude (Perigee), Lifetime, and Frequency

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swoop Altitude</td>
<td>120–220 km</td>
<td>10 km</td>
</tr>
<tr>
<td>Lifetime</td>
<td>0.25–2.0 yr</td>
<td>0.25 yr</td>
</tr>
<tr>
<td>Swoops Per Day</td>
<td>0.1–1.0</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Figure 4-9: Total Propulsion System Mass vs. Swoop Altitude for Simulation H

In order to determine the optimal number of swoops per day, the simulation was
run again with higher fidelity for a constrained mission lifetime of one year. This lifetime was chosen to emphasize the utility of swooping maneuvers in extending lifetime past the minimum CLOSeSat requirement of six months. The results are plotted in Figure 4-10. Using the same propellant mass ratio assumption of 50% discussed in Subsection 4.2.3, a swooping profile can be determined to satisfy the mission requirements. The maximum swooping frequency capable of meeting the derived CLOSeSat propellant mass requirement ($m_{prop} \leq 120$ kg) corresponds to the dark blue markers, representing a frequency of one swoop every 10 days. This frequency crosses the maximum propulsion system mass value at an altitude of about $160 \text{ km}$, the lowest swoop perigee for this CLOSeSat configuration.

Figure 4-10: Total Propulsion System Mass vs. Swoop Altitude for Simulation H
4.4 Optimal CLOSeSat System Design

Based on these simulations and the system trade conclusions derived from each, a preliminary CLOSeSat design was selected as described in Table 4.10. This design was then run through the SSDT one final time to determine its system-level characteristics. Table 4.11 describes the performance and cost metrics associated with this design vector, as calculated by the SSDT, and compares each value with the associated requirement or constraint from Table 3.1 (if applicable). The final CLOSeSat design utilizes a high-$I_{sp}$ [electric] propulsion system for orbit maintenance at low altitudes, and a traditional chemical propulsion system for swoop maneuvering. This bifurcated propulsion system increases complexity and mass, but allows the system to take advantage of the unique properties at both low altitudes (improved resolution) and a higher parking orbit (increased lifetime, improved coverage).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of Planes</td>
<td>4</td>
</tr>
<tr>
<td># of Satellites Per Plane</td>
<td>4</td>
</tr>
<tr>
<td>Parking Orbit Altitude</td>
<td>567 km</td>
</tr>
<tr>
<td>Swoop Altitude</td>
<td>160 km</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>350 s</td>
</tr>
<tr>
<td>Swoop Maneuver</td>
<td></td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>1,500 s</td>
</tr>
<tr>
<td>Orbit Maintenance</td>
<td></td>
</tr>
<tr>
<td>Frequency of Swoops</td>
<td>~weekly</td>
</tr>
<tr>
<td>Off-Nadir Pointing Angle</td>
<td>40°</td>
</tr>
<tr>
<td>Optical Array Size</td>
<td>20,000 pixels</td>
</tr>
</tbody>
</table>
Table 4.11: Performance and Cost for Optimal CLOSeSat Swooping Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Resolution [m]</td>
<td>1.0</td>
<td>≤ 1.0</td>
</tr>
<tr>
<td>Mean Response Time [h]</td>
<td>1.34</td>
<td>≤ 12</td>
</tr>
<tr>
<td>Image Size [km x km]</td>
<td>5 x 5</td>
<td>≥ 5 x 5</td>
</tr>
<tr>
<td>Time to Global Coverage [h]</td>
<td>2.05</td>
<td>~ 12</td>
</tr>
<tr>
<td>Lifetime Duration [yr]</td>
<td>1.0</td>
<td>~ 1.0</td>
</tr>
<tr>
<td>Max. Targets Per Orbit</td>
<td>5 (S&amp;D) or 1 (RT)</td>
<td>5 (S&amp;D) or 1 (RT)</td>
</tr>
<tr>
<td>Primary Mirror Diameter [m]</td>
<td>0.338</td>
<td>≤ 1.0</td>
</tr>
<tr>
<td>Wet Mass Per Spacecraft [kg]</td>
<td>237.4</td>
<td>≤ 240</td>
</tr>
<tr>
<td>Ballistic Coefficient [kg/m²]</td>
<td>863.03</td>
<td>-</td>
</tr>
<tr>
<td>Cost Per Image [FY00$k]</td>
<td>7.09</td>
<td>-</td>
</tr>
<tr>
<td>Cost Per Spacecraft [FY00$M]</td>
<td>5.94</td>
<td>-</td>
</tr>
<tr>
<td>Constellation Lifecycle Cost [FY00$M]</td>
<td>94.9</td>
<td>~ 110</td>
</tr>
</tbody>
</table>
4.4.1 Comparison with Existing Technologies

In comparison to the current state of the art, CLOSeSat represents a progression toward smaller, cheaper, more responsive spacecraft.

ORS-1

The stated goal of CLOSeSat was to cut each cost metric of ORS-1 by at least 50 percent, and CLOSeSat exceeded these expectations. While ORS-1 weighs 450 kg and costs $215 million (for one spacecraft), the entire 16-satellite CLOSeSat constellation would cost only $94.9 million (not including launch and operating costs), and each satellite weighs just over half of ORS-1 (237.4 kg each). This weight savings could further result in a cheaper system by enabling the launch of multiple CLOSeSats on the ESPA ring of a larger launch vehicle. These advancements in producing smaller, cheaper satellites are made possible by operating at lower altitudes, as described in this chapter.

NanoEye

The NanoEye concept bears a striking resemblance to the proposed CLOSeSat design. By operating at very low altitudes, both microsatellites achieve sufficient resolution with small optical systems. (NanoEye’s aperture diameter of 0.25 m is comparable to CLOSeSat’s 0.338 m aperture.) At just $1 million per satellite NanoEye is projected to be six times cheaper than CLOSeSat, presumably because NanoEye lacks a robust propulsion system capable of making swooping maneuvers. Altogether the NanoEye and CLOSeSat designs have much in common, as both strive to develop relatively cheap, small, short-lived satellites that take advantage of low altitude operation and exemplify the tenets of operationally responsive space.
Chapter 5

Conclusions and Recommendations

5.1 Conclusions

Using the Satellite System Design Tool as the primary instrument of analysis, this research concluded that sustainable altitudes as low as 240 km are made possible through the use of continuously thrusting, high-$I_{sp}$ propulsion systems for short mission lifetimes (six months). For a longer lifetime (one year) altitudes as low as 160 km may be reached approximately every ten days through advanced propulsion systems and dynamic orbit changes. The preliminary CLOSeSat system design demonstrates an innovative design resulting in a relatively cheap, small satellite capable of producing high-resolution imagery within hours of notification.

5.2 Summary of Contributions

In summary, the research questions listed in the Motivation section (Section 1.1) have been answered as follows:

- **What is the lowest viable altitude for an optical imaging micro- or minisatellite to survive and produce useful information?** This altitude depends on a number of variables that have been analyzed throughout this study. For the given scenario, CONOPS, and requirements, this minimum
sustained altitude occurs at approximately 240 km for a six month lifetime. This altitude can be lowered to approximately 160 km via swooping maneuvers with a majority of the satellite’s quiescent lifetime spent at a higher parking orbit.

- **What might be a preliminary system-level design of such a satellite?** CLOSeSat, as described in Tables 4.10 and 4.11, demonstrates one potential early design for a small optical imaging satellite at VLEO.

- **Is such a design feasible in the near future given the inherent challenges of operating at such an altitude?** Yes, CLOSeSat represents a feasible design, using a combination of available propulsion systems for orbital maintenance and maneuvering, as well as an aerodynamic design (high ballistic coefficient) to minimize drag.

- **Which emerging technologies might enable such a design to become a reality in the near future?** In addition to the advanced propulsion techniques and aerodynamic designs discussed in this thesis, emerging technologies may further enable low-perigee operation in the near future. These include the use of skipping orbits and variable aerodynamic properties (i.e., changing the spacecraft’s shape throughout its orbit), which could be discussed as an extension to this study (see Subsection 5.3.3 in Future Work).

## 5.3 Future Work

### 5.3.1 Single Satellite Design → Constellation Design

While this study focused primarily on the design of a single CLOSeSat spacecraft, Simulation G demonstrated the potential advantages of altering the baseline constellation. A comparison of coverage and response time statistics for various constellations in VLEO (similar to Table 3.2) would demonstrate the trade in such parameters at
very low altitudes. Further analysis would also focus on cost-responsiveness trades for constellations of various compositions. A long-term CLOSeSat system outlook would predict the usable lifetime of each spacecraft and plan a launch cycle to ensure sustained coverage. This would require an accurate lifetime prediction function.

5.3.2 Inclusion of Lifetime Prediction Function

Though originally planned as an additional tool for CLOSeSat design, the lifetime prediction function was found to be very similar to functionality available commercially through SatLife. Nevertheless, it would be helpful to include code in the SSDT (or create an STK/SatLife interface) that enables a better understanding of satellite lifetimes in relation to altitude, solar cycle, and available propellant. This could be added into the lifetime module using a simple numerical algorithm developed by the Australian Space Weather Agency to model satellite orbit decay [10].

5.3.3 Modeling of Skipping Orbits and Variable Drag Designs

The limitations of time and the SSDT drove this research to focus almost exclusively on swooping maneuvers and propulsion systems as enablers of low-perigee operation. The other two exciting concepts previously discussed, skipping orbits and variable drag designs, deserve a similarly thorough analysis of their potential use. A dynamic drag analysis could be combined with the aforementioned lifetime prediction function to better forecast a given spacecraft’s expected lifetime in the high drag environment of the MLTI. The modeling of a skipping orbit exceeds the current capabilities of STK, so an additional program would be required to simulate this behavior.
Appendix A

Additional Simulations

A.1 Simulation i: Varying Altitude, Ground Resolution, and Specific Impulse

Two-axis trades are helpful in understanding the SSDT and verifying its usefulness in the CLOSeSat design problem, but more intricate trades must be run to demonstrate the coupling of various parameters. Simulation i combines the first three simulations to analyze 192 different possible satellite architectures with varying altitude, off-nadir ground resolution, and specific impulse. Table A.1 describes the varied parameters for this simulation. Figure A-1 depicts these results by comparing the possible altitudes with corresponding cost per image metric. In addition to cost, it is also helpful to understand how these designs correspond to responsiveness. Figure A-2 relates mean response time to altitude for the same set of designs.

Table A.1: Setup of Simulation i: Varying Altitude, Ground Resolution, and Specific Impulse

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–250 km</td>
<td>20 km</td>
</tr>
<tr>
<td>Off-Nadir Ground Resolution</td>
<td>0.8–1.5 m</td>
<td>0.1 m</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>150–450 s</td>
<td>100 s</td>
</tr>
</tbody>
</table>

Due primarily to the decreased coverage at low altitudes, the cost per image increases as altitude decreases. One downside to operating in VLEO is the decreased
field of view visible to the spacecraft as it orbits the Earth more closely. This results in fewer imaging opportunities over the lifetime of the satellite, thus dividing the overall system cost over a smaller number of produced images, and increasing the cost per image. As visualized in Figure A-1, this trade is something to consider when placing satellites at very low altitudes. Although the cost per image may continue to rise, Figure A-2 demonstrates that there is an altitude "sweet spot" in terms of responsiveness at about 210 km, which is further discussed in Simulation F.

A.2 Simulation ii: Testing Elliptic Orbits with Varying Altitude, Ground Resolution, and Specific Impulse

The next simulation increased complexity by including a range of eccentricities. Whereas the nominal circular orbit is rather simple to model, elliptic orbits represent a more difficult problem, as the spacecraft’s velocity changes throughout the orbit. For these more complicated orbits it was especially useful to utilize STK’s or-
As the number of parameters increases, a Pareto optimal front begins to form. This front consists of the optimal design combinations for the given metrics (in this case, mass and cost). Here a potential design along the front is selected, and is found to orbit at an altitude of 210 km with an eccentricity of 0.3, a specific impulse of 450 seconds, and a ground resolution of 1.1 m. The propulsion system of this architecture weighs 6.85 kg, while it costs $879.48 per image.
A.3 Simulation iii: Swooping Maneuvers and Propulsion Systems from Various Parking Orbits

The final CLOSeSat parameter to be analyzed is the swoop altitude, referring to the lowest perigee attained by the spacecraft as it swoops over a given ground target. (In this case the target is located in Kabul, Afghanistan.) Simulation iii varied the parking orbit altitude, specific impulse, and eccentricity with a fixed swoop altitude of 125 km, as shown in Table A.3.

Table A.3: Setup of Simulation iii: Varying Altitude, Specific Impulse, and Eccentricity

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–450 km</td>
<td>50 km</td>
</tr>
<tr>
<td>Specific Impulse</td>
<td>100–450 s</td>
<td>50 s</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0.0–0.5</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Figure A-4 plots the propulsion mass and response time for the simulated designs.

In this scenario a lower parking orbit is desirable, due to the decreased ΔV required
Figure A-4: Total Propulsion System Mass vs. Mean Response Time vs. for Simulation iii
to swoop. This does not take into account reduced lifetime, which is a significant
shortcoming of the SSDT. A local minimum propulsion mass of 5.05 kg is found to
occur at a mean response time of 44,600 seconds (12.39 hours). This corresponds to
the design with a circular parking orbit at 150 km (the lowest altitude simulated)
and a specific impulse of 100 seconds. This result is surprising, because it was the
lowest specific impulse tested in this simulation. Previous simulations indicated that
propulsion systems with higher $I_{sp}$ were almost always preferred for CLOSeSat, but
this is not always the case. This scenario does not require as much propellant to
transfer from the parking orbit to the swooping maneuver, so the advantage from
increased specific impulse does not outweigh the decreased responsiveness resulting
from a weaker propulsion system (less thrust). Unfortunately such a design is not
plausible, as Simulation K will demonstrate, because a parking orbit at an altitude
of 150 km is untenable.
**A.4 Simulation iv: Incorporating the Drag Penalty**

Since the previous simulations did not incorporate a drag model, an additional simulation was required to determine the lower bound of useful operating perigees. Simulation iv ran the baseline CLOSeSat design through a variation of altitudes, ballistic coefficients, and design lifetimes as described in Table A.4. Figure A-5 omits excessive solutions at very low altitudes and constrains the tradespace to propellant masses of less than 1000 kg. Here a possible design at 240 km with $\beta = 550 \text{ kg/m}^2$ and lifetime of 36 days requires 348 kg of propellant mass for orbital maintenance.

**Table A.4: Setup of Simulation iv: Varying Altitude, Ballistic Coefficient, and Lifetime**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>150–300 km</td>
<td>10 km</td>
</tr>
<tr>
<td>Ballistic Coefficient</td>
<td>50–550 kg/m²</td>
<td>100 kg/m²</td>
</tr>
<tr>
<td>Lifetime</td>
<td>0.1–0.5 yr</td>
<td>0.1 yr</td>
</tr>
</tbody>
</table>

Figure A-5: Total Propulsion System Mass vs. Altitude for Simulation iv
Appendix B

MATLAB Code

B.1 SSDT Overview

The MATLAB code used to run the Satellite System Design Tool consists of dozens of functions and subfunctions, and thousands of lines of code. In lieu of including all that code here, only the main program ('sat_system.m'), the initial satellite array that lists all pertinent values ('sat_initialize.m'), the main STK interface function ('sat_stk.m') and the modified propulsion ('sat_prop.m') and lifetime modules ('sat_lifetime.m') for CLOSeSat analysis are listed.

B.1.1 'sat_system.m'

```matlab
function sat_out = sat_system(sat_in)

% SAT_SYSTEM
% sat_out = sat_system(sat_in)

% Function to model the CLOSeSat system, either a single realization of
% the system or an entire family of architectures. Operates on an array
% of 'sat' structures containing the CLOSeSat design parameters. Thus
% sat_in can be a single structure, a vector of structures, or an array
% of structures.
```

101
sat_system runs the subsystem modules on each of the sat structures in the sat_in array. The fully-populated models are returned in the sat_out array, which has the same dimensions and sizes as the sat_in array.

Jared Krueger <jkrue@mit.edu>
Daniel Selva <dselva@mit.edu>
Matthew Smith <m_smith@mit.edu>

13 Oct 2008

Initialize the STK/MATLAB interface
agiInit;
remMachine = stkDefaultHost;

open a socket to STK
conid = stkOpen(remMachine);

Create a Scenario
stkNewObj('/','Scenario', 'CLOSESat');
scenario_path = '/Scenario/CLOSESat/';
stkSetAnimationTimeStep(conid,scenario_path,3);

for keeping track of the run
total = numel(sat_in);
counter = 1;

for i = 1:size(sat_in,1)
    for j = 1:size(sat_in,2)
        for k = 1:size(sat_in,3)
            for l = 1:size(sat_in,4)
                for m = 1:size(sat_in,5)
                    temp = sat_in(i,j,k,l,m);
                
```
% orbit dynamics model
temp = sat_orbit(temp);

% optical payload model
temp = sat_optics(temp);

% First STK module
temp = sat_mini_stk(temp,conid);

% ADCS model
temp = sat_adcs(temp);

% OBDH model
temp = sat_obdh(temp);

% communications subsystem model
temp = sat_comm(temp,conid);

% STK simulation
temp = sat_stk(temp,conid);

% thermal model
temp = sat_thermal(temp);

% lifetime model
temp = sat_lifetime(temp);

% propulsion model
temp = sat_prop(temp);

% power model
temp = sat_power(temp);

% mass budget
temp = sat_mass(temp);
% cost model
temp = sat_cost(temp);

sat_out(i,j,k,l,m) = temp;

% display message
fprintf('%3.0f of %3.0f\n', counter, total);
counter = counter + 1;

% keep a record
save sat_log sat_out;

end
end
end
end

% Close out the stk connection
stkClose(conid);

% Close any connection
stkClose('all');

return;

B.1.2 'sat_initialize.m'

function [sat] = sat_initialize()

% sat_initialize
% sat = sat_initialize;
%
% Function to create a sat structure and initialize to default values.
%
% ----------------------
% Constants
% -----------------

global Deg Rad MU RE OmegaEarth SidePerSol RadPerDay SecDay Flat EEsqrd ... 
    EEarth J2 J3 J4 GMM GMS AU HalfPI TwoPI Zero_IE Small Undefined 

global SolarFlux LightSpeed EarthMagneticMoment gravity k_Boltzmann

wgs84data;

SolarFlux = 1367;       % Solar flux [W/m^2]
LightSpeed = 3.0E8;      % Light speed in m/s
EarthMagneticMoment = 7.96e15; % Magnetic moment of the Earth in Tesla*m^3
gravity = 1000*MU/(RE^2); % gravity on the Earth in m/s^2
k_Boltzmann = 1.37E-23;  % Boltzmann Constant

% -----------------
% User-defined inputs
% -----------------

% system
sat.Lifetime = 1;        % satellite lifetime in years
sat.Ntargets = 5;        % Number of targets for the entire constellation
sat.Mass = 150;          % spacecraft mass [kg]

% structure
sat.Ixx = 90;            % Moment of inertia around x axis [kg*m^2]
sat.Iyy = 60;            % Moment of inertia around y axis [kg*m^2]
sat.Izz = 90;            % Moment of inertia around z axis [kg*m^2]
sat.SurfaceArea = 1;     % Surface area [m^2]
sat.BC = 200;            % Ballistic Coefficient [kg/m^2]

% orbital
sat.Alitude = 567;       % orbit altitude [km]
sat.SwoopAlt = 160;      % perigee of swooping maneuver [km]
sat.NSwoopsPerDay = 0.15; % Frequency of swooping maneuver
sat.Eccentricity = 0;    % orbit eccentricity [#]
sat.Nsats = 4; % total number of satellites [#]
sat.Nplanes = 4; % number of planes
sat.MaxPointing = 40; % maximum pointing angle from nadir [deg]

% optical
sat.GResOff = 1; % ground resolution at max off-nadir angle [m]
sat.GSD = 0.5*sat.GResOff; % Ground Sampling Distance = half of resolution [m]
sat.Wavelength = 500e-9; % observation wavelength [m]
sat.NPixCrosstrack = 20000; % number of CCD pixels along the crosstrack direction [#]
sat.NPixVelocity = 64; % number of CCD pixels along the velocity direction [#]
sat.PixelSize = 10e-6; % size of pixels in the CCD [m]
sat.Nbits = 12; % detector quantization [#]
sat.ArealDensity = 30; % Primary mirror areal density [kg/m^2]

% ADCS
sat.ADCSConf = 1; % 1: RW+MGT(Mom. dump.) 2:RW+THR(M.D.) 3:RW+THR(M.D. + slew S&D only)
sat.Reflectance = 0.6; % Reflectance of the sun radiation on the satellite
sat.MaxSolarAngle = 0; % Worst-case solar incidence angle [deg]
sat.OffsetCPsolar = 0.3; % Offset between the center of pressure and the center of gravity of the s/c for solar pressure [m]
sat.OffsetCPaero = 0.2; % Offset between the center of pressure and the center of gravity of the s/c for aerodynamics [m]
sat.ResidualDipole = 1; % Residual dipole of the s/c [A*m^2]
sat.DragCoefficient = 2; % Aerodynamic drag coefficient of the s/c
sat.RWMarginFactor = 1.5; % Margin for sizing of reaction wheels
sat.ThrusterMomentArm = 0.5; % Moment Arm for calculation of thrusters' torque.
sat.NumberofRW = 4; % Number of reaction wheels (>=3) [#]
sat.NumberofSunSensors = 3; % Number of sun sensors (>=2) [#]
sat.NumberofMagnetometers = 2; % Number of magnetometers (>=2) [#]
sat.NumberofMagneticTorquers = 3; % Number of Magnetic torquers (>=3) [#]
sat.ADCSSpecificImpulse = 70; % specific impulse in s (70s for cold gas, 200s for hydrazine)

% Comm
% Uplink
sat.ULRequiredEbNO = 12.0;  \% Uplink Required Energy per Bit to Noise Spectral Density in dB
sat.ULEbNOMargin = 3.0;  \% Uplink Margin : Eb/No - Eb/NO required
sat.ImplementationLoss = 2.0;  \% Implementation Loss in dB
sat.ULTXPower = 1.0;  \% Uplink Transmitter Power in W
sat.ULTXAntennaDiameter = 2.0;  \% Uplink Transmitter Antenna Diameter in m
sat.ULTXLineLoss = -1.0;  \% Uplink Transmitter Power in dB
sat.GroundAntennaMinElevation = 10;  \% Ground Antenna minimum elevation in deg
sat.ULFrequency = 2.0E9;  \% Uplink Frequency in Hz
sat.ULDataRate = 9600;  \% Uplink Data rate in bps
sat.ULTXAntennaPointingOffset = 0.2;  \% Uplink Transmitter Antenna Pointing Offset in deg
sat.ULRXAntennaEfficiency = 0.55;  \% Uplink Receiver Antenna Efficiency
tsat.TTCRedundancy = 2;  \% Number of elements of each (antenna and transponder)

\% Downlink
sat.DLRequiredEbNO = 10.0;  \% Downlink Required Energy per Bit to Noise Spectral Density in dB
sat.DLEbNOMargin = 3.0;  \% Downlink Margin : Eb/No - Eb/NO required
sat.DLFrequency = 2.2e9;  \% Downlink Frequency in Hz
sat.DLRXAntennaPointingOffset = 0.2;  \% Downlink Receiver Antenna Pointing Offset in deg
sat.DLRXAntennaDiameter = 2.0;  \% Downlink Receiver Antenna Diameter in m
sat.DLTXLineLoss = -1.0;  \% Downlink Transmitter Power in dB

\% OBDH
sat.CompressionFactor = 10;  \% Compression Factor

\% ground station
sat.NGroundStations = 2;  \% number of ground stations
sat.LatGroundStations = [78.2303 -77.8392];  \% Latitude vector of ground stations
sat.LonGroundStations = [15.3928 -193.3331];  \% Longitude vector of ground stations

\% Lifetime
sat.Isp = 1500;  \% specific impulse in [s] (1,500 for electric propulsion)
% Propulsion
sat.SpecificImpulse = 350; % specific impulse in [s] (350s for solid/liquid propulsion)

% Computed outputs
% orbits
sat.Inclination = []; % orbit inclination [deg]
sat.Semimajor = []; % semimajor axis [m]
sat.Velocity = []; % orbital velocity [m/s]
sat.VelocityPer = []; % velocity at perigee [m/s]
sat.VelocityApo = []; % velocity at apogee [m/s]
sat.Period = []; % orbital period [s]
sat.Density = []; % atmospheric density at altitude [kg/km^-3]
sat.MeanAngMotion = []; % mean angular motion, n [rad/s]
sat.Rho = []; % angular radius [deg]
sat.Epsilon = []; % elevation angle of ground target [deg]
sat.Lambda = []; % earth central angle [deg]
sat.Range = []; % range to target [km]
sat.CoverageWidth = []; % coverage width [km]

% ADCS
% Disturbances
sat.TorqueGravity = []; % Disturbance Torque due to gravity gradient in Nm
sat.TorqueMagnetic = []; % Disturbance Torque due to gravity gradient in Nm
sat.TorqueSolar = []; % Disturbance Torque due to solar pressure in Nm
sat.TorqueAero = []; % Disturbance Torque due to aerodynamics in Nm
sat.MaxTorque = []; % Worst case Disturbance in Nm

% Actuators: Reaction Wheels
sat.RWTorque = []; % Torque required from reaction wheel in Nm
sat.RWMomentum = []; % Momentum required from reaction wheel in Nms
sat.RWMaxOmega = []; % Maximum rotation speed of reaction wheel in rad/s

% Actuators: Magnetic Torquers
sat.MGTorquersDipole = []; % Dipole of the magnetic torquers in Am^-2.
%% Actuators: Thrusters

sat.ThrusterForce = []; % Force yielded by thrusters [N]
sat.ThrusterNpulses = []; % Number of pulses done by thrusters during lifetime [#]
sat.ADCSDelta_V = []; % DeltaV necessary for ADCS (thrusters option) [m/s]
sat.ADCSDelta_VD = []; % Proportion from DeltaV due to disturbances [%]
sat.ADCSDelta_VS = []; % Proportion from DeltaV due to slewing requirements [%]
sat.ADCDVSVT = []; % Ratio between DeltaV due to slewing requirements and total DeltaV [%]
sat.ADCSPropellantMass = []; % Propellant mass necessary for ADCS [kg]
sat.MassADCS = []; % ADCS mass [kg]
sat.PeakPowerADCS = []; % ADCS peak power requirements [W]
sat.AvgPowerADCS = []; % ADCS average power requirements [W]
sat.OffPowerADCS = []; % ADCS min power requirements [W]

% optical

sat.Aperture = []; % primary mirror aperture diameter [m]
sat.Flength = []; % focal length [m]
sat.FOVc = []; % instrument cross track field of view [deg]
sat.GResNadir = []; % ground resolution at nadir [m]
sat.AngleHorizSTK = []; % horizontal half angle for STK Sensor [deg]
sat.AngleVertSTK = []; % vertical half angle for STK Sensor [deg]
sat.ImageSize = []; % number of bits per image [bit]
sat.MassOptics = []; % mass of the optical subsystem [kg]
sat.PeakPowerOptics = []; % peak power of the optical subsystem [W]
sat.AvgPowerOptics = []; % avg power of the optical subsystem [W]
sat.OffPowerOptics = []; % off power of the optical subsystem [W]

% comm

sat.ULRXAntennaDiameter = []; % Uplink Receiver Antenna Diameter in m
sat.ULRXAntennaGain = []; % Uplink Receiver Antenna Gain
sat.ULRXAntennaBeamwidth = []; % Uplink Receiver Antenna Beamwidth in deg
sat.DLDataRate = []; % Downlink Data Rate
sat.DLTXPower = []; % Downlink Transmitter Power in W
sat.DLTXAntennaBeamwidth = []; % Downlink Transmitter Antenna Beamwidth in deg
sat.DLTXAntennaGain = []; % Downlink Transmitter Antenna Gain
sat.DLTXAntennaDiameter = []; % Downlink Transmitter Antenna Diameter in m
sat.MassComm = []; % Mass of the communications subsystem [kg]
sat.PeakPowerComm = []; % Peak Power of the communications subsystem [W]
sat.AvgPowerComm = []; % Average Power of the communications subsystem [W]
sat.OffPowerComm = []; % Off Power of the communications subsystem [W]

% power
sat.MassPower = []; % Mass of the power subsystem [kg]
sat.SolarArrayPower = []; % Power generated by the solar arrays [W]
sat.SolarArrayArea = []; % Area of the solar arrays [m^2]

% thermal
sat.MassThermal = []; % Thermal subsystem mass [kg]
sat.PeakPowerThermal = []; % Thermal subsystem peak power requirements [W]
sat.AvgPowerThermal = []; % Thermal subsystem average power requirements [W]
sat.OffPowerThermal = []; % Thermal subsystem min power requirements [W]

% lifetime
sat.BC = []; % Ballistic Coefficient [kg/m^2]
sat.Drag = []; % Drag force [N]
sat.mdot = []; % Constant mass flow rate required for orbit maintenance [kg/s]
sat.MPropOrbit = []; % Propellant mass required for orbit maintenance for lifetime duration [kg]

% propulsion
sat.MassProp = []; % Propulsion subsystem mass [kg]
sat.PeakPowerProp = []; % Propulsion subsystem peak power requirements [W]
sat.AvgPowerProp = []; % Propulsion subsystem average power requirements [W]
sat.OffPowerProp = []; % Propulsion subsystem min power requirements [W]
sat.SwoopTime = []; % TOF of Hohmann swoop [min]
sat.DVperSwoop = []; % Delta V required to complete one swooping maneuver [m/s]
sat.DVLifetime = []; % Delta V required for lifetime of one swoop per day [m/s]
% OBDH

sat.MassOBDH = []; % OBDH subsystem mass [kg]
sat.PeakPowerOBDH = []; % OBDH subsystem peak power requirements [W]
sat.AvgPowerOBDH = []; % OBDH subsystem average power requirements [W]
sat.OffPowerOBDH = []; % OBDH subsystem min power requirements [W]

% structure

sat.Volume = []; % Estimated Spacecraft volume [kg/m^3]
sat.Inertia = []; % Estimated moment of inertia [kg*m^2]
sat.MassStruct = []; % Estimated structural mass [kg]

% system

sat.Cost = []; % spacecraft system cost estimate [FYOO$k]
sat.Power = []; % spacecraft power use [W]

% STK

sat.InTheaterAccessDuration = []; % Representative Access Duration for the In Theater scenario (calc w/ STK)
sat.CoverageTime = []; % time to achieve 100% coverage [min]
sat.FinalCoverage = []; % coverage at end of simulation period
sat.RevisitTimeMax = []; % maximum revisit time [min]
sat.RevisitTimeMean = []; % mean revisit time [min]
sat.ResponseTimeMax = []; % maximum revisit time [min]
sat.ResponseTimeMean = []; % mean revisit time [min]
sat.CommDurationMax = []; % Max Duration of the communication link between GS and satellites
sat.CommDurationMean = []; % Mean Duration of the communication link between GS and satellites
sat.CommDurationMin = []; % Min Duration of the communication link between GS and satellites
sat.CommDurationMin = []; % Total Duration of the communication link between GS and satellites
sat.CommRespTimeMax = []; % Max Response time for the satellites with the ground stations
sat.CommRespTimeMean = []; % Mean Response time for the satellites with the ground stations
sat.NImagesPerDay = []; % Number of images collected per day per target

% Plot

sat.Fields = []; % Parameters to plot by sat_plot.
return;
% end sat_initialize.m

B.1.3 'sat_stk.m'

function [sat_out] = sat_stk(sat_in, conid)

% SAT_STK
% sat = sat_stk(sat)
%
% This function utilizes STK to calculate Figures of Merit such as
% Revisit Time, Coverage, etc. It defines a scenario with one ground
% station with an antenna, Nsats satellites in Nplanes, one optical
% sensor and one communications sensor in each satellite. It computes
% revisit time and time to global coverage. The outputs are copied into
% the satellite structure.
%
% Jared Krueger <jkrue@mit.edu>
% Daniel Selva <dseleva@mit.edu>
% Matthew Smith <m_smith@mit.edu>
%
% 1 Nov 2008

% Constants

dtr = pi/180;
Re = 6378; % Radius of the Earth in km
scenario_path = '/Scenario/CLOSESat/';
scenario_name = 'CLOSESat';
% Inputs

nsats = sat_in.Nsats; % number of sats per plane
nplanes = sat_in.Nplanes; % number of orbital planes
inc = sat_in.Inclination.*dtr; % Inclination in rad
h = sat_in.Altitude;
angle_h = sat_in.AngleHorizSTK; % horizontal half angle for the sensor
angle_v = sat_in.AngleVertSTK; % vertical half angle for the sensor
D_SAT = sat_in.ULRXAntennaDiameter;
n_gs = sat_in.NGroundStations; % ground station coordinates
lat_gs = sat_in.LatGroundStations;
lon_gs = sat_in.LonGroundStations;
f_GHz = sat_in.DLFrequency/1E9;
D_GS = sat_in.ULTXAntennaDiameter; % Diameter of the ground station antenna

% Internal calculations

% Create a ground station, set locations, add a sensor to each
% Inner calculations

for i = 1:n_gs
    facility_name = ['GS' num2str(i)];
    facility_path = ['/Scenario/' scenario_name '/Facility/' facility_name '/'];

    % create a ground station
    stkNewObj(scenario_path, 'Facility', facility_name);

    % set location
    LLApos = [dtr*lat_gs(i); dtr*lon_gs(i)];
    stkSetFacPosLLA(facility_path, LLApos);

    % add a sensor to the ground station
    stkNewObj(facility_path, 'Sensor', 'Antenna');
% set sensor properties
fac_sensor_path = ['./Scenario/ scenario_name ''/Facility/' facility_name ''/Sensor/Antenna'];
stkSetSensor(conid, fac_sensor_path, 'HalfPower', f_GHz, D_GS);
end

% Create a Coverage Grid

coverage_name = 'Coverage1';
stkNewObj(scenario_path, 'CoverageDefinition', coverage_name);
coverage_path = ['./Scenario/ scenario_name ''/CoverageDefinition''/ coverage_name '/'];
%Set coverage properties (latitude bounds) using stkSetCoverageBounds
lat_min = -70;
lat_max = +70;
stkSetCoverageBounds(conid, coverage_path, lat_min, lat_max);

% Create Figure of Merit: Revisit Time

FOM_name1 = 'Revisit_Time';
stkNewObj(coverage_path, 'FigureOfMerit', FOM_name1);
FOM_path1 = [coverage_path 'FigureOfMerit/ ' FOM_name1 ' /'];
stkSetCoverageFOM(conid, FOM_path1, 'RevisitTime');

% Create Figure of Merit: Response Time

FOM_name2 = 'Response_Time';
stkNewObj(coverage_path, 'FigureOfMerit', FOM_name2);
FOM_path2 = [coverage_path 'FigureOfMerit/ ' FOM_name2 ' /'];
stkSetCoverageFOM(conid, FOM_path2, 'ResponseTime');
% Create constellation with nplanes separated Delta_RAAN = 180/nplanes deg
% and nsats separated by Delta_Mean_Anom= 360/nsats deg

% -----------------------------------------------

% constellations

% tStart = 0;
% tStop = 86400.0;
% tStep = 3.0;
% semimajorAxis = (Re + h)*1000;

% Calculate the RAAN and mean anomaly of each satellite
raan = zeros(nsats*nplanes, 1);
MeanAnomaly = zeros(nsats*nplanes, 1);
z = 1;
for i=1:nplanes
    for j=1:nsats
        raan(z) = 180/nplanes.*(i-1);
        MeanAnomaly(z) = 360/nsats*(j-1);
        z = z + 1;
    end
end

for n=1:(nsats*nplanes)
    sat_name = ['Sat' num2str(n)];
%
% Create one satellite
stkNewObj(scenario_path, 'Satellite', sat_name);
satellite_path = ['/Scenario/' scenario_name '/Satellite/' sat_name '/'];

%
% Assign orbit properties to the satellite
stkSetPropClassical(['/Scenario/' scenario_name '/Satellite/' sat_name '/'], ...
    'J2Perturbation', 'J2000', tStart, tStop, tStep, 0, semimajorAxis, 0.0, inc, 0.0, raan(n).*dtr, ...
    MeanAnomaly(n).*dtr);

%
% Add a sensor (payload) to the satellite

%-----------------------------------------------
s = 1;
sensor_name = ['Sensor' num2str(s)];
stkNewObj(satellite_path, 'Sensor', sensor_name);
sensor_path = ['/Scenario/' scenario_name '/Satellite/' sat_name '/Sensor/' sensor_name];

% Set the satellite sensor using custom function (Rectangular)
stkSetSensor(conid, sensor_path, 'Rectangular', angle_v, angle_h);

% Add a sensor (communications) to the satellite
antenna_name = ['Antenna' num2str(s)];
stkNewObj(['/Scenario/' scenario_name '/Satellite/' sat_name '/'], 'Sensor', antenna_name);
antenna_path = ['/Scenario/' scenario_name '/Satellite/' sat_name '/Sensor/' antenna_name];

% Set the communications sensor using custom function (Simple Cone)
stkSetSensor(conid, antenna_path, 'HalfPower', f_GHz, D_SAT);

% Assign satellite's sensor as an asset to the coverage
stkSetCoverageAsset(conid, coverage_path, sensor_path);

% Assign both ground stations as assets for each satellite's antenna
for m = 1:n_gs
    facility_name = ['GS' num2str(m)];
    fac_sensor_path = ['/Scenario/' scenario_name '/Facility/' facility_name '/Sensor/Antenna
stkSetCoverageAsset(conid, antenna_path, fac_sensor_path);
end

end

% ------------------------------------------------------------
% Compute communications metrics
% ------------------------------------------------------------

% COMPUTE COMMUNICATION LINK DURATION (MAX, MEAN, MIN, TOTAL)

% Create a file containing the paths of the antennas
targetsfile = 'C:\Documents and Settings\ja21441\My Documents\MATLAB\CRISIS SVN\targets.txt';
fid = fopen(targetsfile,'w');

for n=1:(nsats*nplanes)
    sat_name = ['Sat' num2str(n)];
    target_path = ['Satellite/' sat_name '/Sensor/Antenna/'];
    fprintf(fid,'%s
',target_path);
end
fclose(fid);

targetsfile = ['"' targetsfile '"'];

% Assign targets to ground station antennas using the file
for i = 1:n_gs
    facility_name = ['GS' num2str(i)];
    fac_sensor_path = ['/Scenario/' scenario_name '/Facility/' facility_name '/Sensor/Antenna'];

    % Assign the targets in the file to the ground station
    AssignTargetToSensor(conid, fac_sensor_path, targetsfile);
end

% Initializations for computing performance
mean_comm_time = zeros(nsats*nplanes,1);
max_comm_time = zeros(nsats*nplanes,1);
min_comm_time = zeros(nsats*nplanes,1);
total_comm_time = zeros(nsats*nplanes,1);

min_resp_time = zeros(nsats*nplanes,1);
max_resp_time = zeros(nsats*nplanes,1);
mean_resp_time = zeros(nsats*nplanes,1);

for i = 1:(nsats*nplanes)
    sat_name = ['Sat' num2str(i)];
    sat_antenna_path = [''/Scenario/' scenario_name '/Satellite/' sat_name '/Sensor/Antenna/'];

Compute Access of each sat to both GS

```matlab
CALL = ['Cov,RM ' sat_antenna_path ' Access Compute "Coverage"'];
RESULTS = stkExec(conid,CALL);
NA = size(RESULTS,1)-2;
dur_access = zeros(NA,1);
for k = 1:NA
    dur_access(k) = sscanf(RESULTS(k+1,:), '%d,X[^,],%f');
end
```

Compute max, mean, min and total access duration for each satellite

```matlab
MEAN_COMM_TIME(i) = mean(dur_access);
MAX_COMM_TIME(i) = max(dur_access);
MIN_COMM_TIME(i) = min(dur_access);
TOTAL_COMM_TIME(i) = sum(dur_access);
```

Define FOM Response Time for each satellite

```matlab
CALL = ['Cov ' sat_antenna_path ' FOMDefine Definition ResponseTime Compute Average'];
stkExec(conid,CALL);
```

Compute RT

```matlab
call = ['Cov ' sat_antenna_path ' Access Compute Export "FOM Value" "C:\Documents and Settings\ja21441\My Documents\MATLAB\CRISIS SVN\results.csv"'];
stkExec(conid,cald);
```

Read the file

```matlab
fid = fopen('C:\Documents and Settings\ja21441\My Documents\MATLAB\CRISIS SVN\results.csv','r');
C = textscan(fid,'%[^,],%f
','Headerlines',7,'BufSize',25000);
resp_time = C{1};
fclose(fid);

% Suppress the 1e6 from the results (points after the last access are
% fixed to 1e6 response time by STK)
l = 1;
for k = 1:length(resp_time)
    if resp_time(k) ~= 1000000
        resp_time2(l,1) = resp_time(k);
        l = l+1;
    else
        break;
    end
end

% Compute min, mean, max response time
min_resp_time(i) = min(resp_time2);
max_resp_time(i) = max(resp_time2);
mean_resp_time(i) = mean(resp_time2);

end

% Compute Optics metrics

% Access calculation
stkComputeAccess(conid, coverage_path);

% Time to 100% coverage
[cov_data, cov_names] = stkReport(coverage_path, 'Percent Coverage');
time = stkFindData(cov_data{2}, 'Time');  % # of seconds past start time
cov = stkFindData(cov_data{2}, '% Accum Coverage');  % accumulated coverage
coverage_time = NaN;
for i = 1:length(cov)
    if cov(i) >= 80
coverage_time = time(i);
break;
end
end

% Percent coverage at the end of simulation period
cov = cov(end);
final_cov = cov(end);

% Revisit time for targets
[rt_data, rt_names] = stkReport(FOM_path1, 'Value By Grid Point');
rt_value = stkFindData(rt_data{3}, 'FOM Value'); % revisit time by grid point
max_revisit_time = max(rt_value);
mean_revisit_time = mean(rt_value);

% Response time for targets
[rt_data, rt_names] = stkReport(FOM_path2, 'Value By Grid Point');
rt_value = stkFindData(rt_data{3}, 'FOM Value'); % revisit time by grid point
max_response_time = max(rt_value);
mean_response_time = mean(rt_value);

% Calculate number of images per day
NImagesPerDay = 86400/mean_revisit_time;

% Assign outputs

sat_out = sat_in;
sat.out.RevisitTimeMax = max_revisit_time;
sat_out.RevisitTimeMean = mean_revisit_time;
sat_out.ResponseTimeMax = max_response_time;
sat_out.ResponseTimeMean = mean_response_time;
sat_out.CoverageTime = coverage_time;
sat_out.FinalCoverage = final_cov;
sat_out.CommDurationMax = max(max_comm_time);
sat_out.CommDurationMean = mean(mean_comm_time);
sat_out.CommDurationMin = min(min_comm_time);
sat_out.CommDurationTotal = sum(total_comm_time);

sat_out.CommRespTimeMax = max(max_resp_time);
sat_out.CommRespTimeMean = mean(mean_resp_time);

sat_out.NImagesPerDay = NImagesPerDay;

% Close all objects except the scenario
% -------------------------------

objects = stkObjNames;
for i = length(objects):-1:2
    stkUnload(objects{i});
end

return;

% end sat_stk.m

B.1.4 ‘sat_prop.m’

function [sat_out] = sat_prop(sat_in)

% Model Inputs
% -------------------------------

h = sat_in.Altitude;
a = sat_in.Semimajor;
hp = sat_in.SwoopAlt;
V = sat_in.Velocity;
m = sat_in.Mass;
Isp = sat_in.SpecificImpulse;
Isp2 = sat_in.Isp;
mp = sat_in.MPropOrbit;
%Drag = sat_in.Drag;
Lifetime = sat_in.Lifetime;
%P = sat_in.Period;
BC = sat_in.BC;
NSwoopsPerDay = sat_in.NSwoopsPerDay;

% Calculations

% Delta_V for deorbiting
% DV = 1000*V*(1-sqrt((2*RE)/(2*RE+h))); % [m/s]

% Propellant mass for swooping
% mp = m*(1-exp(-(DV/(Isp*gravity)))); % [kg]

%Assume one swoop per day
NSwoops = Lifetime*365*NSwoopsPerDay;

%Determine density at swoop altitude
rho = (4*10^-8)*hp^-7.765*(1000)^3;

%Model swooping if applicable
if hp ~= 0
% Delta_V for lower orbit insertion (Hohmann) /swooping
at = (a+RE+hp)/2; %[km]
P = 2*pi*sqrt(at^3/MU);
VA = sqrt(MU/a); %[km/s]
%VB = sqrt(MU/(hp+RE)); %[km/s]
VtA = sqrt(MU*2/a-1/at));
VtB = sqrt(MU*(2/(hp+RE)-1/at));
Drag = 0.5*(rho/(1000^-3))*(VtB*1000^-2*m/BC;
DVA = abs(VtA-VA)*1000; %[m/s]
DVAt = DVA*NSwoops;
mprop = Drag*m*0.1*P/9.81/Isp^2; %[kg]
DVB = gravity*Isp2*log((m+mprop)/m); %Delta V to overcome drag
DVBt = DVB*NSwoops;
DV1 = DVA+DVB; % Delta V per swoop [m/s]
DV2 = DVAt+DVSt; %Lifetime Delta V
TOF = pi/60*sqrt(at^3/MU); %[min]
mp2 = m*(1-exp(-(DVAt/(Isp*gravity)))+m*(1-exp(-(DVSt/(Isp2*gravity)))); % [kg]
else
    mp2 = 0;
end

% Mass and power (TBI!)
mass = mp + mp2 + 3; %[kg] Sum of mass of propellants for orbit maintenance, swooping maneuvers, and structure
power = Drag/0.055*1000; %[W] Power for electric propulsion orbit maintenance

% Model Outputs
%--------------------------------------------------------------
sat_out = sat_in;
sat_out.SwoopTime = TOF;
sat_out.MassProp = mass;
sat_out.PeakPowerProp = power;
sat_out.AvgPowerProp = 0.7*power;
sat_out.OffPowerProp = 0.2*power;
sat_out.DVperSwoop = DV1;
sat_out.DVLifetime = DV2;
return

B.1.5 ‘sat_lifetime.m’

function [sat_out] = sat_lifetime(sat_in)

% sat_lifetime
% sat_out = sat_lifetime(sat_in);
%
% Function to model the CLOSeSat drag and propellant required for orbital maintenance.

% Jared Krueger <jkrue@mit.edu>

% 22 May 2010

%Inputs from sat_in
h = sat_in.Altitude; %[km]
V = sat_in.Velocity; %[km/s]
Vp = sat_in.VelocityPer; %[km/s]
Va = sat_in.VelocityApo; %[km/s]
rho = sat_in.Density; %[kg/km^3]
SA = sat_in.SurfaceArea; %[m^2]
Cd = sat_in.DragCoefficient;
%BC = sat_in.BC; %[kg/m^2]
Lifetime = sat_in.Lifetime; %[years]
m = sat_in.Mass; %[kg]
Isp = sat_in.Isp;
D = sat_in.Aperture;

%Outputs
BC = m/Cd/(pi*(D/2)^2); %[kg/m^2]
Drag = 0.5*(rho/(1000^3))*(V*1000)^2*m/BC; % [N]
mdot = Drag/9.81/Isp; %[kg/s]
sat_out = sat_in;
sat_out.MPropOrbit = mdot*Lifetime*86400*365*m; %[kg of propellant for mission]
sat_out.BC = BC;
sat_out.Drag = Drag;
sat_out.mdot = mdot;

return;
Bibliography


