Tradespace Analysis for Earth Observation Constellations: A Value Driven Approach

by

Eric Magliarditi

B.S., Washington University in St. Louis (2018)
B.S.B.A.,Washington University in St. Louis (2018)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2020

Author . . . .
Department of Aeronautics and Astronautics
August 18, 2020

Certified by . . . .
Olivier de Weck
Professor, Aeronautics and Astronautics and Engineering Systems
Thesis Supervisor

Certified by . . . .
Afreen Siddiqi
Research Scientist, Aeronautics and Astronautics
Thesis Supervisor

Accepted by . . . .
Zoltan Spakovszky
Professor, Aeronautics and Astronautics
Chair, Graduate Program Committee

Tradespace Analysis for Earth Observation Constellations: A Value Driven Approach

by

Eric Magliarditi

Submitted to the Department of Aeronautics and Astronautics
on August 18, 2020, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

In recent years, the Earth Observation (EO) industry has experienced rapid growth and development, but the tools to analyze EO constellations and space systems are lagging behind the curve. In particular, as costs continue to decrease to the point where constellations are treated like commodities, it is critical to look past cost and towards value creation when designing future EO constellations. In particular, there is an increasing need for value driven tradespace methods that can assist in the development of future EO space systems. Although there exist a few value driven approaches towards EO constellation design, they tend to be non-general and rely on detailed information such as expert knowledge or detailed probability distributions. Due to these limitations and the growing need for value driven trade methods, this thesis presents a new quantitative approach, using a novel Value Function, that can be applied generally to EO constellation trade studies.

This thesis hypothesizes that the value generated by an EO constellation can be generalized in that the quantity and quality of data drives value to various stakeholders. By focusing on this simple hypothesis, this thesis will derive the Value Function to highlight these key attributes. The Value Function can be applied to a EO constellation, and using this new metric, various constellations can be compared against one another in a standardized way. To help show this, three case studies are used.

The first case study, Case Study A, looks at two existing EO constellations: Landsat 8 and RapidEye. Their tradespaces are re-examined, and it is shown that the designs that are currently in operation are not optimal. There exists a single satellite Landsat-like architecture that generates 19.44% more value as compared to the current Landsat satellite in orbit. There also exists a five satellite RapidEye-like constellation that generates 35.76% more value as compared to the existing RapidEye constellation.

The second case study, Case Study B, examines the make versus buy decision. In particular, it provides a hypothetical case study to determine if the Governor of California should purchase satellite imagery directly from Planet Labs, or if the state should fund a new constellation. This case study that shows the importance of understanding both costs and benefits when making important and high-investment decisions. For example, this thesis will discover an architecture that has 2 satellites,

where each satellite is similar to a Planet Flock 2p CubeSat, and an orbital geometry that emphasizes California as compared to global coverage. The value ratio of this architecture is 5.01, and it is always greater than the cost ratio. This implies that it is in the state's best interest to build the constellation as opposed to purchasing imagery directly. Other architectures will be shown that do not lead to this same conclusion.

The third case study, Case Study C, examines a proposed Synthetic Aperture Radar small satellite constellation, MicroX-SAR, by exploring a small set of potential architectures. These architectures are compared quantitatively using the Value Framework presented in this thesis, and it is shown that proposed constellations that are Sun-synchronous and at altitudes of 600 km generate the most value for a global region of interest. This information can be used when designing the mission parameters, such as orbital geometry, that will be important for constellation implementation.

Overall, this thesis contributes a new model to evaluate EO constellations that relies on value driven methods as opposed to commonly used cost based metrics.

Thesis Supervisor: Olivier de Weck
Title: Professor, Aeronautics and Astronautics and Engineering Systems

Thesis Supervisor: Afreen Siddiqi
Title: Research Scientist, Aeronautics and Astronautics

Acknowledgments

First and foremost, I would like to thank my advisors: Dr. Oli de Weck and Dr. Afreen Siddiqi. Both of your guidance and support has been instrumental towards my success at MIT. Afreen, you helped guide my research direction, introduced me to new ideas and concepts, and pushed me to be a better researcher. My thesis would not be what it is today, and TAT-C would not be what it is today without your continued support over the past two years. Oli, I cannot express my gratitude enough for taking me in as a student back in 2018. When I was a junior in undergrad, I made a decision to focus more on engineering, and follow my passion for aerospace engineering. I knew it would be a challenging road, and I would need a great deal of assistance moving forward. When you took me in as a student in SERG, you gave me the confidence and support I needed to make the most of my time at MIT. During the summer of 2017, I spent my summer at an investment bank. Three years later, and I am now following my dream and working at SpaceX. This would not have been possible without your support, and I will forever be grateful for that. I will never forget our brainstorming sessions, and I look forward to see what other impressive academic ideas and concepts that you develop in the future.

Before I say thank you to my friends and family, I would like to express my gratitude for the financial support that enabled me to learn and grow over the past two years. The support provided by TAT-C funding during my first year not only gave me the financial means to attend MIT, but the project itself taught me extremely valuable technical skills that I will use for the years to come. My second year studies were funded primarily through the MIT-KACST CCES program, and I am extremely grateful for their financial support. Similar to TAT-C, the technical skills I gained as a research assistant on this project have opened my eyes to a new field of exciting research that I continue to work on even out of academia.

I would also like to say thank you to Beth Marois, Julie Finn, and all the other administrators that help make life at MIT much smoother and less complicated.

I would particularly like to express my gratitude and appreciation to our custodial staff, and in particular, to Craig. Craig, not only would you keep our lab space clean, but I always looked forward to our daily chats at 4pm. It was great to take a break from the work I was completing, and talking to you about all things sports, and even more as we got to know each other better. I truly wish you and your family well, and I am extremely sad that I could not say goodbye in person due to COVID-19. Keep doing what your doing, and make sure to interrupt the lab even when I am gone. It is good for all of the students to take a break and relax! Next time I am in Boston, I will make sure to swing by and say hello!

To my family: Mom, Dad, Nikki & Alexandra - I want you all to know I could not have done it without your love and support. I know it was a bit of a surprise when I told all of you I would be going to grad school, but you all understood just how important this move would be. You all have taught me so much, provided for me in a multitude of ways, and given me everything I needed and more. This is all a family

member can ask for, and I am extremely grateful for this.

Last, but definitely not least, I would like to say thank you to Groux. At the time of writing, we have been together for almost 7 full years. I have gotten to know every side to you, and you have gotten to know every ounce of me. Without your constant support, I would not be where I am today. You know me better than I know myself, and you always know what is going on inside of my head even before I do. This has been pivotal for my growth and development. You know that I tend to stay where it is comfortable, but you know that I love to learn and explore. Because of this, you make me ask the hard questions and push me to become the best version of myself. There are countless times where the only reason I decided to travel, try a new food, etc., was because of your ability to challenge me in an extremely healthy and motivating way. When I was struggling to figure out what to do with my life back in 2017, you helped me realize that going to grad school would be the kick-start I needed. You visited MIT with me during that summer, even if it met more time long distance. You persevered through all of my schooling, and now we finally get to reap the fruits of our efforts. Groux, all I can really say is thank you. You have helped mold me into the person I am today, and I would not be where I am without your guidance, support, and love. I am excited to see what the future has in store for us, and I can't wait to continue the journey with you by my side.

Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Abbreviations

EO Earth Observation

LEO Low Earth Orbit

RGB Red-Green-Blue

TAT-C Tradespace Analysis Tool for Constellations

RAAN Right Ascension of the Ascending Node

GPS Global Positioning System

IR Infrared

UV Ultra Violent

OLI Operational Land Imager

SAR Synthetic Aperture Radar
SNR Signal-to-Noise Ratio

GSD Ground Sampling Distance
GSFC Goddard Space Flight Center
GUI Graphical User Interface

JSON JavaScript Object Notation

FF Full Factorial

GMAT General Mission Analysis Tool

API Application Programming Interface

ECI Earth Centered Inertial Frame

ROI Region of Interest

POI Point of Interest

OMM Orbit Maintenance Module

EDA Effective Data Acquired

GB Gigabyte

FOV Field of View

SSO Sun-synchronous Orbit

NPV Net Present Value

USGS United States Geological Survey

IP Integer Program

ML Machine Learning

THIS PAGE INTENTIONALLY LEFT BLANK

Variables

N_s Number of photocarriers/electrons at the detector

N_t Total number of noise electrons

σ_N Achievable noise equivalent reflectivity

σ_0 Distributed target reflectivity

$\mathbf{R}$ Range vector from the satellite to the target ground point

$\mathbf{S}$ Position vector of the satellite in the Earth centered initial frame

$\mathbf{T}$ Position vector of the target ground point in the Earth centered initial frame

γ Look Angle

θ_i Incidence Angle

ϕ Depression Angle

h Orbital Altitude

R_E Equatorial Radius of Earth

$\mathbf{R^{drv}}$ Derived Range Vector from the *derived* satellite to the target ground point

$\mathbf{S^{drv}}$ Derived Satellite position from the *derived* satellite

θ_i^{drv} Derived Look angle from satellite derived position

N_r Number of read out noise electrons

N_{ph} Number of photons at the detector

Q_E Quantum Efficiency of the Detector

$R_T^{det}|_{ph}$ Rate of photons at the detector

T_i Integration time of ground pixel

$R_T^{sen}|_{ph}$ Rate of photons at sensor aperture

τ_{op} Optics System Efficiency

$R_T^{rad}|_{ph}$ Rate of photons radiated

D_{ap} Aperture Diameter

L_T Total Radiance from the target area

A_{gp} Observation ground pixel area

L_E Radiance from Earth in the directon of target ground pixel

L_S^{uw} Upwelling reflected Solar radiance from the ground pixel

λ_1 Lower end wavelength of operating band

λ_2 Upper end wavelength of operating band

L_λ Plancks spectral blackbody radiance equation

τ_λ^{atm} Wavelength dependent atmospheric loss

Υ Planks constant

c Speed of light

λ Wavelengths

k_B Boltzmann constant

T Target equivalent blackbody temperature

$R_S^{uw}|_{ph}$ Upwelling photon rate from the ground pixel to the satellite

$R_S^{dw}|_{ph}$ Downwelling photon rate at ground pixel

L_S^{dw} Downwelling radiance at target observation ground pixel

r_{Solar} Solar radius

$\mathbf{V_{Sun2T}}$ Vector from Sun to target in ECI frame

L_S Radiance from the Sun

θ_i^{Solar} Solar incidence angle at ground pixel

$\mathbf{P_{Sun}}$ Position vector of the Sun

ρ_{CT} Cross track ground pixel resolution

ρ_{AT} Along track ground pixel resolution

ξ Instantaneous Field-of-View of the detector

d Detector width

f Focal length

T Scene noise temperature

B_T Chirp bandwidth

F_N System noise figure

L_{radar} Radar hardware related losses

L_{atmos} Two-way atmospheric losses

P_{avg} Average transmit power

G_A Gain of antenna

λ_0 Operating center wavelength of the radar

v_s Velocity of satellite

ψ_g Grazing angle to target ground pixel

η_{ap} Aperture efficiency of atenna

D_{az} Dimension of antenna in along-track direction

D_{elv} Dimension of antenna in cross-track direction

d_{sar} Duty cycle

P_T Peak transmit power

T_{eff} Effective pulse width

f_p Pulse repetition frequency

k ROI Index

R ROI Set

w_k ROI Weight

N_k Number of POIs within ROI k

A_i Number of Access Events of POI i

Q_{ij} Data collected of POI i during access event j

μ_{ij} Instrument Quality Metric of POI i during access event j

Chapter 1

Introduction

1.1 Motivation

The aerospace industry has recognized the benefits of deploying distributed small satellite constellations as opposed to single monolithic satellites. With recent technological advents such as component miniaturization, advanced data processing algorithms, and substantially cheaper launch costs, satellite providers have recognized the importance of thinking small and distributed when it comes to developing the next generation of satellite missions.

Some of the biggest gains in small satellite constellations have come from the Earth Observation (EO) sector. For example, Planet Labs Inc., has transformed the EO landscape by providing frequent Earth imagery on a daily basis. By focusing on large scale, small satellite constellations, Planet has been able to quickly adapt to market and environmental changes, and provide users with data that they require. The Planet constellation currently consists of about 150 distributed satellites that include over 130 Planet designed and manufactured Doves, 15 SkySats, and 5 RapidEye satellites [1]. By deploying such a large and diverse fleet of small satellites, Planet has been able to provide users with data and information that is enabling them to better understand the world we live in and is providing policy makers with the power to make more informed decisions [2] [3].

Although small satellite constellations are becoming the norm within the space and EO industries, there is still a lack of general understanding of how to model and design such large scale systems. Design formulation, tradespace exploration, and optimization criteria are just a few concepts that become much more difficult when satellite missions begin to scale in size and complexity. Because of this, it has become apparent that more sophisticated tools are needed in order to design and analyze future constellation designs. Also, it has become clear that standard single-objective metrics such as cost and risk are no longer sufficient to meet the complex goals of mission designers and stakeholders [4]. Thus, new computational tools as well as a more value-focused optimization scheme will be needed in order to design future

constellations that will help in the multitude of EO-driven applications [5]. Some key applications include the following:

1. Agriculture [6]

2. Urban Planning [7]

3. Disaster Relief [8]

4. Climate Change Analysis [9]

The purpose of this thesis is to examine the potential for both computational tools and value-driven optimization within the systems engineering of small satellite constellations. In particular, this thesis will focus on EO constellations as they represent a domain that rapidly growing in importance. With the tools presented in this thesis, future mission designers will hopefully have a much stronger ability to build EO satellite constellations that will be more efficient, have higher performance, and provide greater return on investment - thus helping promote the *intelligent* and rapid growth of small satellite constellations.

1.2   Research Questions

This thesis examines the tradespace of distributed EO constellations from both a methodological and conceptual framework. This will be done by examining and obtaining answers to these primary research questions.

1. How is value defined in the scope of designing and operating EO satellite constellations? In particular, what drives value within the EO design tradespace, to both primary and secondary stakeholders?

2. Can EO constellations be traded against one another using a generalized concept or metric of value that can be used for different applications, spectral bands, and Earth Science disciplines?

3. Assuming a generalized value measure exists, can it be used to help guide search strategies to find optimal EO constellation designs for a given mission?

4. What are the limitations when using a value-based approach towards constellation design, specifically, what level of fidelity or resolution is necessary to be useful and credible?

5. Can the value function presented in this thesis be applied to the design of constellations outside of EO, such as communications or deep space exploration systems?

The first two research questions look qualitatively at the concept of value based tradespace methods, and determine if they are necessary for rigorous analysis. The third and fourth research questions assume a value-based tradespace method exists, but looks at how capable it is in assisting the optimization process. The final research

question looks at synergies between various constellation domains, and whether disparate systems can use similar tools in order to design more intelligent space systems.

1.3   Scope & Research Contributions

This thesis has three primary goals:

1. Clarify the importance of using value-driven tradespace analysis as opposed to standard cost based methods.

2. Give a detailed derivation and walk through of the newly developed value function that has been built and iterated upon over the past two years.

3. Provide users of the NASA sponsored Tradespace Analysis Tool for Constellations (TAT-C) a simple and intuitive documentation to assist in future development and maintenance of the tool.

By focusing on these three goals, this thesis will be focusing on the design and analysis of satellite constellations, primarily with considerations of the EO domain. This thesis will give a comprehensive overview of small satellite EO constellations, with particular emphasis on the instruments or payloads that they host. This thesis will not look at specific technical aspects when it comes to small satellites, such as physical limitations of actuators or heaters, but rather assume that the spacecraft subsystems and components satisfy the baseline mission requirements in order to make intelligent decisions about the constellation as a whole. The only subsystem that will be examined in detail is the payload subsystem given that the most valuable output, information, of EO constellations comes from the instruments on board the satellite bus. The performance of these instruments will strongly influence the results of a successful trade study. This thesis will also not examine the impact of distributed ground stations or various communication delays or total downlink time. Although these are critical elements that can and should be modeled, the goal of this thesis is to examine constellations at a higher level of abstraction, meaning communications and downlink performance will be held constant for each architecture. Beyond the subsystems, constellation characteristics such as orbital altitude & inclination as well as the specific region of interest on the surface of the Earth will be of importance. By examining EO satellite constellations at the systems engineering level, rather than the subsystem level, this thesis will provide users with a tool and methodology to answer critical questions that will impact performance and cost of future EO satellites.

In its totality, this thesis contributes to the field by providing a novel approach to the design and analysis of small satellite constellations in Low Earth Orbit (LEO). In particular, it provides a value driven framework that can be generalized across constellation types for EO missions. This framework will assist in the design optimization of constellations, and will help create the next generation of small satellite EO constellations. The author hopes that this work will be utilized as a helpful starting point and refined further so that generalized valuation measures can be looked at across the heavily cost-focused space sector.

1.4 Thesis Overview

As discussed, this thesis will attempt to answer the above research questions as well as satisfy the goals listed above. In Chapter 2, this work will breakdown current trends in the EO industry. Chapter 2 will also provide a paradigm for EO constellations that will provide terms and definitions that will be critical to the understanding of this thesis. Current and past missions will be used as references in order to exemplify the terms used.

Once the general EO landscape is understood and defined, this thesis will transition to discuss the Tradespace Analysis Tool for Constellations (TAT-C). This tool will be used to simulate constellations and provide the mechanism used to make comparative trades of EO constellations. This chapter will provide a high level overview of the software architecture, as well as discuss its current capabilities and limitations. The chapter will breakdown each module within TAT-C and provide a qualitative and quantitative explanation of each. Breaking down TAT-C will enable future users to utilize the tool and build upon it given that it is published as open source software. Chapter 3 will enable Goal 3 to be satisfied.

Following the TAT-C description, this thesis will then dive into value-based evaluation approaches. Chapter 4 will begin with a detailed literature review, showing current capabilities and gaps when it comes to detailed value modeling. This thesis will then first provide an overview of the original value function derived for EO constellations, and then dive deeper in to the mechanism behind its derivation. Once the value function is defined, Chapter 5 will utilize the value function through three case studies. The first case study will look at two existing EO missions (Landsat and RapidEye), and re-run the tradespace analysis for these missions, but this time using the value function defined in this thesis. The second case study will show the importance of value-based optimization through the classic make vs buy decision. The third case study will conduct a proof-of-concept tradespace simulation for a proposed SAR satellite system.

Lastly, Chapter 6 will discuss future work, with an emphasis on machine learning optimization techniques, and then a detailed summary of the thesis will be provided in Chapter 7. The thesis breakdown can be visualized in Figure 1-1.
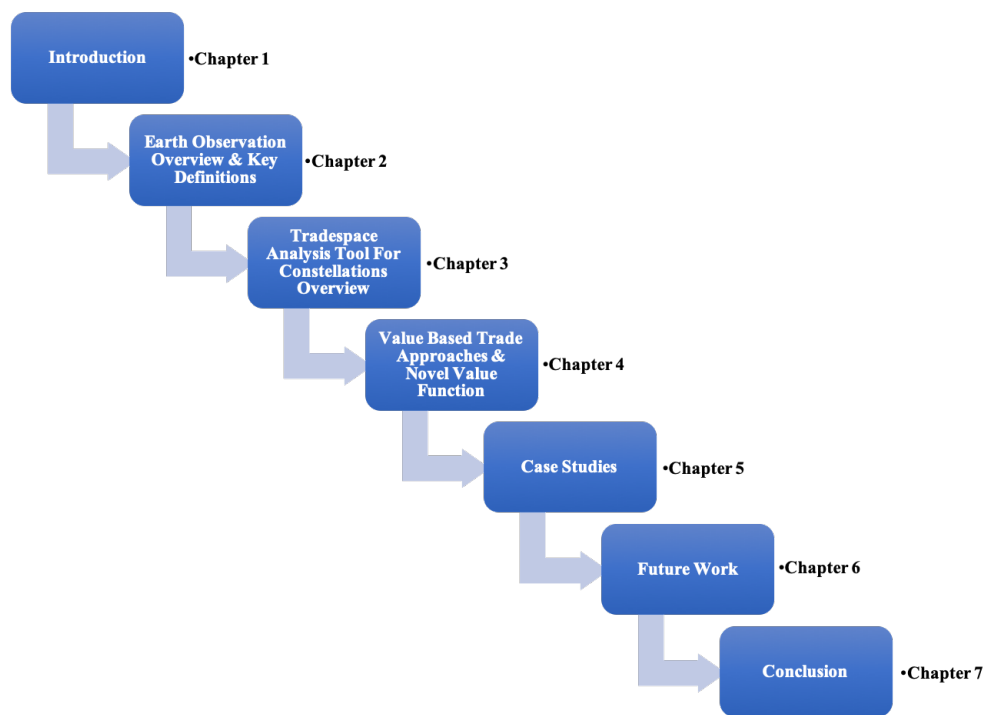
Figure 1-1: Thesis Flow Chart

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 2

Earth Observation Constellations

2.1 Current Earth Observation Market Overview

One of the primary motivating factors for the work in this thesis comes from the fact that the EO market is experiencing tremendous growth, and it is being driven by EO satellite constellations [10], [11]. Given the increased demand for data, and increased demand for highly specific, high-resolution, daily imagery; advanced, optimized constellations will be needed in order to satisfy market demand. It is also worth noting that the increase in computational techniques and capabilities has helped create even more demand for satellite imagery [12], [5]. Machine learning methods such as Convolutional Neural Networks (CNNs) have allowed basic satellite imagery to represent more than just a standard RGB image [13]. Now, substantial insights can be gathered that make the data provided even more valuable than before. We have come a long way from the days in the 1960s when intelligence analysts manually inspected images under a magnifying glass to detect features of interest. Planet Labs is one of the leading satellite operators and data providers in the world. They understood that the market was demanding strong insights from satellite imagery, and have created full scale analytic platforms to provide useful insights to end users [14]. Before diving into the details of specific EO terms that will assist in the remainder of the thesis, a couple of important EO trends need to be discussed because they help put into perspective the rise of EO constellations.

2.1.1 Earth Observation Trends

2.1.1.1 Diminishing Launch Costs

One of the most important trends that has helped enable the rise in the number of EO satellites launched is the large reduction in launch costs. Historically, the cost of launch was a large barrier to entry. Only well established and well funded organizations had the financial ability to launch assets into orbit. Due to advances in technology as well as a rise of market competition, the cost to launch an asset has decreased tremendously. References [15] and [16] provide a great summary of
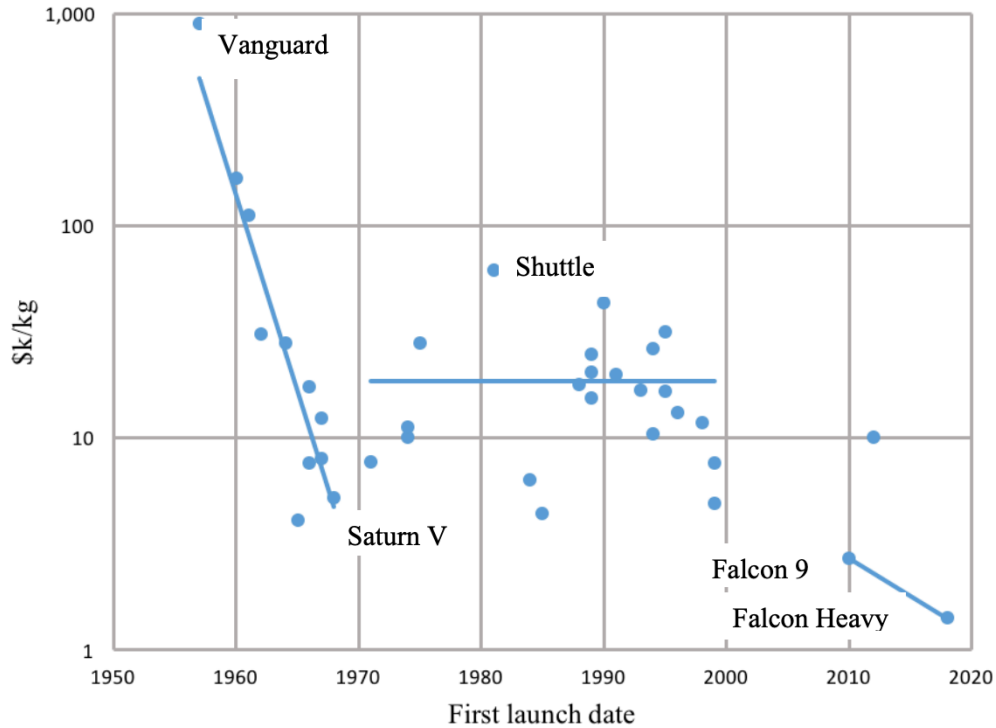
Figure 2-1: Decreasing Launch Costs from 1950 to 2020 [15]

the priming cost reduction mechanisms that have helped fuel this drop in launch cost. Figure 2-1 shows how the cost per kilogram has greatly diminished recently as well. A major contribution was made by SpaceX and their continually improving launch vehicle the Falcon9. Also, international launch providers such as the Indian Space Research Organization and their Polar Satellite Launch Vehicle have been very active in enabling EO constellations such as the Doves to be launched. Overall, the reduction in launch cost has enabled smaller satellite operators such as Planet to enter the landscape and transform the quantity and quality of EO data available to the public.

2.1.1.2 Small Satellite Revolution

In addition to decreasing launch costs, the miniaturization of satellites has also been an important EO trend. Historically, EO satellites have been large and expensive. Landsat 8, an important mission that will be discussed further in Section 2.2.2.1, cost roughly $855 million and has a mass of about 2,600 kg [17], [18]. With cost barriers to entry such as these, it was difficult to provide highly specialized and optimized constellations to end users. However, this all changed with the advent of small satellites, primarily CubeSats. A CubeSat is used to describe a small satellite that has a basic unit in the form of a 10 cm cube [19]. A standard 10 cm CubeSat has a form factor of 1U. Variations in form factor, such as 3U, are quite common and represent a 30 cm x 10 cm x 10 cm satellite. A 3U CubeSat is about the size

of a shoe box, and can have a mass of about 5 kg. When compared to the Landsat 8 satellite , this is a difference in mass of about 520 times. The minimization of EO satellites is important because they will come in at a much smaller cost and can be rapidly iterated in order to constantly add performance and new features that end users desire. Low launch cost and the advent of CubeSats has created a perfect storm where large barriers to entry have essentially been broken down, which has allowed more market competition, which in turn provides better services at lower costs. With these advances, the commercialization of space has become a reality [20]. To show just how important CubeSats have become, Figure 2-2 depicts the number of CubeSats launched from 2001 to 2018. Note, 2018 numbers are expected since the numbers come from Reference [19] which was published in October 2018.
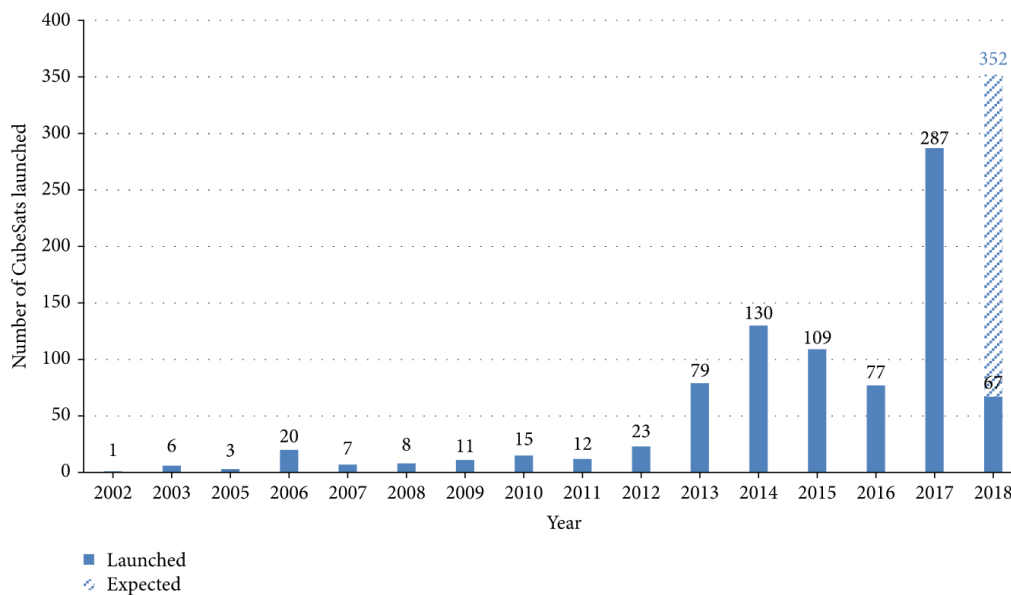
Figure 2-2: Number of CubeSat Launches from 2001 to 2018 [19]

Overall, with new technologies such as CubeSats, lower launch costs, and a rise in market providers, the EO constellation industry as a whole is experiencing substantial technical and financial growth. However, it is important to note that with advances in technologies come added complexities. Larger and more complex EO constellations are becoming the norm, and there are many design issues involved with these constellations [21]. Thus, efficient and intelligent means of comparison will be needed to keep up with the rapid development of EO constellations. As an example, one of the most important trade-offs is the one between revisit frequency and resolution (a proxy for image quality). This trade-off will be discussed in further detail throughout the thesis, but Figure 2-3 provides a plot of this trade-off [22]. In particular, it shows the difference in image resolution and refresh frequency, which is simply the inverse of revisit time.
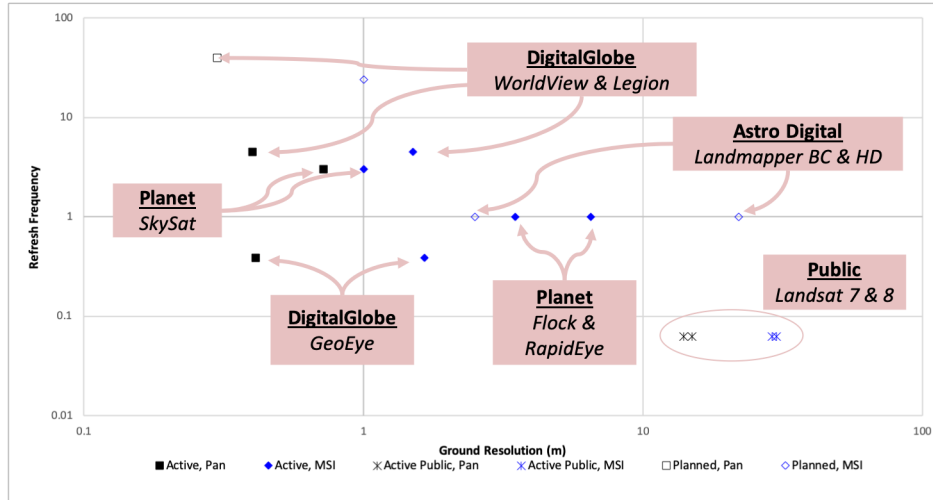
Figure 2-3: Ground Resolution and Refresh Frequency for Selected EO Satellites [22]

2.2 Earth Observation Paradigms & Terminology

In order to understand the individual components involved in EO constellations, it is critical to establish formal terms and definitions that will be used in the remainder of this thesis. Many of these terms will be used in the value function and TAT-C tool that will be discussed in the following chapters. This section will break down two broad concepts, EO constellations and EO instrument/payloads, and dive into specific cases of each. To begin, let's examine the highest level term of the complete satellite system, or a constellation.

2.2.1 Earth Observation Constellations & Architectures

A constellation is defined as a "set of satellites distributed over space intended to work together to achieve common objectives." [23] Given that constellations can vary in the number of satellites, the constellation pattern, the type of constellation, and even the orbital geometry, the design space for constellations becomes complex very quickly. Although there are many types of constellation architectures, this thesis will primarily focus on two constellation architectures: Homogeneous Walker, & Heterogeneous Walker Constellations. However, this thesis will also briefly discuss the following three constellation architectures: Precessing, Train, and Ad Hoc.

2.2.1.1 Homogeneous Walker Constellation

The homogeneous walker constellation was first developed in the 1980s by J.G. Walker [24]. In this type of constellation, all of the satellites within the constellation are placed in a Walker delta pattern where all satellites are in an inclined circular orbit. It is common to reference the total number of satellites as t, and there are s satellites within each p evenly spaced orbital plane [23],[24]. The major assumption within Homogeneous Walker Constellations is that all of the p orbital planes are at the same

inclination, i, and there are the same number of satellites per orbital plane. Due to this structure, the ascending nodes of the orbital planes are distributed uniformly around the equator at intervals of $360/p$ degrees, and within each plane, the satellites are phased evenly at intervals of $360/s$ degrees. Homogeneous walker constellations also assume that all of the orbital planes must have the same relationship to each other in phase difference, meaning the relative phasing must be an integer multiple of $360/t$. Because of this, Homogeneous walkers are usually described in shorthand using i: $t/p/f$ [23]. Homogeneous Walker Constellations are an important geometric pattern, and from a design point of view, mission planners must decide the orbital altitude and inclination, the number of satellites, and the number of orbital planes upfront in order to create a Walker constellation.
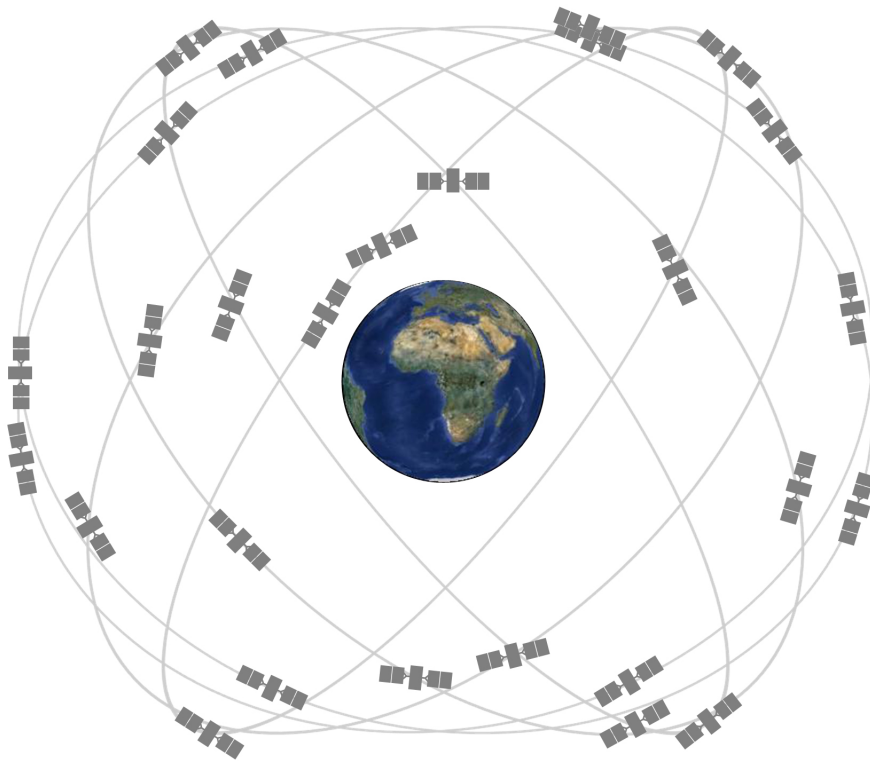
Figure 2-4: GPS Homogeneous Walker Constellation (Additional Non-Operating Satellites Included) i: $t/p/f$ - 55°: $24/6/90$°[25]

2.2.1.2   Heterogeneous Walker Constellation

The Heterogeneous Walker Constellation is a more dynamic case of the Homogeneous Walker Constellation. Like the homogeneous constellation, the satellites are still arranged in circular orbits with an equal number of satellites per orbital plane. However, the main difference between the heterogeneous and homogeneous Walker is that the former allows variations in the orbital planes from an inclination and altitude perspective [26]. These constellations allow designers more flexibility when designing a mission, but can dramatically increase the number of constellation architectures eval-

33

uated for analysis. This is because each combination of altitude and inclination must be evaluated, whereas the altitude and inclination are set as constant parameters for all orbital planes and are therefore uniform in the homogeneous case.

2.2.1.3   Precessing Constellation

Precessing Constellations are a more novel constellation architecture that rely on the gravitational perturbation effect that comes about due to the oblateness of the Earth, or the J_2 effect [26]. The J_2 effect tries to pull the orbital plane of a satellite towards the equatorial plane, but this resulting force does not change the inclination of the orbit, but rather the location of the right ascension of the ascending node (RAAN) of the satellite's orbit [23]. When a satellite is North of the equator, the oblateness of Earth causes a small pulling force southward that produces a torque which in turn changes the direction of the angular momentum vector [23]. This force is amplified at low orbits where the RAAN can rotate up to 7.7° per day [23]. In reality, Precessing Constellations are created by launching a collection of satellites, and initially each satellite is placed in a slightly different orbital plane. Over time, the J_2 effect will cause the RAAN and mean anomaly of these satellites to spread out thus creating the constellation network as seen in the rightmost image in Figure 2-5.
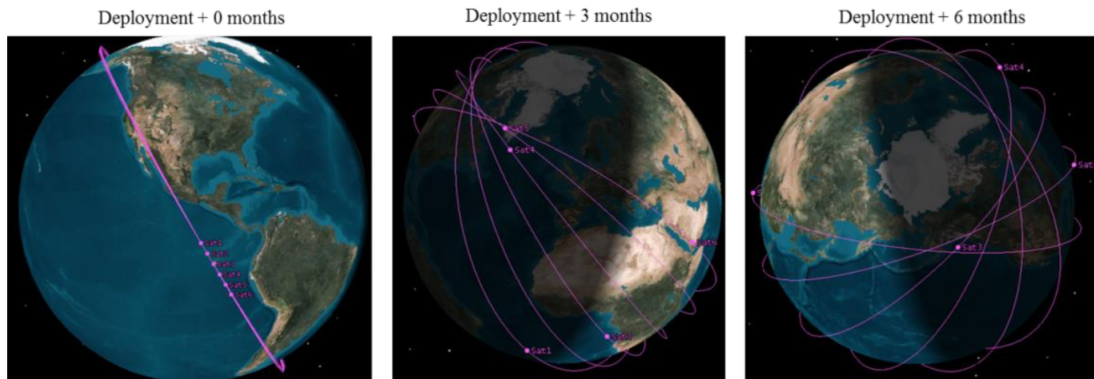
Figure 2-5: Evolution of a Precessing Constellation [26]

2.2.1.4   Train Constellation

Train Constellations are a unique class of constellation architectures where all satellites within the constellation are placed in the same orbital plane. When it comes to phasing, the satellites can be phased uniformly across the plane or separated by a specific phasing angle [27]. What makes Train constellations unique is that all satellites maintain the same ground track. It is important to note that Train constellations are not the same as a repeating ground track orbit that focuses on repeating a specific ground track over an interval of time. The NASA A-Train constellation is an example of a Train constellation that consists of 5 currently operating satellites [28].
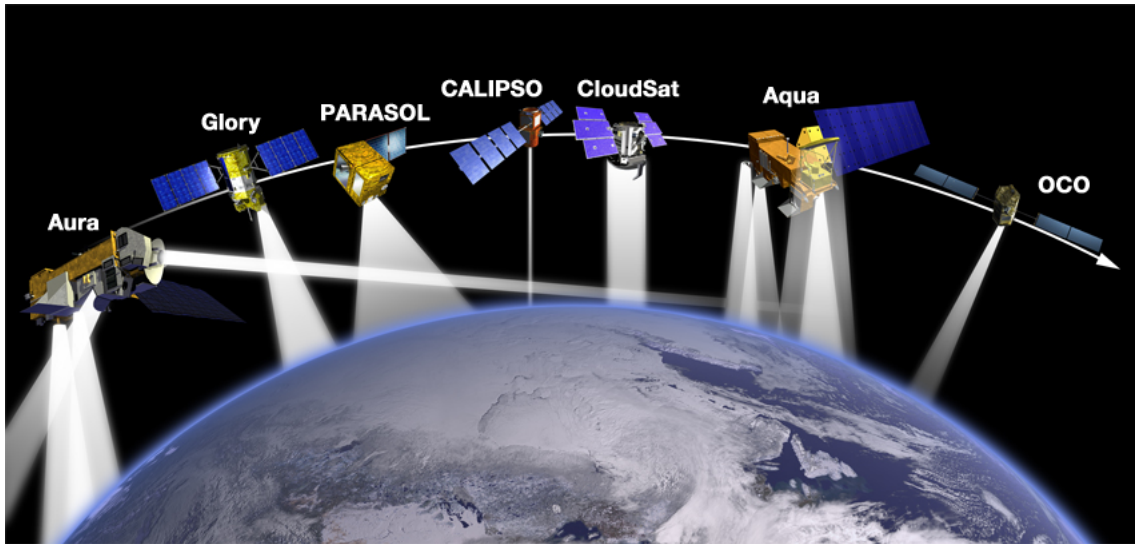
Figure 2-6: NASA A-Train Constellation
Glory & OCO Not Operational, [28]

2.2.1.5 Ad Hoc

The last classification of constellation architectures that will be discussed in this thesis are labeled as Ad Hoc. The Merriam Webster dictionary defines the term ad hoc as "formed or used for specific or immediate problems or needs." [29]. Following this definition, Ad Hoc constellations are created when satellite operators launch new spacecraft as needed. These constellations have become viable due to the fact that CubeSats can ride as secondary payloads on many standard launch vehicles [30]. These missions are built as launch opportunities arise, and can bring substantial cost benefits because of this [26]. Since Ad Hoc constellations can be almost random given that they arise due to piggyback launch opportunities rather than through planning, they have been somewhat challenging to appropriately model.

2.2.1.6 Example Missions

In this section we discuss some representative EO missions that fall into these four constellation architecture types.

Constellation Architecture	Example Mission	Description
Homogeneous Walker	Global Positioning System (GPS)	24 satellites, 6 orbital planes Altitude of 20,200 km Inclination of 55° *i: t/p/f:* 55°: 24/6/90°
Heterogeneous Walker	BeiDou Navigation Satellite System	22 satellites that fly in the GEO, MEO, and inclined geosynchronous orbital planes
Train	A-Train	5 EO satellites of varying nationality Sun-synchronous orbital inclination Altitude of 705 km.
Ad Hoc	Planet Labs Flock	Planet launches their PlanetScope satellites when launch opportunities arise, Over 170 satellites have been deployed.

Table 2.1: Existing Constellation Architecture Examples

2.2.2 Earth Observation Instrument Classification

One of the most performance critical subsystems involved in a EO satellite is the instrument or payload subsystem. This subsystem performs the imaging or the radiometric measurements needed to satisfy mission requirements and objectives. Since this thesis is focusing on EO systems, and in particular, EO imagery, this thesis will focus on two families of instruments: Passive Optical Scanners & Synthetic Aperture Radars.

2.2.2.1 Passive Optical Scanners

This thesis defines passive optical scanner instruments as any EO imaging payload that operates at the visible or near-visible wavelength spectrum. Near-visible wavelengths include infrared (IR) and ultra-violet (UV). A classic example of a passive optical scanner is the Operational Land Imager (OLI) that is currently being used on the Landsat-8 mission [31]. The Landsat-8 mission is an example of a large monolithic satellite mission, but has an immense history given that the program has been imaging the Earth since the 1970s [32]. Figure 2-7 below shows the actual OLI on the left and the schematic view on the right. Table 2.2 also shows the spectral bands and corresponding wavelengths of the bands the OLI operates within.
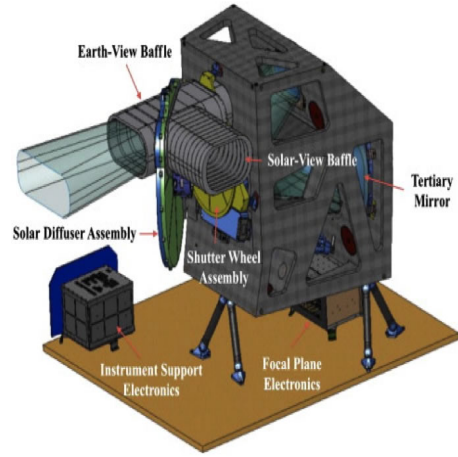
(a) Actual OLI [31]   (b) OLI Schematics [33]

Figure 2-7: Landsat-8 Operational Land Imager

Band Name	Central Wavelength [nm]
Coastal/Aerosol	443
Blue	482
Green	562
Pan	590
Red	655
NIR	865

Table 2.2: OLI Bands & Corresponding Wavelengths

As an example of a small EO satellite constellation, the Planet Labs' PlanetScope satellite contains a much smaller EO passive optical scanner. The PlanetScope satellite is a 3U-form CubeSat, implying that the entire satellite is 30cm x 10cm x 10cm [1]. The PlanetScope passive optical scanner collects imagery in the RGB and near IR bands [34]. Table 2.3 below breaks this down.
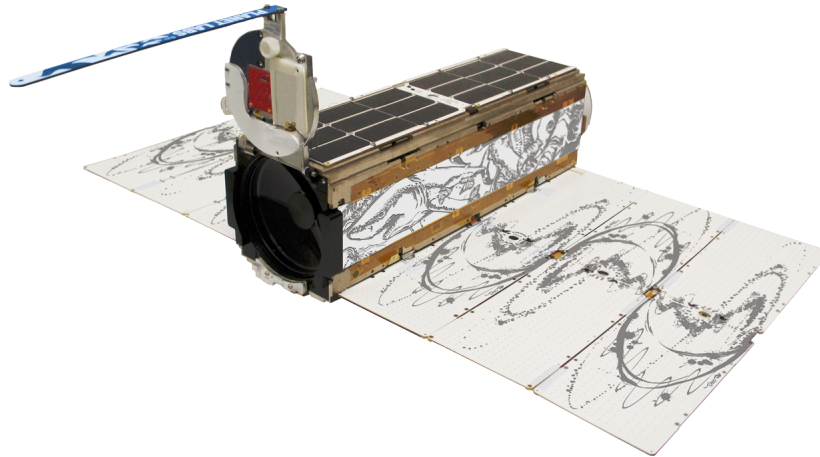
Figure 2-8: PlanetScope Satellite [1]

Band Name	Central Wavelength [nm]
Blue	475
Green	555
Red	657
Red Edge	710
NIR	805

Table 2.3: PlanetScope Visible Bands & Corresponding Wavelengths [34]

2.2.2.2   Synthetic Aperture Radar

Synthetic Aperture Radars (SAR) are a special case of EO imaging instruments that use active radar techniques. What makes SAR so powerful is it can effectively "see" through clouds and fog and even take images during the night [35]. SAR images represent a two dimensional map of the radar reflectivity of a target scene which includes dimensions of range and azimuth. For a more detailed description of SAR, see reference [35]. SAR instruments have been historically more expensive, but given new technologies and cost reductions, more small SAR satellite constellations are being developed and launched, as seen by the recent developments by the SAR startup Capella Space [36]. Figure 2-9 below shows an example image constructed by the SAR instrument from the Sentinel-1 satellite, in comparison to a passive optical scanner natural color image created by the Sentinel-2A satellite. Sentinel-1 operates at the C-band with a central frequency of 5.404 GHz [37]. Table 2.4 breaks down the Sentinel-2A satellite operating bands.
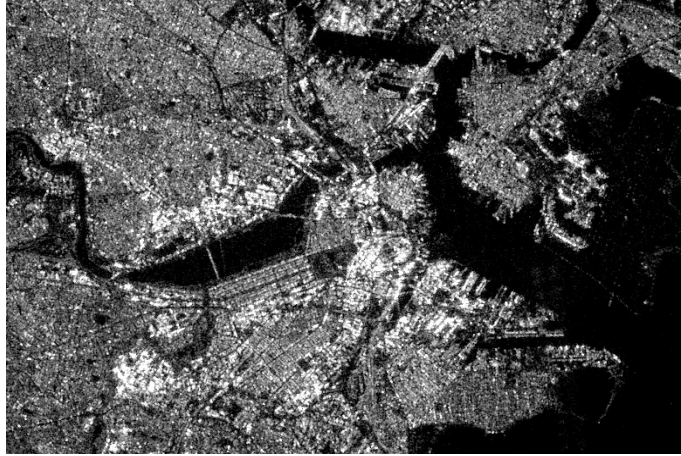
(a) SAR Image of Boston, MA [38]

(b) Passive Optical Scanner Image of Boston, MA [38]

Figure 2-9: SAR vs Passive Optical Scanner Imagery

Band Name	Central wavelength (nm)
Coastal aerosol	442.7
Blue	492.4
Green	559.8
Red	664.6
Vegetation Red Edge I	704.1
Vegetation Red Edge II	740.5
Vegetation Red Edge III	782.8
NIR	832.8
Narrow NIR	864.7
Water Vapor	945.1
SWIR I	1373.5
SWIR II	1613.7
SWIR III	2202.4

Table 2.4: Sentinel-2A Operating Bands [39]

2.2.3 Earth Observation Instrument Performance Metrics

There is a famous idiomatic expression within the business world that claims "You can't manage what you can't measure." What this implies is that no matter the infrastructure, human capital, etc., if the business can't measure its performance, it is doomed to fail. This concept can be directly applied to EO instrument performance. If there is no way to measure the performance of the instruments on board the satellite, then designers will be unable to evaluate constellations in an effective way. At the highest level, the value of payloads comes from the usefulness of their data. For example, if a scientists is interested in observing rainforests in South America, obtaining high quality imagery of the region would be valuable. Imagery that contains clouds, is low resolution, blurry, poorly calibrated, etc., would not perform well for the scientists. Thus, it is critical to evaluate these instruments based upon the *value* they can provide to end users. The following sections will break down the performance metrics for Passive Optical Scanners and SAR instruments.

2.2.3.1 Passive Optical Scanner Performance Metrics

When a passive optical scanner acquires an image in the visible spectrum, there are certain characteristics of this image that users can observe to determine the performance. There are many possible performance measurements, but as seen in Chapter 4, the value function derived in this thesis will utilize a combination of two critical performance metrics: the signal-to-noise ratio (SNR) and the ground pixel resolution. The ground pixel resolution is strongly determined by the point spread function (PSF) of the imager.

The SNR is an important and commonly used performance metric for visual-based imagery. At the most basic level, the SNR represents the ratio of of a desired signal

to the level of noise [40]. To put it in laymen terms, the SNR provides a metric to measure the amount of useful information as compared to irrelevant information (or noise) in the data. When it comes to image data from a passive optical scanner, the SNR can be used as a way for a scientist to understand the *quality* of an image once it is captured from the passive optical scanner. When the SNR is greater than 1, then the image provides the scientist with more useful information since there is more information as compared to noise. If the SNR is less than 1, then there is more noise in the image, which means less useful information for potential end users. For passive optical scanners, equation 2.1 below breaks down how the SNR is calculated [23].

$$SNR = \frac{N_s}{N_t} \tag{2.1}$$

N_s represents the number of photocarriers available, and N_t represents the total number of noise electrons. These two quantities have extensive calculations that relate to a multitude of variables, and this will be discussed further in Section 3.5.4. In summary, the SNR is an important metric because it provides end users with a simple and intuitive metric for the quality of an image. If a particular instrument captures images with higher SNR as compared to an alternative instrument, then scientists and mission planners would like to utilize the first instrument, holding all else constant.

The second critical performance metric for passive optical scanners is the ground pixel resolution. This thesis defines ground pixel resolution in a similar way to the ground sampling distance (GSD). The GSD is defined as the distance on the ground at which the passive optical scanner samples the scene [23]. To put it in a more concrete way, the ground pixel resolution represents the spatial distance of a single pixel on an image. For example, the Landsat-8 OLI has a ground pixel resolution of 30 meters, whereas the PlanetScope imager has a ground pixel resolution of 3 meters [32], [34]. This means that one pixel for the OLI represents an area of 30x30 meters, and the PlanetScope imager represents an area of 3x3 meters. One pixel for Landsat 8 covers approximately 100 pixels in a PlanetScope image.

As discussed above, the ground pixel resolution is dependent on the PSF. It relies on the PSF because no matter how good the quality of an imager, there are fundamental limits in resolution due to diffraction [41]. Diffraction causes a point source image to not appear as a focal plane but as a pattern of circles that get dimmer away from the center. This pattern is referred to as the PSF [41]. A helpful way to think about the PSF is to think of it as an impulse response of an imaging system, i.e. how the image system reacts when a single point source is in view [42]. Thus, the ground resolution is limited by the PSF of an imaging system. This can be expressed through the quality factor which is defined as the ratio between the pixel size and the diameter of the PSF. Figure 2-10 shows this relationship and how the relative sizing of the ground pixel and the PSF impact overall image quality. Given that the PSF impacts the fundamental physics of observation systems, it is easy to see how this concept can

have wide reaching consequences on ground pixel resolution and even SNR. This is beyond the scope of this thesis, but it is a critical concept that should be addressed further in future work.
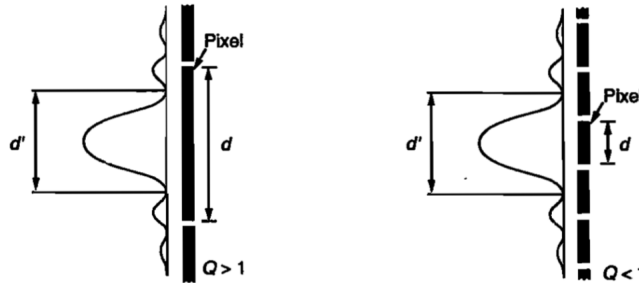
Fig. 9-12. **Effect of Varying Quality Factor.** Different sizing of the detector pixel with respect to the point spread function is shown by varying the Quality Factor, Q. A large quality factor results in the relative sizing in the diagram on the left and a low quality factor results in the relative sizing in the diagram on the right.

Figure 2-10: PSF Quality Factor Explanation [41]

The *lower* the ground pixel resolution, the *higher* the image resolution is. The higher the image resolution, the more distinct features can be examined. On average, higher resolution imagery is much more useful to end users, however this can vary depending on the specific use cases of the end user. To clearly demonstrate this point, Figure 2-11 shows the an image from the Landsat OLI, and the same image taken from the PlanetScope imager. Ground pixel resolution is determined from the focal length and aperture diameter of the passive optical scanner, as well as the orbital altitude and pointing direction of the satellite. Section 3.5.3 will go into further details of the orbital geometry involved and break down a more detailed formula for the ground pixel resolution. One area where Landsat 8 is superior to the imagery provided by PlanetScope's imager is the color calibration. While taken at different times one can clearly see that the colors in Figure 2-11 of the same area are quite different. The issue of calibration of distributed EO constellations is outside the scope of this thesis but could be addressed in future research.

(a) Landsat-8 OLI Boston, MA Image [43]
30 meter Resolution

(b) PlanetScope Boston, MA Image [43]
3 meter Resolution

Figure 2-11: Image Resolution Comparison

2.2.3.2 Synthetic Aperture Radar Performance Metric

Although the SNR of an image can be calculated for SAR instruments, this thesis will use a variation of SNR to measure SAR instrument performance. This thesis will utilize the achievable noise equivalent reflectivity, or σ_N for SAR instruments . This metric is also known as the Noise Equivalent Sigma Zero, or σ_{NEZ0}. This metric gives us the scaled inverse of the SNR of the SAR generated image, as equation 2.2 shows [44].

$$\sigma_N = \frac{\sigma_0}{SNR} \tag{2.2}$$

In this equation, σ_0 is the distributed target reflectivity. Given that this metric is a scaled inverse of the SNR, then smaller values implies a higher quality of image, or better performance of the instrument. Smaller σ_N implies large SNR, which as in the passive optical scanner case, means a better quality image and stronger instrument performance. It is also important to note that it is quite common for σ_0 to be negative. Because of this, smaller values also means a larger negative σ_N. σ_N is typically measured in units of decibels. Figure 2-12 shows how a changing σ_N can drastically impact the quality of the resulting image and reflect the performance of the instrument. The images are taken of the Capitol building in Washington D.C., and the lower the σ_N, the better the resolution and the more information is available from the image [44].

(a) Capital Building SAR Image [44]
$\sigma_N < $ -30 dB

(b) Capital Building SAR Image [44]
$\sigma_N = $ -15 dB

Figure 2-12: SAR Image Resolution Comparison

2.3 Summary

This section gave an overview of the EO market & landscape, as well as provided a technical breakdown of common EO paradigms and terminologies. With this foundation in place, this thesis will now move to describe a software tool, TAT-C, that will incorporate these EO concepts in order to help enable the tradespace analysis of EO satellite constellations.

Chapter 3

Trade-Space Analysis Tool for Constellations

3.1 Introduction & Software Overview

The Tradespace Analysis Tool for Constellations is the end product of a multi-year research effort led by the NASA Goddard Space Flight Center (GSFC). Outside of the GSFC, this project was developed by multiple team members from a few institutions that included the Stevens Institute of Technology, the Bay Area Environmental Research Institute (in collaboration with NASA Ames Research Center), Texas A&M University, and the Massachusetts Institute of Technology. Given the rise of distributed spacecraft missions and EO constellations, TAT-C was developed in order to provide a framework that enables Pre-Phase A constellation design exploration and optimization with respect to specified a-priori science goals [45]. To do this, TAT-C was built using a modular framework that centers on 6 key modules: Orbits & Coverage, Instrument, Cost & Risk, Launch, Tradespace Search Executive, and Value. Each of these modules takes in specific input information and generates specific output files that either feed into other modules or are returned to the end user. Figure 3-1 depicts the complete modular architecture for TAT-C.

3.1.1 TAT-C Execution Process

Users can run TAT-C through a command line interface, as well as through a graphical user interface (GUI). At the heart of TAT-C is a tradespace search JavaScript Object Notation (JSON) file that defines the search parameters that will be evaluated by the various modules. All the user has to do is generate this JSON file, either through manual creation or through the GUI, and TAT-C will take care of the rest. This JSON file consists of values for each of the Knowledge Base Object Schemas that were generated for TAT-C. Object schemas are helpful because they help standardize the module inputs and outputs. This enables human and machine readable documents and more formal semantic definitions for easy interoperability
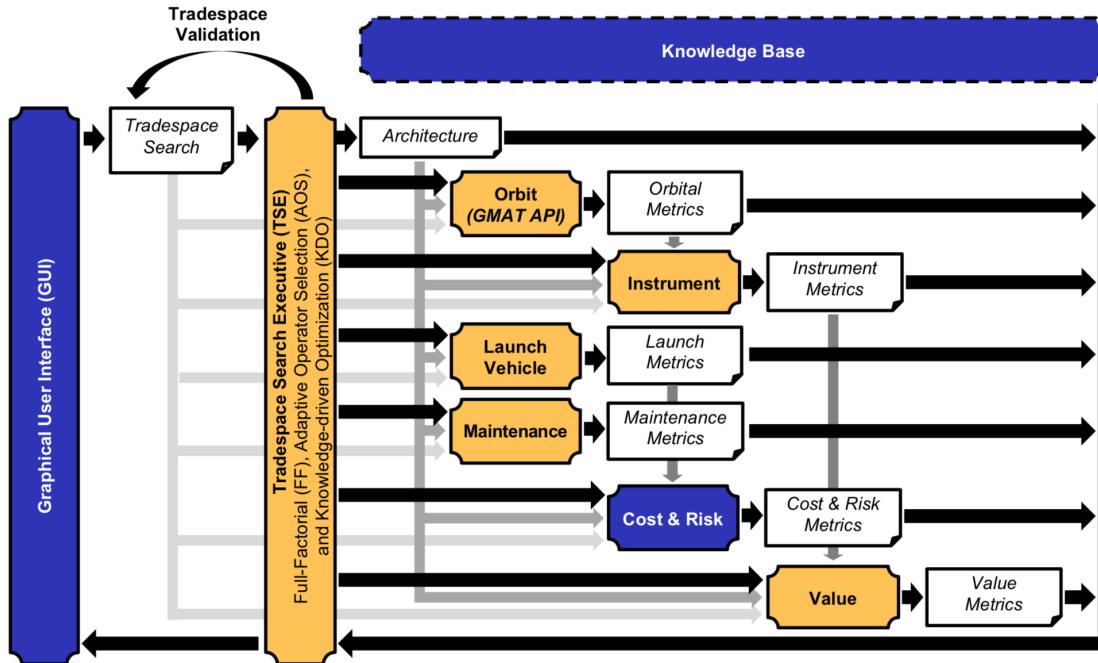
Figure 3-1: TAT-C Architecture [46]

between modules [47]. There are a wide variety of variables that the user can control in order to increase or decrease the size of the design space. For a detailed list of these variables and their corresponding object schema, please visit the following website: `https://tatckb.org/schema.html`. Some of the most important variables that users are able to explore include:

- Mission Start Date

- Mission Duration

- Constellation Architecture Type

- Number of Satellites

- Number of Orbital Planes

- Satellite Bus

- Instrument Specifications

- Ground Stations Used

- Search Strategy

Once all of the objects are defined in the JSON file, depending on the search strategy utilized, TAT-C generates a collection of constellation architectures that each module will evaluate. Once TAT-C is complete, the user will have an immense amount of data that relates to each constellation. The GUI contains some pre-built post-processing, but since the end user has access to all lower-level data outputs of TAT-C from each

module, they are then able to perform post-processing to their own unique standards. In summary, TAT-C was created in order to assist in the design and development of complex EO constellations by creating a tool that efficiently searches the exploration space for novel and interesting designs.

3.2   Current Capabilities

TAT-C is a powerful modeling tool, however, there are some important limitations to its capabilities that are critical to know before running a tradespace search. In order to better understand TAT-Cs capabilities, this thesis will enumerate the valid options for key design variables.

To start, let's examine the highest level of the design space, or the type of constellation architecture employed. The following table lists the constellation architectures that can currently be selected within the TAT-C platform. Note that a hybrid constellation implies a constellation that deploys a subset of satellites to one constellation architecture such as a Train and another subset of satellites to a different architecture such as Ad-Hoc. Note all of the constellation architectures were discussed in Section 2.2.1

Constellation Architecture
Homogeneous Walker
Heterogeneous Walker
Train
Ad-Hoc
Hybrid

Table 3.1: TAT-C Constellation Architecture Capability

A constellation's orbit is an important design choice and there are many variables involved with selecting the orbital trajectory of a satellite. The following table breaks down the key orbital variables and their respective bounds [48].

Orbital Variable	Bounds
Altitude	300-1000 km
Inclination	0° - 180°
True Anomaly	0° - 360°
RAAN	0° - 360°

Table 3.2: TAT-C Orbital Geometry Capability

The following instrument types are currently supported through the Instrument Module. Note a Basic Sensor represents any instrument that is not a Passive Optical Scanner or a SAR. This was modeled to assist in edge cases that users may create.

Instrument Type
Passive Optical Scanner
Synthetic Aperture Radar
Basic Sensor

Table 3.3: TAT-C Instrument Capability

Lastly, there exist three types of search strategies that end users can specify when running TAT-C. These search strategies tell TAT-C how to explore the tradespace, either using optimization or not.

Search Strategy
Full Factorial
Evolutionary Algorithm
Knowledge Driven Optimization

Table 3.4: TAT-C Search Strategy Capability

Given these parameters, there exists a large design space that is filled with many interesting and novel constellations. It is also worth noting that given the way TAT-C is formulated, it is also possible to create a "mega" constellation that consists of multiple standard constellations. This essentially means that a complete constellation can consist of multiple smaller constellations, where each small constellation may represent a different constellation type, deploy a different instrument, etc. This allows for more complex architectures to be generated, thus increasing the effectiveness of the tool overall! That being said, it is worth noting two key limitations in the TAT-C formulation.

3.2.1   TAT-C Limitations

The first limitation is that a specific constellation architecture can only utilize a single type of spacecraft. This means that if an end user wants to simulate a constellation with two satellites, where each satellite is unique, TAT-C cannot create a single tradespace search to express this. However, there is a standard workaround to this issue, and it has to deal with the "mega" constellations discussed previously. A user can specify two sub-constellations within a single "mega" constellation, where each sub-constellation contains one of the unique satellites. This is a practical work around, but makes it somewhat more difficult to model a heterogeneous constellation in terms of the satellites involved with the architecture. To make this even more clear, one can think of an example where a designer specifies three types of unique satellites, and TAT-C will generate a single constellation architecture for each satellite, rather than a single architecture that deploys all three satellites. If the end user is only examining how orbital altitude will affect performance, for example, and they only want to examine a 400 and 500 km altitude orbit, a total of 6 architectures will be

generated. Two architectures, one for each altitude, will be created for each unique satellite.

The second limitation is similar to the first limitation; only a single instrument can be modeled on a spacecraft. Real world satellites tend to place several instruments or sensors on the same satellite bus. Unfortunately, TAT-C only allows a single instrument to be placed within a satellite. This was a design choice made early in the TAT-C development process, and an important limiting factor. However, the user still has the ability to trade various instruments by specifying the varying instruments in the initial tradespace search JSON file. An architecture will be generated for each instrument, and if all else is held constant, the end user can make an educated trade on the instruments by examining each architecture in a post-processing sense.

3.3 Tradespace Search Exploration Process

One of the most important aspects of any tradespace exploration is the methodology used to generate and explore the design space. In TAT-C, there are two classes of search strategies: Full Factorial (FF) and Optimized.

FF enumeration is intuitive but quite costly. A FF search implies examining all possible combinations of an architecture generated by TAT-C. In a hypothetical example, a user specifies 10 different orbital altitudes, 20 orbital inclinations, and 40 satellites to trade on. A FF search will generate each combination of the three variables, resulting in 8000 different architectures. Although the FF search strategy is exhaustive, simulating 8000 architectures is computationally burdensome (see Table 3.6 in Section 3.4.2 for an execution time discussion). It would take a long time and then the user would have to parse through each architecture in order to find the set of architectures that they deem optimal.

Optimized search strategies on the other hand do not explore the entire design space. Rather, using heuristics or gradient-based methods, optimized search strategies will explore the trade space in an "intelligent" manner in order to find architectures that meet objective criteria described by the end user. Unfortunately, for most optimized search strategies, there are no algorithms and cases that guarantee global optimality, but empirically these methods can still produce relevant results that are superior to FF search, without exploring the entire design space. TAT-C deploys two types of optimized search strategies: an evolutionary algorithm & knowledge driven optimization.

This thesis will not dive into the specific details of these two optimized search strategies given that the work is outside of the scope the thesis. However, it is worth noting these algorithms for future use of the tool. At a very high level, the evolutionary algorithm utilizes variable length chromosomes to help find new architectures, and the knowledge driven search strategy uses existing knowledge from a database to help find better architectures faster [46]. The following references also provide excellent explanations of the TAT-C specific optimized search algorithms: Reference [49] &

Reference [50].

To show the power of these optimized algorithms, the following figures depict a FF search vs a search that deploys an evolutionary algorithm, respectively. The FF search generates 5000 architectures, where as the evolutionary algorithm only had to search through 500 architectures in order to converge to an 'optimal' solution [51]. This tradespace compares the lifecycle cost of an EO constellation versus the revisit time (lower is better for both). The utopia point in this case is in the lower left of this plot towards the origin.
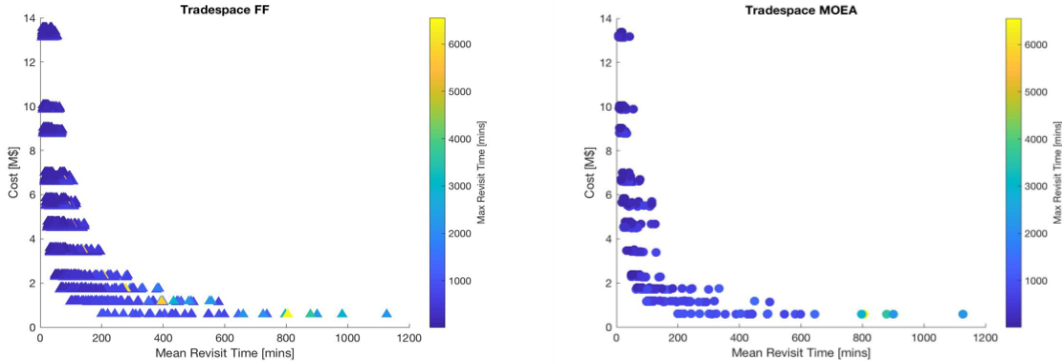
(a) Full Factorial - 5000 Generated Architectures [46]

(b) Evolutionary Algorithm - 500 Generated Architectures [46]

Figure 3-2: TAT-C Search Strategy Comparison

3.4 Orbits & Coverage Module

At the foundation of TAT-C is the Orbits & Coverage Module, also referred to as the Orbit Maintenance Module (OMM). This module completes all of the orbit propagation and calculates all critical coverage metrics such as revisit time for each architecture generated by the tradespace search executive. This module completes its orbital propagation calculations by making a direct call to the General Mission Analysis Tool (GMAT) through an application programming interface (API). GMAT was created and maintained through NASA GSFC, where it is was designed to help model, optimize, and estimate satellite trajectories [52]. Typically, GMAT is a standalone software program that anyone can download due to its open source license. However, given that the development of TAT-C was lead at GSFC, a more light-weight API was developed for TAT-C use.

At the most basic level, the orbits module allows direct numerical simulation, written and compiled in C++, of the constellation architectures that are being evaluated. The propagation uses a point mass model of the Earth and also considers gravitational effects such as J_2-perturbations [46]. Given the user's requirements, it is also possible to include atmospheric drag in the simulation. This is particularly helpful for low altitude EO constellations where atmospheric drag can cause orbits to degrade

quite rapidly without station-keeping [23]. For a detailed breakdown of how the gravitational perturbations and atmospheric drag are modeled, see Reference [48]. During the numerical simulation, the orbits module propagates a satellite forward in time by a particular time constant, and then thanks to the physics-based models built internally within GMAT, calculates the eccentricity, inclination, semimajor axis, argument of perigee, RAAN, mean anomaly, altitude, the x,y,z components of the satellite's position in the Earth Centered Inertial (ECI) frame, and the velocity components in the ECI frame as well. As stated above, it builds into the model orbital degradation effects, and the module can even specify how the satellites can maintain their orbit through phasing maneuvers or Hohmann transfers, given propellant reserves [48]. To give an example of this, Figure 3-3 below shows an output file from the orbits module for a generic satellite that records each of these orbital quantities.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Time[s]	Ecc[deg]	Inc[deg]	SMA[km]	AOP[deg]	RAAN[deg]	MA[deg]	Lat[deg]	Lon[deg]	Alt[km]	x[km]	y[km]	z[km]	vx[km/s]	vy[km/s]	vz[km/s]
2	56	0.0009109	97.4016	6878.14	359.998	0.00063883	3.5553	3.54535	50.2246	493.832	6858.68	-54.7849	422.317	-0.471759	-0.979694	7.54154
3	112	0.0009109	97.4016	6878.14	359.995	0.00127766	7.11057	7.09011	49.7634	494.11	6819.14	-109.36	843.013	-0.941701	-0.974062	7.49807
4	168	0.0009109	97.4016	6878.14	359.993	0.00191648	10.6658	10.6337	49.2952	494.569	6753.42	-163.516	1260.47	-1.40802	-0.9647	7.4258
5	224	0.0009109	97.4016	6878.14	359.991	0.00255531	14.2209	14.1755	48.8161	495.203	6661.75	-217.046	1673.09	-1.86891	-0.951643	7.32501
6	280	0.0009109	97.4016	6878.14	359.989	0.00319414	17.776	17.7149	48.322	496.003	6544.51	-269.745	2079.28	-2.32261	-0.934943	7.19609
7	336	0.0009109	97.4016	6878.14	359.986	0.00383296	21.3309	21.2514	47.8082	496.957	6402.13	-321.411	2477.49	-2.76736	-0.914663	7.03955

Figure 3-3: TAT-C Orbits & Coverage Sample Output

In order to get a more detailed picture of the entire Orbits & Coverage module, the following figure taken from Reference [48] provides an excellent block diagram of the entire system and breaks down its inputs and outputs.
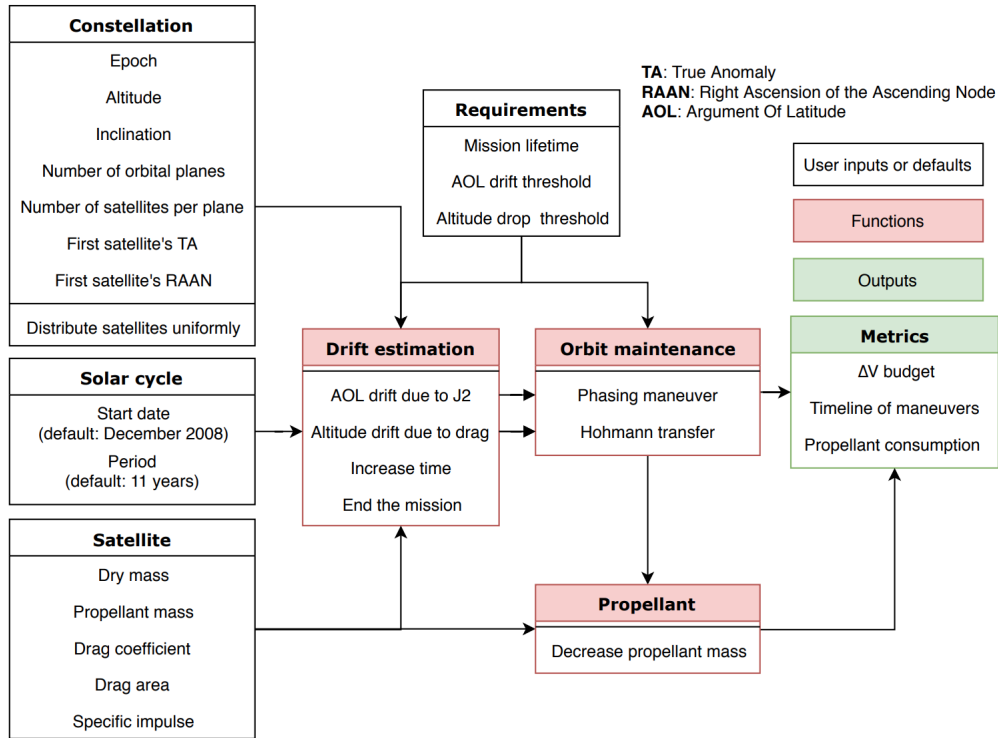
Figure 3-4: Orbits & Coverage Module Block Diagram [48]

3.4.1 Regions of Interest & Grid Spacing

What makes the orbits module an even more powerful tool is that it also calculates coverage metrics, in addition to performing the general satellite propagation calculations. In order to do this, the module produces a set of grid points within a user defined region of interest. A user defines a region of interest (ROI) by defining boundaries on the Earth's surface through latitude and longitude values. Note that this implies ROIs have to be a closed rectangle. Table 3.5 and Figure 5-25 depict a ROI that covers California and most of Nevada.

Figure 3-5: Example ROI Visual

	Latitude	Longitude
Min	33	-124
Max	42	-115

Table 3.5: ROI Description

Note that a user can define multiple ROIs which can be helpful when determining coverage metrics for various independent regions. A user can also examine global coverage by setting the region of interest latitude bounds to be (-90,90) and longitude bounds to be (-180,180).

Within a ROI, the orbits module produces a set of internal grid points where it calculates specific coverage metrics. This can best be explained graphically with the following diagram that shows the grid points overlayed on the entire Earth, which represents a global ROI.
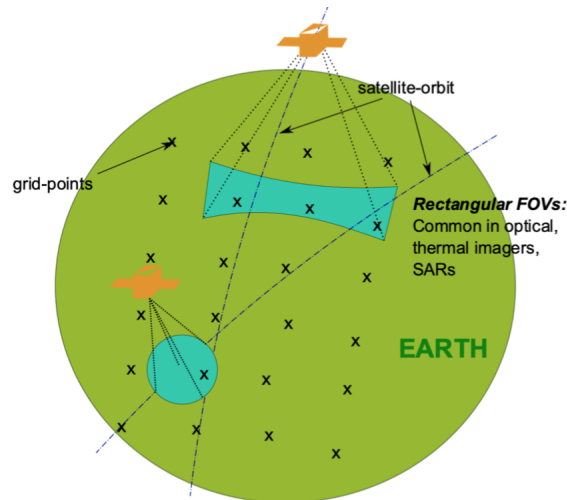
Illustration of satellite orbit and coverage

Figure 3-6: Orbits Module Grid Representation

Representing individual grid points, which can also be represented as a unique point of interest (POI), does come with an important limitation. This limitation has to deal with the *sparsity* of the grid generated by the orbits module.

3.4.2 Effects due to Grid Sparsity

In an ideal world, the grid generated by the orbits module would be dense. Each grid point would represent a unique latitude, longitude pair. Unfortunately, this would be very computationally expensive for the coverage portion of the Orbits and Coverage module. It would be inefficient to create such a dense grid. However, the alternative also represents a limitation of the model. Having a sparse grid implies that there are very few grid points generated within a ROI. At first glance, this may not seem to be a problem. However, the coverage module calculates coverage metrics **only when a grid point is within the field of view of the instrument on the satellite** [53]. To help break this down, it is important to understand what an access event is.

An access event is defined as an observation made of a point on the ground by an instrument on a satellite. These events are *only* recorded when a grid point is within the field of view of an instrument. In the theoretical case where the grid is infinitely

dense, the instrument would pick up a grid point at every single instance during its orbit. This is essentially what happens in real life (without worrying about slight variations in timing, etc.). Since the grid tends to be sparse for computational reasons, the total number of access events will be less than what would occur in the physical world, meaning the number of access events will be *underestimated.*

Not only will there be an underestimated number of access events, but the resulting data could also become somewhat skewed due to the sparsity of the grid. The field of view of the sensor can be broken down into ground pixels that capture the data on the Earth's surface. Another way to think about when an access event is recorded is to think that a grid point, or POI, has to be within one of the ground pixels in the sensor field of view. Figure 3-7 below shows a similar picture to Figure 3-6, but now shows what happens when a grid point is not within the ground pixel generated by the instrument. The figure has been dramatically simplified and the sensor only has three ground pixels. Each blue dot in the diagram represents a POI.
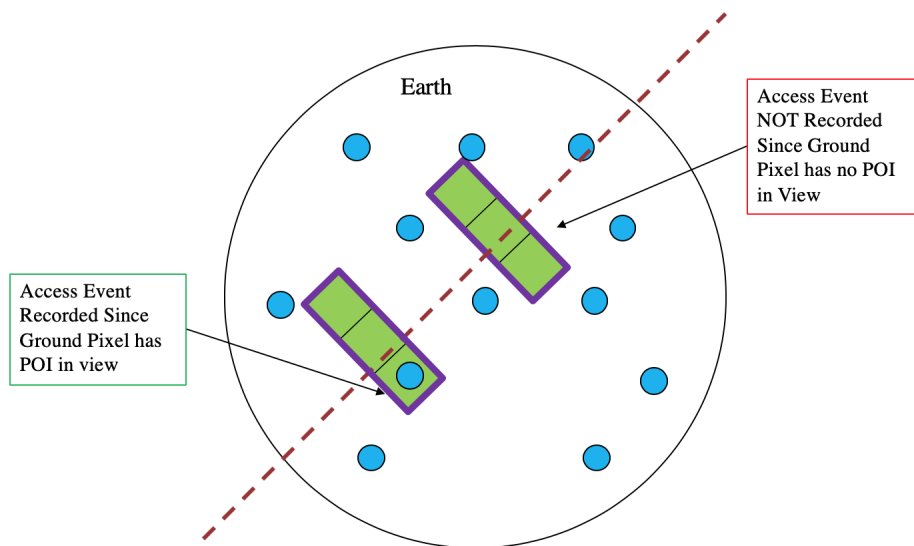
Figure 3-7: Mock Up of Sparse Grid Effect on Orbits & Coverage Module

Since an access event is not being recorded, overall coverage metrics such as revisit time, percent coverage, and response time can become skewed. These metrics are critical not only to the end user, but they feed into the Instrument Module as direct inputs. In Chapter 4, the Value module will also discuss how it takes inputs from the Instrument module, meaning small numerical errors can propagate throughout the TAT-C program.

Although the above section breaks down the important limitation of the Orbits & Coverage Module, there still is a lot of value in this model. The model itself has been validated through detailed comparisons to its private counterpart, Analytical Graphics, Inc.'s Systems Tool Kit, which is used extensively in industry. Also, TAT-C gives the user the ability to manually adjust the number of grid points during simulation. This is a powerful feature because it allows users to create higher fidelity simulations

if they can afford the extra computational time. To emphasize this point further, a simple TAT-C simulation was created in order to understand the relationship between grid spacing and execution time. Using the Landsat-8 satellite specifications, a global ROI, a 10 day simulation time, and a single satellite placed in an SSO orbit at 705 km in altitude, the execution time was recorded for various grid times. A Mac computer was used for this study with 8 GB of RAM and an Intel Core i5-2540M @ 2.60GHz processor. Table 3.6 below depicts the results. Note the third column shows the multiple of each row as compared to the run with 1000 grid points.

Grid Points	Execution Time [s]	Factor
1000	332	1
2000	741	2.23
4000	1257	3.78
6000	2176	6.54
8000	2762	8.31
10000	2814	8.47

Table 3.6: TAT-C Example Execution Time Table

As the above table shows, the execution time increases essentially linearly with the number of grid points. However, the absolute time for analysis is quite high, meaning the number of grid points should be set to meet specific time requirements. Also, since TAT-C was designed as a architecture comparison tool, any numerical errors within the orbits module will impact all constellation architectures, which means direct comparison of architectures is still valid.

3.4.3 Orbits & Coverage Outputs

Figure 3-3 showed the output of the orbits module, specifically the outputs that relate to the position and velocity of the satellite that is being simulated. The orbits module also produces coverage metrics for each access event that is recorded. Table 3.7 depicts the metrics calculated for each access event by the orbits module. These metrics, in addition to the satellite position and velocity, is recorded in a csv file called *accessInfo.csv where the * represents the satellite in question.

Access Event Metric	Description
POI	The POI index that is observed during the access event
Duration	The duration of the access event in seconds
Access From [Days]	Start of the access event referenced to the starting epoch
Time [Days]	Time when the access event is recorded

Table 3.7: Access Event Metrics

Given that grid points, or POIs, are important, the average, minimum, & maximum access time and revisit time are also calculated for each POI. In addition, the total number of POI access events and a metric called the time to coverage is recorded. Note that time to coverage is defined as the time in seconds before the first access event is recorded of a particular POI. Each architecture generates a file labeled lcl.csv that contains this information.

Finally, the orbits module generates a JSON file labeled gbl.json that creates aggregate statistics over each POI in order to find the average, maximum, & minimum for the following coverage metrics.

Global Coverage Metric	Description
Access Time	The POI index that is observed during the access event
Data Latency	The duration of the access event in seconds
Coverage	Percent of POIs observed out of all generated POIs
Downlink Time per Pass	Start of the access event referenced to the starting epoch
Number of Ground Station Passes per Day	Number of ground station passes recorded per day
Number of POI Passes	Number of access events recorded for a POI
Response Time	Time between access event and completion of downlink
Revisit Time	Time between access events for a particular POI
Time to Coverage [s]	Time before the first access event is recorded for a POI
Downlink Time per Day	Total access time spent per day when access event over a ground station occurs

Table 3.8: Global Coverage Metrics from the Orbits & Coverage Module. The Value Function Utilizes the Coverage Metric

3.5   Instrument Module

The goal of the Instrument Module within TAT-C is to provide instrument related metrics, such as SNR, to help emphasize the importance of the end products involved with a specific constellation architecture. The Instrument Module uses numerical methods in order to evaluate specific data metrics, rather than analytical approaches due to limitations in orbital approximations [54]. At the highest level, the Instrument module takes as input the satellite position and velocity vectors, the orientation of the satellite, the position of the target ground point, the instrument specifications such as field of view (FOV), and space environmental conditions. Reference [54] provides an excellent breakdown of these inputs in a much more detailed way. Once these inputs are provided to the module software, the data metrics are calculated through physics-based models. Depending on the instrument used, different data metrics are output to the end user. Figure 3-8 depicts the overall input-output relationship for the instrument module.

Currently the instrument module supports three classes of instruments: Passive Optical Scanners, Synthetic Aperture Radars, and Basic Sensors. The first two have been discussed previously in this thesis, where as the basic sensor represents a sim-
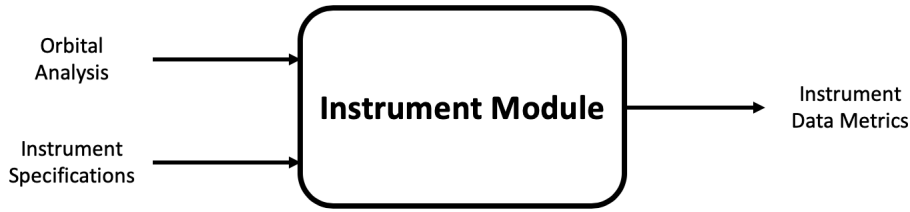
Figure 3-8: Instrument Module Block Diagram

ple model for any sensor instrument that is not a Passive Optical Scanner or SAR instrument. Due to its generality, it does not provide very meaningful information to the end user. Because of this, this thesis will not discuss it in further detail.

3.5.1 Instrument Module Limitations

The instrument Module is a powerful tool that greatly influences the overall performance of TAT-C, but it does have some important limitations. The first major limitation is that it requires a great deal of knowledge about an instrument's specifications. Given that TAT-C is a pre-phase A tool, some mission planners may not have this information for an instrument that has not been developed. Thus, existing reference instruments must be used if information about a developing sensor is not available at the time of analysis. Fortunately, there are many instrument specifications publicly available which enable users to sample various instruments.

Another key limitation of the instrument module has to deal with orbital geometry. Data metrics are evaluated for a specific ground pixel based on the geometry of the satellite and target ground point. This can best be represented graphically. For a simple Matrix imager, it is possible to model its ground footprint as a series of detectors on the Earth's surface. These detectors can be represented in a 2x2 grid, where the number of rows and columns of detectors is determined by the instrument specifications. Each individual detector cell, i.e. a row and column element, represents a ground pixel. The instrument module breaks down metrics for each individual ground pixel, but with a minor caveat. Although two ground pixels maybe in view of the sensor, *only* ground pixels that are observed purely at a side-looking angle, meaning they are in the cross-track direction, are utilized in data-metric calculations [54]. Figure 3-9 provides a graphical representation of this phenomena. In the figure, at time $t = 1$, only ground pixel 1 is utilized for data metric calculations, where as at $t = 2$, only ground pixel 2 is utilized for data metrics. Also, the modeled instruments do not have the ability to look-ahead or look-back to keep a target point in view as the satellite moves through its orbit.

(a) GP 1 Observation at Time 1 [54]   (b) GP 2 Observation at Time 2 [54]
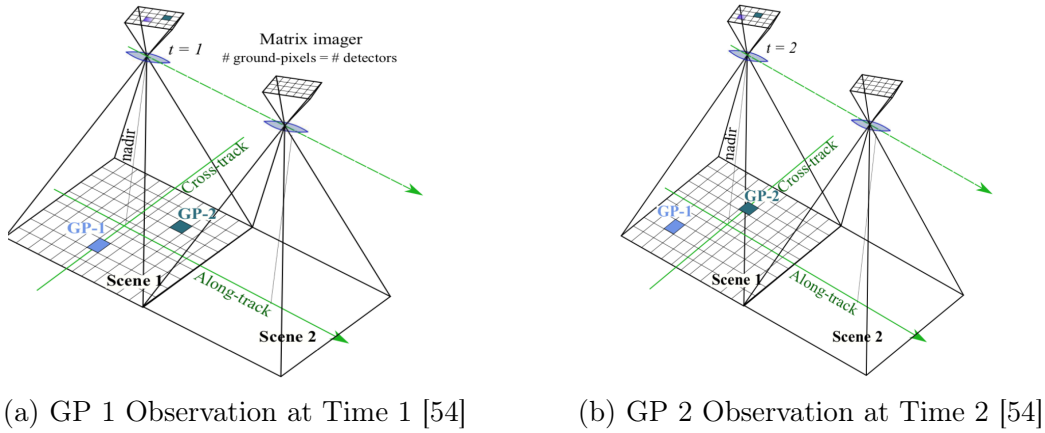
Figure 3-9: Instrument Module Side Look Angle Requirement

Modeling specific instrument performance is non-trivial, and thus limitations with this model are expected. However, this thesis will assume that due to the specificity and completeness of the entire model, the results obtained from the module are still considered useful.

3.5.2   Instrument Module Outputs

Before diving into the calculations for the key data metrics provided by the instrument module, it is worth noting the various outputs for each specific instrument. This thesis will focus primarily on the SNR and σ_N, but the instrument module produces the following outputs for each instrument.

Passive Optical Scanner	SAR	Basic Sensor
SNR	$\sigma_{\mathbf{N}}$	Observation Range
Dynamic Range	Observation Zenith Angle	Solar Zenith
Noise-Equivalent-Delta Temperature	**Ground Pixel Resolution**	
Ground Pixel Resolution		

Table 3.9: Instrument Module Outputs by Sensor Class. Value Function will utilize variables in bold.

3.5.3   Satellite Target Viewing Geometry

Prior to breaking down the instrument performance metric calculations, a detailed understanding of the satellite viewing geometry utilized by TAT-C is needed. Figure 3-10 below shows a simple diagram of a satellite in Earth orbit and the key angles due to its looking geometry.
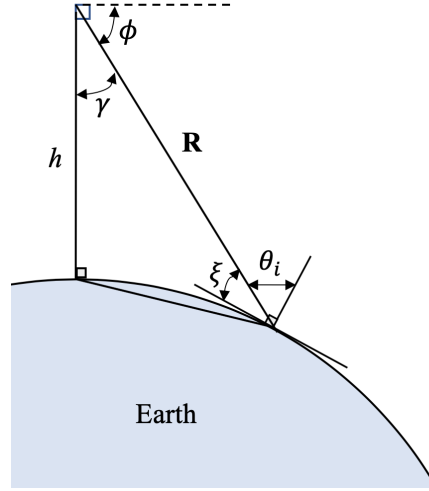
Figure 3-10: EO Satellite Viewing Geometry Angles
Adapted from [55]

The main viewing angles for this satellite and Earth geometry are as followed: γ which represents the look angle or nadir angle; θ_i represents the incidence angle as viewed from the ground; and ϕ represents the depression angle. Before the equations for these terms can be presented, two other variables are also critical to understand, **T** and **S**. **T** is the position vector of the target point on the Earth, in an Earth centered inertial (ECI) frame. Put simply, **T** provides the coordinates of the ground point being examined by a satellite. **S** is the position vector of the satellite also in the ECI frame. **S** provides the coordinates of the satellite for orbital analysis. Using these terms, it is possible to mathematically construct the viewing angles.

$$\mathbf{R} = \mathbf{T} - \mathbf{S} \tag{3.1}$$

$$\gamma = \arccos\left(\frac{\mathbf{R}}{\|\mathbf{R}\|} \cdot \frac{-\mathbf{S}}{\|\mathbf{S}\|}\right) \tag{3.2}$$

$$\theta_i = \arcsin\left(\sin\gamma \frac{R_E + h}{R_E}\right) \tag{3.3}$$

$$\phi = 90° - \gamma \tag{3.4}$$

The above picture is the first step in understanding the overall satellite viewing geometry. Due to non-spherical Earth effects as well as discrepancies in the orbit propagation timing due to computational constraints, TAT-C utilizes a representation of the satellite geometry called the *derived* satellite position [54]. The *derived* satellite position is the position of the satellite at which the line from the target ground pixel

to the satellite is perpendicular to the satellite velocity (at zenith). This derived position works due to the large curvature of Earth and short time intervals involved in the calculations [54]. Figure 3-11 provides a useful graphical understanding of this *derived* satellite position.
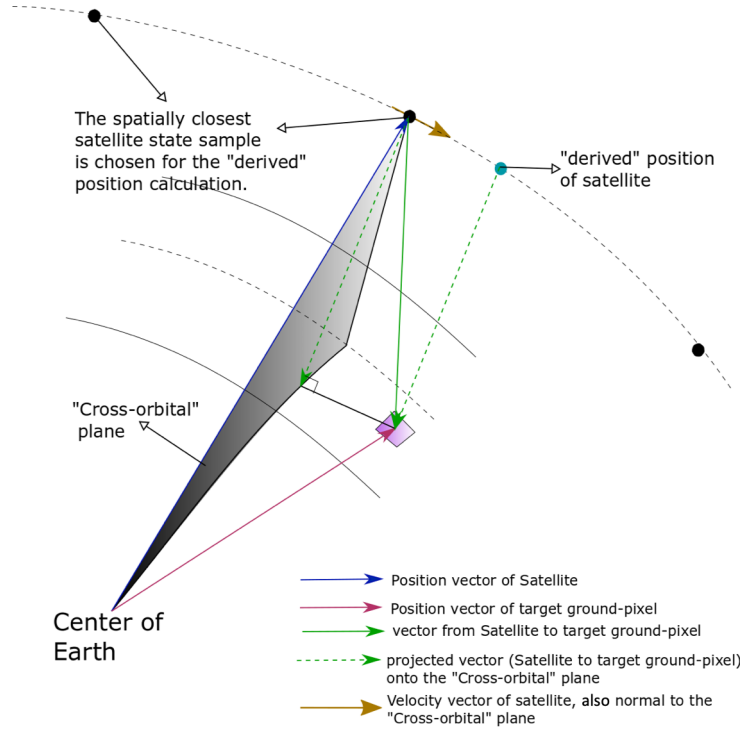
Figure 3-11: *Derived* Satellite Position Breakdown [54]

With this understanding of the *derived* satellite position, it is possible to update **R** and **S** using the equations below. Note, these updated terms provide the means to calculate the derived look angle, θ_i^{drv}, which will be utilized in upcoming SNR calculations.

$$\mathbf{R^{drv}} = \mathbf{R} - (\mathbf{R} \cdot \mathbf{v_{sc}})\mathbf{v_{sc}} \tag{3.5}$$

$$\mathbf{S^{drv}} = \mathbf{T} - \mathbf{R^{drv}} \tag{3.6}$$

3.5.4 Passive Optical Scanner Metric Calculations

In the previous chapter, equation 2.1 showed how to calculate the SNR. The remainder of the equation derivation is as follows. Note, all equations come from [23].

$$SNR = \frac{N_s}{\sqrt{N_s + N_r^2}} \qquad \text{Signal to Noise Ratio} \tag{3.7}$$

$$N_s = N_{ph}Q_E \qquad\qquad \text{Signal Electrons} \qquad\qquad (3.8)$$

The above equations calculate the SNR for an access event and rely on the number of signal electrons at the detector and the number of noise electrons.

$$N_{ph} = R_T^{det}|_{ph}T_i \qquad\qquad \text{Photons at Detector} \qquad\qquad (3.9)$$

$$R_T^{det}|_{ph} = R_T^{sen}|_{ph}\tau_{op} \qquad\qquad \text{Rate of Photons at the Detector} \qquad\qquad (3.10)$$

$$R_T^{sen}|_{ph} = \left(\frac{R_T^{rad}|_{ph}}{\|\mathbf{R}\|^2}\right)\left(\frac{D_{ap}}{2}\right)^2\pi \qquad \text{Rate of Photons at Sensor Aperture} \qquad (3.11)$$

$$R_T^{rad}|_{ph} = L_T A_{gp} \qquad\qquad \text{Rate of Photons Radiated} \qquad\qquad (3.12)$$

$$L_T = L_E + L_S^{uw} \qquad\qquad \text{Total Radiance from Target} \qquad\qquad (3.13)$$

The above equations are used to calculate the radiance to signal electrons. Notice how the Aperture Diameter, D_{ap}, and vector between the satellite and the target, $\mathbf{R}$, are included.

$$L_E = \int_{\lambda_1}^{\lambda_2} L_\lambda \tau_\lambda^{atm} \cos\theta_i^{drv} \qquad \text{Radiance from Earth in Direction of Target} \qquad (3.14)$$

$$L_\lambda = \frac{2\Upsilon c^2}{\lambda^5}\frac{1}{\exp\frac{\Upsilon c}{\lambda k_B T} - 1} \qquad \text{Planks Spectral Body Radiance Equation} \qquad (3.15)$$

$$L_S^{uw} = \frac{R_S^{uw}|_{ph}}{4\pi A_{gp}} \qquad\qquad \text{Upwelling Solar Radiance from Target} \qquad\qquad (3.16)$$

$$R_S^{uw}|_{ph} = R_S^{dw}|_{ph} \cos \theta_i^{drv} \qquad \text{Upwelling Photon Rate from Target} \qquad (3.17)$$

$$R_S^{dw}|_{ph} = L_S^{dw} A_{gp} \frac{\pi r_{Solar}^2}{\|\mathbf{V_{Sun2T}}\|^2} \qquad \text{Downwelling Photon Rate at Target} \qquad (3.18)$$

$$L_S^{dw} = L_S \cos \theta_i^{Solar} \qquad \text{Downwelling Radiance at Target} \qquad (3.19)$$

$$L_S = \int_{\lambda_1}^{\lambda_2} L_\lambda \tau_\lambda^{atm} \qquad \text{Radiance from Sun} \qquad (3.20)$$

$$\theta_i^{Solar} = \arccos\left(\frac{\mathbf{T} \cdot \mathbf{V_{Sun2T}}}{\|\mathbf{T}\|\|\mathbf{V_{Sun2T}}\|}\right) \qquad \text{Solar Incidence Angle at Target} \qquad (3.21)$$

$$\mathbf{V_{Sun2T}} = \mathbf{T} - \mathbf{P_{Sun}} \qquad \text{Vector from Sun to Target} \qquad (3.22)$$

The above equations determine the radiance with Earth as a reflector of Solar Energy. It assumes Earth is a black body and a Lambertian surface [54]. These calculations are critical to get a sense for how the Sun impacts image quality. The last four equations are important because they help determine the ground pixel resolution, which is an important quality metric itself. The area of a ground pixel is also used in equation 3.18.

$$A_{gp} = \rho_{CT}\rho_{AT} \qquad \text{Observation Target Area} \qquad (3.23)$$

$$\rho_{AT} = \xi\|\mathbf{R^{drv}}\|^2 \qquad \text{Along-Track Ground Pixel Resolution} \qquad (3.24)$$

$$\rho_{CT} = \xi\frac{\|\mathbf{R^{drv}}\|^2}{\cos \theta_i^{drv}} \qquad \text{Cross-Track Ground Pixel Resolution} \qquad (3.25)$$

$$\xi = \frac{d}{f} \qquad\qquad \text{Instantaneous Field of View} \qquad\qquad (3.26)$$

As is evident, there are multiple equations that need to be solved in order to calculate the SNR for a particular ground point. Given the complexity involved with these equations, it is a worthwhile exercise to point out which variables come from user-defined inputs and those that are generated from the Orbits & Coverage module. Understanding where the variables come from is important because by knowing which values are user-defined, end users can tune these parameters in order to achieve a specific level of instrument performance. In the following table, each variable involved with the SNR calculation will be marked as either an instrument specification or a variable that comes from the specific orbital geometry. One variable that is important to note is λ, or the Wavelength. Typically, a satellite sensor will produce data across multiple bands and wavelengths. In TAT-C, the user specifies a single band, and a central wavelength for this band, for an instrument. Thus, if a user wants to model separate bands, then they must create a "new" instrument, rather than using a single instrument with multiple bands. Although this is a structural limitation, there is a simple work around that can assist in analysis.

Instrument Specification Variable	Description
d	Detector Width
f	Focal Length
ξ	Instantaneous FOV
λ	Wavelength
D_{ap}	Aperture Diameter
τ_{op}	Optic System Efficiency
Q_E	Quantum Efficiency
N_r	Read Out Noise Electrons

Table 3.10: Instrument Specific SNR Calculation Inputs

Orbital Geometry Variables	Description
θ_i^{drv}	Derived Look Angle
$\mathbf{R^{drv}}$	Derived Range Vector
$\mathbf{T}$	Position Vector of Target Ground Point
$\mathbf{P_{Sun}}$	Position Vector of Sun
r_{Solar}	Solar Radius
τ_λ^{atm}	Wavelength Dependent Atmospheric Loss
$\mathbf{R}$	Range Vector from Satellite to Target

Table 3.11: Orbital Geometry SNR Calculation Inputs

To emphasize the importance of these independent variables, in particular the user defined instrument specifications, the TAT-C tool was used to simulate two architectures. Each architecture contained a single satellite placed at an orbital altitude of 500 km and in a SSO inclination representing PlanetScope and RapidEye. The only variation between the architectures were the instruments used. In particular, the focal length, aperture diameter, number of read out electrons, and FOV are quite different for these instruments. Table 3.12 shows the differences in the key instrument variables between each instrument.

	PlanetScope [34], [56]	RapidEye Imager [56]
d [m]	0.0000075	0.0000065
f [m]	1.14	0.633
Along Track FOV	1.9773°	0.006°
Cross Track FOV	2.9662°	7°
λ [m]	4.75e-7	4.75e-7
D_{ap} [m]	0.091	0.145
τ_{op}	0.6	0.85
Q_E	0.5	0.45
N_r	277	1688

Table 3.12: Instrument Specifications for PlanetScope & RapidEye

After running a 30 day simulation where the region of interest was the entire Earth, Figures 3-12 and 3-13 display the SNR distributions for the PlanetScope and Rapid-Eye instruments respectively. Note how both have similar distribution shapes, but vary in the numerical magnitude of SNR. These results are just meant as an illustrative example, and show how small variations in instrument parameters can influence SNR numerical results.
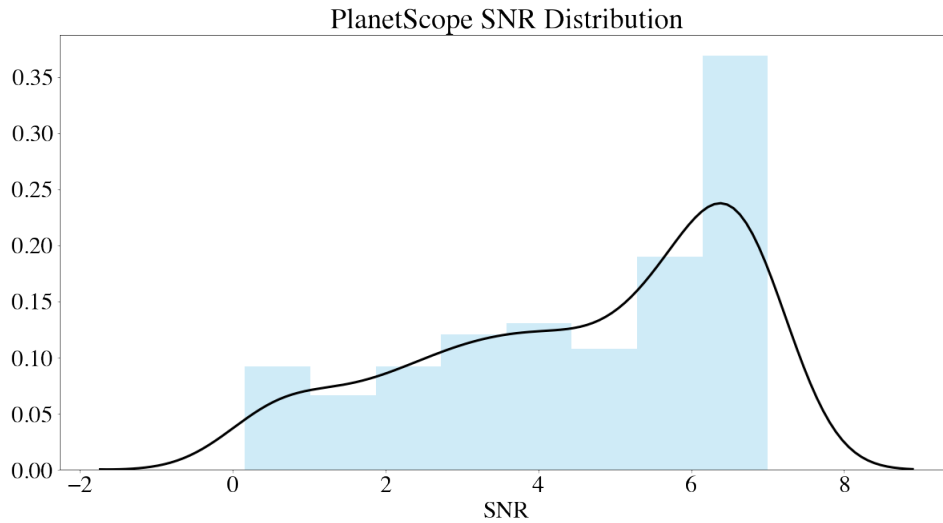
Figure 3-12: PlanetScope SNR Distribution
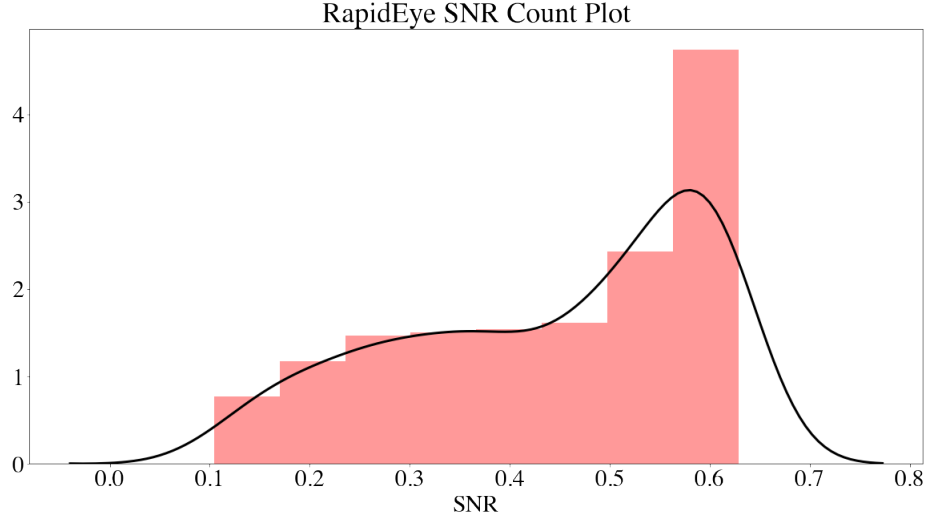
Figure 3-13: RapidEye SNR Distribution

3.5.5 Synthetic Aperture Radar Metric Calculations

The following explains how equation 2.2 calculates σ_N, also known as σ_{NEZ0}. The remainder of the equation derivation is as follows. Note, all equations come from [44].

$$\sigma_N = \frac{265\pi^3 kT}{c}\|\mathbf{R}\|^3 v_s \cos\psi_g \frac{B_T F_N L_{radar} L_{atmos}}{P_{avg} G_A^2 \lambda_0^3} \tag{3.27}$$

$$G_A = 4\pi \frac{\eta_{ap} D_{az} D_{elv}}{\lambda_0^2} \qquad \text{Antenna Gain} \tag{3.28}$$

$$P_{avg} = d_{sar} P_T \qquad \text{Average Transmit Power} \tag{3.29}$$

$$d_{sar} = T_{eff} f_p \qquad \text{Duty Cycle} \tag{3.30}$$

$$\psi_g = 90° - \theta_i^{drv} \qquad \text{Grazing Angle to Target} \tag{3.31}$$

Although there are fewer equations as compared to the Passive Optical Scanner SNR calculation, these equations represent complex physics based calculations that help model the detailed and intricate active SAR instrument. As done in the prior section, the following tables break down the SAR instrument-specific variables and the orbital geometry variables that impact σ_N respectively.

Instrument Specification Variable	Description
B_T	Chirp Bandwidth
η_{ap}	Atenna Aperture Efficiency
D_{az}	Dimension of Antenna Along-Track
D_{elv}	Dimension of Antenna Cross-Track
λ_0	Operating Center Wavelength of Radar
P_T	Peak Transmit Power
L_{radar}	Radar Hardware Loss
f_p	Pulse Repetition Frequency
F_N	System Noise Figure

Table 3.13: Instrument Specific σ_N Calculation Inputs

Orbital Geometry Variables	Description
θ_i^{drv}	Derived Look Angle
$\mathbf{v_s}$	Velocity of Satellite
$\mathbf{R}$	Range Vector from Satellite to Target

Table 3.14: Instrument Specific σ_N Calculation Inputs

3.6 Cost, Risk, & Launch Module

3.6.1 Cost & Risk Module

When conducting EO constellation architecture trade studies, understanding the cost and risk associated with a constellation is important. Within TAT-C is a built-in Cost & Risk module that generates multiple cost outputs and various risk categories. Given that this thesis is focused on the value generated by an architecture, the author recommends reviewing reference [27]. In reference [27], the author breaks down the TAT-C Cost & Risk module in extensive detail. For the purposes of this thesis, the outputs of the Cost & Risk module is of more importance. Tables 3.15 and 3.16 below depict the cost-based outputs and the risk outputs respectively.

Cost Category
Ground Cost
Hardware Cost
Integration and Test Cost
Launch Cost
Non-Recurring Cost
Operations Cost
Program Cost
Recurring Cost
Lifecycle Cost

Table 3.15: TAT-C Cost Outputs

69

Risks
Lack of flight heritage
Infant Mortality
Loss of coverage due to loss of one or more satellites
Orbital debris collision
Interrupted installation
Formation change due to atmospheric drag
Improper satellite spacing due to atmospheric drag
Instrument design flaw
Instrument deployment failure
Premature instrument failure
Radiation damage to instrumentation
Power subsystem degradation
Improper attitude alignment
Propellant related accident
Thermal subsystem failure

Table 3.16: TAT-C Risk Outputs

For each cost category, a cost estimate is provided as well as a calculated standard error for that cost. Note that cost categories including Integration and Test, Ground, Operations, and Program are calculated by taking set percentages of total satellite bus cost. Recurring and Non-Recurring costs are calculated by taking summations of all cost categories, but as Research, Development, Test & Evaluation costs and first-unit costs respectively. For each risk category, a likelihood and consequence is provided. These are ranked on a 1 through 5 ordinal scale, where 1 implies low likelihood or low consequence, and 5 implies high likelihood or high consequence. Overall, the Cost & Risk Module within TAT-C is a very helpful tool that provides mission planners with cost and risk data that is critical from a pre-phase A standpoint.

3.6.1.1 Cost Model Limitations

Although the cost model in TAT-C is highly detailed and relies on historical data, it has two primary limitations. The first limitation is that it is not high-fidelity for very small satellites, or CubeSats. One of the primary variables that the cost model looks at is the mass of a spacecraft. Table 3.17 breaks down the cost models utilized based on the spacecraft mass. As the table shows, a decision tree is used for the cost model when the spacecraft mass is less than 20 kg. The decision tree used is built on very little data and thus the results are inflated. Upcoming Section 5.2.6 discusses this in more detail and proposes a naive approach in order to obtain more accurate costs for CubeSats.

Cost Model	Minimum Mass [kg]	Maximum Mass [kg]
Unmanned Space Vehicle Cost Model	1000	-
Small Satellite Cost Model	20	1000
CubeSat Decision Tree Cost Model	-	20

Table 3.17: Cost Model by Spacecraft Mass

The second limitation of the cost model is the learning curve. The learning curve is applied in order to reflect the fact that it becomes easier and less costly to build the same satellite. The lower the learning curve rate, the more cost savings are realized. Reference [27] discusses the learning rate in great detail, and reference [57] discusses how the learning curve rate for small satellites should be set to around 0.662. In the cost model within TAT-C, the learning curve is set based on the number of identical satellites produced. Table 3.18 shows the learning curve rate for three tiers of production.

Learning Curve Rate	Minimum Number of Satellites	Maximum Number of Satellites
0.95	0	10
0.90	10	50
0.85	50	-

Table 3.18: Cost Model Learning Curve Rates

As Table 3.18 shows, the lowest learning curve rate that is currently applied is 0.85, and this is only when more than 50 satellites are produced! For large spacecraft, these learning curve rates are more realistic, but as reference [57] shows, CubeSats require a much smaller learning curve rate. These high learning curve rates can have large implications for large constellations, and is thus considered a limitation within the scope of this thesis. As a result the costs for large constellations of satellites <20 kg in mass may be overestimated by TAT-C.

3.6.2   Launch Module

One of the most important cost drivers for a constellation comes from the one time cost of launching the satellites into orbit. Although launch costs continue to decrease due to increased competition and launch providers, it still can cost up to $100 Million to launch a payload into orbit. When launching satellites into orbit, the key determining factors are the target altitude, inclination, and payload mass. The higher the altitude, the larger the payload mass, and the larger the inclination plane change relative to the launch site latitude the more propellant is used which translates to higher costs. Given the importance of understanding the launch cost, the author of this thesis developed a simple optimal launch solver that works within the bounds of TAT-C. This solver is called from the Launch Module, and quickly searches the available launch vehicles in the launch vehicle database in order to find the lowest cost launch option for a constellation.

The launch vehicle database is critical given that the search algorithm looks through the library of vehicles in order to find an optimal cost. The database was constructed by examining all currently operational launch vehicles and examining their respective user guides in order to learn information about payload performance. This is important because many launch vehicles are specified to perform at a specific inclination, and the payload mass varies as a function of altitude. This can be see in Figure 3-14 that reflects the Falcon 9 altitude versus payload mass chart. These charts are critical because various altitudes and inclinations can have large differences in payload mass available to reach target orbit. Since LEO orbits for EO constellations are rarely highly eccentric, these altitude versus payload mass charts are almost always linear. Because of this, the author of this thesis assumed all payload mass - altitude relationships were linear.

4.1.1. Low Earth Orbit
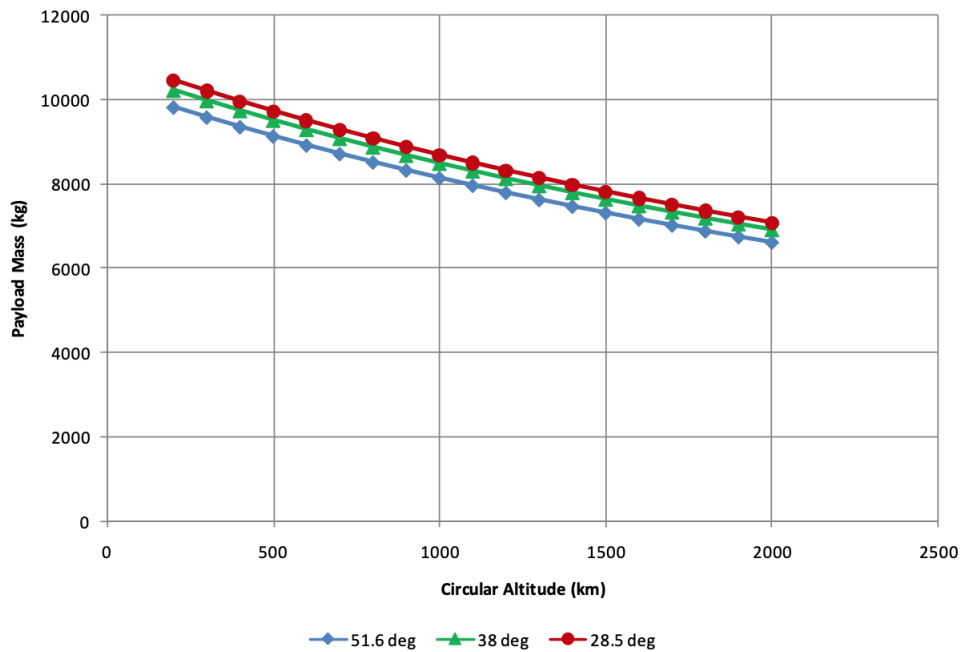
Figure 3-14: Falcon 9 Payload Mass vs Altitude vs Inclination [58]

Table 3.19 displays the features needed for each launch vehicle within the database. Note, end users are able to define new vehicles within TAT-C.

Feature	Units	Description
name		Name of the Launch Vehicle
acronym		More descript name of the Launch Vehicle - includes inclination
inclination		Inclination capability of the Launch Vehilce
maxAltitude	km	Maximum altitude of a circular orbit of the Launch Vehicle given the inclination
maxAltitudePayloadMass	kg	Maximum payload capacity of the Launch Vehicle at the highest altitude
minAltitude	km	Minimum altitude of a circular orbit of the Launch Vehicle given the inclination
minAltitudePayloadMass	kg	Maximum payload capacity of the Launch Vehicle at the lowest altitude
mass	kg	Total Vehicle Wet Mass
finalStageDryMass	kg	Dry mass of the Launch Vehicle's Final Stage
finalStagePropellantMass	kg	Propellant Mass of the Launch Vehicle's Final Stage
finalStageSpecificImpulse	s	Specific impulse of the Launch Vehicle's Final Stage
finalStageBurnDuration	s	Burn time of the Launch Vehicle's Final Stage
finalStageRestartable		Whether or not the Launch Vehicle's Final Stage is restartable
maxNumberRestarts		Maximum number of restarts of the upper stage engine. Not well documented, default to 5 if restartable
reliability		Launch Vehicle reliability: number of successes/total launch attempts
cost	Millions	Total cost to launch the vehicle.
operatingNation		The country that operates the launch vehicle
fairingDiameter	meters	Total fairing diameter of the launch vehicle
fairingHeight	meters	Total fairing height of the launch vehicle

Table 3.19: Launch Module Database Features & Descriptions

The only other consideration that is important to note is that of a primary versus secondary launch. This thesis defines a primary launch as a launch that is completely owned by a single satellite operator. A primary launch implies only a single payload, that can consist of multiple spacecraft, and the entire launch cost is paid for by the satellite operator. There are new small launcher companies coming online, such as Rocket Labs, that make this a realistic option even for small satellites. A secondary launch is a launch where a satellite operator essentially ride-shares on a launch vehicle. For secondary launches, this thesis assumed that the satellite(s) payload can only make up at most 49% of the launch vehicle payload capacity and volume. This is because the secondary payload does not constitute the majority of the payload mass by definition. From a cost perspective, this thesis assumes that secondary payloads pay the launch provider the cost per kilogram where the cost is the standard primary launch vehicle cost. Because of this, there are substantial cost savings when a secondary launch is used. The following two sections break down the solver used to find the optimal launch cost for both primary and secondary launches.

3.6.2.1 Primary Launch Formulation

The solver used to find the lowest cost launch manifest is a simple integer program (IP). Although there are more launch providers, there still do not exist a large, over 100, set of vehicles which makes performing a large search quite easy and computationally efficient. Note that in implementation, the maximum number of launches required for a specific launch vehicle is calculated and each launch vehicle instance is considered a unique vehicle. This implies more than one of the same vehicle can be used. Let's first break down the variables involved in the IP formulation. Table 3.20 shows the variables and their corresponding variable abbreviation. It is worth noting that Variable Y_j is an indicator variable that is used to ensure that the costs obtained for a complete manifest represents only vehicles that have satellites on board.

Variable	Description
m	Number of Launch Vehicles
n	Number of Satellites
X_{ij}	1 if Assign Satellite i to rocket j
Y_j	1 if Launch Vehicle j is Utilized
C_j	Cost of Launching Vehicle j
VC_j	Payload Volume of Launch Vehicle j
$Capacity_j$	Payload Mass Capacity of Launch Vehicle j
M_i	Mass of Satellite i
V_i	Volume of Satellite i

Table 3.20: Primary Launch Module Variables

Using these variables, the equations below depict the IP formulation for a primary payload launch. The formulation is very similar to the classic knack-sack problem. Equation 3.32 represents the minimization function and can be expressed in words as the minimum cost over all launch vehicles that were selected for launch. Equation 3.33 represents the constraint that only one launch vehicle can be assigned to a satellite. Equation 3.34 represents the constraint that the satellites placed on a launch vehicle must have a mass less than that of the launch vehicle payload capacity. Similarly, equation 3.35 represents the constraint that the satellites placed on a launch vehicle must take up a volume less than that of the launch vehicle volume capacity. Finally, equation 3.36 forces the decision variables to be binary which makes it a strict IP.

$$min \sum_{j=1}^{m} C_j Y_j \tag{3.32}$$

$$s.t \sum_{j=1}^{m} X_{ij} = 1 \qquad \text{for all satellites } i \tag{3.33}$$

$$\sum_{i=1}^{n} X_{ij} M_i \leq Capacity_j Y_j \qquad \text{for all Launch Vehicles } j \tag{3.34}$$

$$\sum_{i=1}^{n} X_{ij} V_i \leq VC_j Y_j \qquad \text{for all Launch Vehicles } j \tag{3.35}$$

$$X_{ij}, Y_j \in \{0, 1\} \tag{3.36}$$

3.6.2.2 Secondary Launch Formulation

The secondary launch formulation uses the same variables as the primary launch formulation, but with one revision. The variable C_j, which previously represented the cost of launching a Launch Vehicle j, now represents *the cost per kilogram* of launching Launch Vehicle j. Also, variables $Capacity_j$ & VC_j represent 49% of vehicle payload capacity and volume capacity respectively. The equations below depict the revised IP for the secondary formulation. Equation 3.37 represents the minimization function and can be expressed in words as the sum over the cost per kilogram for each launch vehicle multiplied by the total payload mass on that launch vehicle. Note that an indicator variable is not needed because if no satellites are assigned to a vehicle, then the second part of the summation will be equal to 0. Equation 3.38 represents the same constraint as equation 3.33, i.e. a satellite can only be placed on a single launch vehicle. Equations 3.39 and 3.40 are the same as equations 3.34 and 3.35 but without the indicator variable. Lastly, equation 3.41 forces the decision variable to be binary which forces the formulation to be a strict IP.

$$min \sum_{j=1}^{m} C_j \sum_{i=1}^{n} X_{ij} M_i \tag{3.37}$$

$$s.t \sum_{j=1}^{m} X_{ij} = 1 \qquad \text{for all satellites } i \tag{3.38}$$

$$\sum_{i=1}^{n} X_{ij} M_i \leq Capacity_j \qquad \text{for all Launch Vehicles } j \tag{3.39}$$

$$\sum_{i=1}^{n} X_{ij} V_i \leq VC_j \qquad \text{for all Launch Vehicles } j \tag{3.40}$$

$$X_{ij} \in \{0, 1\} \tag{3.41}$$

3.6.2.3 Launch Module Summary

In summary, the Launch Module is a very simple and easy to understand model that enables the end user to determine the optimal launch manifest from a cost perspective.

In the real world, there are more constraints such as launch site and changes to payload mass due to payload adapters, but given the publicly available data, these were not included in the formulation. Future iterations of the launch module should include these factors. Also, the Launch Module can be improved by mapping the altitude versus payload mass function explicitly, rather than making a linear assumption. In the distant future, there may become a large set of launch vehicles and a new bin-packing algorithm maybe needed to reduce computational time. Currently, the Launch Module requires about one second of CPU time to execute which means it is capable of calculating an optimal launch manifest for thousands of architectures if needed. It is also worth noting that the user can add new launch vehicles, disable the search process if they must use a specific launch vehicle, and mark each constellation architecture as primary or secondary. These end user tags are critical in that they provide the user with more flexibility from a programming perspective.

3.7   Summary

As Chapter 3 showed, TAT-C is a complex and sophisticated tool that can help mission designers and planners find more optimal constellation architectures. It provides a large quantity of useful data, and is quite flexible to specific user needs. It does have some important limitations, but given the scale and overall performance, TAT-C is a tool that can add substantial value and help improve the design process overall. TAT-C is intended be open source which will not only allow anybody to use the tool, but it will also enable users to tune the software to fit specific needs. This will make it even more capable, which will add more value to the end user. The author looks forward to the open-source release so the community will be able to utilize the program to its full potential.

Chapter 4

Value Driven Trade Approaches

4.1   Value Function Overview

Value modeling allows mission planners and designers the ability to look past simple cost metrics when exploring the constellation design tradespace. Value modeling has historically been quite specific and focuses on monetary flows or stakeholder requirements [59], [60]. These methods are intuitive and are built on the principle of defining value from the lens of a specific user, which makes sense given that value is subjective from person to person or organization to organization. However, this thesis crafts a new value model that can decipher value by examining spacecraft and instrument design choices. By creating a value formulation that is dependent on mission parameters, mission planners will have the ability to compare constellation architectures that evaluate performance rather than cost, and across scientific domains or specific user requirements. The power of a generalized value model is that it can be used by any potential end user, which is why the value model developed in this thesis was built in tandem with the TAT-C tool. In order to make the Value Function general, this thesis developed the following hypothesis.

The value of an Earth Observation constellation architecture is driven primarily by the *quantity* and *quality* of data that is returned to the end user. Thus, architectures that generate large quantities of *high quality, useful* data are more valuable than architectures that produce less useful data.

This hypothesis can be extended further to understand the key differences between constellations that return large quantities of lower quality data versus constellations that return smaller quantities of higher quality data. This key trade-off is balanced in the real world, where various EO providers, such as Planet, Maxar, and the federal government (Landsat), provide different variations in quality and quantity of EO data. Currently, the market determines how much value each constellation provides, thus enabling us to understand where end users put more emphasis - quantity or quality.

At the core of the hypothesis is the quantity and quality of data collected by a constellation architecture. Thus any Value Function formulation will need to determine accurate proxies for these terms. In the following sections, we will discuss existing literature on the topic, and then derive the proposed Value Function that will utilize information generated by TAT-C in order to find suitable proxies for quantity and quality of data, and therefore value.

4.2 Literature Review

Before diving into the details of the proposed Value Function in this thesis, it is critical to understand the previously published approaches to quantifying value within the scope of EO constellations. To start, the National Academies of Sciences, Engineering, and Medicine discussed in 2015 the need for methods and metrics that NASA can use to make programmatic decisions about the scope and design of EO systems [61]. To put this in layman terms, it was only five years ago when the National Academy of Sciences determined that value-based trades were critical to designing the next generation of EO systems. Historically, there has been a large emphasis on cost. Designers attempted to satisfy mission requirements with the smallest cost. This was exemplified by two main factors. The first was sophisticated instruments and parts were larger in a physical sense, and thus required more mass and volume which made the overall satellite system larger, more complex, and more expensive. To make matters worse, larger satellites are harder to get into orbit, meaning they were more expensive to launch as well. Given these dynamics, it makes sense that there was a strong emphasis to minimize cost while satisfying mission requirements. However, due in part to miniaturization, smaller and yet effective instruments, and rapidly decreasing launch costs, as discussed in section 2.1.1, it is easier and cheaper than ever to get assets into orbit. These trends have pushed mission designers and planners to examine value-based metrics and methodologies. Examining the previous literature, this thesis categorizes previous work in the following categories: value network modeling, value of information, commercially-driven value, rules-based approaches, and case studies.

4.2.1 Value Network Modeling

The idea of value network modeling, also known as value flow mapping, was proposed by Dr. Bruce Cameron in 2007 and extended by Dr. Wen Feng in 2010 [62], [63]. In Cameron's PhD thesis, he postulated that it is possible to compare space exploration architectures by considering the various stakeholder value delivery chains in order to prioritize certain objectives [62]. Value network models help map the indirect benefit delivered to stakeholders, and a numerical method is used to prioritize paths through the network. These value networks allow space exploration architectures to help make critical decisions that drive value across organizations and national agencies.

In order to understand this work, it is critical to understand what a stakeholder is. Cameron uses the following definition that comes from Freeman's *Strategic Manage-*

ment: A Stakeholder Approach [64] which defines stakeholders as those entities that have an interest in the value creating organization. By examining the direct and indirect stakeholders, Cameron's work utilizes network diagrams to represent the value flows across each stakeholder or agent. Once these networks are established, it is possible to determine the network flows that maximize stakeholder value. These methods work well for large public space architectures because they do not have easily derived requirements and non-technical needs force the examination of various indirect value loops [65]. An example value network map is displayed below.
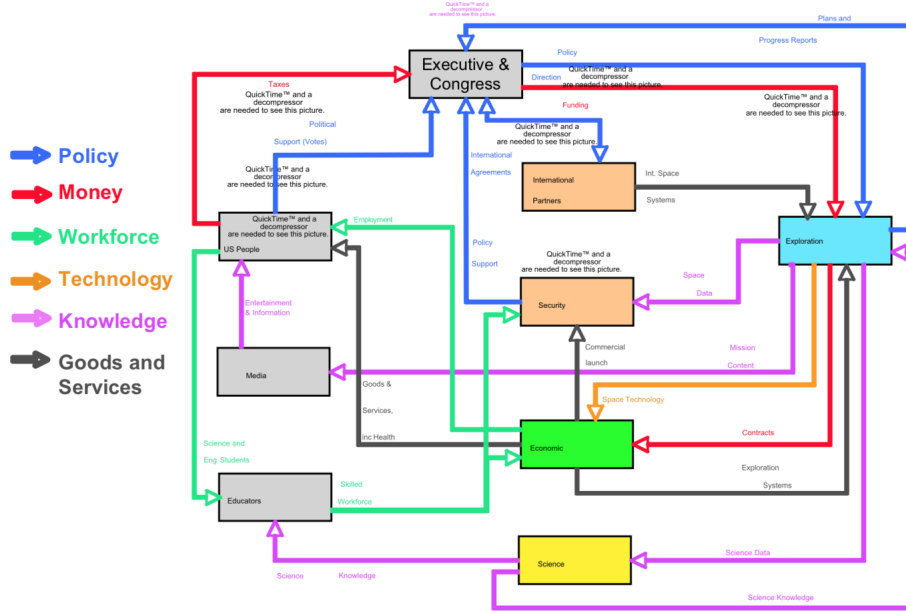
Figure 4-1: Value Map for Government-Based Space Architecting [62]

These value network maps do an excellent job distinguishing various stakeholders and creating a very well defined map that describes where value creation occurs. This is useful because it forces mission planners to focus on value-driving activities to stakeholders, rather than standard technical requirements or mission objectives. By definition, the optimal architecture is the one that maximizes stakeholder value, which this work identifies.

The primary limitation of this model is what also makes it intuitive and robust: the need to define all stakeholders. Understanding all of the direct and indirect stakeholders apriori is challenging. Not only that, stakeholders can change during the mission, which can change the optimal architecture selection. In the case of EO constellations, it would be possible to map out the various stakeholders, but this would become quite complex as constellation architectures become more complicated and sophisticated. Due to this fact, this thesis believes that value network modeling is not the preferred way to calculate the value of an EO constellation. Ideally, a more general model could be built that uses theoretical data from potential architectures in order to drive the decision making process. Another way to think about this is

that Stakeholder Value Networks focus more on the demand side of EO, whereas this thesis believes a more supply-side based approach can generate more fruitful and general results.

4.2.2   Value of Information

A very interesting way to capture the value of EO constellation is to examine how much value is *lost* if the constellation does not exist. By examining this gap, it is possible to quantify, even in a dollar sense, the value of a EO constellation. This idea was postulated in by Brathwaite & Saleh, or reference [60]. In this paper, the authors utilize a Bayesian framework to determine the overall system value. They view space systems as information sources, and stakeholders as information sinks. The key idea behind the authors' work is that the information generated by a constellation allows stakeholders to update their beliefs and thus make more optimal decisions on the ground. The increase in expected payoffs can thus be ascribed to the value of the system. Brathwaite & Saleh propose a new metric, the *Value of Design*, to quantify the increased value.

The authors provide a concrete example where an EO architecture provides hurricane information to oil rig operators in the Gulf of Mexico. The information allows the oil rig operator to have a better estimate of the forecasted track of a hurricane. By understanding this information, the operators are able to make a more informed decision on whether they should close the rig or keep it open. Closing the rig will incur lost revenue opportunities. Thus it is possible to price the value of having the more accurate forecasting model. What makes this methodology interesting is that is very easy to trade against specific EO constellation systems. For example, various instruments can be traded on, as long as it is possible to extrapolate how the various instruments impact the decision criteria metric, in this case the track of the hurricane. The model developed relies extensively on Bayesian probabilistic inference, thus it is able to generate the *Value of Design* for various EO constellation designs, as seen in Figure 4-2.

Brathwaite & Saleh's work utilizes a well thought-out and detailed model in order to define the value of an EO constellation. The idea behind the model is intuitive, and the fact that it can provide value in terms of dollars is valuable. However, their work does come with some limitations. For one, their model requires probability models of stakeholder's beliefs, as well as economic models of expected pay-offs. These can be very difficult to evaluate, especially at the very beginning of the design process. Theoretically this model could be applied to situations where the economic piece is not directly known, but without further work, it would be difficult to establish the value generated of non-priced-based services. Overall, Brathwaite & Saleh's work provides an intuitive and clever way to approach the value process. Their work allows for various trades, but can only be traded upon within the scope of a single case or specific application, for example the hurricane tracking. This has the potential to lead to an underestimate of the value of an EO system as new uses of such data are discovered almost every day. Thus, a more general model that examines the value

of information, but does not rely on probabilistic models and a specific output scope would be desirable.
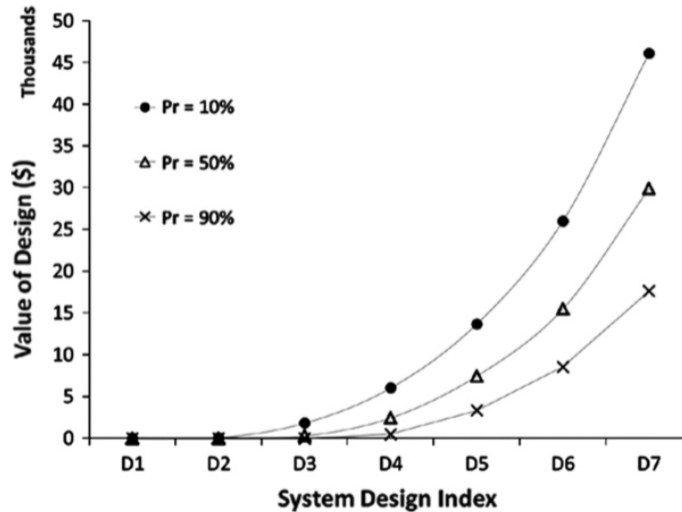
Figure 4-2: Value of Design Trade [60]

4.2.3 Commercially-driven Value

An alternative way to measure the value of a space system is through direct revenue generation activities, and calculating the Net Present Value (NPV) of the system. NPV has its historical roots in the financial sector, where it is commonly used to value companies, investment opportunities, and other capital budgeting and capital allocation activities [66]. At the highest level, the NPV is simply the difference of the present value of all expected future cash inflows and the present value of all future cash outflows. Another way to think about it is as the present value of revenue minus the present value of costs. In the upcoming section 4.3.3.3, a more detailed description of present value is given in order to assist in the value function derivation.

Geng et al. utilize the NPV concept and apply it to a telecommunication space system [67]. In their work, they develop a complex revenue model of the spacecraft, as well as a detailed spacecraft lifecycle cost model. What makes their model interesting from a trade space exploration point of view is that they incorporate the key design choices such as the number of transponders and mission lifetime in order to determine which system produces the most NPV. The NPV model is quite useful because it not only shows which spacecraft design maximizes NPV, but also shows the expected time to break even, i.e. when NPV is equal to 0. Figure 4-3 shows their paper's calculated NPV as a function of the number of transponders. Figure 4-4 shows a NPV plot that explains the time required to make NPV positive, in particular when two different launch costs are assumed.

The benefit of this type of model is that it is backed by extensive financial theory, and

the value accrued can be expressed in dollar terms. However, the biggest limitation to this type of model is that the system must generate positive cash flows in order to be used. Information can serve as a proxy for cash flows, but then other problems arise that are addressed in the value of information work described above [60]. Since not all EO constellations will be generating cash flows directly, i.e. the constellation owners will not be selling data services, it is difficult to use this model in practice (note that there are EO data providers that sell data gathered from their constellation, and thus this type of methodology would work quite well for them). Constellations that provide data as a public service, on the other hand, such as the European Sentinel satellites which are part of the Copernicus program generate their data as a public service. Again, a more general method of calculating value is needed in order to overcome the limitations of this type of model.
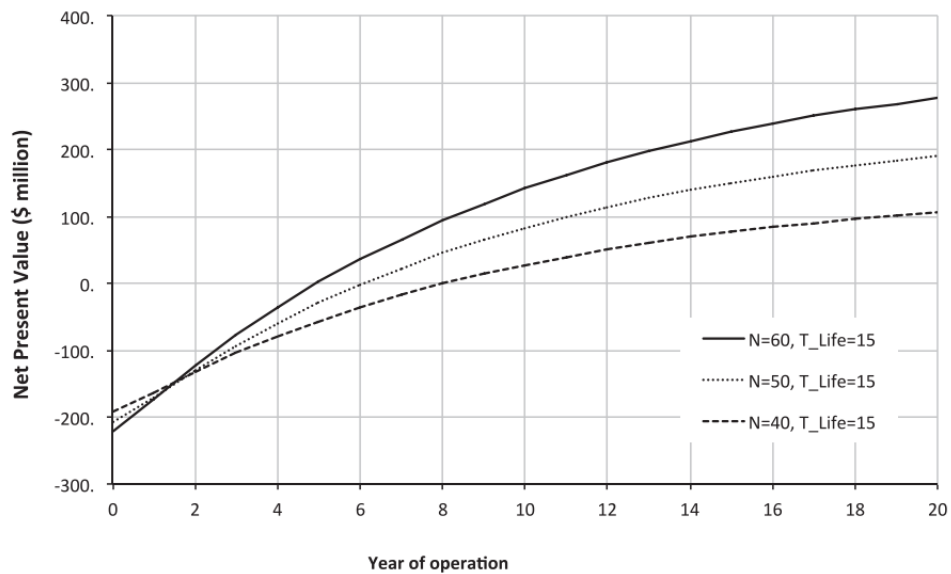
Figure 4-3: NPV of Spacecraft with 3 Levels of Transponders [67]
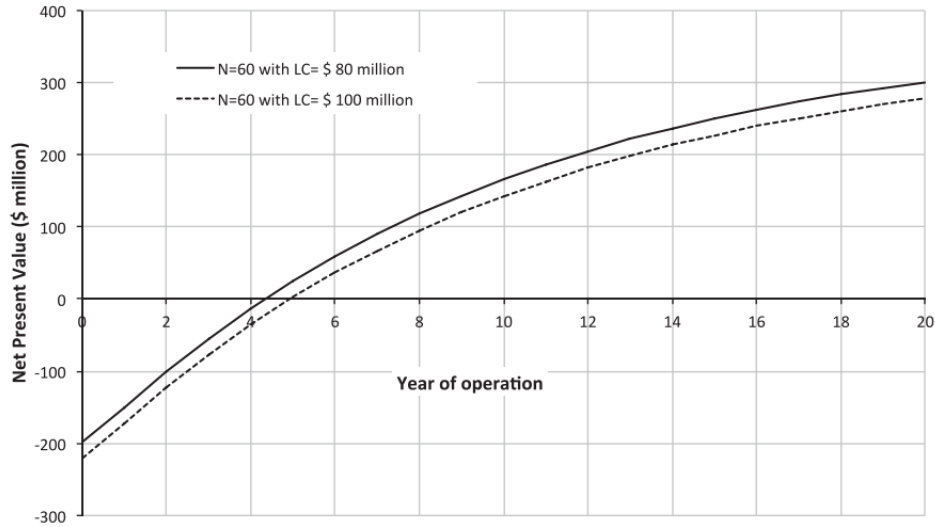
Figure 4-4: NPV of Spacecraft with 2 Launch Cost Situations [67]

4.2.4 Rules-Based Value Trades

Outside of direct value calculations, an alternative methodology to approach the value problem is through rules-based systems. This methodology towards value discovery was introduced by Dr. Daniel Selva in his PhD thesis [68]. His work was also discussed in the following article: Reference [69]. It is important to note that Selva's work served as one part of the inspiration of the TAT-C tool, and he was actively involved with the TAT-C formulation and implementation. In particular, he and his team focused on the search strategies used in TAT-C, as discussed in section 3.1.1.

Selva's work used rules-based systems in order to explore the architectural trade space of EO constellations. In particular, Selva defined value as the ability to satisfy mission requirements and satisfy the needs of the Earth Science Decadal Survey. Similar to TAT-C, Selva formulated the architecture problem as a combinatorial optimization problem. The key decision variables where what instruments to use, which instruments to place on a satellite, and mission scheduling. As Selva explains, the rules-based expert system uses expert knowledge to generate logical rules. In the scope of Selva's paper, these rules represent knowledge from domain experts that help guide the optimal heuristic search process to find feasible and value driving architectures. Figure 4-5 helps depict this process graphically.

Selva's method is sophisticated and does an excellent job exploring the EO constellation trade space. However, the largest limitation of the model is that rules-based systems require expert knowledge, and modeling expert knowledge is time consuming and expensive. Not only that, but by relying on rules derived by expert knowledge, more novel or complex architectures may be neglected due to the tendency to focus on what has worked in the past rather than what will work in the future. Thus a value metric that does not rely on expert knowledge would be beneficial because it
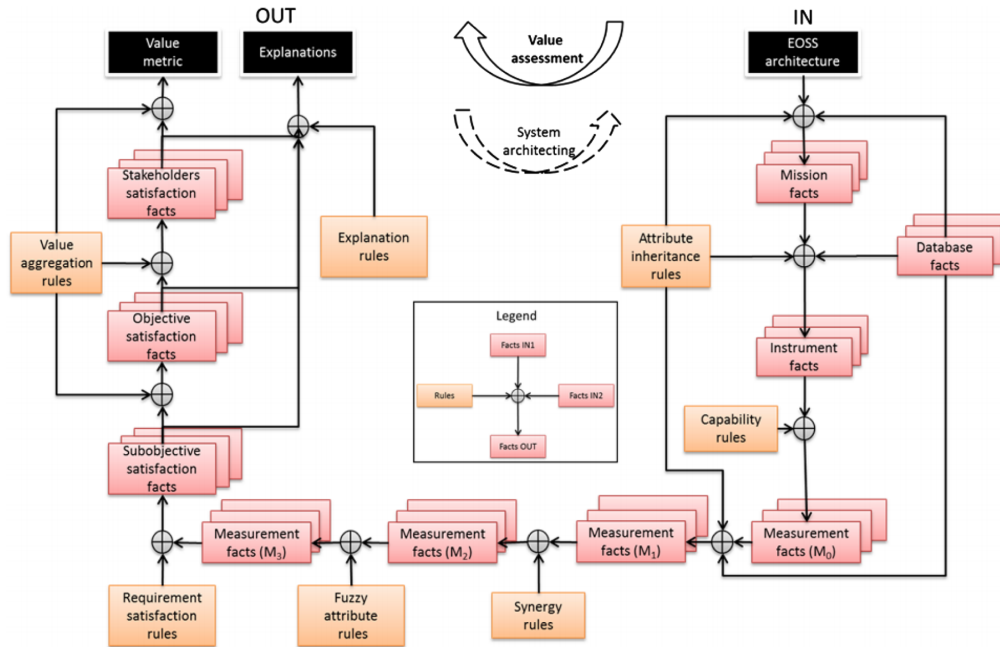
Figure 4-5: Selva Rules-Based Architecture Exploration Process [68]

could help discover new architectures that may have been sidelined in the past.

4.2.5 Case Studies

Similar to the value of information approach, another common way to determine the value of a specific EO space system is through detailed case studies that attempt to measure the direct and indirect benefits that arise due to the system. One of the largest studies was conducted by the ACIL Allen Consulting group in 2015. This group attempted to quantify the benefits that can be attributed to EO data, specifically within Australia [70]. The results of this study are depicted in the following figure.

TABLE ES 1 – SUMMARY OF FINDINGS							
Activity	Sectors	Impact	Nature of benefit	2015	2025	Potential employment impacts 2015	Potential employment impacts 2025
				$million	$million		
Weather forecasting	Agriculture, aviation	Economic	Increase in value added	75.8	199.1	903	1,474
Ocean observation	Great Barrier Reef	Social/ environmental	Sustainable ecosystem	825.0	1275.0	7593	10,146
	Petroleum and shipping	Economic	Increase in value added	13.3	29.7	11	26
Landscape and land cover monitoring	Government	Social/ environmental	Sustainable land cover	20.7	39.2		
	Property	Economic	Increase in value added	64.4	128.8	384	768
Agriculture	Crops and pastoral, biosecurity	Economic	Increase in value added	84.7	754.7	402	3,583
Water management	Water sector	Economic	Increase in value added	39.5	80.0		
	Water resources	Social/ environmental	Enhancing habitat and catchment ecology	14.9	14.9		
Natural disasters	Insured costs plus social costs	Economic	Increase in value added	213.0	495.0		
Mining	Licence to operate	Economic	Increase in value added	5.0	6.7		
Total Economic				495.7	1,694.0	9,293	15,997
Total social/ environmental				860.6	1,329.1		
NOTE. THIS PROJECT SCOPE DID NOT INCLUDE ECONOMY-WIDE MODELLING OF THE DIRECT ECONOMIC EFFECTS ON GDP AND INCOMES. THE TOTALS THEREFORE ARE DIRECT EFFECTS AND NOT OUTCOMES FOR THE ECONOMY AS A WHOLE.							
SOURCE: ACIL ALLEN							

Figure 4-6: ACIL Allen Summary of Findings [70]

As Figure 4-6 shows, the total economic value add due to EO data was roughly $500 Million in 2015, and the total social and environmental impact was about $860 Million. These are significant numbers, and reflect just the impact on Australia! ACIL Allen went about this study by estimating the consumer surplus through various willingness to pay studies as well as by examining the impact of EO services on the productivity of downstream users of data. This study also looked at what would happen if there was a loss of service, similar to the value of information approach. In summary, the ACIL Allen case study provided a detailed breakdown of the value created by EO data products. Unfortunately, the methods used in the case cannot be used at the individual EO architecture level, and are thus not feasible for the scope of EO constellation trades.

Another case study example comes from the United States Geological Survey (USGS). The USGS examined the value that users derive from use of Landsat data products by sending a survey to users who use medium resolution imagery in any capacity in their work [71]. Over 2,500 people responded to the USGS survey, and through the

study, the USGS learned about the value of Landsat data primarily by postulating about what users would do if Landsat data no longer existed. They also examined the willingness to pay for replacement imagery if Landsat data ceased to be provided as well. This can be shown in the Figure 4-7. Similar to the ACIL Allen study, these methods cannot be used at the architecture level, and thus are infeasible when conducting large scale trades.
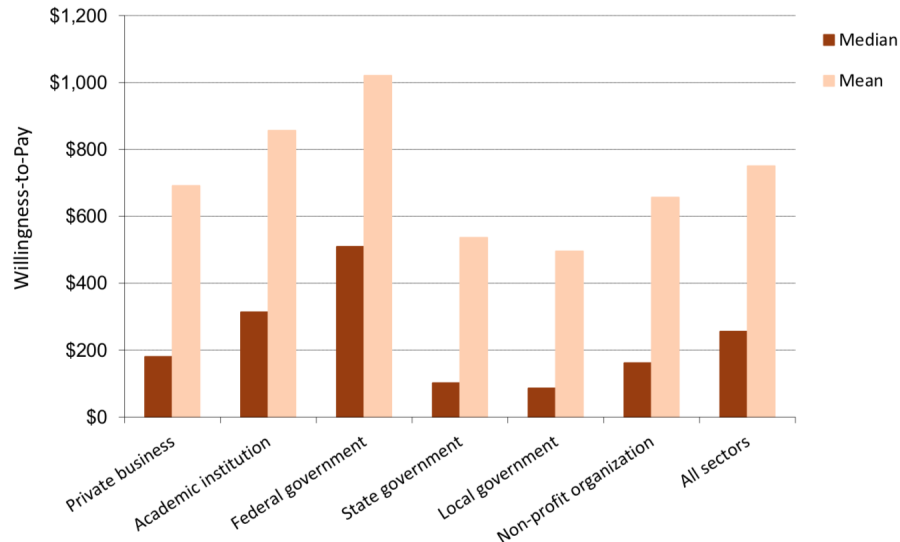
Figure 4-7: Willingness to Pay per Image Scene for Imagery to Replace Landsat Imagery [71]

4.2.6 Literature Review Summary

As the previous section described, there have been efforts to utilize a value framework to help mission planners and designers make more optimal decisions when it comes to EO constellation architecting. Table 4.1 provides a summary of each primary method, as well as their key limitations.

With the knowledge of the existing literature, it is evident that a standard value based trade approach currently does not exist. A highly general model that can be applied to hundreds or thousands of architectures is a crucial need in the industry. The following section will derive a Value Function that overcomes the limitations of the previous methodologies, and help establish a value based approach that the National Academy of Sciences is currently searching for.

Existing Value Approach	Description	Limitations
Value Network Modeling	Utilizes stakeholder value flows to determine optimal architecture	Need to understand all stakeholders and various connections
Value of Information	Looks at the benefits that arise from having more information	Need probabilistic and economic models to capture all architecture dynamics
Commercially-driven Value	Use NPV theory to model architecture value	Hard to expand beyond commercial, revenue generating spacecraft
Rules-Based Value	Uses expert knowledge to guide the optimal search of constellation architectures	Modeling expert knowledge is time and resource consuming and may not be feasible during early design stages
Case Studies	Examine benefits of EO after the fact and applied to large scale systems	No way to scale down to the individual architecture level

Table 4.1: Literature Review Summary

4.3 Value Function Derivation

4.3.1 Important Terminology

Before diving into the Value Function derivation, it is important to discuss two terms. The first term, region of interest, was discussed in section 3.4.1. From the lens of the end user, ROIs are set up to define the geographical bounds of interest. For example, if the US government decided to create an EO constellation that emphasized the continental United States, then it would set the ROI to represent the latitude and longitude values that bound this geographic area. It is also possible to set up multiple ROIs. By expanding the previous example, let's say the United States government wants to emphasize the continental United States and Alaska. This implies there are two discrete ROIs. ROIs are important because they define the area of interest, i.e. the area most *valuable* to the end user. It is possible to set the ROI to represent the entire globe, but more useful information can often be gathered, especially in an optimization sense, when a user sets a clear ROI or set of ROIs of a more regional nature. From now on, this thesis will index a single ROI with the subscript k, and represent the set of all ROIs with the variable R.

However, this leads to an important follow-up question. What if a user wants to emphasize a subset of ROIs, more than others? This is a natural follow-up, but can be quickly solved by applying weights to each individual ROI. If the end user weighs each ROI uniformly, then the weights can be all set to 1. Following the above example, if the US wants to emphasize the value coming from observing the continental United States, it would assign a weight to the continental area that is higher than the weight assigned to Alaska. From now on, this thesis will represent the weight of a single ROI with the variable w_k and note that all weights across all ROIs need to sum to 1 for normalization purposes.

The second term has been developed to represent the combination of *quality* & *quantity* of data. This term is called the Effective Data Acquired (EDA) [4]. Given that each ROI can have its own unique quantity and quality of data gathered, or simply put, its own EDA, this thesis will represent the EDA of a particular region k with the following notation: EDA_k.

4.3.2 Function Overview

This thesis proposes that the value of an EO constellation architecture can be described with a simple summation formula.

$$\text{Architecture Value} = \sum_{k}^{R} w_k EDA_k \qquad (4.1)$$

As alluded to in the previous section, since each ROI is independent, the ROIs will be evaluated individually and the data collected, quality of this data, and the weight of the ROI will be used to create a value-based metric. w_k is dimensionless, whereas the EDA has units of bits [note that these units can be scaled to whatever form the end user views as useful such as Megabytes or Gigabytes]. This functional form is simple, but also very intuitive. Value is driven by the weighted summation of the quantity and quality of data, which is conveniently wrapped in the EDA variable. Thus, the structure, complexity, and usefulness of this proposed value function comes almost entirely from the way EDA is defined and calculated.

4.3.3 Effective Data Acquired Calculation

Prior to diving into the EDA equations, a disclaimer must be given. Due to the limitations of TAT-C, in particular the grid spacing effects described in section 3.4.2, as well as the nuanced nature of capturing the quantity and quality of data, the EDA formulation has rapidly evolved over the last year and a half. Constant iterations and improvements have been made in order to make the most robust and accurate model as possible. The functional form of the EDA could change in the future, thus the formulation that will be proposed should be examined as a living model that can and most likely will be updated in the future. The author of this thesis utilizes versions in order to keep track of the various iterations. At the time of writing, the EDA functional form represents the 2nd complete version, or simply version 2.0.

The EDA of ROI k can be expressed as followed:

$$EDA_k = \sum_{i=1}^{N_k} \sum_{j=1}^{A_i} (Q_{ij}\mu_{ij}) f(j;\theta) \qquad (4.2)$$

Let's break down each of the variables described in the above equation.

Variable [Units]	Description
N_k	Total Number of Points of Interest observed by the Architecture over Region k
A_i	Total Number of Access Events where Point of Interest i is observed
Q_{ij} [Bits]	Data Collected of Point of Interest i during Access Event j
μ_{ij}	Instrument Quality Metric of Point of Interest i during Access Event j
$f(j;\theta)$	Access Event Scaling Function

Table 4.2: Variable Description for EDA Calculations

N_k and A_i are purely dependent on the way the Orbits & Coverage module generates the grid points within each ROI as well as how the orbits are propagated during the numerical simulation. It is also worth noting that the longer the simulation in terms of the overall mission duration, which is set by the end user, the larger A_i will be. A_i is an important variable because it serves as a proxy for time. As the j index in the EDA function increases towards A_i, this by definition also implies that the time of the mission is increasing. This is important because by having a proxy for time, it is possible to consider timing related effects such as discounting or scaling. $f(j;\theta)$ represents this time scaling function and it will be discussed in section 4.3.3.3. Variables Q_{ij} and μ_{ij} are more nuanced variables and will be explained in sections 4.3.3.1 and 4.3.3.2 respectively.

4.3.3.1 Q_{ij} Breakdown

Understanding how much data is collected is critical for understanding how much value a constellation is generating. Unfortunately, this is a non-trivial problem. For one, the data collected by a satellite instrument does not automatically translate to data received by an end user. There are many limiting factors such as downlink issues, command and data handling issues, and many more that make it difficult to acquire an image and transmit this raw data down to the ground immediately. Also, due to the grid spacing issues within the Orbits & Coverage module, the way access events are recorded can impact the quantity of data. Thus, it is important to note that the quantity of data metric proposed in this thesis, Q_{ij}, serves as a proxy of the actual data collected by the satellite. The actual amount of data collected could be more or it could be less. Usually this is a cause of concern when examining this in an absolute sense. However, the goal of the value model is to produce a model that can be used in architecture *comparison*. Because of this fact, absolute data quantity is not as important as relative comparison. What this means is that even though the model may not produce an absolutely accurate representation of data quantity, if architecture A collects 5 times as much data as architecture B, then it is rational to conclude that architecture A collected more data than architecture B. Note, for architecture selection purposes it is not important to know how much data

was actually collected by architecture A, rather, it is just worth noting the relative amount of data collected between the two architectures. With this knowledge in mind, it is now possible to derive the equations needed to generate the proxy amount of data collected.

$$Q_{ij} = \text{Number of Ground Pixels} * \text{Bits per Pixel} \tag{4.3}$$

$$\text{Number of Ground Pixels} = \frac{\text{Access Event Footprint Area}}{\text{Ground Pixel Area}} \tag{4.4}$$

$$\text{Ground Pixel Area} = \rho_{CT}\rho_{AT} \tag{4.5}$$

$$\text{Access Event Footprint Area} = \text{Footprint Length} * \text{Footprint Width} \tag{4.6}$$

$$\text{Footprint Length} = 2h \tan \frac{CrossTrackFOV}{2} \tag{4.7}$$

$$\text{Footprint Width} = 2h \tan \frac{AlongTrackFOV}{2} \tag{4.8}$$

As the above equations show, the amount of data collected during access event j and of POI i is a function of the bits per pixel of the instrument, the ground pixel resolution of that access event, $\rho_{AT/CT}$, and the footprint area of the image on the Earth's surface.

The bits per pixel of the instrument is provided in the instrument specifications, and will be held constant assuming a single instrument is being analyzed. The bits per pixel describes the quantization of the imager, or the number of bits contained in a single pixel. The footprint area, or the size of the image captured, is a function of the Cross Track Field of View (FOV) and the Along Track FOV. Although the instrument module only calculates ground pixels that are observed in the cross-track direction, when determining the footprint area both along and cross track must be considered. The instrument module limitation will impact when an access event is recorded, but when an event is recorded, obtaining an accurate measure of the footprint area requires both along and cross track directions. Similar to the bits per pixel, these two terms are also specified in the instrument specifications meaning they are held constant for a single instrument. The footprint area is also a function of the altitude of the satellite. This is important to note because all else being constant, the higher the altitude the larger the footprint, and thus the larger quantity of data collected. However, as the altitude increases, the ground pixel area decreases. These two forces tend to cancel each other out which is important because it means that higher altitudes do not explicitly imply larger quantities of data collected, which is

true in practice. The footprint area equation above represents a rectangular sensor type. In practice, an instrument could have a rectangular sensor or even a conical sensor. These can be expressed visually in Figure 3-6 that is also shown below for reference.
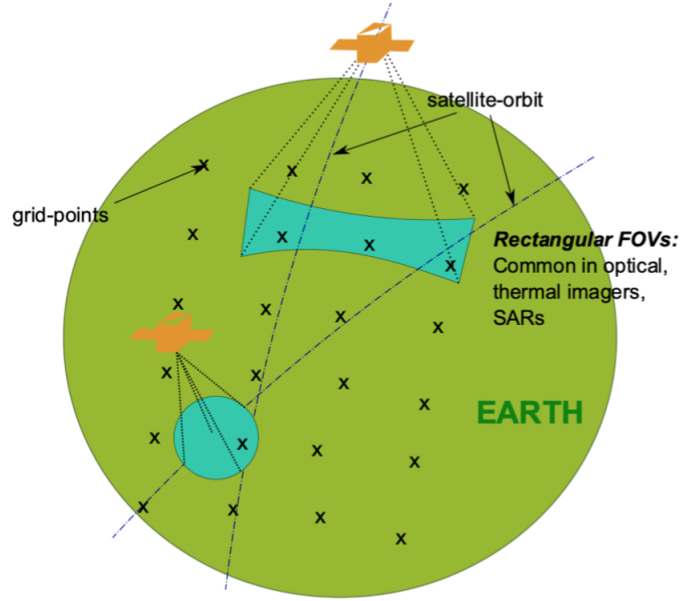
Illustration of satellite orbit and coverage

Figure 4-8: Representation of Conical and Rectangular Sensor

For a conical sensor, the footprint area can be calculated as followed.

$$\text{Conical Radius} = h \tan \frac{Conical FOV}{2} \tag{4.9}$$

$$\text{Conical Sensor Footprint Area} = \pi (\text{Conical Radius})^2 \tag{4.10}$$

In summary, the amount of data collected at each access event for a single POI is primarily driven by the orbital mechanics and the instrument specifications. These two factors do impact the actual quantity of data in reality, thus Q_{ij} can be looked at as a reasonable proxy for the amount of data collected.

4.3.3.2 μ_{ij} Breakdown

The second component of the Value Function term EDA_k is to consider the quality of data that has been acquired. Each time an instrument takes an image of the Earth, this data can be high or low quality. Higher quality data is preferred, in that scientists and end users can use it for more useful purposes. The reason why this quality metric is labeled the Instrument Quality metric, is because it varies depending on the instrument class.

For Passive Optical Scanners, this thesis proposes the use of the SNR. For a detailed description and functional decomposition of the SNR, see sections 2.2.3.1 and 3.5.4 respectively. Based on the way the EDA metric is formulated, see equation 4.2, the higher the instrument quality metric, the higher the EDA and thus architecture value, all else constant. This makes sense because a higher quality of data implies more value to the end user. Because increasing SNR implies higher quality of data, this thesis sets μ_{ij} to be equivalent to the SNR at an access event j for a POI i.

$$\mu_{ij} = SNR_{ij} \qquad \text{For Passive Optical Scanners} \qquad (4.11)$$

For Synthetic Aperture Radars, this thesis proposes the use of the Achievable noise equivalent reflectivity, also known as the σ_N or σ_{NEZ0} as the instrument quality metric. However, unlike SNR, where increases in the SNR imply an increase in image quality, for σ_N, the reverse is true. The lower the σ_N, the higher the image quality. In order to accurately capture these dynamics, μ_{ij} must be cleverly transformed to ensure that the proper value hypothesis dynamics still function properly. μ_{ij} for SAR instruments is defined as followed

$$\mu_{ij} = -(\sigma_N)_{ij} + C \qquad \text{For Synthetic Aperture Radars} \qquad (4.12)$$

The first and most important question that comes from this formulation is what is the C value in the equation. Here, C represents a constant value that will be applied to all architectures that are being compared. The reason for C is to create a functional structure that allows the EDA to increase while data quantity is increasing, and data quality is increasing. In the SAR case, data quality increases when the instrument quality metric, σ_N, decreases. This can best be expressed visually in the following figure. Figure 4-9 depicts the μ_{ij} function for SAR instruments. In this case, it shows it for a C set to 30.
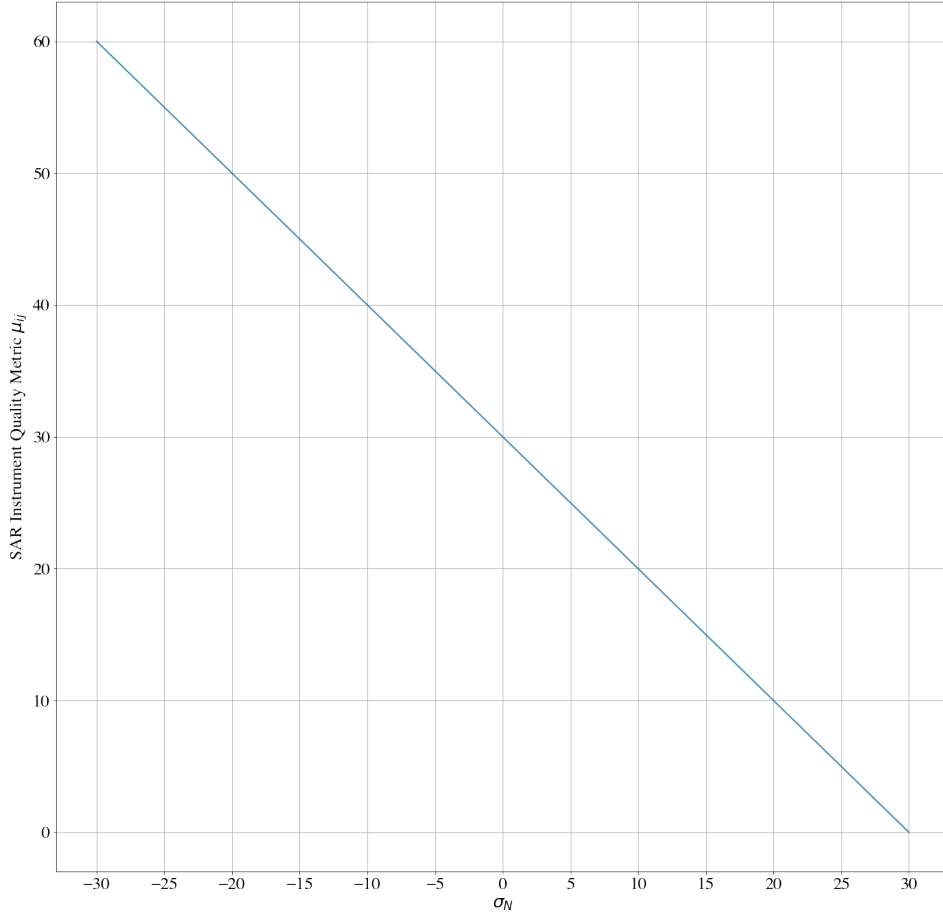
Figure 4-9: Example μ_{ij} Functional Form for SAR Instruments

The above figure shows how the SAR instrument quality will increase even as σ_N decreases, thus achieving the desired functional form. Currently, this thesis sets C to be the maximum value of σ_N across all architectures for the given simulation. This is by no means the best way to set C, but enables numerical results to be bounded within a reasonable range. This concept will be revisited in Section 5.3, which discusses a SAR case study.

In summary, this thesis uses two common and well understood instrument performance metrics in order to capture the quality of data that is being obtained by the architecture. These metrics were used not only due to their large use in the field, but because they also are outputs of the Instrument Module from TAT-C. This means the Value Module has all of the necessary pieces from a computational standpoint from the standard outputs of TAT-C.

4.3.3.3 Scaling Factor: $f(j; \theta)$ Breakdown

The last core component of the EDA function is the time scaling factor, or $f(j; \theta)$. As the formula suggests, this scaling factor is a function of the jth access event of POI i, parameterized by a value θ. The parameter θ will be discussed shortly. Given the

way the function is created as well as in practical implementation, all of the access events for a POI i are sorted according to their time from the start of mission. This implies $j = 1$ is the first access event for POI i and $j = A_i$ is the last access event for POI i.

This time function was inspired by an important concept in Finance: the time value of money [72]. In the financial world, assets received today are worth more than assets received tomorrow. This is because assets that are received today can be deployed and put to work in order to generate returns prior to receiving assets tomorrow. In the standard form, the time value of an asset can be expressed mathematically. Let's say an asset, A, is received in T days time. If the daily interest rate r is assumed, and the time horizon is set to T days, then the value of this asset today, A_0, can be written as follows in its time discretized form. (Note: a continuous time version contains the exponential e^{-rt}).

$$A_0 = \frac{A}{(1 + r)^T} \tag{4.13}$$

As this formula shows, as long as the interest rate is greater than 0, then the value of an asset received in the future is less than the asset value A_0 had we simply received it today. This formulation can also be generalized for various other asset flows - such as information - that occur at each time step.

$$A_0 = \sum_{t=1}^{T} \frac{A_i}{(1 + r)^t} \tag{4.14}$$

Following on this analogy, this thesis also postulates that the value of data gathered by a satellite today is worth more than data gathered tomorrow, assuming constant downlink latency. The more data a scientist or end user has today, the higher likelihood that they can deploy models, algorithms, etc., that generate value to various stakeholders. Just how in the financial world different agents have different interest rates to express their sensitivity to time, different end users could also have different time based rates as well. For example, given that glaciers move very slowly, scientists that observe glacier movements have a smaller sensitivity to time compared to scientists who track hurricanes since hurricanes move at 10 m/s whereas glaciers move at less than 0.0001 m/s, meaning data taken today that is also received today is only slightly more valuable than data taken today received tomorrow [73]. Following the financial example, this would imply an interest rate close to 0. Hurricane monitoring is the opposite, where timely data is absolutely key to mitigate loss of life and property. In this case, r would be set quite high. The question then becomes how does the time value of data work its way into the EDA formulation, i.e. how is $f(j; \theta)$ defined?

When developing the $f(j; \theta)$ function, simplicity and physical intuition are paramount. Following the financial analogy, the scaling function could be set as follows:

$$f(j;\theta) = \frac{1}{(1+\theta)^j} \qquad (4.15)$$

As j increases, the scaling factor f gets smaller assuming a constant θ. To help depict this, a simple example case can be created. By setting θ to be equal to 0.05, and A, which can be thought of as $Q_{ij}\mu_{ij}$, to 10, then we can plot the cumulative sum similar to how the EDA is formulated.
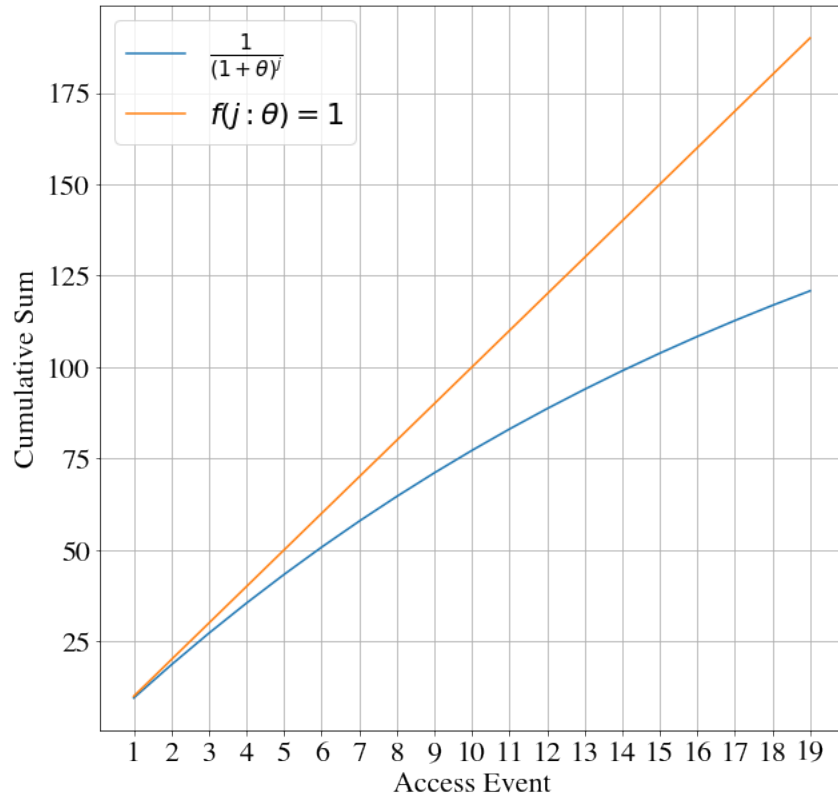
Figure 4-10: Scaling Factor Represented as Time Value of Data

The above figure not only depicts the formulation that is similar to the financial time value of money concept, but also compares it to a scaling function that is set to 1, meaning there is no scale applied. This is important because as the curves show, the scaling function is never above the non-scaled version. This mathematically makes sense, but may not make much sense in a physical intuition sense. What this means is that the first one or two access events, for example, may be *most* valuable to the end user. In fact, these two access events may provide more value than when no scaling factor is applied at all. However, as j increases, the value of additional observations decreases as well. This concept is best known as having diminishing returns, meaning each subsequent contribution to the EDA decreases until the EDA overall roughly levels out. Equation 4.15 exhibits this behavior, but it does not allow for that increase in value that is sought after at the key beginning stages. In order to exhibit this behavior, the function must be updated as followed.

$$f(j; \theta_1, \theta_2) = \frac{1}{(1 + \theta_1)^{j - \theta_2}} \tag{4.16}$$

As the equation shows, there are now two parameters, θ_1 and θ_2. θ_1 controls the rate, where as θ_2 gives the function the ability to increase at the beginning by making the exponent negative, and then decrease overall. Figure 4-11 depicts this graphically, where again A was set to 10, and θ_1 was set to 0.05 and θ_2 was set to 5.
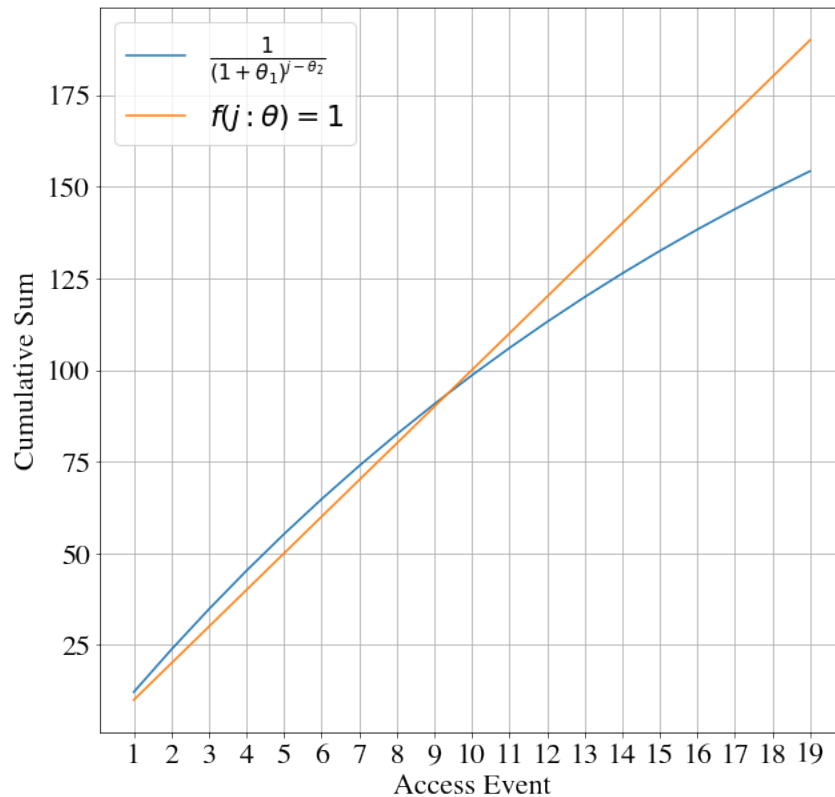
Figure 4-11: Scaling Factor with Increasing Returns at

96

Although this functional form captures the dynamics that are desired, there are now two parameters that must be determined. In order to maintain the functional form and reduce the parameter dimension, the following functional form will be used from now on in order to describe the scaling function.

$$f(j; \theta) = \frac{1}{(j+1) * \theta} \tag{4.17}$$

This new equation provides the same diminishing returns, while also enabling higher value to be associated to earlier access events. The only negative of this functional form is that it no longer adheres to the standard time value of data concept. This is because lower values of θ actually *increase* the value for early access events. Using the same A as for Figures 4-10 & 4-11, the following figure was generated to depict how the cumulative sum changes as a function of j for varying levels of θ. As Figure 4-10 shows, depending on the θ used, the amount of value attributed to the early onset access events can be greatly influenced. What is important to examine is where each θ curve intersects the linear curve. These critical points help determine how much increased value to attribute to a certain number of access events. In actual implementation, the θ value can be set by the end user, or even tuned using standard validation practices in order to find an optimal value.

As a concrete example, using Figure 4-12 as reference, Table 4.3 shows example θ values for two applications: hurricane and glacier analysis.

Application	θ Range	Rationale
Hurricane Tracking	0.1 - 0.2	Majority of the value comes from first few images.
Glacier Analysis	0.5 - Linear	Glacier data today is minimally more valuable than data in the future.

Table 4.3: θ Ranges for Hurricane & Glacier Applications

One final point is that A_i, which already serves as a proxy of time, also indirectly contains information about the number of satellites in a constellation. As the number of satellites in a constellation increases, the number of access events for a given POI will also increase since there are more eyes in the sky able to observe it.

4.3.3.4   EDA Formulation Summary

Pulling together the above analysis, the EDA function can be re-written as follows:

$$EDA_k = \sum_{i=1}^{N_k} \sum_{j=1}^{A_i} (Q_{ij} SNR_{ij})(\frac{1}{(j+1) * \theta}) \quad \text{For Passive Optical Scanners} \tag{4.18}$$
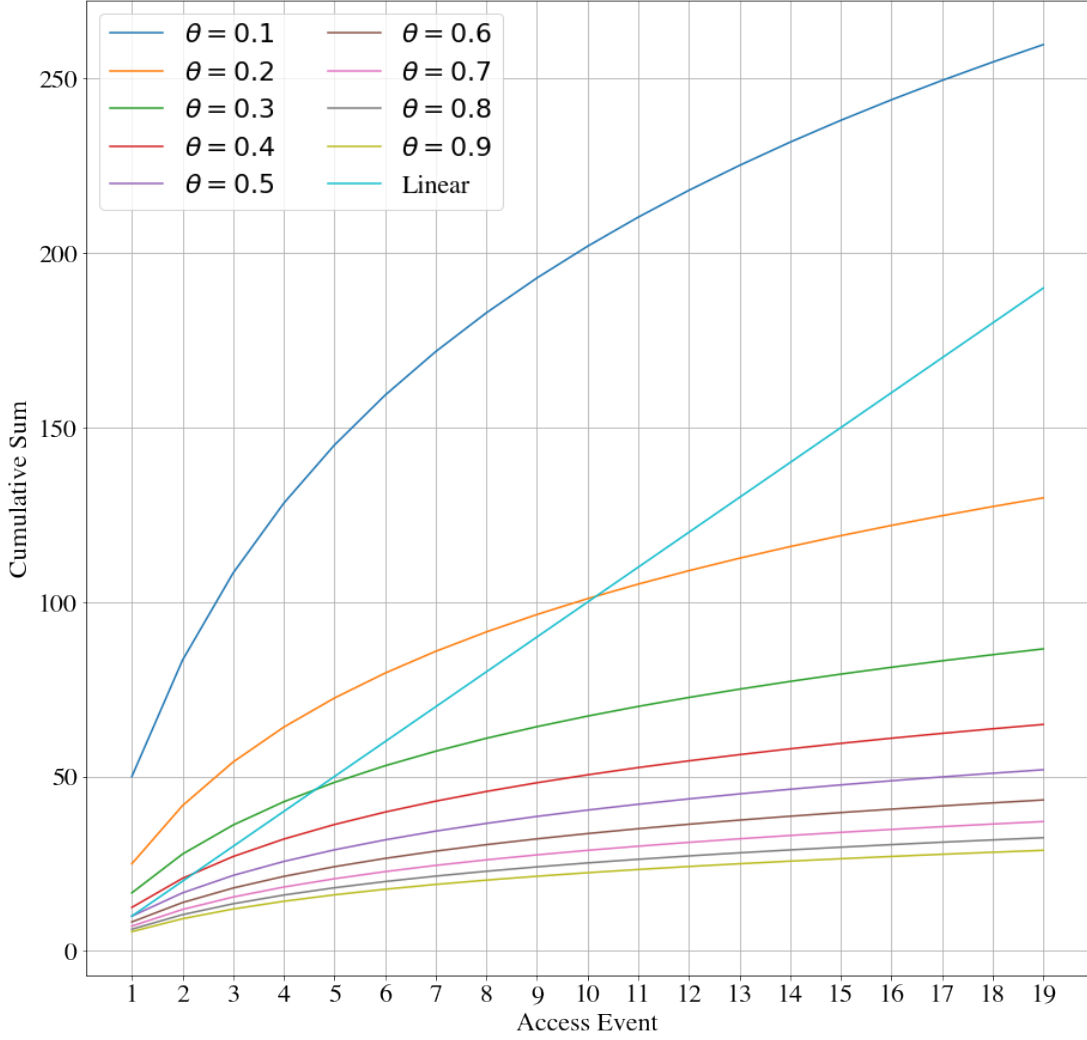
Figure 4-12: Simplified Scaling Factor Functional Decomposition

$$EDA_k = \sum_{i=1}^{N_k} \sum_{j=1}^{A_i} (Q_{ij} \frac{1}{|(\sigma_N)_{ij}|}|)(\frac{1}{(j+1) * \theta}) \quad \text{For Synthetic Aperture Radars} \quad (4.19)$$

Thus, the value of a single EO constellation architecture that consists of R ROIs can also be re-written as:

$$\text{Architecture Value} = \sum_{k}^{R} w_k \sum_{i=1}^{N_k} \sum_{j=1}^{A_i} (Q_{ij} \mu_{ij})(\frac{1}{(j+1) * \theta}) \quad (4.20)$$

The additional outer sum here sums the value of the EO observation architecture across all regions of interest specified on the surface of the Earth.

4.4 Value Function Limitations

Although the Value function as formulated above does an effective job capturing the various dynamics that impact value at a general level, there still exist some key limitations to the current formulation.

4.4.1 Expert Knowledge

One of the most important limitations to the value model is that it does not use previous expert knowledge about EO constellations to calculate value. This is by design - the model was developed to not require previous knowledge, but this can still cause certain problems that are worth addressing. According to the value formulation, EO constellations that gather more high quality data are valued more than those that gather less high quality data. This in a sense gives value to the supply side of Earth Science data as opposed to the demand side. To calculate the quantity and quality of data, Q_{ij} and μ_{ij} respectively, calculations that involve orbital geometry and instrument specifications are used. However, these calculations do not consider important geometric concepts such as how Sun-synchronous Orbits (SSO) allow almost constant lighting conditions when the satellite is overhead [23]. This can be useful for a wide variety of applications that use EO observation data [74]. Ideal geometric concepts like SSO can trickle through μ_{ij}, but this single metric may not capture all of the valuable dynamics involved. Utilizing expert knowledge can be an efficient way to address the value trade approach, similar to how Selva used it to develop rules-based systems [68]. Although this is a limitation of the model, the lack of explicit representation of expert knowledge enables the metric to be used in the absence of expert knowledge which can be costly to acquire.

4.4.2 Missing Data

The value function formulation can be looked at as a post processing tool, thus the inputs to this model are critical. Using a tool such as TAT-C provides the necessary data to calculate the value metric, however, if a user does not have access to TAT-C, then it may be difficult to utilize the value function. It cannot be calculated using simple hand calculations, even though it may be approximated that way. This is emphasised by the need for instrument specific observation data which is non-trivial to calculate. This makes large scale use of the value metric, which is ideal because it will enable continuous improvement, more challenging. However, given the reliance on data from TAT-C, the value function has been implemented directly into the code base so end users can calculate value while executing TAT-C in its entirety.

4.4.3 Relative vs. Absolute Results

Although the Value Function produces results that have a familiar unit of measurement, the actual numerical representation may not be particularly helpful to the end user. If one architecture produces 1,000 GB of value as defined by the Value Function,

this may seem like a large quantity, but it can only be compared against other architectures in a comparative sense. Thus it is a relative metric, not an absolute metric. On the surface, this is not a major limitation, but it is worth noting so that future users do not get confused by the scale of results. Once a particular architecture has been selected during pre-Phase A using TAT-C a more detailed analysis and absolute data output predictions can be made.

Chapter 5

Value Function Case Studies

In order to better understand the underlining dynamics of the Value Function, this thesis completed a set of case studies. The first set of case studies, which will now be referred to as Case Study A, focus on existing EO constellations. In particular, Case Study A will perform a value-driven trade on the existing Landsat and RapidEye constellations. This will provide validation that the value framework developed here is applicable to existing architectures. The second set of case studies, which will be referred to as Case Study B, will focus on a particular use case: the make or buy decision (should an organization purchase off-the-shelf imagery or launch a dedicated EO constellation?). Case Study B will highlight the importance of value-driven trade studies in order to make the critical make or buy decision from an EO data perspective. Finally, Case Study C will perform a small architecture exploration for a proposed SAR small satellite constellation. This case study will show how the Value Framework can be applied to future missions and those with SAR instruments.

5.1 Case Study A: Existing Earth Observation Mission Trade Analysis

Case Study A will examine two existing EO constellations: Landsat and RapidEye. Although Landsat is not a multi-spacecraft constellation, its extensive use, importance, and publicly available information make it a suitable candidate for analysis. The RapidEye constellation consists of 5 small-size & identical EO satellites, which makes it a very interesting case study to evaluate [1].

5.1.1 Landsat

Before diving into the results of the Landsat trade analysis, it is important to understand the Landsat satellite and how the trade analysis was developed. For this case study, the Landsat 8 satellite was utilized as a reference. The Landsat 8 mission was created in order to continue the United States government's effort to maintain EO

data continuity [18]. The Landsat 8 satellite was launched on February 11, 2013 by the United Launch Alliance from the Vandenberg Air Force Base in California [75]. The primary mission objectives of Landsat 8 included the following [76].

- Collect and archive multispectral & thermal data affording to seasonal coverage of the globe for a period of no less than five years

- Ensure that the Landsat Data Continuity Mission data are consistent with data from previous Landsat Missions

- Distribute Landsat data products to users at no cost

The following two tables show the high level attributes of the Landsat 8 satellite, which include spacecraft mass and orbital geometry, and the instrument specifications for the OLI onboard instrument [18], [77]. Note the wavelength used represents a single image band, visible blue, for analysis.

Attribute	Value
Spacecraft Mass	2623 kg
Altitude	705 km
Inclination	SSO - 98.22°

Table 5.1: Landsat 8 High Level Attributes

Instrument Specification OLI	Value
Mass [kg]	450
d [m]	0.000036
f [m]	0.8451
Along Track FOV	0.00244°
Cross Track FOV	15°
λ [m]	4.82e-7
D_{ap} [m]	0.132
τ_{op}	0.9
Q_E	0.9
N_r	20
Bits per Pixel	12

Table 5.2: Landsat 8 Instrument Specifications

Now that we have a better understanding of the existing Landsat satellite, it is possible to design a trade study to re-evaluate the existing mission design in a value-focused framework. In particular, this thesis examines how the high level attributes of orbital altitude and inclination impact the value function, and whether the existing design of Landsat 8 is the optimal design from a value-focused mindset.

From a trade space perspective, this thesis utilizes 5 design variables. Table 5.3 breaks down the design variables and the values that were traded upon for each. Table 5.4 shows the design parameters that were held constant for the analysis.

Design Variable	Traded Values
Number of Satellites	[1, 2, 3, 4]
Number of Orbital Planes	[1, 2]
Orbital Altitude	[500, 550, 600, 650, 700, 710, 750, 800]
Inclination	[0°, 30°, 60°, 90°, SSO]

Table 5.3: Landsat Trade Analysis Design Variables

Design Parameter	Constant Values
Constellation Type	Homogeneous Walker
Region of Interest	Global
Simulation Duration	40 days
Maximum Grid Size	1000 Grid Points

Table 5.4: Landsat Trade Analysis Constant Design Parameters

Given the range of design variables, 320 unique architectures were generated using the TAT-C program. It is worth noting that more values of inclination, altitude, etc., could have been used, but this would have dramatically increased the size of the tradespace. Also, the ROI for each architecture was held constant to encompass the entire globe, and the type of constellation was also set constant to represent a Homogeneous Walker. Heterogeneous Walker constellations, although more interesting due to the increased combinatorial space, become computationally extensive, and thus were not considered. Also in a heterogeneous constellation other parameters such as GSD and image quality may be more difficult to compare to Landsat 8 and thus undermine the data continuity part of the mission.

5.1.1.1 Results

It is first worth examining the distribution of grid points, or POIs, generated by TAT-C during this case study. As explained prior, the target region was set to be the entire globe, and the maximum number of grid points was set to 1000. Because of this, Figure 5-1 displays the Earth as a flat globe, specifically an equirectangular projection, and where each POI was centered from a latitude and longitude perspective.
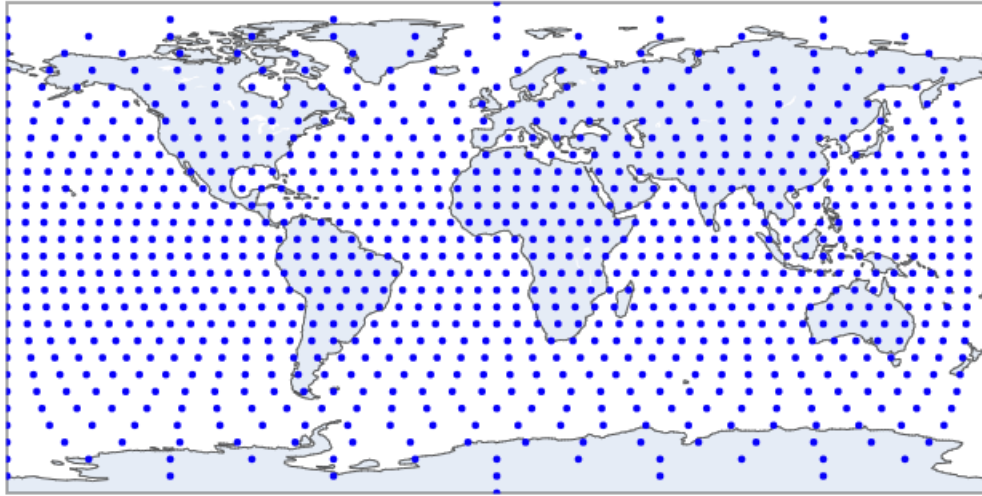
Figure 5-1: TAT-C Global Grid Point Distribution (1000 points)

One of the best ways to analyze the Value Function metric is to plot it against the estimated lifecycle cost of the architecture. Figure 5-2 below depicts this relationship. Note, that this plot is showing the relationship between architecture value and cost when θ from the access event scaling function is set to 0.1. Changing the θ value will simply change the scale of the Architecture Value, but the relative shape of the graph will stay constant. See Appendix A for detailed representations of the Architecture Value plots for various values of θ. It is also worth noting that in the below figures, the variation of colors for each data point is based on a specific attribute of the constellation. For example, in the first plot shown, Figure 5-2, the color is based on the altitude of the satellites in the constellation. In the figure, the actual Landsat-8 architecture is labeled with the purple X on the left side. Visualizing where this architecture exists within the tradespace is important and will be the main focus in the upcoming discussion. Also, a polynomial line of best fit of degree 2 was added to each graph to help display the non-linear nature of the Value Function as a function of lifecycle cost (which is primarily driven by the number of satellites - see Section 3.6.1 for a detailed breakdown of the TAT-C cost module). This shows a relationship of diminishing returns as would be expected. Also higher altitude orbits appear more valuable, presumably because of improved POI revisit frequency as discussed below in more detail. The coefficient of determination, or R^2, is also displayed for this line of best fit.
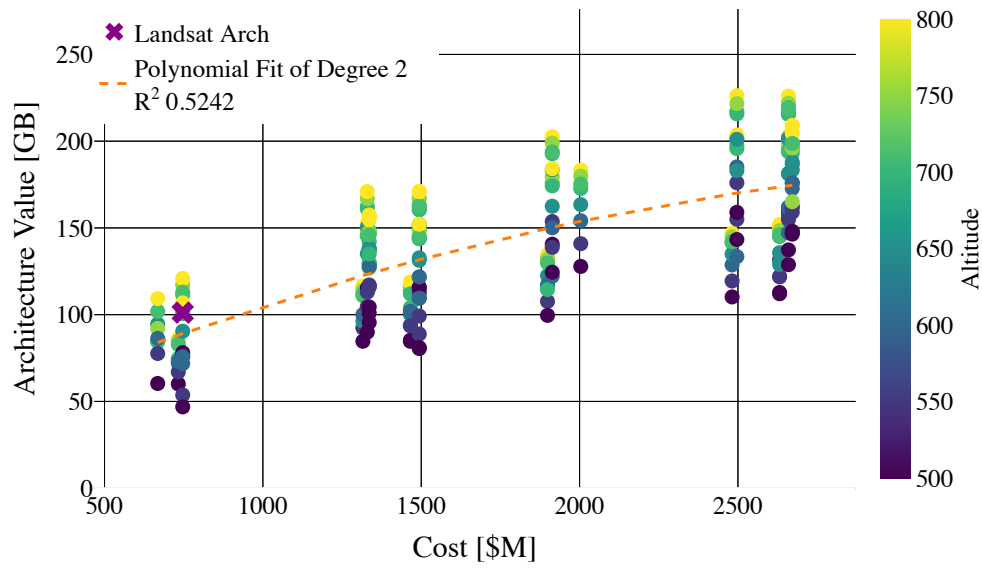
Figure 5-2: Landsat Architecture Value Analysis
Color Scheme: Altitude

The following three figures show the architecture value and cost relationship, just as Figure 5-2, but the color of each data point is based on the inclination of the constellation orbit, the number of POIs observed per architecture, and the average revisit time for each architecture respectively. From these charts, it becomes clear that the Pareto-optimal set of architectures exist when specific conditions are met. If the number of satellites is held constant, the optimal architectures always have an inclination of 90° and an altitude of 800 km. The following section will take a deeper look into this phenomena.
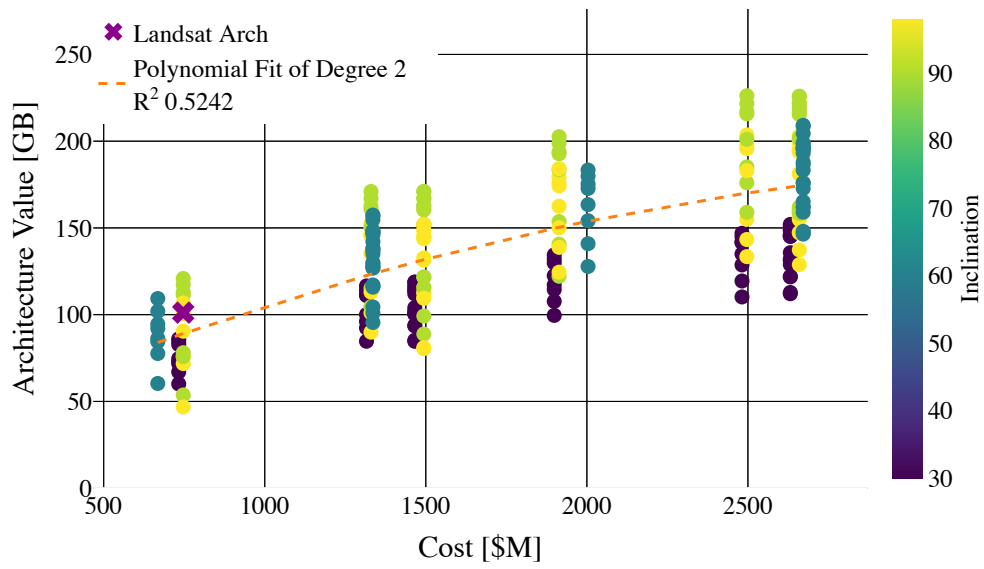
Figure 5-3: Landsat Architecture Value Analysis
Color Scheme: Inclination
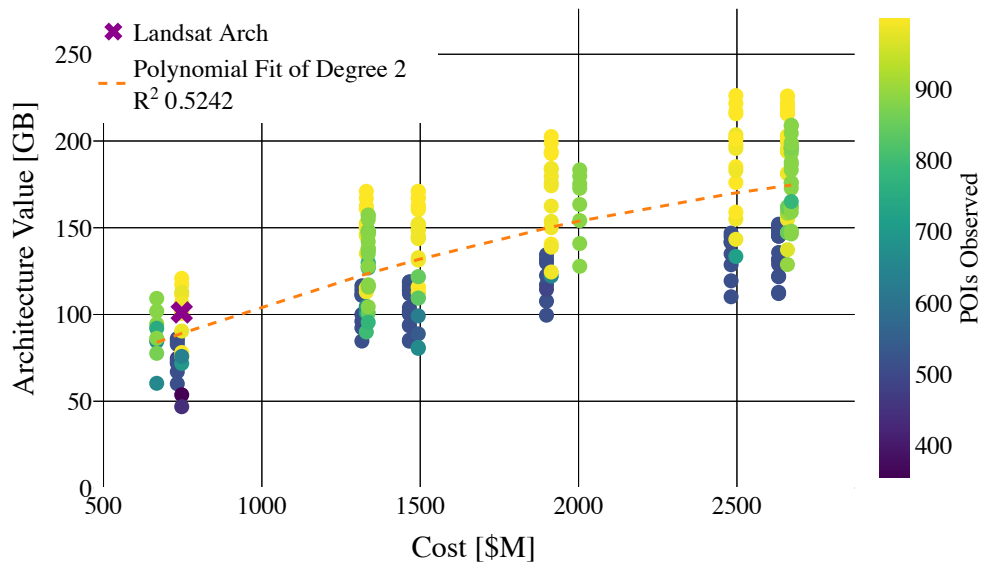
Figure 5-4: Landsat Architecture Value Analysis
Color Scheme: POI Observations

With regards to the number of POIs observed, there is no guarantee that all 1000 POIs will be observed during a simulation, especially if specific orbital elements do not prohibit certain areas to be observed or the simulation time is not sufficient.
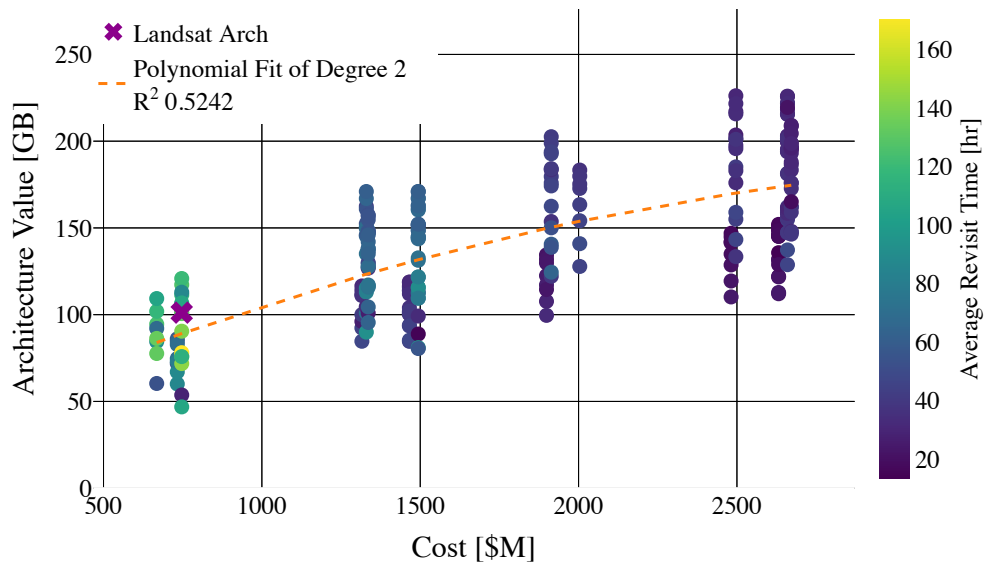
Figure 5-5: Landsat Architecture Value Analysis
Color Scheme: Revisit Time

Another useful visualization is to display the architecture value as a function of average revisit time. Average revisit time is a very common metric that mission designers care about, so understanding the dynamics involved is quite helpful. Figure 5-6 displays this relationship. Again, as in Figure 5-2, θ from the access event scaling function is set to 0.1. Note how in Figure 5-6, the average revisit time tends to decrease as the number of satellites increase. (The worst case revisit time for the actual Landsat 8 satellite is 16 days [18]. This difference is due to the grid spacing and the way the Orbits & Coverage module calculates revisit.)
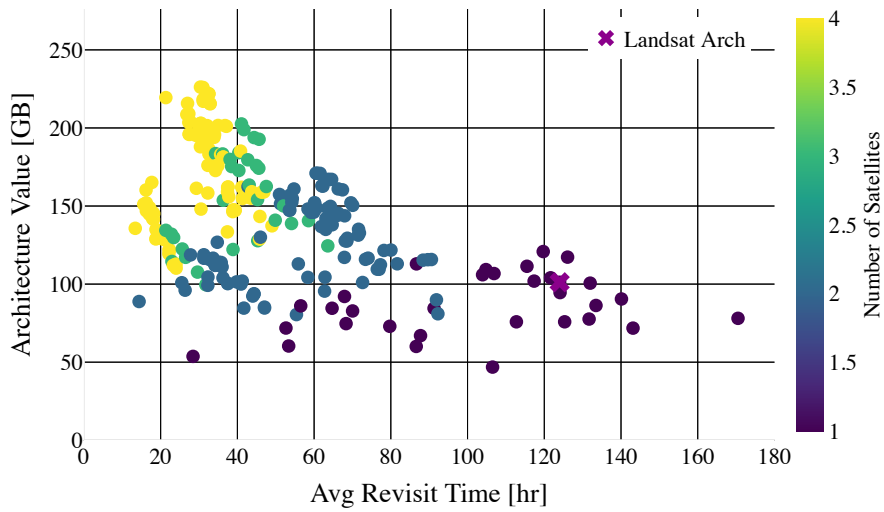
Figure 5-6: Landsat Value - Revisit Time Analysis
Color Scheme: # of Satellites

After exploring the data, another useful way to evaluate the relationship between value and revisit time is by showing the inclination of each architecture as the color scheme of the figure. Figure 5-7 depicts this, and helps explain the stratification within the data.
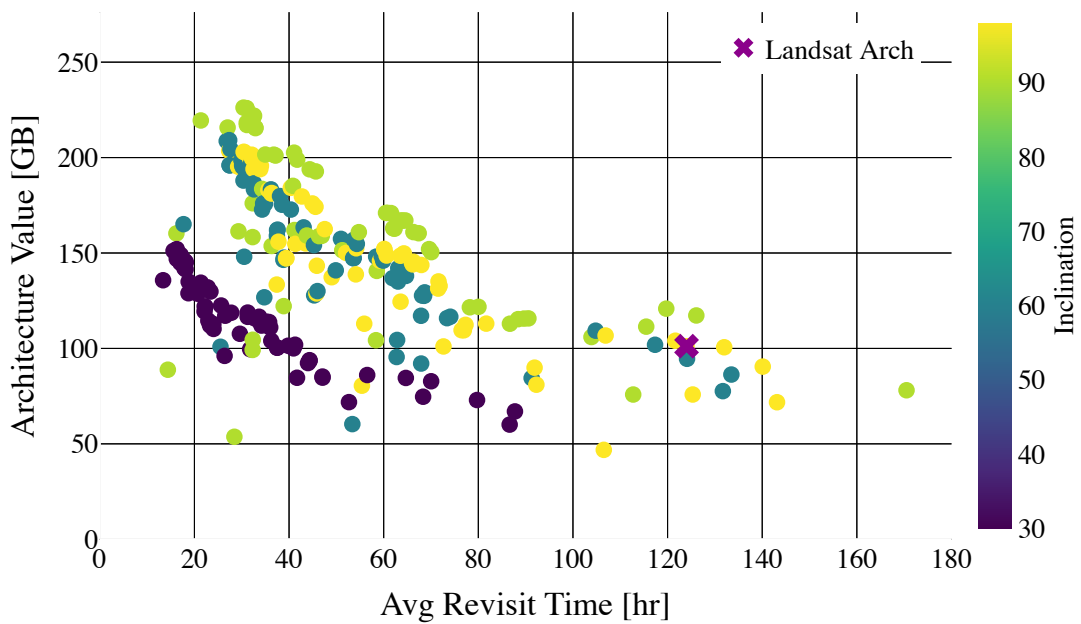
Figure 5-7: Landsat Value - Revisit Time Analysis
Color Scheme: Inclination

5.1.1.2   Discussion

From all of the figures displayed above, it is evident that the existing Landsat constellation architecture is not optimal within the value framework presented in this thesis. To break this down, it is important to isolate variables in order to see which variables were having the strongest effect. Given that the Landsat constellation consists of a single spacecraft, this thesis will pay close attention to the simulated architectures that only contain a single satellite.

This trade analysis generated 32 architectures that have a single satellite. The actual Landsat architecture had the 10th highest architecture value of these 32 architectures. It is again worth noting that the actual numerical value generated by the Value Function is not relevant in an absolute sense, rather, the relative comparison of value across architectures is important. By normalizing the value generated by the actual Landsat 8 architecture (architecture index 232), it is easy to see how much more or less value each architecture generates, as seen in Figure 5-8. From this figure, the best architecture, as defined by the value function, generates 19.44% more value as compared to the actual Landsat architecture. Thus for the same lifecycle cost, architecture 304 gets roughly 20% more value per dollar spent! Let's dive deeper into this to understand why this maybe the case.

	Architecture	Architecture Value [GB]	Normalized
0	arch-304	120.843333	119.44
1	arch-264	117.268994	115.91
2	arch-224	112.951842	111.64
3	arch-184	111.407155	110.11
4	arch-296	109.291849	108.02
5	arch-312	106.763509	105.53
6	arch-144	105.978421	104.75
7	arch-272	104.034211	102.83
8	arch-216	101.913118	100.73
9	arch-232	101.173471	100.00
10	arch-192	100.612601	99.45
11	arch-136	94.563696	93.47
12	arch-256	92.078826	91.01
13	arch-152	90.466509	89.42
14	arch-96	86.310727	85.31

Figure 5-8: Top 15 Single Satellite Architectures
Actual Landsat Highlighted Blue

Architecture	Normalized Value	Cost [$M]	POIs Observed	Revisit Time [hr]	Inclination	Altitude
arch-304	119.44	746.89	999	119.72	90	800
arch-232	100.00	746.89	986	124	98	710

Table 5.5: Landsat Architecture Compared to Best Single Satellite Architecture

By examining Table 5.5, a couple of things become apparent. The first is that architecture 304 observed 13 more POIs as compared to the actual Landsat architecture. (Note that if the simulation time was increased, we would expect all POIs to be observed by the Landsat reference architecture.) Given the way the value function is formulated, this will cause more value to be accrued since more POI observations implies more data generated. However, 13 more POIs only represents a 1.32% difference. The next key set of parameters to examine are the inclination and altitude of the respective orbits. Architecture 304 has a higher altitude and a perfect polar orbit. Initially, it is difficult to understand how this would cause a large difference in value, but by digging deeper into how this impacts the SNR distribution, a clear difference arises.
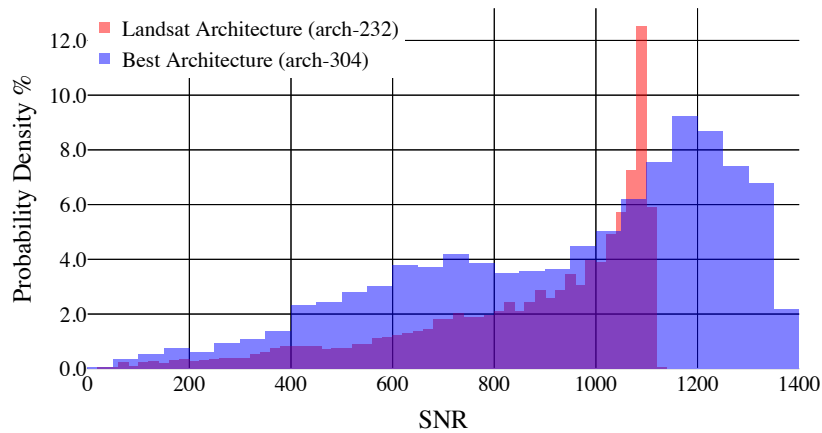
Figure 5-9: Landsat Architecture SNR Comparison to Best Single Satellite Arch.

Figure 5-9 shows the actual Landsat SNR distribution in red and architecture 304 distribution in Blue. Notice how architecture 304 has higher density of larger SNR observations, which implies higher quality data. This is an important discovery, and helps drive the bulk of the difference between architecture 304 and the actual Landsat architecture. As section 3.5.4 breaks down, there are a multitude of factors that go into calculating the SNR for a single observation. Thus it is hard to make a statement about how the SNR is impacted by altitude and inclination given they impact a wide variety of competing sub-functions including $\rho_{CT/AT}$ and θ_i^{Solar}. However, what is evident is that the max SNR value for the Landsat architecture is 1140 db, and the max SNR for architecture 304 is 1400. This is a 22.8% difference. Equation 3.12 shows that the GP Area is directly proportional to N_s, which is the numerator in the SNR equation. N_s is also in the denominator. Since N_r is an instrument specification, we can back out N_s for each architecture. Table 5.6 depicts the N_s values for architecture 232 and 304 and the percent difference.

Architecture	N_s
232	1299999.8769609
304	1960399.9184006
Difference	51%

Table 5.6: N_s Data Table for Landsat Architectures

As the above table shows, there is a large difference in the number of photons at the detector for each architecture. This is driven primarily by the ground pixel area, but can also be influenced by $\mathbf{R}$ and T_i, the integration time of the ground pixel. What is clear from this comparison is that higher altitudes and the polar orbit allows for more signal photons to be detected, which results in a larger SNR distribution.

Attribute	Architecture 304	Architecture 232	Percent Difference
Altitude [km]	800	710	12.676%
Average Cross Track GP Resolution [m]	34.2882001	30.4500131	12.605%
Average Along Track GP Resolution [m]	34.1770352	30.341798	12.64%
Average GP Area $[m^2]$	1171.86902	923.908144	26.838%
Average GP Area $[km^2]$	0.00117187	0.00092391	26.838%
Footprint Area $[km^2]$	29.211887	23.0089254	26.959%
Number of Ground Pixels	24927.6041	24903.9101	0.095%

Table 5.7: Landsat and Best Single Spacecraft Q_{ij} Detailed Breakdown

To further compare these two architectures, it is also important to examine the inputs that feed into the Q_{ij}, or data quantity calculation. The major inputs for the data quantity calculation, as seen in section 4.3.3.1, include the altitude of the satellite, the ground pixel resolution, and footprint area. Table 5.7 compares these attributes for architecture 304 and the Landsat architecture, and shows how they differ. What is important to note is that although the ground pixel resolution is 26.84% higher, thus worse, for architecture 304, the difference in altitude and footprint area implies more ground pixels covered in each POI access event. These two forces counteract

one another, and yield a slight difference in the number of ground pixels observed per access event which also helps explain the difference in value obtained by each architecture. Overall, the big driving factor behind the enhanced performance from a value framework of architecture 304 as compared to the actual Landsat 8 architecture 232 is the superior SNR distribution. The amount of data collected is also higher due to more POI observations driven by the higher altitude and an increased number of ground pixels gathered per access event, but these impact the spread in a smaller fashion.

While thinking about the SNR distribution, this thesis will now briefly discuss the impact of band choice in the tradespace. For this case study, a single band was used, the blue band, at a central wavelength, λ, of 4.82e-7 meters. The way the Value Function is formulated, λ will not impact the quantity of data collected, but will impact the SNR calculated at each orbital pass. In particular, λ directly impacts the equation that calculates the radiance from Earth in the direction of the target, or L_E. (For further breakdown of this equation, see Equation 3.14) While, this does not directly imply that certain bands provide more quality over others, it is a useful exercise to see how the Value metric and SNR distribution would change if the λ used was different. To show this, architecture-304, the best single satellite architecture, was re-run but using the red, green, and NIR bands. The λ for these bands are shown below in Table 5.8.

Band Name	Central Wavelength [m]
Red	6.45e-7
Green	5.55e-7
NIR	8.65e-7

Table 5.8: Alternative Bands for Architecture Analysis

As stated prior, the band wavelength will impact the SNR of an observation. For architecture-304, the following plots show the SNR distribution compared to the standard blue band that was used for the analysis in this case study.
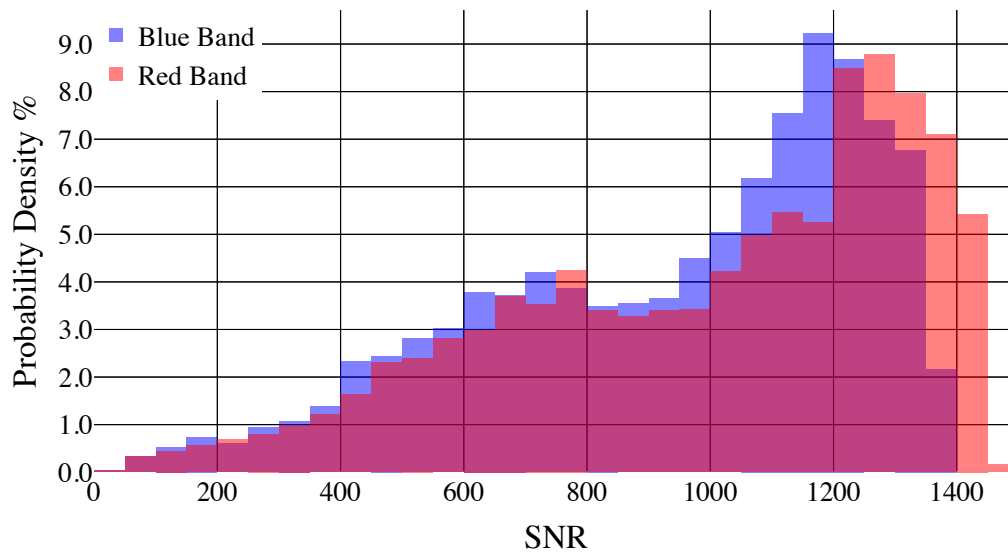
Figure 5-10: Architecture 304 Red Band Comparison
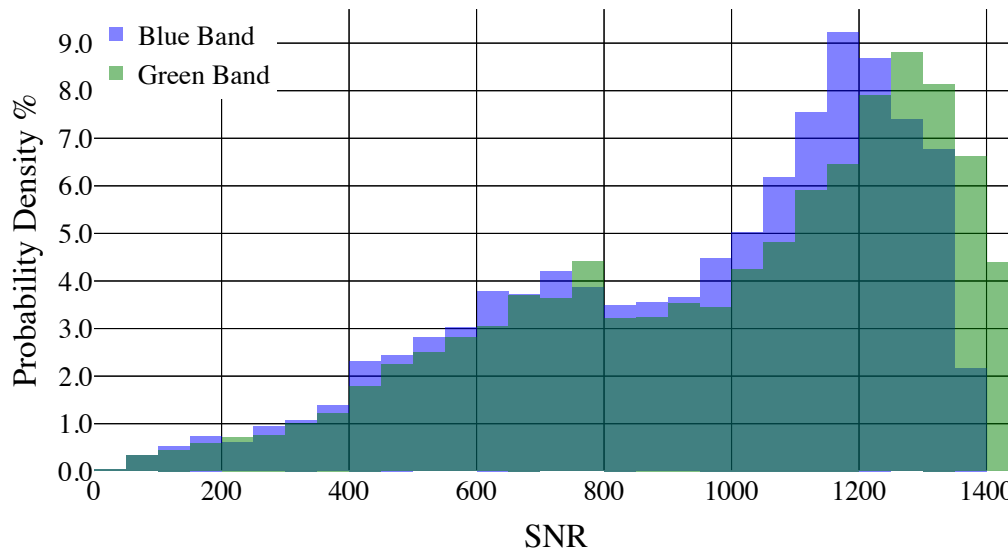
Figure 5-11: Architecture 304 Green Band Comparison
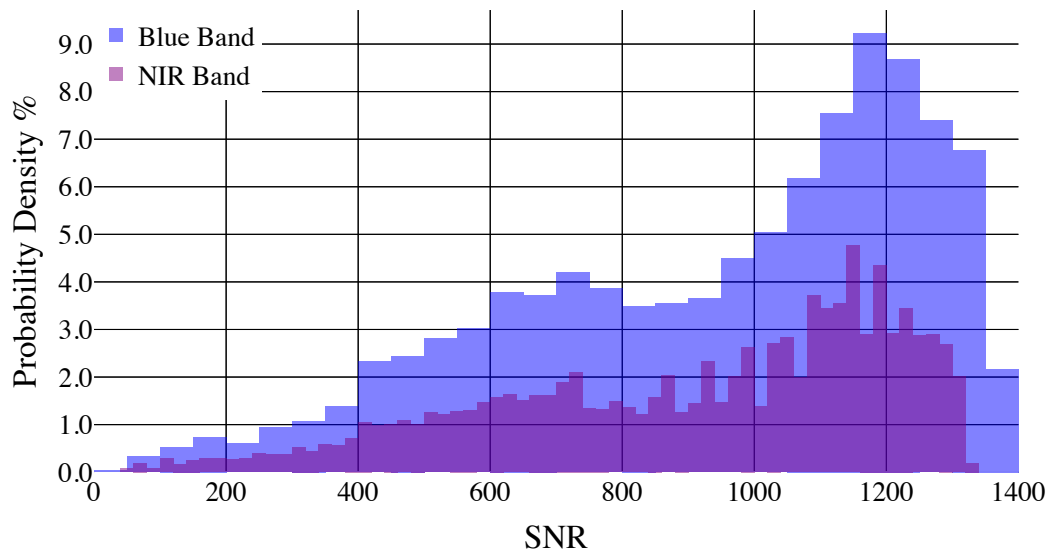
Figure 5-12: Architecture 304 NIR Band Comparison

Figure 5-13 shows the SNR distribution plot but comparing all four bands: blue, red, green, and NIR.
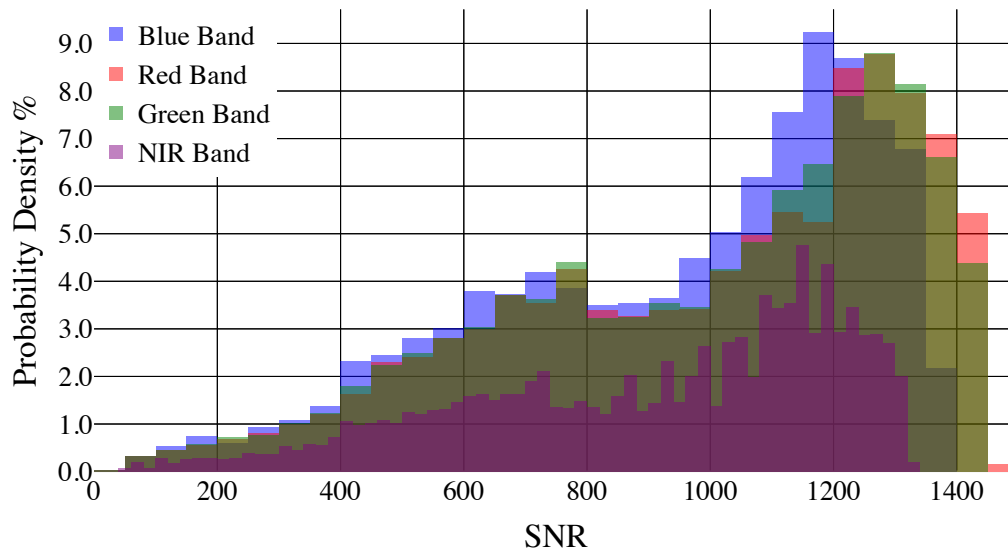
Figure 5-13: Architecture 304 All Band Comparison

As the above figures show, changing the band used for analysis can have both minor and large changes to the overall SNR distribution. This would also imply that the value obtained by each architecture would change as well. Table 5.9 below shows the amount of value gathered by each architecture, and how it compares to the blue band architecture 304.

Band	Architecture Value [GB]	Normalized Value to Blue Band	Normalized λ to Blue Band
Red	127.8048347	105.76%	133.82%
Green	126.9076275	105.02%	115.15%
NIR	116.4014198	96.32%	179.46%

Table 5.9: Architecture Value Comparison for 3 Alternative Instrument Operating Bands

Equation 3.14 shows that a simple change in λ, all else constant, does not impact the SNR calculation in a direct linear fashion. When examining Table 5.9, what is critical to focus in on is that different bands will generate different amounts of value, both positively and negatively. The pure amount of value will differ by a few percentage points, but it does create a change. This analysis should not be used to say a band is superior to another band, but rather show how the numerical results are impacted by an important instrument parameter. Future work will discuss the importance of evaluating multi-band instruments.

To finalize the Landsat tradespace analysis, it is important to view the architectures that had the highest architecture value across the entire tradespace. Figure 5-14 displays the top 20 architectures along with the actual Landsat architecture highlighted in blue.

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Avg Revisit Time [hr]	Inclination	Altitude	Satellites	Orbital Planes	Normalized Value
0	arch-309	226.174679	2496.575843	999	30.376179	90.0	800.0	4	1	223.55
1	arch-310	225.893933	2659.575843	999	31.005207	90.0	800.0	4	2	223.27
2	arch-311	225.821556	2659.575843	999	30.973269	90.0	800.0	4	2	223.20
3	arch-270	221.894289	2659.575843	999	32.600477	90.0	750.0	4	2	219.32
4	arch-271	221.881209	2659.575843	999	32.486250	90.0	750.0	4	2	219.31
5	arch-269	221.668485	2496.575843	999	32.493632	90.0	750.0	4	1	219.10
6	arch-231	219.439849	2659.575843	999	21.351701	90.0	710.0	4	2	216.89
7	arch-230	218.284511	2659.575843	999	31.052358	90.0	710.0	4	2	215.75
8	arch-229	217.075559	2496.575843	999	31.096206	90.0	710.0	4	1	214.56
9	arch-191	215.743412	2659.575843	999	27.009342	90.0	700.0	4	2	213.24
10	arch-189	215.737805	2496.575843	999	32.860734	90.0	700.0	4	1	213.24
11	arch-190	215.432049	2659.575843	999	32.901480	90.0	700.0	4	2	212.93
12	arch-303	209.131841	2671.575843	882	27.345276	60.0	800.0	4	2	206.71
13	arch-302	208.651106	2671.575843	882	26.822753	60.0	800.0	4	2	206.23
14	arch-301	204.459757	2671.575843	882	27.610000	60.0	800.0	4	1	202.09
15	arch-317	203.556272	2496.575843	986	27.338039	98.0	800.0	4	1	201.20
16	arch-318	202.969463	2659.575843	986	30.395524	98.0	800.0	4	2	200.62
17	arch-319	202.928500	2659.575843	986	30.518584	98.0	800.0	4	2	200.57
18	arch-308	202.576819	1913.681882	999	41.066076	90.0	800.0	3	1	200.23
19	arch-150	201.595130	2659.575843	999	34.973390	90.0	650.0	4	2	199.26
214	arch-232	101.173471	746.893961	986	124.008427	98.0	710.0	1	1	100.00

Figure 5-14: Top 20 Architectures as Ranked by Value Function

From the above table, some very important factors come into focus. The first is that almost all of the top 20 best architectures have 4 satellites, i.e. the maximum amount allocated during the tradespace search. This makes intuitive sense because more satellites can capture more data. However, it is worth noting that these architectures do not have the best ratio of architecture value to number of satellites. Given the scaling function used, there is not a linear relationship between the number of satellites and architecture value. Table 5.10 shows the top 20 architectures, but with their respective ratio of value per satellite. What immediately becomes apparent is the actual Landsat architecture has a value per satellite 78% greater than the best architecture overall, even though the best architecture generates 123.55% more value overall. This is all driven by the scaling function utilized when calculating value, and helps create these diminishing returns.

Architecture	Architecture Value [GB]	Satellites	Value per Satellite
arch-309	226.174679	4	56.543670
arch-310	225.893933	4	56.473483
arch-311	225.821556	4	56.455389
arch-270	221.894289	4	55.473572
arch-271	221.881209	4	55.470302
arch-269	221.668485	4	55.417121
arch-231	219.439849	4	54.859962
arch-230	218.284511	4	54.571128
arch-229	217.075559	4	54.268890
arch-191	215.743412	4	53.935853
arch-189	215.737805	4	53.934451
arch-190	215.432049	4	53.858012
arch-303	209.131841	4	52.282960
arch-302	208.651106	4	52.162776
arch-301	204.459757	4	51.114939
arch-317	203.556272	4	50.889068
arch-318	202.969463	4	50.742366
arch-319	202.928500	4	50.732125
arch-308	202.576819	3	67.525606
arch-150	201.595130	4	50.398782
arch-232	101.173471	1	101.173471

Table 5.10: Value per Satellite for top 20 Landsat-like architectures

Returning to Figure 5-14, another key insight is that polar inclined orbits are the most value generating architectures. In total 14 out of the top 20 architectures all have polar orbits. Given the complex components that build out the value function, it is difficult to make a direct claim about how inclination impacts architecture value. However, it is evident from this study that for Landsat-like architectures, polar orbits tend to provide the best value. It is important to note that this may not be the case for all global-looking constellations, and particularly for more specific ROIs. Thus, all that can be said is polar orbits work best when conducting a Landsat comparative trade study. An important point is that perfectly polar orbits at i=90 degrees may lead to improved POI revisit statistics but at the expense of giving up the sun-synchronous orbit. Thus while more POI imagery is obtained by non-SSO orbits the range of illumination conditions may be different making the data continuity part of the mission objective of Landsat more difficult. This may or may not be acceptable to mission planners. If SSO orbits are required, it is worth noting that the existing Landsat 8 architecture is nearly optimal in the single satellite case. Table 5.11 below shows how the two architectures with a larger altitude produce only 5% and 4% more value respectively. Figure 5-15 shows the architecture value-cost plot for all SSO architectures as well.

Architecture	Normalized Value	Cost [$M]	POIs Observed	Revisit Time [hr]	Inclination	Altitude
arch-312	105.53	746.89	986	106.898845	98	800
arch-272	104.034211	746.89	986	121.648983	98	750
arch-232	100.00	746.89	986	124	98	710

Table 5.11: SSO Orbit, Single Satellite Landsat Architecture Comparison
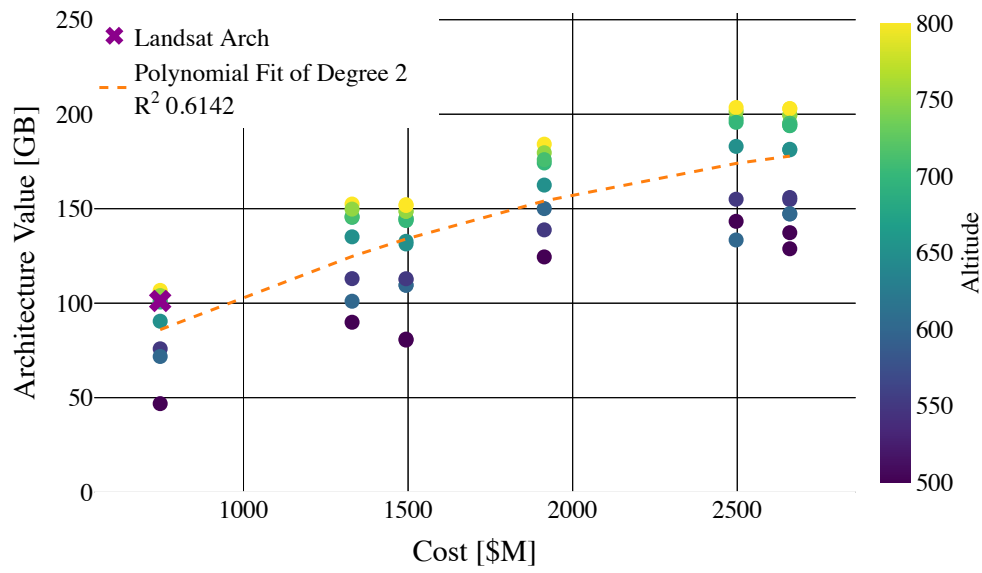
Figure 5-15: SSO Architecture Value Analysis

Also, higher altitude orbits tend to produce more value in this framework. This is most likely due to the increased number of ground pixels, and thus greater data collected. Again, this is only within the scope of this Landsat trade study, so no generalized comments can be made. In summary, the value function as formulated in this thesis creates a metric that is useful when making relative architecture comparisons. Important and useful trends arise due to the value function that are helpful for mission designers and planners. When comparing the value metric to lifecycle cost, the metric also provides the ability to see which architectures provide the most value given a level of cost, which means value for money is maximized. This thesis will now conduct a similar trade study, but with the RapidEye constellation.

5.1.2 RapidEye

For the second component of Case Study A, the RapidEye EO constellation was examined. The RapidEye constellation is an important EO system in that it was the first fully commercial operational class EO system that used multiple spacecraft [78]. The primary objective of the RapidEye system is to provide EO products and services to a large network of users across the globe [79].

The following two tables summarize the high level attributes of the RapidEye constellation, which include spacecraft mass and orbital geometry, and the instrument specifications for the RapidEye Earth Imaging System instrument on board [80], [81], [82]. Again, the visible blue band was used for analysis.

Attribute	Value
Number of Spacecraft	5
Spacecraft Mass	150 kg
Altitude	630 km
Inclination	SSO - 97.9°

Table 5.12: RapidEye Constellation High Level Attributes

Instrument Specification	Value
Mass [kg]	43
d [m]	0.0000065
f [m]	0.633
Along Track FOV	0.0006°
Cross Track FOV	7°
λ [m]	4.75e-7
D_{ap} [m]	0.145
τ_{op}	0.85
Q_E	0.45
N_r	1688
Bits per Pixel	12

Table 5.13: RapidEye Instrument Specifications

From a trade perspective, this thesis utilizes the same 5 design variables as for the Landsat case study. Table 5.14 breaks down the design variables and the values that were traded upon for each. Table 5.15 shows the design parameters that were held constant for the analysis.

Given the design variables, 448 unique architectures were generated using the TAT-C program. It is worth noting that more satellites were examined due to the size of of the RapidEye satellites. The RapidEye satellite is about 15 times smaller than the Landsat 8 satellite, which makes it an excellent large-scale, multi-spacecraft trade study candidate. Note that the upcoming Results and Discussion sections will follow a similar pattern as those from the Landsat case study.

Design Variable	Traded Values
Number of Satellites	[2, 3, 4, 5, 6, 7, 8]
Number of Orbital Planes	[1, 2]
Orbital Altitude	[500, 550, 600, 630, 650, 700, 750, 800]
Inclination	[30°, 60°, 90°, SSO]

Table 5.14: RapidEye Trade Analysis Design Variables

Design Parameter	Constant Values
Constellation Type	Homogeneous Walker
Region of Interest	Global
Simulation Duration	40 days
Maximum Grid Size	1000 Grid Points

Table 5.15: RapidEye Trade Analysis Constant Design Parameters

5.1.2.1 Results

The grid point generation for this trade study is the same as the Landsat case study. See Figure 5-1 for the global grid map. Similar to the Landsat results, Figure 5-16 depicts the architecture value to cost relationship. Note, that this plot is showing the relationship between architecture value and cost when θ from the access event scaling function is set to 0.1. Also, Figures 5-17, 5-18, 5-19 display the same relationship but the color scheme is based on the inclination of the constellation orbit, the number of POIs observed per architecture, and the average revisit time for each architecture respectively.
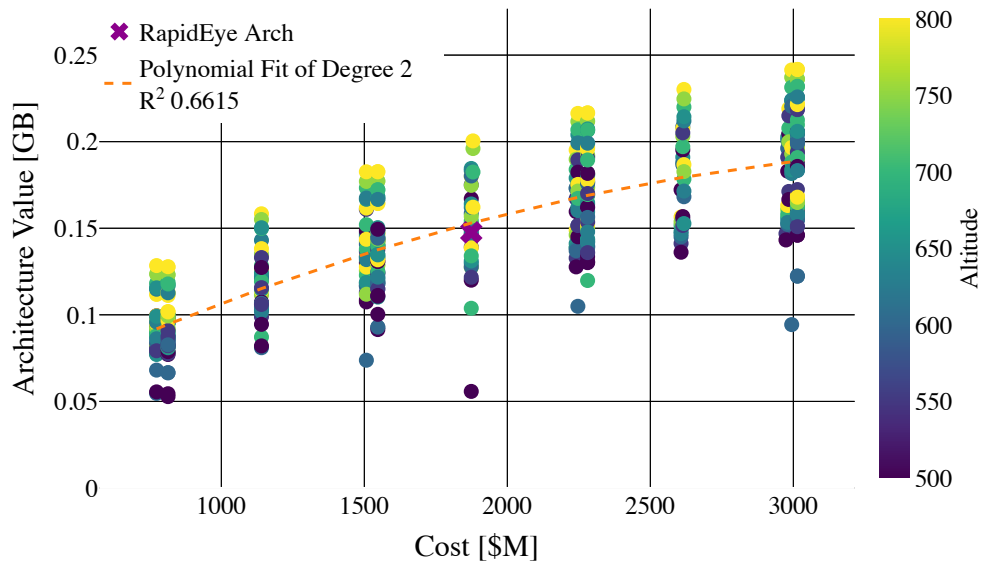
Figure 5-16: RapidEye Architecture Value Analysis
Color Scheme: Altitude

120
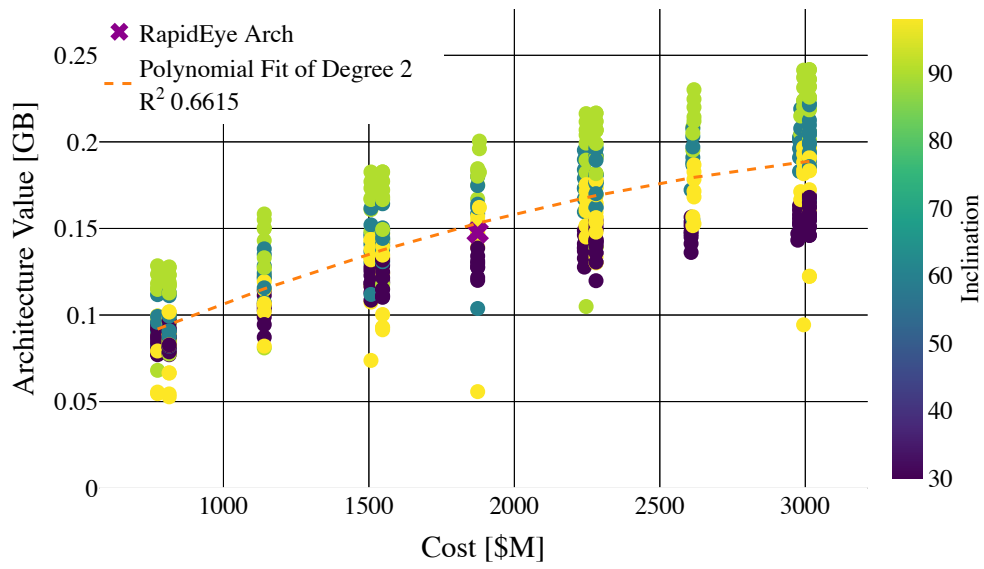
Figure 5-17: RapidEye Architecture Value Analysis
Color Scheme: Inclination
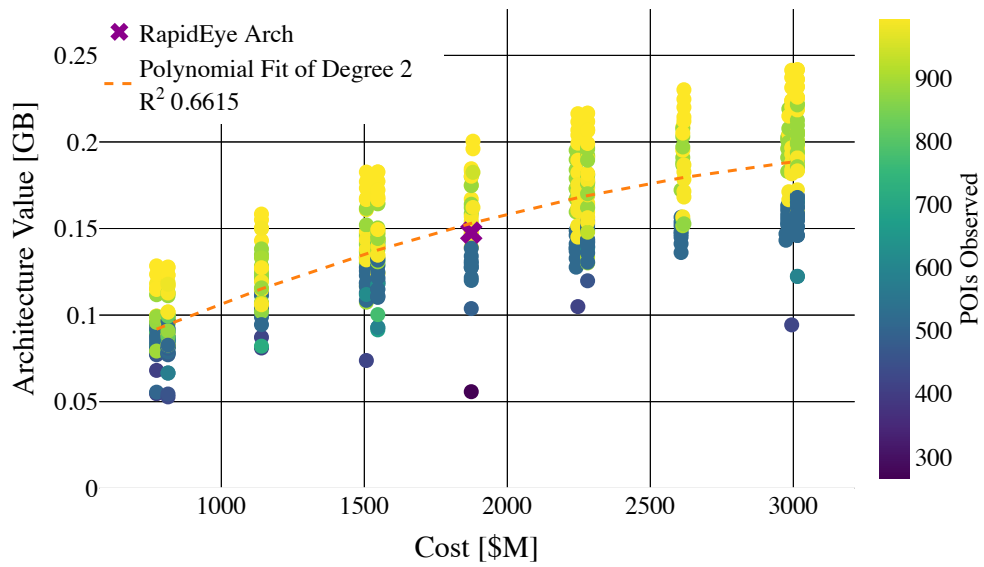
Figure 5-18: RapidEye Architecture Value Analysis
Color Scheme: POIs Observed
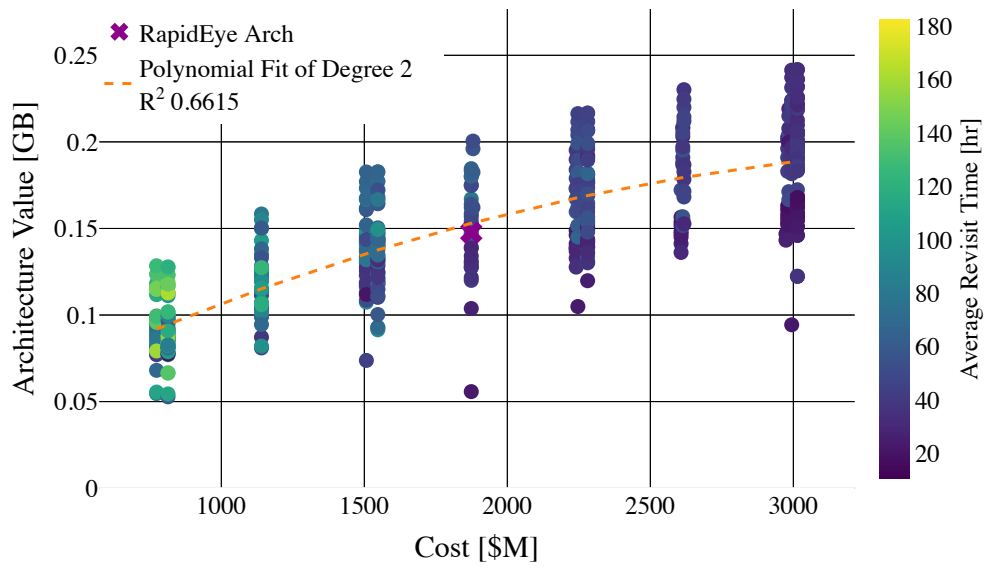
Figure 5-19: RapidEye Architecture Value Analysis
Color Scheme: Revisit Time

The following figures depict the relationship between architecture value and the average revisit time. Again, as in Figure 5-16, θ from the access event scaling function is set to 0.1.
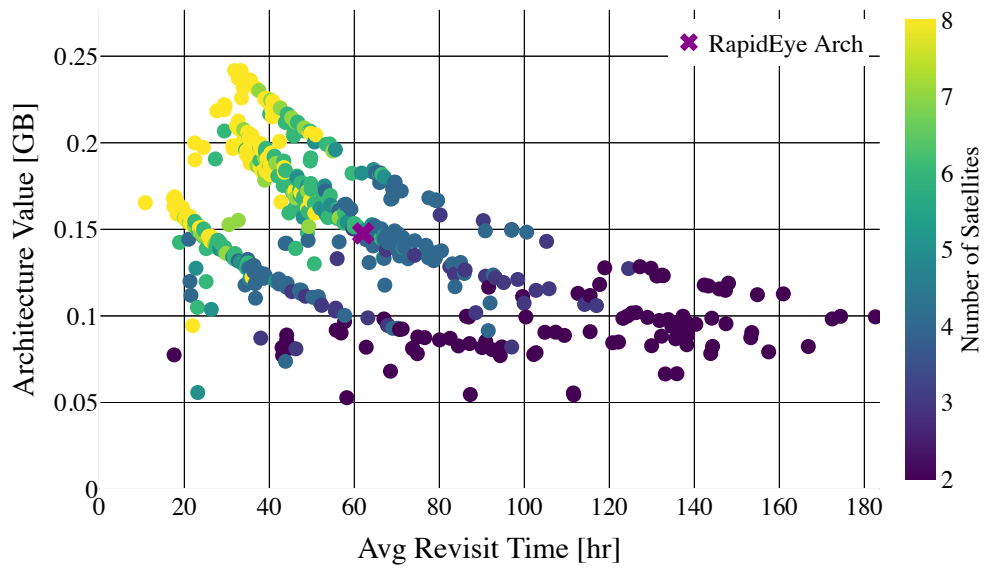
Figure 5-20: RapidEye Value - Revisit Time Analysis
Color Scheme: # of Satellites
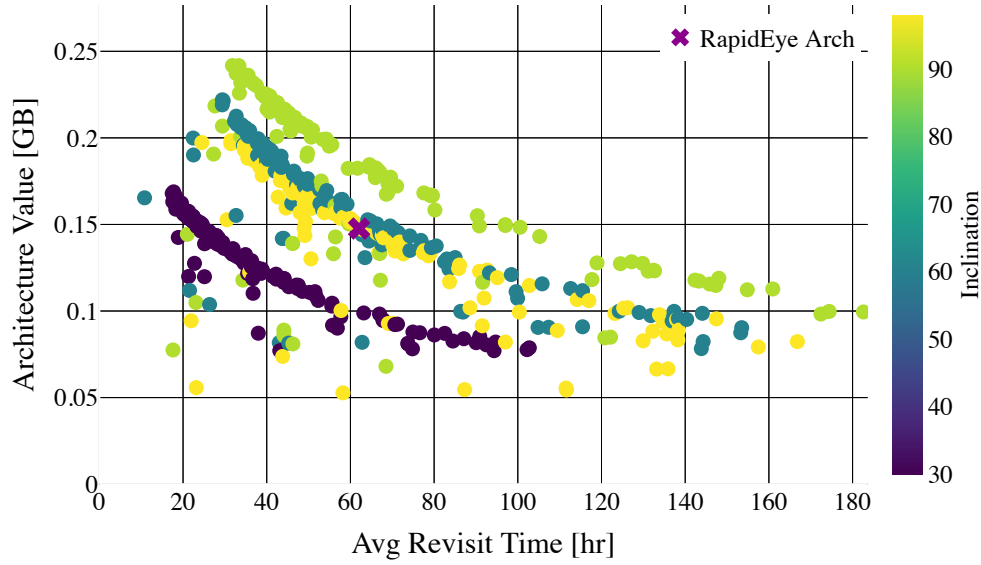
Figure 5-21: RapidEye Value - Revisit Time Analysis
Color Scheme: Inclination

5.1.2.2 Discussion

It is evident that the existing RapidEye constellation architecture is not optimal within the value framework presented in this thesis. Let's first examine the architectures that contain the same number of satellites as the existing RapidEye constellation.

This trade analysis generated 32 architectures that consist of 5 satellites. Of these 32 architectures, the actual RapidEye constellation (architecture 232) generated the 19th highest value. The top 20, five-satellite architectures, thus including the actual RapidEye constellation, and their resulting information can be seen in Figure 5-22. From this figure, the best architecture, as defined by the value function, generates 35.76% more value as compared to the actual RapidEye architecture. Thus for the same lifecycle cost, architecture 457 gets roughly 35% more value per dollar spent. This can also be seen in Table 5.16.

Architecture	Normalized Value	Cost [$M]	POIs Observed	Revisit Time [hr]	Inclination	Altitude
arch-457	135.76	1879.889448	993	50.615307	90	800
arch-232	100.00	1873.889448	986	62.069149	98	630

Table 5.16: Actual RapidEye Architecture Compared to Best Five Satellite Architecture

123

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Avg Revisit Time [hr]	Inclination	Altitude	Satellites	Orbital Planes	Normalized Value	Value per Satellite
0	arch-457	0.200441	1879.889448	993	50.615307	90.0	800.0	5	1	135.76	0.040088
1	arch-397	0.196030	1879.889448	993	55.540658	90.0	750.0	5	1	132.77	0.039206
2	arch-277	0.184529	1873.889448	993	64.544566	90.0	650.0	5	1	124.98	0.036906
3	arch-217	0.183246	1873.889448	993	64.257069	90.0	630.0	5	1	124.11	0.036649
4	arch-337	0.182467	1879.889448	939	61.693707	90.0	700.0	5	1	123.59	0.036493
5	arch-157	0.180195	1873.889448	993	67.042560	90.0	600.0	5	1	122.05	0.036039
6	arch-442	0.179959	1873.889448	882	46.556209	60.0	800.0	5	1	121.89	0.035992
7	arch-97	0.175085	1873.889448	993	53.015186	90.0	550.0	5	1	118.59	0.035017
8	arch-382	0.174769	1873.889448	882	48.891218	60.0	750.0	5	1	118.37	0.034954
9	arch-37	0.167029	1873.889448	993	79.038451	90.0	500.0	5	1	113.13	0.033406
10	arch-262	0.164017	1873.889448	882	54.388078	60.0	650.0	5	1	111.09	0.032803
11	arch-472	0.162151	1879.889448	986	52.056333	98.0	800.0	5	1	109.83	0.032430
12	arch-202	0.162088	1873.889448	882	52.713491	60.0	630.0	5	1	109.78	0.032418
13	arch-142	0.158600	1873.889448	882	53.350667	60.0	600.0	5	1	107.42	0.031720
14	arch-412	0.156898	1873.889448	986	47.947978	98.0	750.0	5	1	106.27	0.031380
15	arch-352	0.154233	1873.889448	986	58.304567	98.0	700.0	5	1	104.46	0.030847
16	arch-82	0.152693	1873.889448	882	64.123282	60.0	550.0	5	1	103.42	0.030539
17	arch-292	0.149290	1873.889448	986	62.399266	98.0	650.0	5	1	101.12	0.029858
18	arch-232	0.147642	1873.889448	986	62.069149	98.0	630.0	5	1	100.00	0.029528
19	arch-22	0.146370	1873.889448	882	66.924617	60.0	500.0	5	1	99.14	0.029274

Figure 5-22: Top 20 Five Satellite Architectures
Actual RapidEye Highlighted Blue

Similar to the analysis conducted in the Landsat case study, by examining Table 5.16 a couple of things become apparent. The biggest differences between the two architectures are the orbital inclination and altitude. Although the inclinations are somewhat similar, the altitudes are quite different. Let's see how this impacts the SNR distribution.
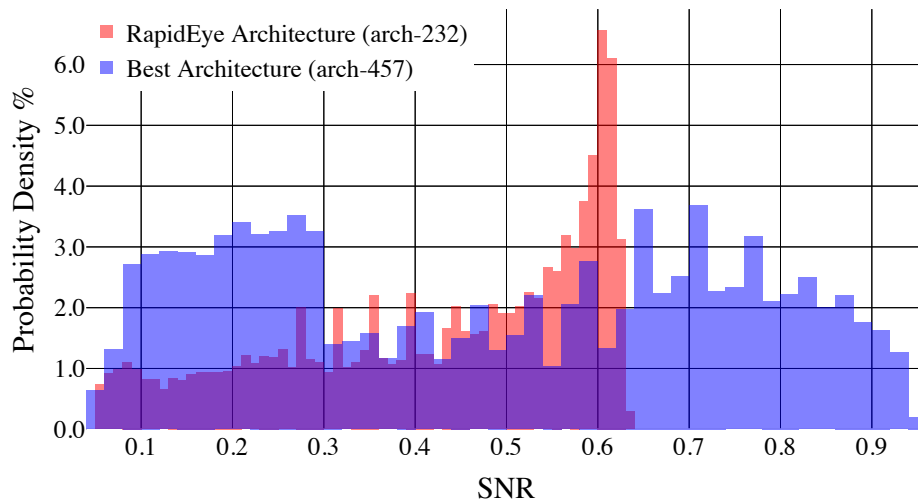
Figure 5-23: RapidEye Architecture SNR Comparison to Best Five Satellite Arch.

The SNR distribution for architecture 457 and the actual RapidEye constellation are quite different. It is important to point out that since both of these architectures

124

represent 5 satellites, these distribution plots show the SNR for all access events over all satellites. The first thing to notice is that the value of the SNR for these architectures is much lower as compared to the SNR from the Landsat case study. There is no inherent positive or negative to this, but it helps show the differences between the two instruments. Also, architecture 457 has almost a uniform SNR distribution, where as the actual RapidEye constellation has a more right-tail heavy distribution. This is most likely due to the SSO orbit and lower altitude as compared to architecture 457. However, the big difference between each SNR distribution is the fact that architecture 457 obtained access events that had a SNR greater than 0.65, whereas the actual RapidEye constellation did not. This increase in SNR is critical and directly impacts the value generated by each architecture.

Attribute	Architecture 457	Architecture 232	Percent Difference
Altitude [km]	800	630	26.984%
Average Cross Track GP Resolution [m]	8.219407092	6.473648748	26.967%
Average Along Track GP Resolution [m]	8.224508712	6.478516155	26.951%
Average GP Area $[m^2]$	67.60058524	41.939638	61.185%
Average GP Area $[km^2]$	0.00006760	0.00004194	61.185%
Footprint Area $[km^2]$	3.2916	2.0413	61.250%
Number of Ground Pixels	48692.4794	48673.0759	0.04%

Table 5.17: RapidEye and Best Five Spacecraft Q_{ij} Detailed Breakdown

Table 5.17 shows the Q_{ij} attributes for architecture 457 and the RapidEye architecture. Although there are large differences between average ground pixel area and footprint area, the number of ground pixels is relatively constant across each architecture. This implies that the largest factor behind the enhanced performance from a value framework of architecture 457 as compared to the actual RapidEye architecture is the superior SNR distribution. The amount of data collected is also higher due to more POI observations and an increased number of ground pixels gathered per access event, but these impact the spread in a smaller fashion.

Given that the RapidEye trade space evaluated architectures both larger and smaller, it is possible to identify architectures that provide the same value for less cost. The Table 5.19 breaks down 16 architectures that come in at a lower cost as compared to the actual RapidEye constellation, and generate at least 97.5% of the value. As the table shows, there are 16 architectures that produce similar, if not slightly more value as compared to the actual RapidEye constellation, for substantially less cost. Table 5.18 shows the architecture information from architectures 273 and 333 as compared to the actual RapidEye architecture. (Note that absolute cost data is not numerically accurate due to the small size of each satellite. Section 5.2.6 will break this down further.

Architecture	POIs Observed	Inclination	Altitude	Satellites	Normalized Value	Normalized Cost
arch-273	986	90.0	650.0	3	101.79	60.85
arch-333	993	90.0	700.0	3	101.52	60.85
arch-232	986	98.0	630.0	5	100.00	100

Table 5.18: RapidEye Low Cost Architecture Attributes

Architecture	Normalized Value	Normalized Cost	Satellites
arch-273	101.79	60.8538	3
arch-333	101.52	60.8538	3
arch-34	100.50	80.4269	4
arch-259	99.54	80.4269	4
arch-199	98.50	80.4269	4
arch-261	101.82	82.5615	4
arch-35	101.20	82.5615	4
arch-36	101.01	82.5615	4
arch-260	101.00	82.5615	4
arch-201	99.88	82.5615	4
arch-470	99.65	82.5615	4
arch-471	99.41	82.5615	4
arch-200	98.25	82.5615	4
arch-95	97.73	82.5615	4
arch-141	97.70	82.5615	4
arch-320	97.56	82.5615	4
arch-232	100.00	100.00	5

Table 5.19: Architectures that Generate at least 97.5% of Value at a Lower Cost

To conclude the RapidEye tradespace analysis, let's examine the architectures with the best architecture value across the entire tradespace. Figure 5-24 displays the top 20 architectures along with the actual RapidEye architecture highlighted in blue. Similar to the Landsat analysis, the architectures with the highest value come from those with the most satellites. However, when the constellations with the best ratio of value per satellite are examined, the top 20 architectures change drastically. Table 5.10 shows the top 20 architectures as ranked by the value per satellite. Unlike the Landsat case, the highest ranked architecture as scored by architecture value, architecture 464, also has a higher value per satellite as compared to the RapidEye constellation - 0.0302 vs. 0.0295. But the architecture with the best value per satellite, architecture 450, has a value per satellite almost double that of architecture 464 - 0.064 vs 0.03.

Similar to the Landsat analysis, polar inclined orbits are the most value generating architectures. It is noticeable that 18 out of the top 20 architectures all have polar orbits. Also, higher altitude orbits tend to produce more value in this framework. This is most likely due to the increased number of ground pixels, and thus greater data collected. Although these are similar results to the Landsat case study, it is

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Avg Revisit Time [hr]	Inclination	Altitude	Satellites	Orbital Planes	Normalized Value	Value per Satellite	Normalized Cost
0	arch-464	0.241735	3014.223117	993	31.803212	90.0	800.0	8	2	163.73	0.030217	160.85
1	arch-463	0.241689	3014.223117	993	33.300164	90.0	800.0	8	2	163.70	0.030211	160.85
2	arch-462	0.241414	2994.223117	993	33.316845	90.0	800.0	8	1	163.51	0.030177	159.79
3	arch-402	0.237191	2994.223117	993	32.689892	90.0	750.0	8	1	160.65	0.029649	159.79
4	arch-403	0.236184	3014.223117	993	35.264579	90.0	750.0	8	2	159.97	0.029523	160.85
5	arch-404	0.236131	3014.223117	993	35.558977	90.0	750.0	8	2	159.93	0.029516	160.85
6	arch-343	0.232008	3014.223117	993	33.782473	90.0	700.0	8	2	157.14	0.029001	160.85
7	arch-344	0.231756	3014.223117	993	36.765129	90.0	700.0	8	2	156.97	0.028970	160.85
8	arch-342	0.231366	2994.223117	993	36.844781	90.0	700.0	8	1	156.71	0.028921	159.79
9	arch-461	0.230156	2617.445227	993	37.535209	90.0	800.0	7	1	155.89	0.032879	139.68
10	arch-224	0.225843	3014.223117	993	33.459268	90.0	630.0	8	2	152.97	0.028230	160.85
11	arch-223	0.225643	3014.223117	993	39.169713	90.0	630.0	8	2	152.83	0.028205	160.85
12	arch-283	0.225365	3014.223117	993	38.882680	90.0	650.0	8	2	152.64	0.028171	160.85
13	arch-284	0.225283	3014.223117	993	39.711282	90.0	650.0	8	2	152.59	0.028160	160.85
14	arch-401	0.224694	2617.445227	993	40.284625	90.0	750.0	7	1	152.19	0.032099	139.68
15	arch-282	0.224015	2994.223117	993	40.795561	90.0	650.0	8	1	151.73	0.028002	159.79
16	arch-222	0.223732	2994.223117	993	39.560666	90.0	630.0	8	1	151.54	0.027966	159.79
17	arch-449	0.221985	3014.223117	882	29.397843	60.0	800.0	8	2	150.35	0.027748	160.85
18	arch-448	0.221392	3014.223117	882	29.599046	60.0	800.0	8	2	149.95	0.027674	160.85
19	arch-162	0.220389	2994.223117	993	42.510633	90.0	600.0	8	1	149.27	0.027549	159.79
235	arch-232	0.147642	1873.889448	986	62.069149	98.0	630.0	5	1	100.00	0.029528	100.00

Figure 5-24: Top 20 RapidEye-like Architectures as Ranked by Value

not sufficient evidence to make a generalized claim about architecture value and its relationship with orbital altitude and inclination. Upcoming section 5.2 will prove this by showing a counter example to this existing relationship.

In summary, using the value framework provided in this thesis, a new family of architectures were generated that provide more value as compared to the existing RapidEye constellation.

5.1.3 Differences Between Landsat & RapidEye

Now that both Landsat and RapidEye have been evaluated, an important factor is worth discussing. A direct comparison between Landsat and RapidEye in terms of overall architecture value was not the goal of this case study. However, a quick glance at the results shows that while RapidEye can visit significantly more POIs over time, Landsat provides about 600 times more value than RapidEye in an absolute sense. By definition, the value metric should not be compared in absolute terms, however, the scale of architecture value was much difference for each constellation class. This is driven primarily by the SNR. The average SNR across all Landsat architectures and access events was 884.79 db, whereas the average SNR for RapidEye was 0.48 db. From an architecture value comparison, the average Landsat architecture value was 142.51 GB, where as the average RapidEye architecture value was 0.15 GB. These are very different numbers, and are even more meaningful because more satellites were used in the RapidEye case. This shows that the implicit quality of the instrument used is quite important. Although Landsat is a large and not as responsive (in terms of revisit) constellation, the quality of the images are far superior in a SNR sense. The GSD of the images is higher, but the overall image quality is much better. This

127

Architecture	Normalized Value	Architecture Value	Satellites	Value per Satellite
arch-450	86.99	0.128436	2	0.064218
arch-451	86.55	0.127789	2	0.063894
arch-452	86.36	0.127506	2	0.063753
arch-390	83.68	0.123548	2	0.061774
arch-392	83.47	0.123238	2	0.061619
arch-391	83.33	0.123036	2	0.061518
arch-332	80.54	0.118909	2	0.059454
arch-330	80.04	0.118172	2	0.059086
arch-331	79.81	0.117828	2	0.058914
arch-272	79.43	0.117279	2	0.058639
arch-271	78.97	0.116597	2	0.058299
arch-270	78.28	0.115579	2	0.057790
arch-210	77.74	0.114783	2	0.057391
arch-436	76.54	0.113003	2	0.056502
arch-211	76.35	0.112725	2	0.056362
arch-212	76.03	0.112251	2	0.056126
arch-435	75.70	0.111768	2	0.055884
arch-437	75.30	0.111173	2	0.055587
arch-453	107.25	0.158343	3	0.052781
arch-393	105.02	0.155049	3	0.051683
arch-232	100.00	0.147642	5	0.029528

Table 5.20: RapidEye Top 20 Value per Satellite Architectures

makes intuitive sense since the instrument onboard the Landsat spacecraft is much larger and more sophisticated as compared to the much smaller instrument on board the RapidEye satellite (a similar claim can be made with the imager on the Planet satellite as well).

5.2   Case Study B: Make or Buy Decision

This case study will look at the all important make vs. buy decision that high level decision makers need to make when evaluating cost and benefits [83]. The make vs. buy decision is when a manager is faced with either building a product in house or purchasing the same product from a third party vendor. In this case study, the make or buy decision will focus on an EO constellation that provides information through the data products derived from the satellite system. To focus on a specific idea, as well as to show the importance of ROI optimization, this case study will analyze at the costs and benefits from the point of view of the governor of California.

5.2.1   The Need & Decision

The California governor desires high resolution imagery of the entire state in order to assist in agricultural management, disaster preparation, and general resource allocation. The governor has two options. The first is to simply purchase EO imagery from Planet Labs, in particular, from their Flock 2p constellation. The second option is to contract a third party to build and operate a new EO constellation specifically optimized for the needs of California. The reason for designing a new constellation would be the Planet Labs Flock 2p is not designed to provide optimal coverage of California, and the governor wants to ensure the data is the best it can possible be. By comparing the estimated lifecycle cost of the new California-customized constellation with the costs of purchasing imagery from a general purpose constellation, and also comparing the difference in value from each option, the governor will be able to make an educated decision to move forward.

5.2.2   Region of Interest

One of the most important attributes of this case study relates to the specific area of interest. This case study was modeled in order to find the family of EO constellations that provide the most value to a specific ROI. In the previous section, the Landsat and RapidEye constellation were trying to capture the most value at a global level. In this case study, the constellations are trying to capture the most value within the state of California. The specific ROI was referenced in section 3.4.1. Figure 5-25 and Table 3.5 were shown in that section, but are displayed below for reference. Also, given this specific ROI, the POIs generated by TAT-C can be expressed visually, as in Figure 5-26. These POIs are critically important because they represent the only parts of the Earth that generate value to the end user, i.e. in this case the governor of California. Using a specific ROI provides an excellent use case for the value framework presented in this thesis since it highlights value generated only within a particular region, not the entire Earth overall. This will become evident in the upcoming case study results.

Figure 5-25: Example ROI Visual for California

	Latitude	Longitude
Min	33	-124
Max	42	-115

Table 5.21: ROI Description: Lat/Lon intervals
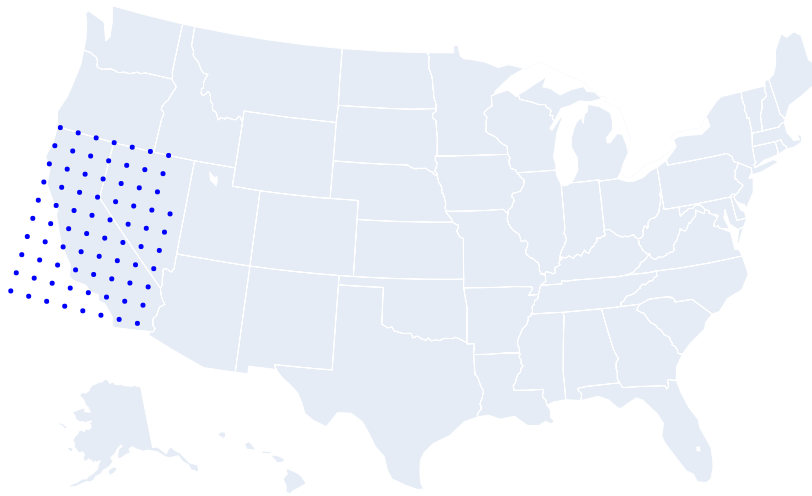
Figure 5-26: Grid Points Generated by TAT-C for California ROI

5.2.3 Planet Labs Flock 2p

On June 22, 2016, Planet Labs launched 12 Flock 2p satellites into a SSO orbit at an altitude of 500 km [84]. Planet currently operates a much larger fleet of satellites, but for the purpose of this case study, this thesis will only evaluate Flock 2p. This flock of satellites represent the 13th design iteration of the Planet Dove spacecraft, and images in the RGB and NIR bands. The red band is centered at 655 nm, the green band is centered at 545 nm, the blue band is centered at 475 nm, and the NIR band is centered at 835 nm [56]. The satellites were placed in an SSO orbit in order to take images at the same sun angle. The constellation will pass each spot on the ground at the same time every day which makes it easier to observe trends over time [84]. Let's examine the specifications of the spacecraft and the imaging instrument [1], [85], [34]. One of the most important attributes of the Planet Flock is the size of

the spacecraft. Each spacecraft represents a 3U cubesat. Compare the mass of each spacecraft, 5 kg, to that of the Landsat satellite, 2623 kg, (a ratio of about 1:500) and it is quite evident to see the difference in the satellites.

Attribute	Value
Number of Spacecraft	12
Spacecraft Mass	5 kg
Altitude	500 km
Inclination	SSO

Table 5.22: Planet Labs Flock 2p Constellation High Level Attributes

Instrument Specification	Value
Mass [kg]	1.6329
d [m]	0.0000075
f [m]	1.14
Along Track FOV	1.9773°
Cross Track FOV	2.9662°
λ [m]	4.75e-7
D_{ap} [m]	0.091
τ_{op}	0.6
Q_E	0.5
N_r	277
Bits per Pixel	12

Table 5.23: Planet Labs Flock 2p Instrument Specifications

5.2.4 Custom Constellation

A large trade space was designed in order to find an optimal constellation for the California ROI. In order to make a better comparative study, the Planet Flock 2p Dove was used as the satellite for analysis. Given that the spacecraft is a CubeSat built with many off the shelf parts, this thesis believes it is reasonable to trade on the same Planet spacecraft for analysis [86]. Thus, the satellites used for the trade will have the same attributes and instrument specifications as seen in tables 5.22 and 5.23. For the trade study, the number of satellites as well as the orbital altitude and inclination will be traded on. The complete set of design variables and their respective values can be seen in Table 5.24, and the simulation parameters can be seen in Table 5.25.

5.2.5 Planet Labs Image Cost

As stated prior, we assume that the two options for the California Governor are to purchase imagery from Planet or to design and build its own specific constellation.

Design Variable	Traded Values
Number of Satellites	[2,3,4,5,6,8,10,12,14]
Orbital Altitude	[400,450,500,550,600,650,700,750,800]
Inclination	[10°,20° 30°, 35°, 40°, 50°, 60°, 80°, 90°, SSO]

Table 5.24: Custom Constellation Trade Analysis Design Variables

Design Parameter	Constant Values
Constellation Type	Homogeneous Walker
Region of Interest	California
Simulation Duration	50 days
Maximum Grid Size	10000 Grid Points

Table 5.25: Custom Constellation Trade Analysis Constant Design Parameters

The lifecycle cost of the built constellation will be discussed in the following section. First, this thesis will discuss the cost of Planet Labs imagery.

Due to the private nature of Planet, it is difficult to get an exact price estimate. However, the Minnesota Department of Transportation conducted a study in 2018 and received a ball-park estimate of roughly $500,000 per year [87]. This ball-park cost was for their SkySat imagery which is their best resolution imagery. Although this imagery is better resolution and quality as compared to the Flock 2p, since the state of Minnesota is about half of the size of California, this thesis will utilize the ball-park price of $500,000 per year as the benchmark for purchasing Planet imagery.

5.2.6 CubeSat Costing

As discussed in Section 3.6.1, the cost estimates generated by TAT-C are not ideal for CubeSats. The model was built using data from larger spacecraft, and thus do not reflect the most up-to-date cost numbers and learning curve effects for CubeSats. Because of this, a new cost methodology is needed in order to create a better analysis for this case study.

In order to go about this, this thesis examined existing or proposed CubeSat missions that publish cost data. Table 5.26 shows CubeSat missions with their relevant costs, CubeSat size, and the number of satellites in the mission. All data was obtained from the Nanosats Database [88]. As the table shows, there is a wide distribution of costs as well as not a large quantity of data which makes it hard to create a robust model. It is useful to know summary statistics from this table, though. Table 5.27 shows the mean project cost, the mean cost per satellite, and the mean cost per form factor (1U size).

Since the Planet Labs Flock 2p Satellite is a 3U-form CubeSat, this model will look closer at the data that represents this type of satellite. Table 5.28 shows the mean project cost, the mean cost per satellite, and the mean cost per form factor (size)

when just examining 3U form factor CubeSats. These costs do not include launch costs.

Project	CubeSat Size	Number of Satellites	Project Cost
Clyde Orbcomm	3U	2	$5,900,000
Deorbitsail	3U	1	$3,200,000
HERMES-SP	3U	6	$3,950,000
Intuition-1	6U	1	$1,200,000
LLITED	1.5U	2	$2,520,000
MarCO	6U	2	$18,500,000
PEARLS	6U	8	$5,950,000
SEAM	3U	1	$3,000,000
Buccaneer	6U	1	$2,500,000
CAPSTONE	12U	1	$13,700,000
DAILI	6U	1	$2,850,000
DUPLEX	6U	1	$1,700,000
Hyperion 1	12U	1	$3,850,000
IOD Mission 6	3U	4	$1,880,000
Kleos Scouting Mission	6U	4	$2,830,000
Link-16 CubeSat (XVI)	12U	1	$10,000,000
OPS-SAT	3U	1	$2,660,000
TEMPEST-D	6U	1	$8,200,000
VPM	6U	1	$4,500,000

Table 5.26: CubeSat Projects with Total Cost Breakdown (FY2019)

Cost Category	Mean Value
Project Cost	$5,204,736.84
Cost per Satellite	$3,863,135.96
Cost per Form Factor	$667,613.3

Table 5.27: Statistics on CubeSat Database Costs

Cost Category	Mean Value
Project Cost	$3,742,000.0
Cost per Satellite	$2,493,666.67
Cost per Form Factor	$831,222.22

Table 5.28: Cost Statistics on 3U CubeSats

Utilizing the information from Table 5.28, this thesis will apply a very simple and naive method when generating architecture costs. First, the average cost per satellite will be multiplied by the total number of satellites in the constellation. After this, the launch cost generated by the secondary launch solver within TAT-C will be added in order to generate a proxy for the lifecycle cost. The equations below depict this formulation.

$$\text{Total Cost Part 1} = \text{Average Cost per Satellite} * \text{Number of Satellites} \tag{5.1}$$

$$\text{Total Cost Part 2} = \text{Secondary Launch Cost Estimate} \tag{5.2}$$

$$\text{Total Estimated Cost} = \text{Total Cost Part 1} + \text{Total Cost Part 2} \tag{5.3}$$

This is by no means an exhaustive exercise, but given the lack of detailed cost data, it is quite difficult to create a more complex and accurate model. That being said, organizations such as the Jet Propulsion Laboratory and The Aerospace Corporation are currently working on developing a CubeSat cost model. However, what is important to recognize is that given the CubeSat nature of the constellations generated in this case study, this new cost model will reflect a much more realistic cost estimate. TAT-C will produce cost estimates in the billions of dollars due to the large number of spacecraft involved, which is not valid and not helpful when analyzing the make vs. buy decision using 1U-12U class satellites. Using this new cost methodology, the most expensive constellation comes in at an estimated cost of just over $50 million, where as the TAT-C estimate for this same architectures is just shy of $2 billion. This is a large difference, and the new cost methodology for CubeSats reflects a much more reasonable cost as compared to the model within TAT-C.

5.2.7 Results & Analysis

Based on the way the trade space was designed, 810 constellation architectures were generated. Note that for the remainder of this section, a θ value of 0.1 was utilized for the access event scaling function. Different values of θ will not impact the relative difference between architectures, but the overall scale of the architecture value obtained per architecture. Architecture 659 represents the current Planet Flock 2p constellation. Table 5.29 below depicts the architecture value generated from this architecture. Again, this architecture value will be treated as a baseline.

Planet Flock 2p
45.129 GB

Table 5.29: Planet Flock 2p Architecture Value

Figure 5-27 depicts the distribution of architecture values generated in the case study. The red dashed line represents the value produced by the existing Planet Flock constellation.
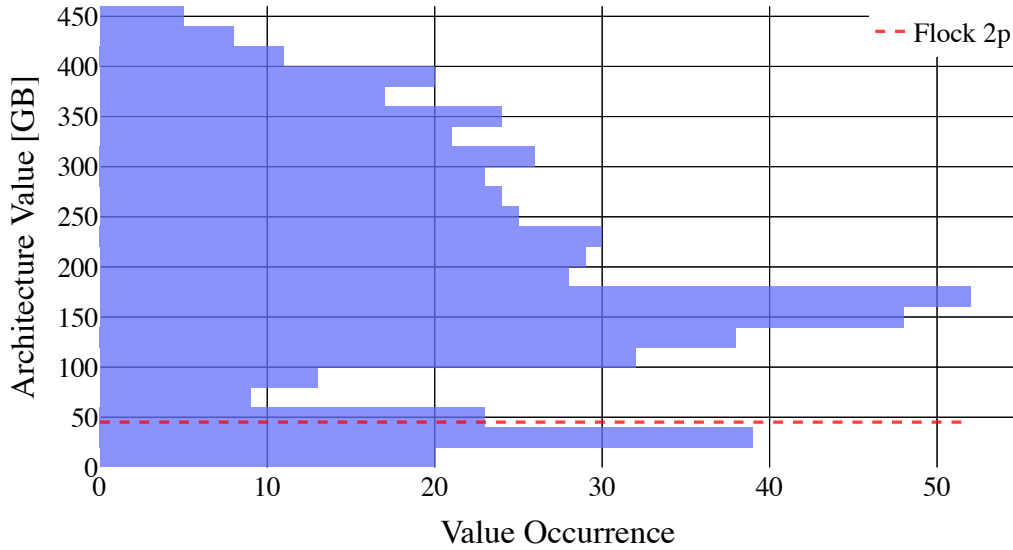
Figure 5-27: Case Study B Trade Space Architecture Value Distribution

From the figure, it is quite evident that the majority of the constellations generated in the trade study provide more value as compared to the Planet Flock 2p. To be more specific, 87.25% of the architectures generated produce more value. Table 5.30 breaks down the architecture value summary mean and median for the trade space.

Statistic	Value in GB
Mean	202.04
Median	188.17

Table 5.30: Case Study B Mean & Median Statistics

The following figure helps visualize the architecture value by plotting it as a function of the number of satellites. Also, the variation of colors for each data point is based on an attribute of the constellation. Figure 5-28 color is based on the orbital altitude. Figure 5-29 color is based on the orbital inclination. These figures are helpful because they show that inclinations of 40° to 50°, in combination of altitudes of about 600 km produce the most value when only one cluster of architectures, from a number of satellites perspective, is evaluated. Another way to think about these results is to realize that the highest value generating architecture, architecture 765, produces about 1000% more value as compared to the Planet Flock, when only considering California as ROI. This is a 10 times multiple compared to baseline and is quite substantial.

136
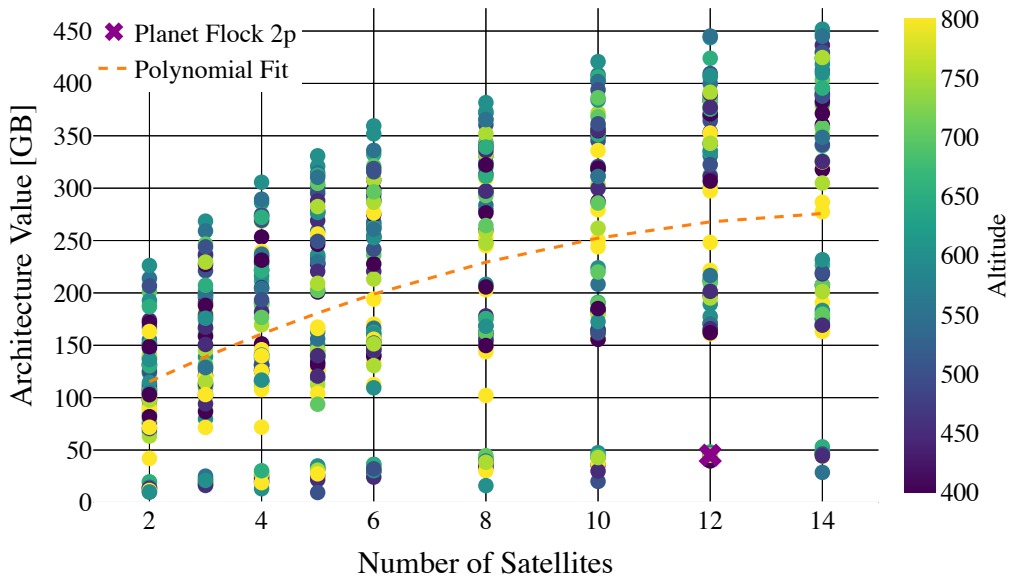
Figure 5-28: Architecture Value - # of Satellites Analysis
Color Scheme: Altitude
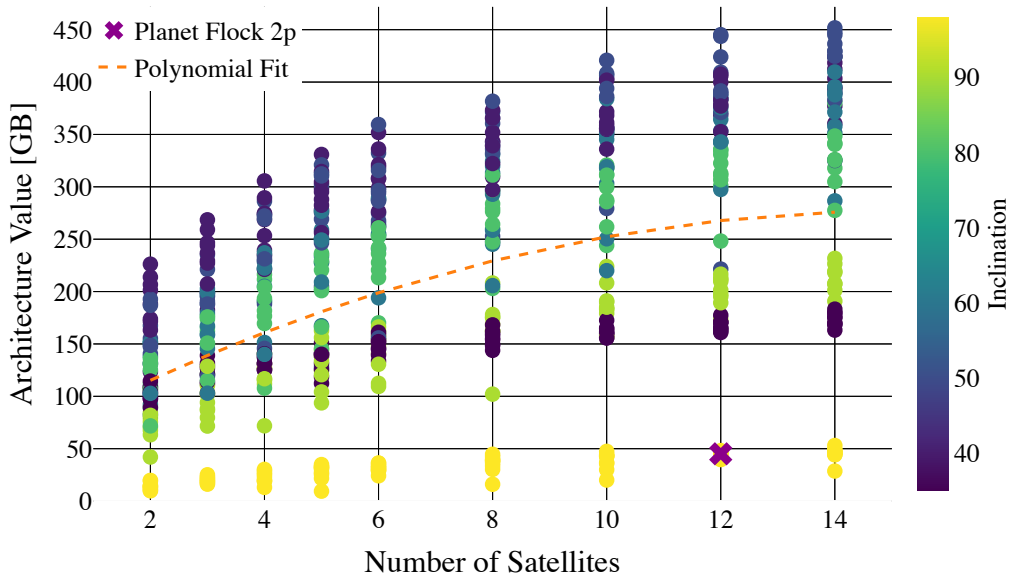
Figure 5-29: Architecture Value - # of Satellites Analysis
Color Scheme: Inclination

Let's investigate why architecture 765 produces more value than the baseline. Similar to Case Study A, understanding the SNR distribution is quite important. Figure 5-30 shows the SNR distribution of both the Planet Flock 2p constellation and the distribution from architecture 759.
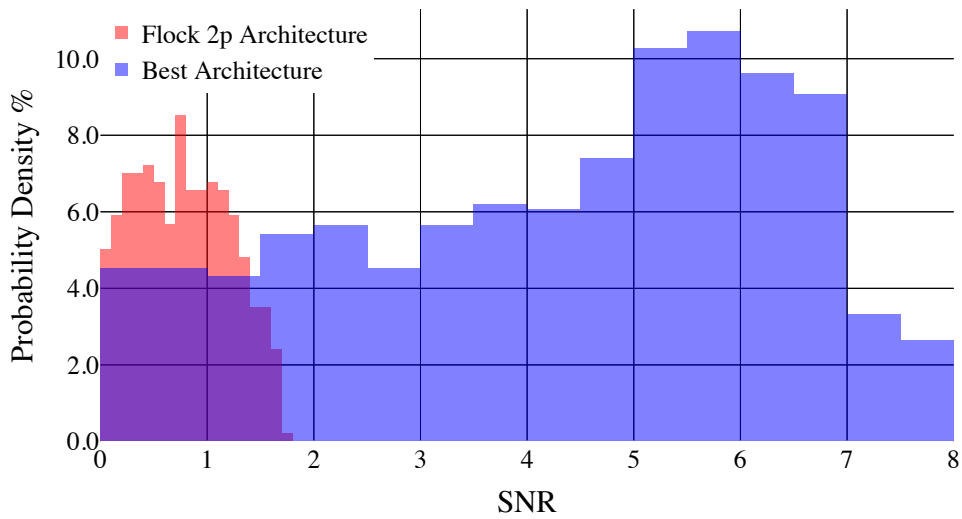
Figure 5-30: Case Study B SNR Distribution Comparison

The above figure shows a large difference in the SNR distribution. Architecture 765 produces much more high SNR access events which helps explain the significant difference in value.

5.2.8 Make vs Buy Analysis

In order to perform a robust make vs. buy analysis, a detailed financial picture is needed from both a cost and benefit perspective. In order to do this within the scope and data of this case study, some simplifying assumptions will need to be made, particularly on the cost.

As discussed prior, this case study will use a proxy cost of Planet data of $500,000 per year for a particular client like the State of California. This case study will examine a time horizon of 10 years. A horizon of 10 years was selected because it represents a midpoint of sorts for the orbital lifetime of a Planet Satellite. This number comes from Planet Labs and their analysis of the orbital lifetime of their CubeSats as a function of altitude [89].
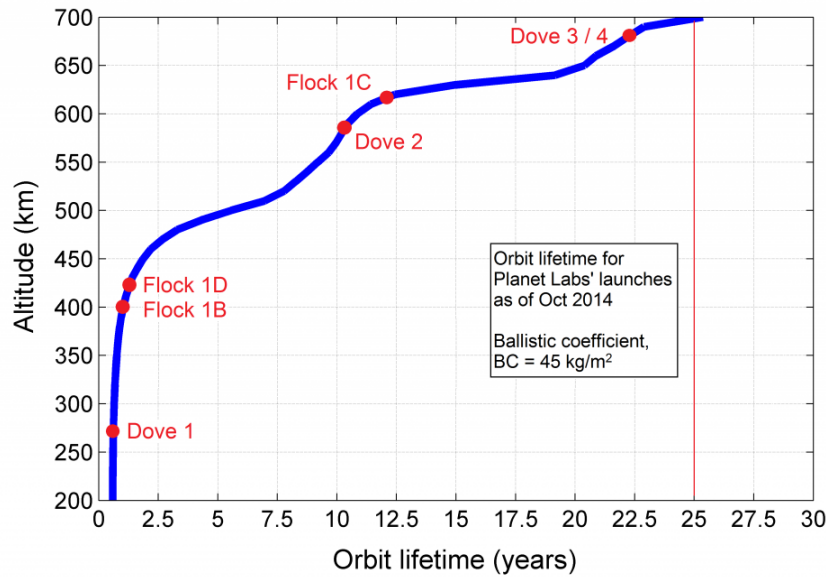
Figure 5-31: Planet Labs Estimated Orbital Lifetime [89]

Figure 5-31 depicts the relationship as calculated by Planet. Since Planet's current spacecraft don't have propulsion they can't raise their orbits to do station-keeping. Since the constellations generated vary in altitude, they will all have different orbital lifetimes. To simplify the case study, the 10 year orbital lifetime was utilized.

Now that the time horizon has been established, a more robust financial picture can be obtained. Using standard cash flow methods, a total non-discounted and discounted cost can be associated to the purchasing of Planet data and the building of a constellation. Purchasing of the Planet data is the more straightforward case. The only question is what to set as the cost growth rate of purchasing Planet data each year. This thesis will assume a cost growth rate per year of 2%. This is target inflation rate set by the Federal Reserve and thus is deemed a reasonable assumption [90]. A discount rate is also needed for discounted cash flow analysis. Rather than arbitrarily selecting a single discount rate, this thesis examined the costs for a wide range of discount rates. The discount rates used varied from 0% to 15% in 1% intervals. Using these rates, Table 5.31 depicts the total discounted cost of purchasing Planet data for the state of California for a 10 year period.

Discount Rate	Total Cost
0%	$5,474,860.50
1%	$5,228,757.19
2%	$5,000,000.00
3%	$4,787,113.99
4%	$4,588,767.24
5%	$4,403,755.47
6%	$4,230,988.45
7%	$4,069,478.03
8%	$3,918,327.50
9%	$3,776,722.19
10%	$3,643,921.06
11%	$3,519,249.36
12%	$3,402,091.96
13%	$3,291,887.48
14%	$3,188,123.06
15%	$3,090,329.64

Table 5.31: Total Discounted Cost of Purchasing Planet Data for 10 Years (FY 2020)

Now that the cost associated with purchasing Planet data is better understood, it is critical to understand how a comparable cost figure will be generated for each architecture. In order to do this, some simplifying assumptions must be made. The first set of assumptions relate to the cost per satellite and launch cost as explained in Section 5.2.6. A 2% cost growth rate is also used for the building of a constellation. The major question to evaluate is what to set the recurring costs to be. Note, recurring costs can include data processing, analysis, and maintaining data architectures. This is because the launch and satellite costs are all assumed to hit in the first year. Due to this situation, this thesis applied a percentage of the satellite costs to generate the yearly recurring costs. Similar to the discount rate, rather than arbitrarily selecting a single percentage of satellite costs, a range was evaluated and examined. Recurring cost percentages of 1% to 10% were examined in increments of 1%.

To provide a concrete example, let's first examine architecture 765. This architecture generated the most value, roughly 10 times as much value as the standard Planet Flock 2p when focusing on this particular ROI. The table below depicts the yearly cost analysis. Note how it is broken down into launch, satellite, and recurring costs. For this example, the recurring cost percentage was set to 5% and the discount rate was also set to 5%.

Year	0	1	2	3	4	5	6	7	8	9
Launch Cost	$4,868,106.00	$-	$-	$-	$-	$-	$-	$-	$-	$-
Satellite Cost	$34,911,333.38	$-	$-	$-	$-	$-	$-	$-	$-	$-
Recurring Cost	$-	$1,745,566.67	$1,780,478.00	$1,816,087.56	$1,852,409.31	$1,889,457.50	$1,927,246.65	$1,965,791.58	$2,005,107.41	$2,045,209.56
Yearly Cost	$39,779,439.38	$1,745,566.67	$1,780,478.00	$1,816,087.56	$1,852,409.31	$1,889,457.50	$1,927,246.65	$1,965,791.58	$2,005,107.41	$2,045,209.56
Total Non-discounted Cost	$56,806,793.64									
Yearly Discounted Cost	$39,779,439.38	$1,662,444.45	$1,614,946.03	$1,568,804.72	$1,523,981.73	$1,480,439.39	$1,438,141.12	$1,397,051.38	$1,357,135.62	$1,318,360.32
Total Discounted Cost	$53,140,744.14									

Table 5.32: Architecture 765 Cost Analysis Recurring Cost Percentage of 5% Discount Rate of 5%

With this cost breakdown, it is now possible to make a direct comparison. To do this, the relationship between the following ratios will be evaluated:

$$\text{Value Ratio} = \frac{\text{Value Generated from Custom Architecture}}{\text{Value Generated from Planet Labs Flock 2p}} \tag{5.4}$$

$$\text{Cost Ratio} = \frac{\text{Cost of Custom Architecture}}{\text{Cost of Imagery from Planet Flock 2p}} \tag{5.5}$$

Using these ratios, the governor of California may decide to build a constellation if the Value Ratio is greater than the Cost Ratio. A purely cost-driven decision would be to only build the custom constellation when the Cost Ratio is less than 1.0. There was not a single architecture generated that had a Cost Ratio less than 1.0. Architecture 8 had the lowest Cost Ratio, with a value of 1.61.

The Value Ratio for architecture 765 is 10.01. Since the discount rate and recurring cost percentage are variable, Figure 5-32 depicts the Cost Ratio table for all of the discount rates and recurring cost percentages. To help visualize this, Figure 5-32 colors Cost Ratios that are less than the Value Ratio in green, and Cost Ratios that are greater than the Value Ratio in red.

		Reccurring Cost %								
	$ 12.07	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
	1%	8.23	8.85	9.47	10.08	10.70	11.32	11.94	12.56	13.18	13.80
	2%	8.57	9.19	9.80	10.42	11.04	11.65	12.27	12.88	13.50	14.12
	3%	8.92	9.54	10.15	10.76	11.37	11.99	12.60	13.21	13.83	14.44
	4%	9.28	9.89	10.50	11.11	11.72	12.33	12.94	13.55	14.16	14.77
	5%	9.64	10.25	10.85	11.46	12.07	12.67	13.28	13.89	14.49	15.10
	6%	10.01	10.61	11.21	11.82	12.42	13.02	13.63	14.23	14.83	15.44
	7%	10.38	10.98	11.58	12.18	12.78	13.38	13.98	14.58	15.18	15.78
Discount Rate	8%	10.75	11.35	11.94	12.54	13.14	13.74	14.33	14.93	15.53	16.12
	9%	11.13	11.72	12.31	12.91	13.50	14.10	14.69	15.28	15.88	16.47
	10%	11.51	12.10	12.69	13.28	13.87	14.46	15.05	15.64	16.23	16.82
	11%	11.89	12.48	13.07	13.65	14.24	14.83	15.41	16.00	16.59	17.18
	12%	12.28	12.86	13.44	14.03	14.61	15.20	15.78	16.36	16.95	17.53
	13%	12.66	13.25	13.83	14.41	14.99	15.57	16.15	16.73	17.31	17.89
	14%	13.05	13.63	14.21	14.79	15.36	15.94	16.52	17.09	17.67	18.25
	15%	13.45	14.02	14.59	15.17	15.74	16.31	16.89	17.46	18.04	18.61

Figure 5-32: Architecture 756 Cost Ratio Table

As Figure 5-32 shows, there are few instances when building architecture 756 is more optimal than purchasing Planet data directly. This is an important discovery, and shows the value framework developed in this thesis can be used to make a critical decision that many city, state, and federal governments are making.

To show another example, Figure 5-33 shows the Cost Ratio Table for architecture 44. Architecture 44 consists of 2 satellites placed in 600km altitude and 40° inclination orbit. Architecture 44 has the best value per satellite in the trade space, and a Value Ratio of 5.01. This is clearly evident in that the entire Cost Ratio Table is highlighted green meaning the constellation should always be built! Having fewer satellites would increase the value per satellite, but the inclination of the orbit is also very advantageous when observing the California ROI.

						Reccurring Cost %					
	$ 1.64	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
	1%	1.11	1.20	1.28	1.37	1.46	1.55	1.64	1.73	1.82	1.90
	2%	1.15	1.24	1.33	1.42	1.51	1.59	1.68	1.77	1.86	1.95
	3%	1.20	1.29	1.38	1.46	1.55	1.64	1.73	1.81	1.90	1.99
	4%	1.25	1.34	1.42	1.51	1.60	1.68	1.77	1.86	1.95	2.03
	5%	1.30	1.38	1.47	1.56	1.64	1.73	1.82	1.90	1.99	2.08
	6%	1.35	1.43	1.52	1.61	1.69	1.78	1.86	1.95	2.04	2.12
	7%	1.40	1.48	1.57	1.65	1.74	1.82	1.91	2.00	2.08	2.17
Discount Rate	8%	1.45	1.53	1.62	1.70	1.79	1.87	1.96	2.04	2.13	2.21
	9%	1.50	1.58	1.67	1.75	1.84	1.92	2.01	2.09	2.18	2.26
	10%	1.55	1.63	1.72	1.80	1.89	1.97	2.05	2.14	2.22	2.31
	11%	1.60	1.68	1.77	1.85	1.93	2.02	2.10	2.19	2.27	2.35
	12%	1.65	1.73	1.82	1.90	1.98	2.07	2.15	2.23	2.32	2.40
	13%	1.70	1.79	1.87	1.95	2.03	2.12	2.20	2.28	2.37	2.45
	14%	1.75	1.84	1.92	2.00	2.08	2.17	2.25	2.33	2.41	2.50
	15%	1.81	1.89	1.97	2.05	2.14	2.22	2.30	2.38	2.46	2.54

Figure 5-33: Architecture 44 Cost Ratio Table

5.2.9 Summary

Utilizing the value framework developed in this thesis supported a critical business decision to be made in a more rigorous framework than is typically considered. It is relatively straightforward to understand the costs associated with purchasing EO data or building and launching a EO satellite constellation, but understanding the benefits obtained from both alternatives is non-trivial. The value framework developed in this thesis enabled direct comparison of both the make and buy options. By comparing Value and Cost Ratios, decision makers can clearly understand that if the make option costs X times as much as the buy option, it may still be worthwhile since the make option provides Y times as much value as the buy option. Overall, Case Study B proved that the value framework can be used in a critical real world setting and help give the insights needed to make large scale decisions that has the potential to impact a large swath of individuals.

5.3 Case Study C: Synthetic Aperture Radar Small Satellite Mission

This case study will focus on a small SAR satellite constellation. Traditionally, SAR instruments were quite complex and large and thus were not prime candidates for small satellite constellations. However, due to advancements in SAR technology and part miniaturization, several small satellite SAR constellations have been proposed and are being developed [91]. Once such small SAR satellite that has been developed is the Mi croX-SAR satellite. This satellite is being developed in Japan by Synspective [91]. The satellite designers have developed a satellite concept that has a mass of 130 kg and has the potential of acquiring 1-3 meter resolution SAR imagery. This satellite plans on launching at the end of 2020. At the time of writing, Synspective has not publicly announced an orbital geometry for the mission.

Although the satellite has been designed, this thesis would like to examine the architecture tradespace for a global constellation of MicroX-SAR satellites. By simulating various constellation architectures and applying the Value Framework, future mission planners can determine the optimal architecture design to maximize value.

The following table summarize instrument specifications for the SAR instrument on board the MicroX-SAR satellite [91].

Instrument Specification	Value
Mass [kg]	130
Pulse Width [m]	31e-5
Data Rate [Gb/s]	2
Antenna Along Track Dimension [m]	4.9
Antenna Cross Track Dimension [m]	0.7
Antenna Aperture Efficiency	0.5
Peak Transmit Power [kW]	2
Chirp Bandwidth [MHz]	75
Minimum Pulse Repetition Frequency [kHz]	3
Maximum Pulse Repetition Frequency [kHz]	8
System Noise Figure [db]	4.3
Bits per Pixel	16

Table 5.33: MicroX-SAR Instrument Specifications

From a trade perspective, this thesis utilizes 3 design variables. Table 5.34 breaks down the design variables and the values that were traded upon for each. Table 5.35 shows the design parameters that were held constant for the analysis.

Given the design variables, 96 unique architectures were generated using the TAT-C program. The upcoming Results and Discussion sections will follow a similar pattern as those from Case Study A.

Design Variable	Traded Values
Number of Satellites	[1,2, 3, 4, 5, 6]
Orbital Altitude	[400, 500, 600, 700]
Inclination	[30°, 60°, 90°, SSO]

Table 5.34: MicroX-SAR Analysis Design Variables

Design Parameter	Constant Values
Constellation Type	Homogeneous Walker
Region of Interest	Global
Simulation Duration	40 days
Maximum Grid Size	3000 Grid Points

Table 5.35: MicroX-SAR Trade Analysis Constant Design Parameters

5.3.1 Results

Before jumping into the results, it is important to understand the grid generated by TAT-C for this case study. Unlike Case Study A, this case utilized 3000 grid points, as opposed to 1000. For the global region, $41,252$ grid points are possible, meaning 3000 represents a small portion. Keeping the grid points low greatly reduces the computation burden, and thus decreases the execution time. However, unlike Case Study A that obtained high fidelity results with 1000 grid points, the complex modeling involved with SAR instruments required more grid points to be generated in order to capture a larger number of events where an access event was actually recorded. Thus, figure 5-34 shows the grid points generated for this case study.
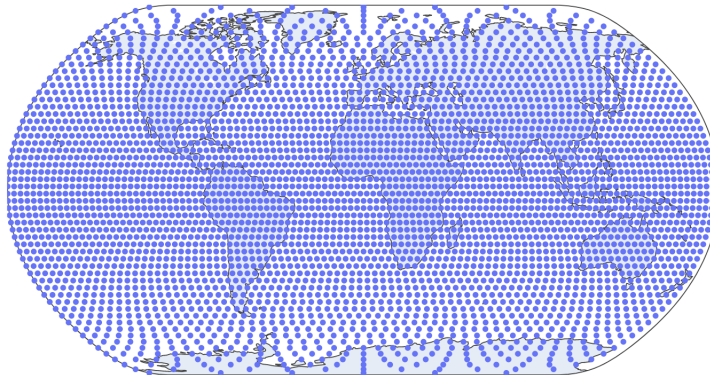
Figure 5-34: Grid Points Generated for Case Study C

Just as in Case Study A, this case study will depict the architecture value to cost relationship. Note that cost is directly related to the number of satellites deployed in an architecture, meaning the stratification in the upcoming plots represent 1, 2, 3, 4, 5, and 6 satellite architectures respectively. θ for the following plots is set to 0.1. Also, as Section 4.3.3.2 explains, the C utilized in the instrument quality calculations for this case study was set to 0. All of the observations recorded in this case study had a σ_N less than 0, which enabled this thesis to set C accordingly. Figure 5-35 shows architecture value plot and the color scheme is based on the altitude of the orbit. Figure 5-36 shows the plot but with the inclination as the color scheme. Figure 5-37 also depicts the value cost relationship but with the number of POIs observed by each architecture.
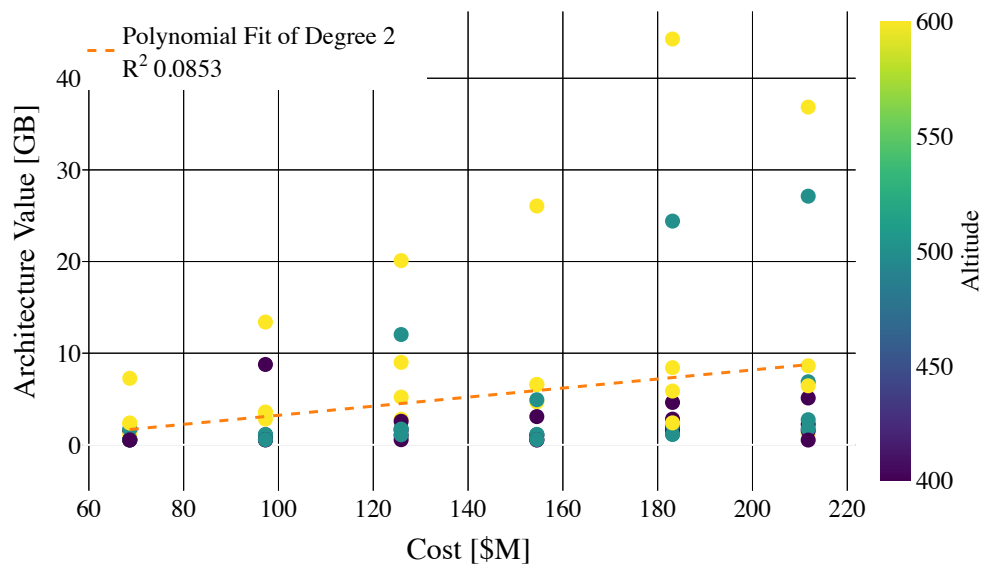
Figure 5-35: MicroX-SAR Architecture Value Analysis
Color Scheme: Altitude
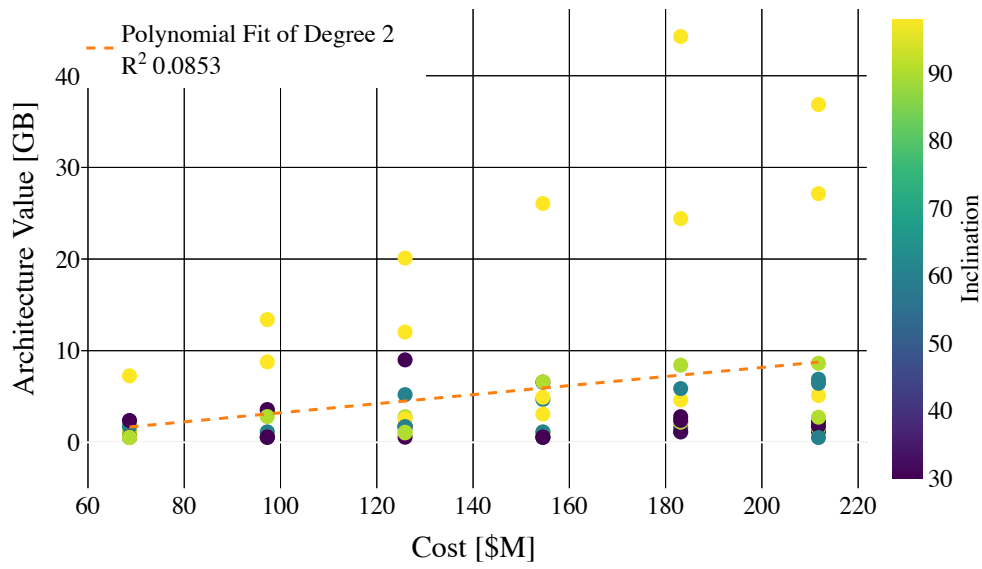
Figure 5-36: MicroX-SAR Architecture Value Analysis
Color Scheme: Inclination
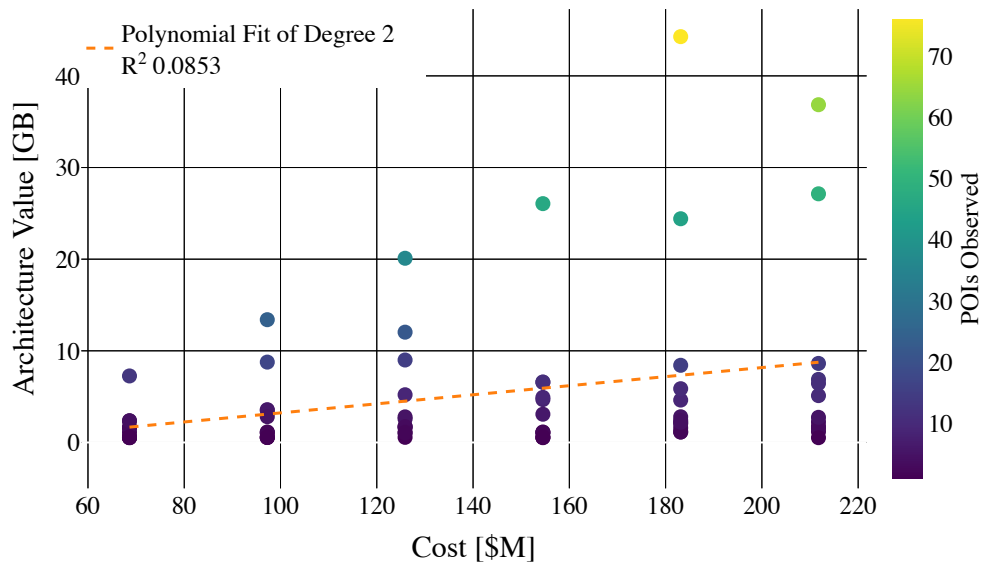
Figure 5-37: MicroX-SAR Architecture Value Analysis
Color Scheme: POIs Observed

As explained above, the number of grid points needed to be increased in order for architectures to record a higher number of access events. Also, as Figure 5-37 depicts, the maximum number of POIs observed by any architecture in this simulation is just over 70. With a simulation time of 40 days, this seems highly unlikely. This is most likely due to the complex nature of how access events are recorded for SAR instruments within the TAT-C module. In particular, the instrument specifications

currently being used can cause issues when calculating whether or not an observation has occurred. Future work should focus on highlighting the key bounds of TAT-C for this instrument class. Due to the issues discussed, many architectures did not observe the same POI more than one time. Thus, unlike in Case Study A, architecture value plots as a function of revisit time are not shown.

5.3.2 Discussion

MicroX-SAR is a proposed mission meaning this thesis will not benchmark the architectures with a specific nominal architecture. However, it is still possible to examine the current tradespace and develop some interesting conclusions. Let's first examine the architectures that generate the most value. Figure 5-38 displays the top 20 architectures in the tradespace.

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Inclination	Altitude	Satellites	Value per Satellite
0	arch-70	44.282194	183.131536	76	98.0	600.0	5	8.856439
1	arch-71	36.854070	211.757843	64	98.0	600.0	6	6.142345
2	arch-47	27.131790	211.757843	49	98.0	500.0	6	4.521965
3	arch-69	26.060947	154.505229	46	98.0	600.0	4	6.515237
4	arch-46	24.411100	183.131536	44	98.0	500.0	5	4.882220
5	arch-68	20.100106	125.878922	36	98.0	600.0	3	6.700035
6	arch-67	13.396960	97.252614	24	98.0	600.0	2	6.698480
7	arch-44	12.035508	125.878922	22	98.0	500.0	3	4.011836
8	arch-50	8.998555	125.878922	15	30.0	600.0	3	2.999518
9	arch-19	8.769454	97.252614	17	98.0	400.0	2	4.384727
10	arch-65	8.626647	211.757843	14	90.0	600.0	6	1.437774
11	arch-64	8.420445	183.131536	15	90.0	600.0	5	1.684089
12	arch-66	7.258968	68.626307	13	98.0	600.0	1	7.258968
13	arch-35	6.882892	211.757843	12	60.0	500.0	6	1.147149
14	arch-63	6.596718	154.505229	11	90.0	600.0	4	1.649179
15	arch-51	6.566546	154.505229	11	30.0	600.0	4	1.641637
16	arch-59	6.442247	211.757843	11	60.0	600.0	6	1.073708
17	arch-58	5.867228	183.131536	10	60.0	600.0	5	1.173446
18	arch-56	5.208048	125.878922	9	60.0	600.0	3	1.736016
19	arch-23	5.111046	211.757843	10	98.0	400.0	6	0.851841

Figure 5-38: Top 20 MicroX-SAR Architectures

From the above figure and the SAR Architecture Value plots, it becomes evident that value is accrued primarily by architectures with a sun-synchronous orbit, or an inclination around 98°. Similar to Case Study A, architectures with higher orbits also tend to produce more value. This makes sense given that higher orbits tend to have larger sensor footprints which imply higher quantities of data collected. However, in this particular case study, it is almost entirely driven by the number of POIs observed by an architecture.

For this simulation, the average number of POIs observed across all architectures is 9.236. This is quite low (the rationale for which is explained in the previous subsection, section 5.3.1. Due to this, this thesis also will examine the architecture value per POI observed. Figure 5-39 shows the top 20 architectures, but measured by value per POI observed.

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Inclination	Altitude	Satellites	Value per Satellite	Value per POI [GB]
0	arch-65	8.626647	211.757843	14	90.0	600.0	6	1.437774	0.616189
1	arch-50	8.998555	125.878922	15	30.0	600.0	3	2.999518	0.599904
2	arch-63	6.596718	154.505229	11	90.0	600.0	4	1.649179	0.599702
3	arch-52	2.397297	183.131536	4	30.0	600.0	5	0.479459	0.599324
4	arch-51	6.566546	154.505229	11	30.0	600.0	4	1.641637	0.596959
5	arch-49	3.578614	97.252614	6	30.0	600.0	2	1.789307	0.596436
6	arch-48	2.379966	68.626307	4	30.0	600.0	1	2.379966	0.594992
7	arch-53	1.181327	211.757843	2	30.0	600.0	6	0.196888	0.590664
8	arch-29	1.766728	211.757843	3	30.0	500.0	6	0.294455	0.588909
9	arch-26	1.766728	125.878922	3	30.0	500.0	3	0.588909	0.588909
10	arch-54	2.350283	68.626307	4	60.0	600.0	1	2.350283	0.587571
11	arch-58	5.867228	183.131536	10	60.0	600.0	5	1.173446	0.586723
12	arch-59	6.442247	211.757843	11	60.0	600.0	6	1.073708	0.585659
13	arch-57	4.683729	154.505229	8	60.0	600.0	4	1.170932	0.585466
14	arch-55	3.500143	97.252614	6	60.0	600.0	2	1.750071	0.583357
15	arch-70	44.282194	183.131536	76	98.0	600.0	5	8.856439	0.582660
16	arch-56	5.208048	125.878922	9	60.0	600.0	3	1.736016	0.578672
17	arch-71	36.854070	211.757843	64	98.0	600.0	6	6.142345	0.575845
18	arch-28	1.148224	183.131536	2	30.0	500.0	5	0.229645	0.574112
19	arch-24	0.574101	68.626307	1	30.0	500.0	1	0.574101	0.574101

Figure 5-39: Top 20 Architectures Measured by Architecture Value per POI Observed

The Figure above paints an interesting picture. Architecture 70 had the highest raw architecture value score, but it is not even in the top 10 architectures when measured by value per POI observed. Given that there are many architectures with very few POIs observed, future work should focus on running a longer duration simulation in order to see if similar results, from a value per POI observed perspective, are gathered. Although higher altitude orbits are still dominating, only one architecture, architecture 71, has an SSO in this top 20 list. While examining the architectures with the worst value per architecture, as seen in Figure 5-40, this thesis realized that the spread in architecture value per POI observed was quite low.

	Architecture	Architecture Value [GB]	Cost [$M]	POIs Observed	Inclination	Altitude	Satellites	Value per Satellite	Value per POI [GB]
52	arch-6	0.546562	68.626307	1	60.0	400.0	1	0.546562	0.546562
53	arch-42	0.544842	68.626307	1	98.0	500.0	1	0.544842	0.544842
54	arch-37	0.540569	97.252614	1	90.0	500.0	2	0.270285	0.540569
55	arch-36	0.540569	68.626307	1	90.0	500.0	1	0.540569	0.540569
56	arch-10	1.591457	183.131536	3	60.0	400.0	5	0.318291	0.530486
57	arch-8	1.591375	125.878922	3	60.0	400.0	3	0.530458	0.530458
58	arch-15	1.046375	154.505229	2	90.0	400.0	4	0.261594	0.523188
59	arch-13	1.046375	97.252614	2	90.0	400.0	2	0.523188	0.523188
60	arch-9	0.522408	154.505229	1	60.0	400.0	4	0.130602	0.522408
61	arch-11	0.522398	211.757843	1	60.0	400.0	6	0.087066	0.522398
62	arch-17	1.557715	211.757843	3	90.0	400.0	6	0.259619	0.519238
63	arch-12	0.517647	68.626307	1	90.0	400.0	1	0.517647	0.517647
64	arch-19	8.769454	97.252614	17	98.0	400.0	2	4.384727	0.515850
65	arch-14	1.028986	125.878922	2	90.0	400.0	3	0.342995	0.514493
66	arch-18	1.542635	68.626307	3	98.0	400.0	1	1.542635	0.514212
67	arch-22	4.627188	183.131536	9	98.0	400.0	5	0.925438	0.514132
68	arch-21	3.082737	154.505229	6	98.0	400.0	4	0.770684	0.513789
69	arch-20	2.568942	125.878922	5	98.0	400.0	3	0.856314	0.513788
70	arch-23	5.111046	211.757843	10	98.0	400.0	6	0.851841	0.511105
71	arch-16	2.043550	183.131536	4	90.0	400.0	5	0.408710	0.510887

Figure 5-40: Bottom 20 Architectures Measured by Architecture Value per POI Observed

The total spread in architecture value per POI observed is just over 0.1 GB. This is a much tighter and reasonable range as compared to the spread in raw architecture value, which is 43.76 GB. As stated prior, the limited number of POIs observed across the simulation reduce the effect of trading pure architecture value. However, normalizing value by the number of POIs observed help show interesting trend. For example, it is interesting to examine two architectures that vary greatly in raw architecture value, but are relatively similar in value per POI observed: architectures 23 and 70. Table 5.36 below shows these two architectures and their raw architecture value and value per POI observed.

Architecture	Architecture Value [GB]	POIs Observed	Architecture Value per POI Observed [GB]
Architecture 70	44.282194	76	0.582660
Architecture 23	5.111046	10	0.511105

Table 5.36: SAR Architecture Comparison

The difference in raw architecture value can be explained almost entirely in the difference between the number of POIs observed, but the value per POI observed is quite similar. In fact, architecture 70 only has an architecture value per POI observed 14% greater than architecture 23. Thus, there must be an underlying factor in architecture 23 that is enabling each observation to have almost as much value as architecture 70. It is also worth noting that architecture 23 is in a SSO orbit, just like architecture 70, but is at a lower altitude of 400 km.
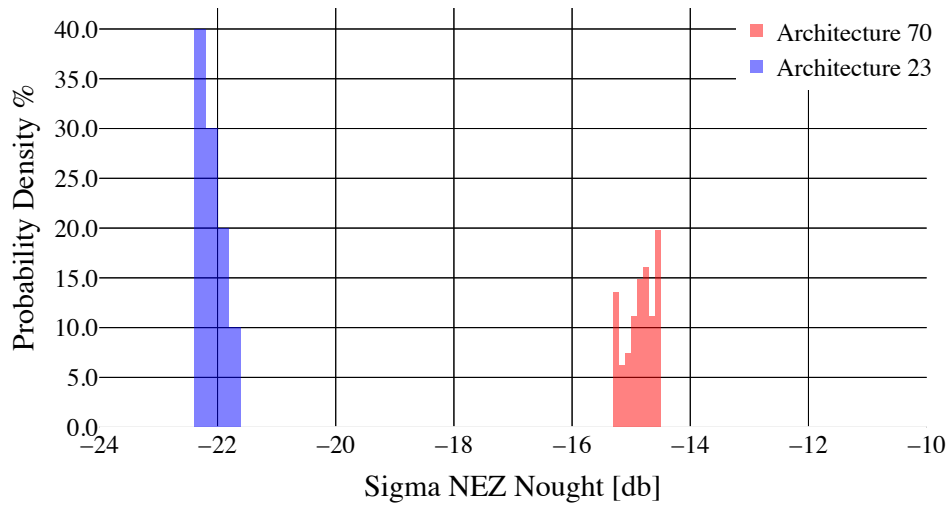
Figure 5-41: σ_N Distribution Plot

In order to answer this question, it is important to look at the distribution of σ_N produced by each architecture. Figure 5-41 shows a distribution plot for σ_N for each architecture.

The above figure shows how the distribution of σ_N for architecture 23 is more optimal as compared to the distribution for architecture 70. Remember, the more negative σ_N implies a higher quality of data acquired. To think about this another way, the average σ_N value for architecture 23 and 70 is -22.12 and -14.84 db respectively. The average σ_N for architecture 23 is 49% lower, or higher quality, than architecture 70. This difference shows the importance of trading off both the quantity and quality of recorded access events of a satellite system. Architecture 70 provides a large quantity of solid data, where as architecture 23 is providing much less data, but at a higher quality. The value metric provides users with a metric to help better understand this trade-off.

5.3.3   Summary

The previous case study focused on the proposed MicroX-SAR satellite system. SAR instruments are of increasing importance, and thus value-driven metrics for this class of instrument is vital. The Value Framework provided in this thesis was utilized to show the relative differences in a small family of architectures. Comparing SAR constellations is more complex as compared to a standard passive optical scanner, however, the Value Framework was able to distinguish high performing architectures from low performing architectures. This information is valuable to end users and can help the mission planners design the next generation MicroX-SAR constellation.

Chapter 6

Future Work

The previous two chapters derived the Value Function and applied it to two case studies to show the impact the Value Function can have on real-world Earth imaging applications. The Value Function was shown to help in the trade space analysis of EO constellations, and it can be used to get a better understanding of cost and benefits. However, the Value Function is not a perfect metric, and future work will be needed in order to overcome limitations and add features that will enable it to become even more useful. Future work should focus on overcoming current modelling limitations, combining Machine Learning with the Value Function, and incorporating calibration effects for increased functionality. Given that the limitations were already discussed in Section 4.4, Machine Learning and calibration effects will be discussed here.

6.1   Machine Learning Optimization

Machine Learning (ML) is a rapidly developing field, and is being applied to a wide variety of applications and domains. At the highest level, ML attempts to make decisions and predictions based on data [92]. ML algorithms find patterns in data, and then apply these patterns to new data in order to make predictions. What makes ML such an important part of the field of Artificial Intelligence is that it can be applied to a wide variety of tasks, across multiple domains. Some of the most common tasks include classification and regression. The power of ML comes from the fact that different domains, such a the medical field and self driving cars, can use similar ML algorithms in order to make predictions about the likelihood of a tumor or the likelihood of a pedestrian being present, respectively. One of the best ways to think about ML is through an input-output representation. Traditional programming uses human knowledge and expertise to devise a set of rules. Data is fed into these systems, and output is generated. There is typically no probability associated with the resulting answers. ML does almost the opposite. ML systems take in data, as well as the output, and produce the rules to go from input to output. This can be seen visually in Figure 6-1.
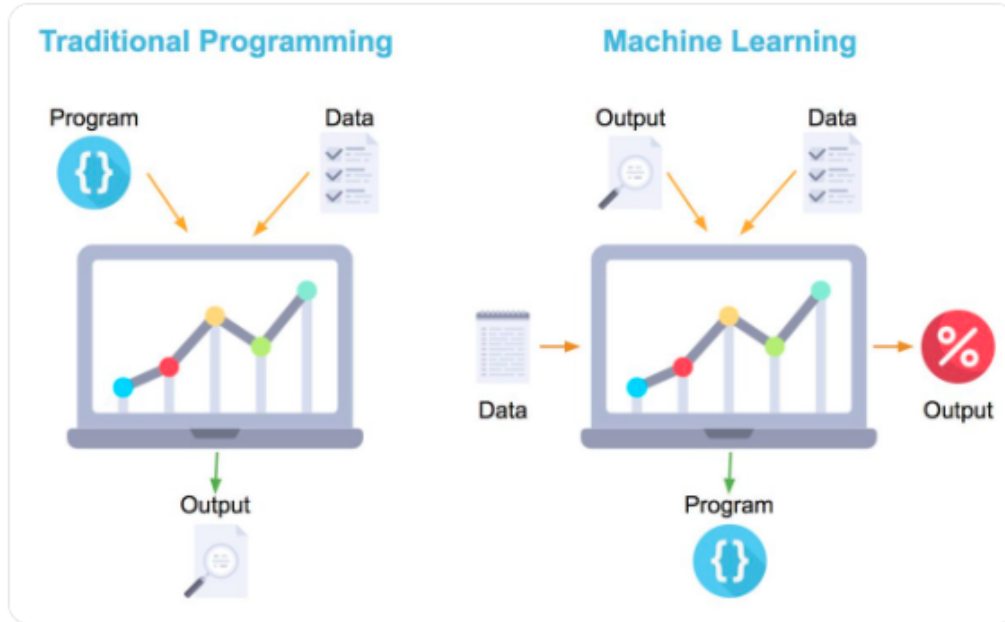
Figure 6-1: Machine Learning vs Classical Programming [93]

In reference to TAT-C, the trade space exploration process has the ability to use ML to help guide the optimal search process. This is helpful because it will help find which set of design variables are the most important, in reference to a specific objective. With regards to the Value Function, TAT-C has the ability to utilize the Value Function as a metric to help guide the optimized search. The user can set the primary objective to be the Value Function, and using the optimal search strategies, architectures will be generated that have higher architecture value. This capability currently exists, and can be utilized today. There is another way ML can be used to help make the Value Function even more powerful.

Within the scope of the Value Function itself, ML can be used in the following way. ML will be applied to the scaling function, or $f(j; \theta)$ within the Value Function. The author purposely parameterized the scaling function with the θ so it could be applied to ML systems in the future. Currently, the scaling function is set by the user. This thesis proposed a single way to describe the scaling function, shown below for reference.

$$f(j : \theta) = \frac{1}{(j+1)\theta} \tag{6.1}$$

This formulation came about by understanding the importance of diminishing returns, but using ML, the form of the scaling function can be *learned*, rather than assumed. In order to do this, a training dataset will be needed for each existing satellite constellation, and a large assumption will need to be made. The assumption that must be made is that the optimal, value-generating architecture for an existing

154

constellation is the one that is currently operating or has operated in the past. As an example, the five satellite, 630 km SSO orbit of RapidEye would be considered the most value generating architecture.

If this assumption is made, then the next step is to use this information to build a training database. Each data point of the database will consist of a generated constellation architecture. An initialized θ will be supplied, and the Value Function will be calculated for each architecture using that θ. Once this process is complete, standard back-propagation will be used in order to update θ and push it towards its optimal value. Figure 6-2 depicts this process.
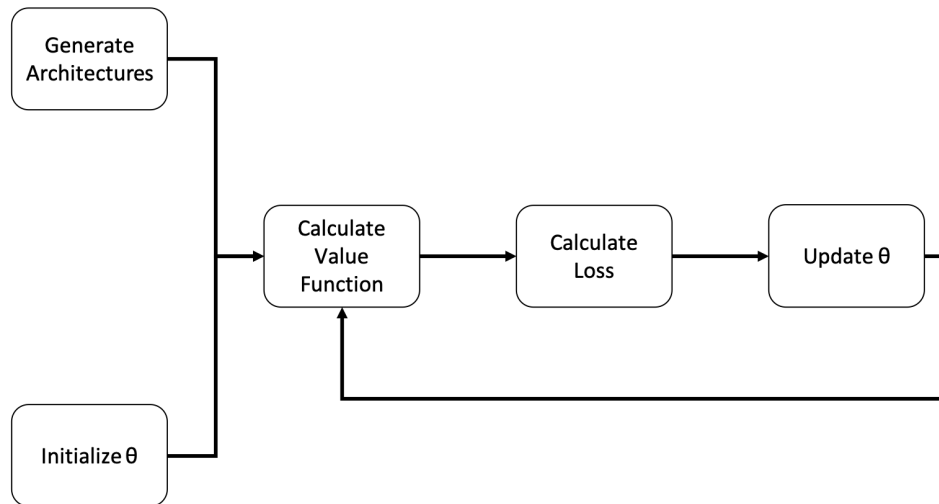

Figure 6-2: Example ML Workflow for Future Work

To do this, the ML system needs an output prior. Ordinal ranking can be used for this task since the assumption was made that the best architecture is the existing one. Each architecture that is not the existing constellation can be ranked poorly, let's say with a 0. The existing architecture will be ranked 1. Using ordinal regression, or ranking regression, techniques, the model should be able to learn what causes the architecture to have the specific ranking scheme. Standard logistic regression could also be used depending on the output style of the training data. This type of technique will require a large quantity of data, which may not make it feasible to use in the next few years. However, as CubeSat constellations continue to be deployed, the number of constellations to analyze will increase and make this training scheme more realistic.

6.2   Radiometric Calibration Effects

Another key piece of future work has to deal with incorporating calibration effects into the Value Function derivation [94]. Instrument calibration is important. Poor

calibration will greatly reduce the quality of the image captured. This refers mainly to the radiometric calibration of the RGB and NIR bands. What is the percent error between the colors recorded by the imager and the actual images on the ground? Within the scope of the Value Function, instrument calibration effects will directly impact μ_{ij}. What will need to be determined in future work is the magnitude of calibration, i.e. how much quality is lost per unit of calibration loss. Calibration loss can be thought of as a number between 0 and 1, where close to 0 implies minimal losses due to good calibration, and a numbers close to 1 implies large quality losses due to poor calibration. Determining how calibration loss effects value will be critical, and then the calibration loss can be directly applied to the μ_{ij} term.

One way to determine the effect of the calibration loss uses ML. Using a pre-trained or off-the-shelf building or road detection algorithm, for example, the performance of the model can be looked at with well calibrated and poorly calibrated images [95]. It will be important to control for calibration in the training of the model, but using the change in performance of correct object classification (false positives, false negatives) as a proxy for the reduction in calibration could be a useful and relatively straightforward way to determine the magnitude of the calibration effect. Future work should focus on applying this technique and exploring other ideas for how to quantity the magnitude of calibration losses.

6.3   Extending the Value Framework

An exciting area of future work comes from expanding the current value framework to include non-Earth observing payload systems. Specifically, it would be interesting to look at constellations that combine both EO payloads as well as telecommunication payloads. To do this, the value framework would have to be extended in order to quantify the value derived from the system. Luckily, valuing telecommunication payloads is a more straightforward and current methods exist, as discussed in Section 4.2. As constellations evolve and payloads become more sophisticated and complex, it will make logical sense to deploy space systems that combine both communication transmitters and EO systems. For example, SpaceX is currently deploying 1000's of small telecommunication satellites that will provide internet access to individuals. Given that these satellites will be covering the globe well due to the sheer number of spacecraft in orbit, it is easy to think of a world where the company decides to put an imager on the spacecraft in addition to the telecommunication payload. This would require additional engineering, but would enable SpaceX to become one of the largest EO providers while also achieving its telecommunication mission goals. These future hybrid-constellations are exciting and are ripe for future research, especially in value driven trade approaches.

Another key extension of the Value Framework would be to focus on modelling instruments that have multiple bands. Currently, both TAT-C and the Value Framework only examine a single band at a time. This makes intuitive sense because the instrument in question is not being traded on directly, however, many payloads exist

that observe frequencies in a wide range of operating wavelengths. Understanding the synergies of bands, and the weights applied to each band will be critical in order to extend out the Value Framework for multiple operating bands.

6.4   TAT-C Multi-Payload Capability

The last area of future work includes adding multiple payload analysis to the TAT-C software tool. TAT-C is currently available for use by NASA employees, and it will hopefully be open sourced in the near future. Although TAT-C is a powerful tool, one of the biggest limitations is the fact that only one instrument can be placed on a satellite. Many satellite systems have multiple payloads, and modeling these dynamics is key to understanding overall constellation value. Reference [69] discusses the importance of payload synergies and the non-trivial exercise that must be worked through in order to effectively model multiple payload value. Future work should focus on implementing this feature in TAT-C, and then extending the value framework in incorporate multiple payloads as well.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 7

Summary & Conclusions

This thesis examined the tradespace analysis of EO constellations through a value-focused lens. It first examined EO terms and paradigms that are important to the EO industry at large. This thesis provided a detailed breakdown of the TAT-C tool, and then dove into value-driven trade approaches. It provided a literature review on existing value approaches, and then described a new formulation that can be used. Three case studies were then discussed in order to show the scope and usefulness that can come from using the newly derived Value Function.

In order to evaluate the contributions of this thesis, it is important to revisit the goals that were discussed in Chapter 1. This thesis had three goals that can be summarized as followed:

1. Show the importance of using value driven tradespace analysis.

2. Provide a detailed derivation and breakdown of the Value Function.

3. Give future users of TAT-C a form of documentation that explains how to use the tool from end to end.

In Section 4.1, this thesis discussed why there is a need for value-driven trades. This is even more important in today's world where larger and more complex constellations are being designed and developed every year. Section 4.2 also showed how the valuation of satellite constellation is of research interest, and there are varying methodologies used to calculate space system value. This large research effort combined the discussion on value trade approaches and the limitations imposed when using cost based trade approaches, as seen in Section 3.6.1.1, show that the first goal of the thesis was achieved.

As seen above, the second goal was to provide a detailed derivation of the Value Function. The bulk of Chapter 4 was spent discussing the Value Function, and in particular, its derivation. This thesis broke down each individual element, and discussed not only its importance, but the rationale for where it came from. It is important to note that the Value Function described in this thesis is the culmination

of almost two years of work and many iterations. Because of this fact, this author strongly believes that each element that is involved with the Value Function has an important piece to play and that it directly relates to the generalized concept of value that was hypothesized. The detailed description and derivation of the Value Function in Chapter 4 shows how the second goal of this thesis was satisfied.

Finally, the entirety of Chapter 3 was spent discussing how TAT-C was designed and built, and the various inputs and outputs of the software system. Users should be able to use Chapter 3 of this thesis to better understand what is going on underneath the hood of TAT-C. Because of this, this thesis can stand in as a form of documentation that users can utilize when debugging or configuring TAT-C. Because of this, goal 3 was satisfied which means this thesis contributed to each of the three primary thesis goals.

It is also worth noting that by satisfying the thesis goals, the majority of the research questions were analyzed as well. Referring back to Section 1.2, the first four questions focused on understanding where value comes from within the EO domain, whether or not it is possible to build a value-driven framework, if the Value Function can be used to guide optimal search strategies, and are the limitations such that useful results can still be gathered. Chapter 4, as well as the case studies in Chapter 5, answered the first four research questions. This thesis made clear the need for value driven frameworks, discussed where value comes from within the EO domain, talked about how the Value Function can be used to guide optimal searches, especially with the power of TAT-C optimized search functionality, and discussed the limitations of the existing cost-driven model. The last research question focused on expanding the value framework to constellation design outside of EO. Unfortunately, this thesis was not able to directly answer this question. That being said, as Section 6.3 explains, the author hopes that future work will be conducted that can examine whether or not a similar model framework can be used for a larger set of constellation domains.

One of the most interesting questions in Earth Observation today is the trade off between large quantity but low quality, and low quantity but high quality data. This implies the difference between large constellations of small and cheap satellites, and constellations of a few larger and more sophisticated spacecraft. With the value-driven framework developed here, coupled with the use of TAT-C, this question can be examined in a rational way during early conceptual design of new EO missions.

In summary, this thesis showed the importance of value-driven trade approaches in the rapidly evolving EO constellation domain, and created a new value framework in order to evaluate a large set of possible distributed Earth Observation architectures. One of the most important takeaways of this thesis is that not only can mission planners and designers use the Value Function to help find more optimal constellations, but high-level decision makers also can use and *understand* this model in order to make decisions that are critical. Case Study B in Chapter 5 shows a perfect example of this. As EO data becomes more prolific, cities, states, and federal governments will become more interested in it and the scope and use-cases will grow dramatically. Understanding costs and benefits is absolutely critical, and before this thesis developed

the Value Framework, understanding benefits was non-trivial.

It is the author's hope that this framework continues to be improved and can be well understood so future users have the ability to apply it to their specific problems in the future. Using the Value Function model allows more optimal EO constellations to be discovered, which will help push the already fast-growing EO industry further into the future and help improve life on our planet.

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix A

Case Study A Landsat Plots with Varying θ
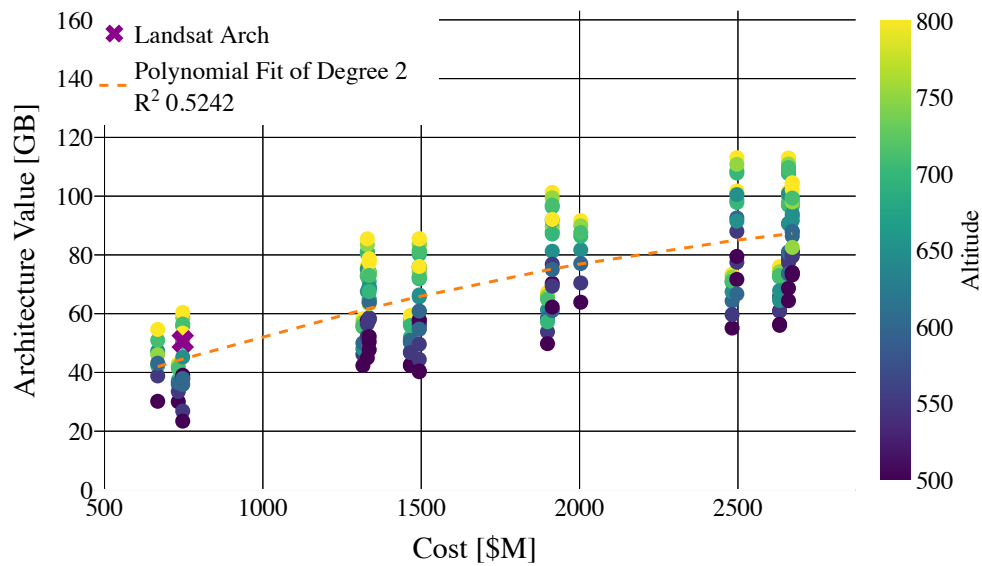
Figure A-1: Landsat Architecture Value Analysis
Color Scheme: Altitude
Scaling θ: 0.2
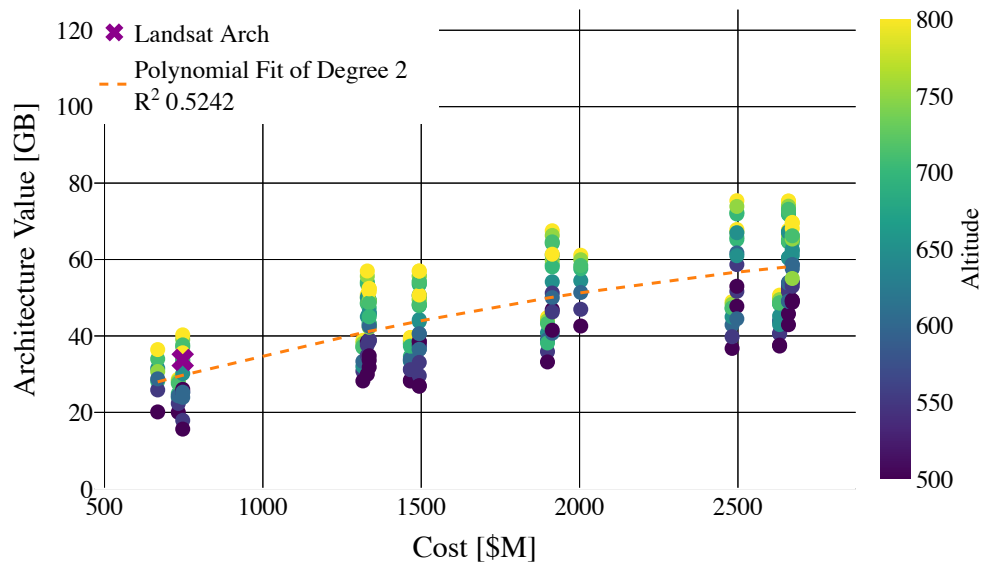
Figure A-2: Landsat Architecture Value Analysis
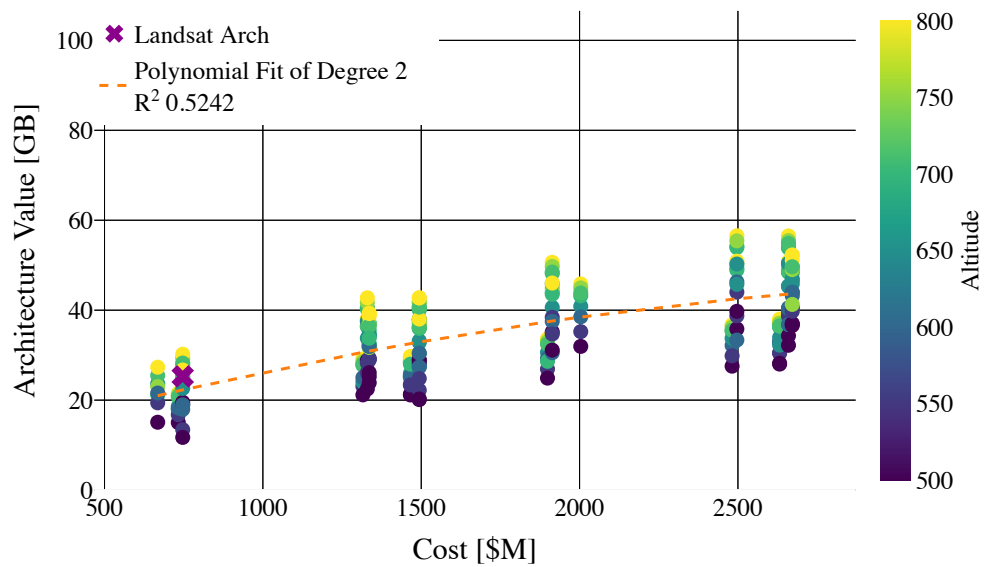Color Scheme: Altitude
Scaling θ: 0.3

Figure A-3: Landsat Architecture Value Analysis
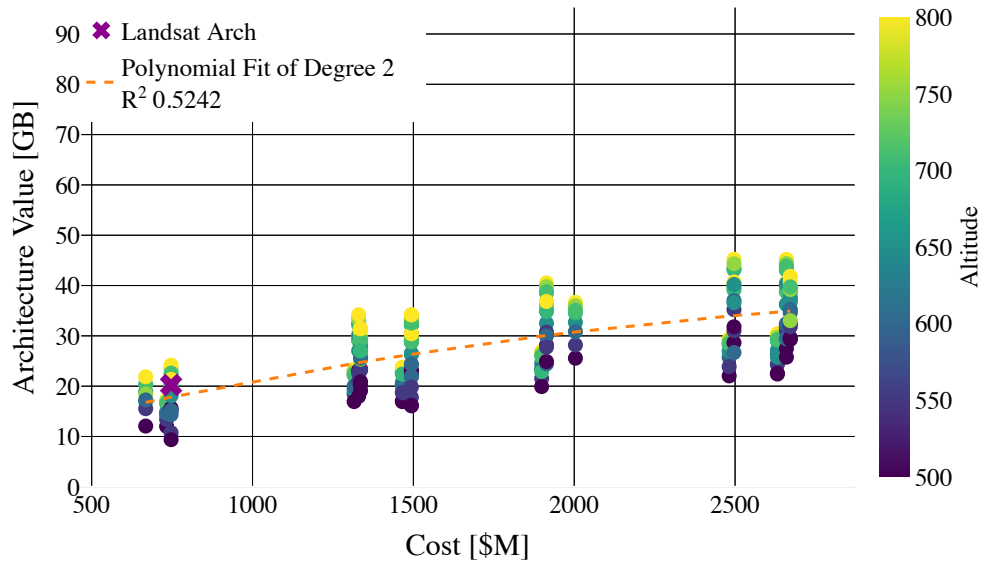Color Scheme: Altitude
Scaling θ: 0.4

Figure A-4: Landsat Architecture Value Analysis
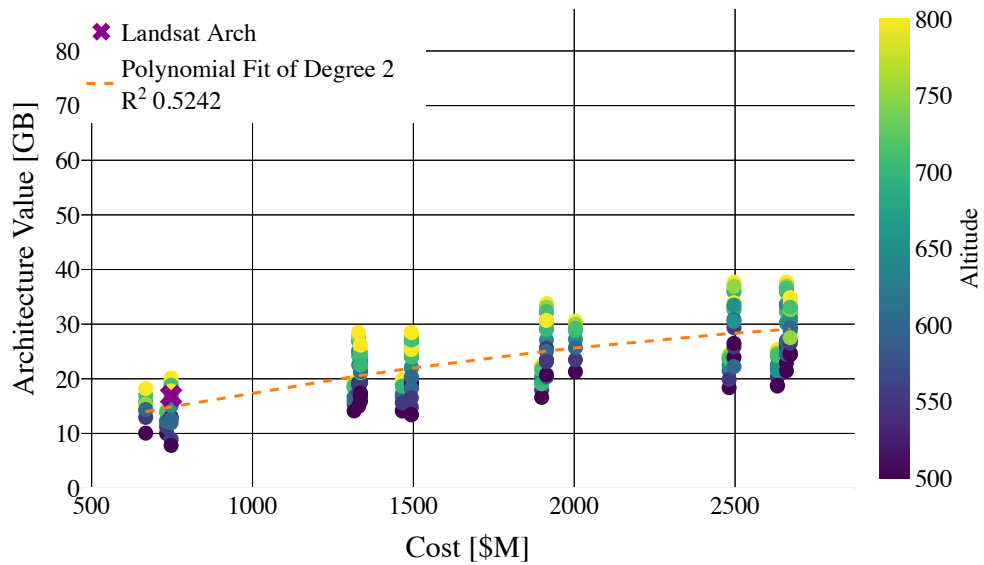Color Scheme: Altitude
Scaling θ: 0.5

Figure A-5: Landsat Architecture Value Analysis
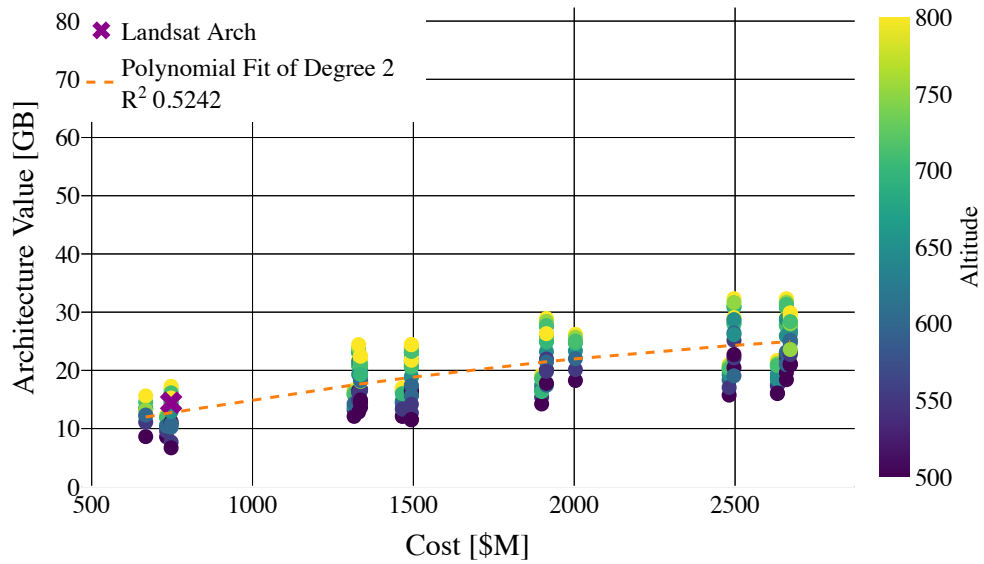Color Scheme: Altitude
Scaling θ: 0.6

Figure A-6: Landsat Architecture Value Analysis
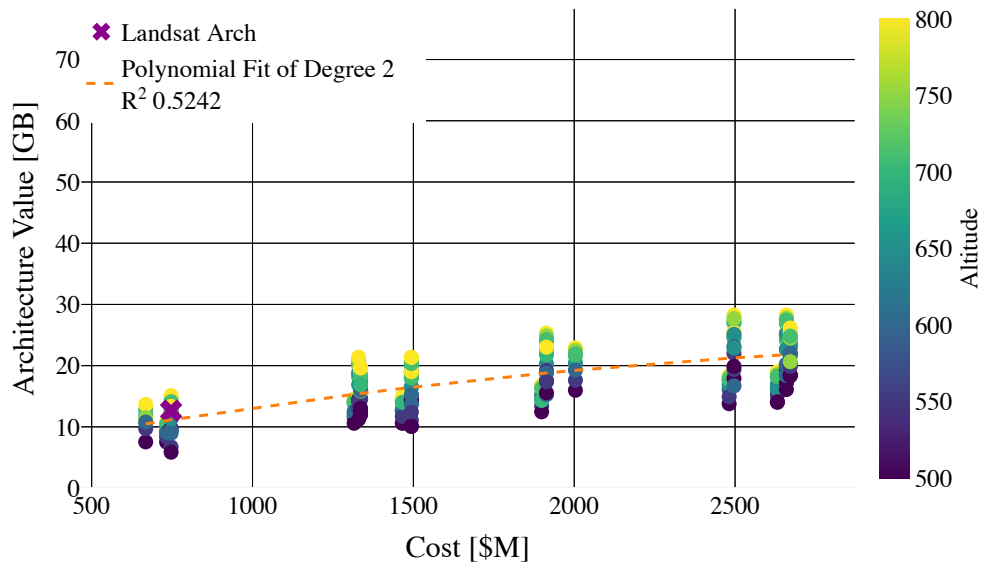Color Scheme: Altitude
Scaling θ: 0.7

Figure A-7: Landsat Architecture Value Analysis
Color Scheme: Altitude
Scaling θ: 0.8
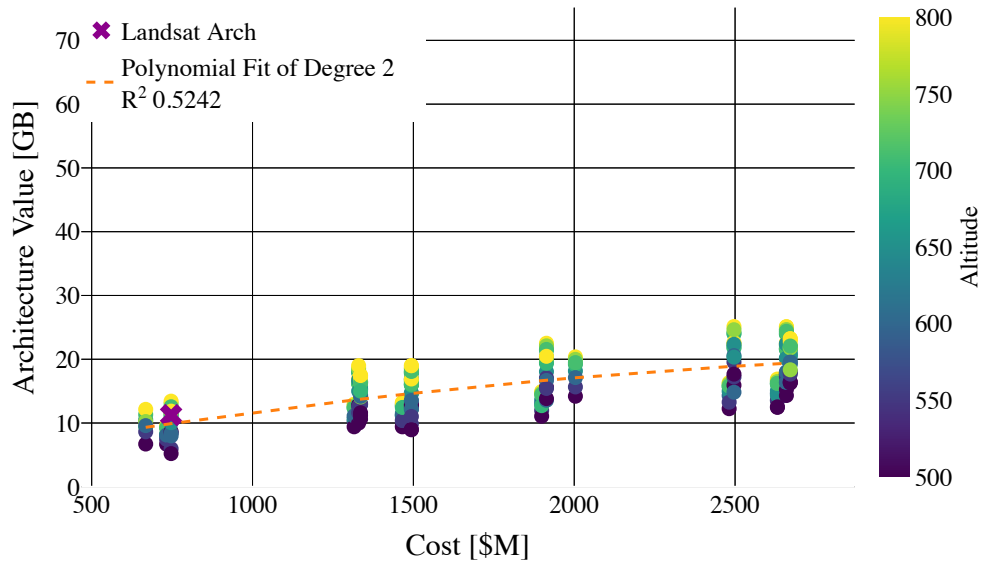
Figure A-8: Landsat Architecture Value Analysis
Color Scheme: Altitude
Scaling θ: 0.9

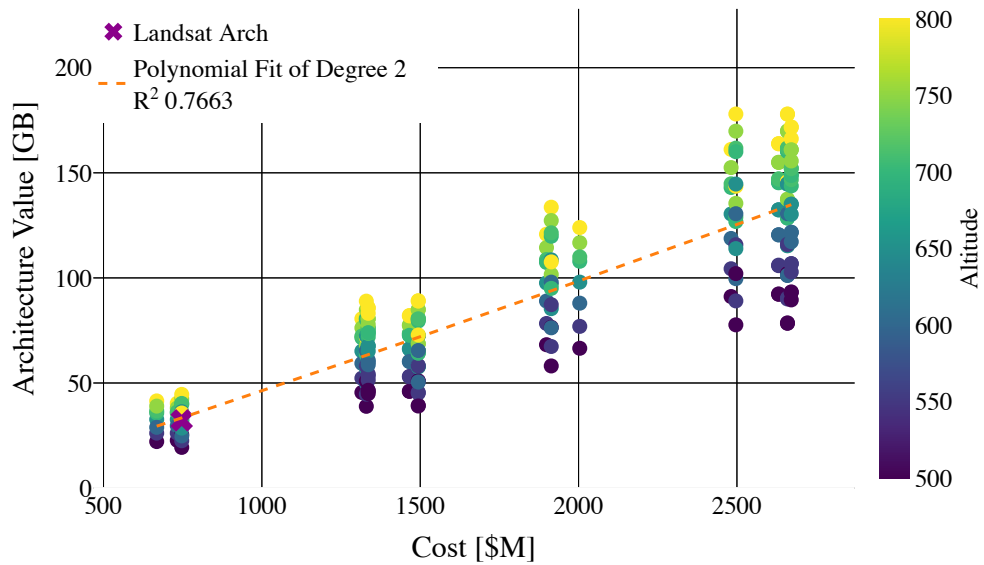
Figure A-9: Landsat Architecture Value Analysis
Color Scheme: Altitude
No Scalling Applied

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] Planet Labs Inc. Planet Imagery and Archive, 2020. Available: `https://www.planet.com/products/planet-imagery/`.

[2] Tanya Harrison. 60 Planet Use Cases to be Presented by Scientists at the American Geophysical Union Event Next Week, 2020. Available: `https://www.planet.com/pulse/60-planet-use-cases-to-be-presented-by-scientists-at-the-american-geophysical-union-event-next-week/`.

[3] Robbie Schingler. State of California and Planet Announce Groundbreaking Initiative to Support Action on Climate Change, 2020. Available: `https://www.planet.com/pulse/state-of-california-and-planet-announce-groundbreaking-initiative-to-support-action-on-climate-change/`.

[4] A. Siddiqi, E. Magliarditi, and O. d. Weck. Valuing New Earth Observation Missions for System Architecture Trade-Studies. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 5297–5300, July 2019.

[5] E. Magliarditi, A. Siddiqi, and O. de Weck. Remote Sensing for Assessing Natural Capital in Inclusive Wealth of Nations: Current Capabilities and Gaps. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 4411–4414, 2019.

[6] Annemarie Klaasse. Earth Observation for Sustainable Development: Agriculture and Rural Development, September 2018. Available: `https://olc.worldbank.org/system/files/3.%20Theme1_Klaasse_AgriculturalProductivity%20by%20Annemarie.pdf`.

[7] Open Earth Observations for Sustainable Urban Development, Feb 2020. Available: `https://www.earthobservations.org/article.php?id=410`.

[8] Guy Schumann, Dalia Kirschbaum, Eric Anderson, and Kashif Rashid. *Role of Earth Observation Data in Disaster Response and Recovery: From Science to Capacity Building*, pages 119–146. Springer International Publishing, Cham, 2016. Available: `https://doi.org/10.1007/978-3-319-33438-7_5`.

[9] Hua-Dong Guo, Li Zhang, and Lan-Wei Zhu. Earth Observation Big Data for Climate Change Research. *Advances in Climate Change Research*, 6(2):108 – 117, 2015. Available: `https://doi.org/10.1016/j.accre.2015.09.007`.

[10] Debra Werner. Forecasts Call for Rapid Growth in Earth observation market, Jan 2019. Available: `https://spacenews.com/forecasts-call-for-rapid-growth-in-earth-observation-market/`.

[11] Reportlinker. Global Earth Observation Satellite, Data and Service Market Generated $7.17 billion in 2018 and is Estimated to Grow at a CAGR of 9.10% During 2018-2023, Apr 2019. Available: `https://www.prnewswire.com/news-releases/global-earth-observation-satellite-data-and-service-market-generated-7-17-billion-in-2018-and-is-estimated-to-grow-at-a-cagr-of-9-10-during-2018-2023--300837805.html`.

[12] Martin Sudmanns, Dirk Tiede, Stefan Lang, Helena Bergstedt, Georg Trost, Hannah Augustin, Andrea Baraldi, and Thomas Blaschke. Big Earth data: disruptive changes in Earth observation data management and analysis? *International Journal of Digital Earth*, 0(0):1–19, 2019. Available: `https://doi.org/10.1080/17538947.2019.1585976`.

[13] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel. Multi-Task Learning for Segmentation of Building Footprints with Deep Neural Networks. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1480–1484, 2019.

[14] Jenna Mukuno. How Swift Geospatial Is Creating Positive Change In Forest Management, 2020. Available:`https://www.planet.com/pulse/how-swift-geospatial-is-creating-positive-change-in-forest-management/`.

[15] Harry Jones. The Recent Large Reduction in Space Launch Cost. 2018.

[16] Mike Safyan. Rocket Launch Trends Roaring Into The 2020s, 2020. Available: `https://www.planet.com/pulse/rocket-launch-trends-roaring-into-the-2020s/`.

[17] William Harwood. NASA Launches $855 Million Landsat Mission, Feb 2013. Available: `https://www.cbsnews.com/news/nasa-launches-855-million-landsat-mission/`.

[18] EO Portal. Landsat-8: Landsat Data Continuity Mission. Available: `https://directory.eoportal.org/web/eoportal/satellite-missions/l/landsat-8-ldcm`.

[19] Thyrso Villela, Cesar A Costa, Alessandra M Brandão, Fernando T Bueno, and Rodrigo Leonardi. Towards the Thousandth CubeSat: A Statistical Overview. *International Journal of Aerospace Engineering*, 2019:5063145, 2019. Available: `https://doi.org/10.1155/2019/5063145`.

[20] Small Satellites, Big Market: Sky's the Limit for CubeSats. Available: `https://www.protolabs.com/resources/blog/small-satellites-big-market-sky-s-the-limit-for-cubesats/`.

[21] S. Marcuccio, S. Ullo, M. Carminati, and O. Kanoun. Smaller Satellites, Larger Constellations: Trends and Design Issues for Earth Observation Systems. *IEEE Aerospace and Electronic Systems Magazine*, 34(10):50–59, 2019.

[22] Olivier de Weck, Afreen Siddiqi, Matthew Moragues, Alejandro Trujillo, George Lordos, and Mehak Sarang. Commercial Space Technology Roadmap, 2019. Unpublished.

[23] J.R. Wertz, D.F. Everett, and J.J. Puschell. *Space Mission Engineering: The New SMAD*. Space technology library. Microcosm Press, 2011.

[24] J.G. Walker. Satellite Constellations. *Journal of the British Interplanetary Society*, 1984.

[25] Navigation National Coordination Office for Space-Based Positioning and Timing. Space Segment. Available: `https://www.gps.gov/systems/gps/space/`.

[26] Sreeja Nag, Steven Patrick Hughes, and Jacqueline Lemoigne. Navigating the Deployment and Downlink Tradespace for Earth Imaging Constellations. 2017.

[27] Veronica Foreman. Emergence of second-generation low earth orbit satellite constellations : a prospective technical, economic, and policy analysis, 2018.

[28] David Winker. Cloud-Aerosol Lidar and Infrared Pathfinder Satellite Observations. Available: `https://www-calipso.larc.nasa.gov/about/atrain.php`.

[29] Merriam-Webster Online. Merriam-Webster Online Dictionary, 2009. Available: `http://www.merriam-webster.com`.

[30] A. Marinan, A. Nicholas, and K. Cahoy. Ad hoc cubesat constellations: Secondary launch coverage and distribution. In *2013 IEEE Aerospace Conference*, pages 1–15, March 2013.

[31] Edward J. Knight and Geir Kvaran. Landsat-8 Operational Land Imager Design, Characterization and Performance. *Remote Sensing*, 6(11):10286–10305, 2014. Available: `https://www.mdpi.com/2072-4292/6/11/10286`.

[32] USGS. Landsat Missions. Available: `https://www.usgs.gov/land-resources/nli/landsat/landsat-8?qt-science_support_page_related_con=0#qt-science_support_page_related_con`.

[33] James R. Irons, John L. Dwyer, and Julia A. Barsi. The Next Landsat Satellite: The Landsat Data Continuity Mission. *Remote Sensing of Environment*, 122:11 – 21, 2012. Landsat Legacy Special Issue.

[34] Planet Labs Inc. Planet Imagery Product Specification: PlanetScope & RapidEye, October 2016. Available: `https://`

www.planet.com/products/satellite-imagery/files/1610.06_Spec%
20Sheet_Combined_Imagery_Product_Letter_ENGv1.pdf.

[35] Armin W. Doerry and Fred M. Dickey. Synthetic Aperture Radar. *Opt. Photon. News*, 15(11):28–33, Nov 2004. Available: `http://www.osa-opn.org/abstract.cfm?URI=opn-15-11-28`.

[36] Sandra Erwin and Sandra Erwin. Capella Space to launch seven radar satellites in 2020 as it prepares for commercial operations, Dec 2019. Available: `https://spacenews.com/capella-space-to-launch-seven-radar-satellites-in-2020-as-it-prepares-for-commercial-operations/`.

[37] User Guides - Sentinel-1 SAR - Definitions - Sentinel Online. Available: `https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/definitions`.

[38] European Space Agency. Copernicus Sentinel Data, 2020.

[39] Sentinel-2a (10m) Satellite Sensor. Available: `https://www.satimagingcorp.com/satellite-sensors/other-satellite-sensors/sentinel-2a/`.

[40] Will Gragido, Johnl Pirc, Nick Selby, and Daniel Molina. Chapter 4 - Signal-to-Noise Ratio. In Will Gragido, Johnl Pirc, Nick Selby, and Daniel Molina, editors, *Blackhatonomics*, pages 45 – 55. Syngress, Boston, 2013. Available: `http://www.sciencedirect.com/science/article/pii/B9781597497404000046`.

[41] J.R. Wertz and W.J. Larson. *Space Mission Analysis and Design*. Space Technology Library. Springer Netherlands, 1999.

[42] G.S. Kino and T.R. Corle. *Confocal Scanning Optical Microscopy and Related Imaging Systems*. Elsevier Science, 1996. Available: `https://books.google.com/books?id=tNJX4wwZPaAC`.

[43] Planet Team 2017. Planet Application Program Interface: In Space for Life on Earth. Available: `https://api.planet.com`.

[44] Armin Doerry. *Performance Limits for Synthetic Aperture Radar - second edition*. 02 2006.

[45] J. Le Moigne, P. Dabney, O. de Weck, V. Foreman, P. Grogan, M. Holland, S. Hughes, and S. Nag. Tradespace analysis tool for designing constellations (tat-c). In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1181–1184, July 2017.

[46] Paul Grogan & Jonathan Verville. Tradespace Analysis Tool for Constellations AIST-16-0107 Final Review. Final Report given to NASA administrators, 2019.

[47] P. T. Grogan. Modeling Challenges for Earth Observing Systems of Systems. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 5289–5292, July 2019.

[48] Andris Slavinskis, Sreeja Nag, and Joel Mueting. An Initial Analysis of the Station Keeping Tradespace for Constellations. *IEEE Aerospace Conference Proceedings*, 03 2019.

[49] Nozomi Hitomi, Hyunseung Bang, and Daniel Selva. Adaptive Knowledge-Driven Optimization for Architecting a Distributed Satellite System. *Journal of Aerospace Information Systems*, 15(8):485–500, 2018.

[50] N. Hitomi and D. Selva. Constellation optimization using an evolutionary algorithm with a variable-length chromosome. In *2018 IEEE Aerospace Conference*, pages 1–12, March 2018.

[51] Pau Garcia Buzzi, Daniel Selva, Nozomi Hitomi, and William J. Blackwell. Assessment of constellation designs for earth observation: Application to the TROPICS mission. *Acta Astronautica*, 161:166 – 182, 2019. Available: `http://www.sciencedirect.com/science/article/pii/S0094576518317338`.

[52] NASA Goddard Space Flight Center. General Mission Analysis Tool, 2018.

[53] V. Ravindra and S. Nag. Fast Methods of Coverage Evaluation for Tradespace Analysis of Constellations. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pages 1–13, 2019.

[54] V. Ravindra and S. Nag. Instrument Data Metrics Evaluator for Tradespace Analysis of Earth Observing Constellations. *IEEE Aerospace Conference 2020*, pages 1–20, 2020.

[55] D. Bickel, B. Brock, and C. Allen. Spaceborne SAR study: LDRD 92 final report. [Synthetic Aperture Radar (SAR)]. 02 1992.

[56] Planet Labs Inc. RapidEye Imagery Product Specifications, January 2016. Available: `https://www.planet.com/products/satellite-imagery/files/160625-RapidEye%20Image-Product-Specifications.pdf`.

[57] S. Nag, J. LeMoigne, and O. de Weck. Cost and risk analysis of small satellite constellations for earth observation. In *2014 IEEE Aerospace Conference*, pages 1–16, 2014.

[58] SpaceX. Falcon 9 Launch Vehicle Payload User's Guide, 2008. Available: `https://www.spaceflightnow.com/falcon9/001/f9guide.pdf`.

[59] Dragos Maciuca, Jonathan Chow, Afreen Siddiqi, Olivier de Weck, Santiago Alban, Larry Dewell, Adam Howell, Joseph Lieb, Benjamin Mottinger, Julee Pandya, Jaime Ramirez, Alvar Saenz-Otero, Paul Yang, Andrew Zimdars, Salma Saeed, David Miller, and Scott Hubbard. *A Modular, High-Fidelity Tool to Model the Utility of Fractionated Space Systems*. Available: `https://arc.aiaa.org/doi/abs/10.2514/6.2009-6765`.

[60] Joy Brathwaite and Joseph H. Saleh. Bayesian Framework for Assessing the Value of Scientific Space Systems: Value of Information Approach with Applica-

tion to Earth Science Spacecraft. *Acta Astronautica*, 84:24 – 35, 2013. Available: `http://www.sciencedirect.com/science/article/pii/S0094576512004250`.

[61] Engineering National Academies of Sciences and Medicine. *Continuity of NASA Earth Observations from Space: A Value Framework*. The National Academies Press, Washington, DC, 2015. Available: `https://www.nap.edu/catalog/21789/continuity-of-nasa-earth-observations-from-space-a-value-framework`.

[62] Bruce G. Cameron. *Value Network Modeling: A Quantitative Method For Comparing Benefit Across Exploration Architectures*. PhD thesis, 2007.

[63] Wen Feng, Edward Crawley, Olivier de Weck, Rene Keller, and Bob Robinson. Dependency Structure Matrix Modelling for Stakeholder Value Networks. 07 2010.

[64] R. Freeman and John Mcvea. A Stakeholder Approach to Strategic Management. *SSRN Electronic Journal*, 01 2001.

[65] Bruce G. Cameron, Edward F. Crawley, Geilson Loureiro, and Eric S. Rebentisch. Value flow mapping: Using networks to inform stakeholder analysis. *Acta Astronautica*, 62(4):324 – 333, 2008. Available: `http://www.sciencedirect.com/science/article/pii/S0094576507002585`.

[66] Will Kenton. Net Present Value (NPV), Feb 2020. Available: `https://www.investopedia.com/terms/n/npv.asp`.

[67] F. Geng, J. H. Saleh, A. Tien, and R. A. Herd. Beyond Cost Tools: Spacecraft Net Present Value and the Hosted Payload Paradigm. *IEEE Transactions on Aerospace and Electronic Systems*, 51(4):3348–3363, Oct 2015.

[68] Daniel S. Valero. *Rule-Based Systems Architecting of Earth Observation Satellite Systems*. PhD thesis, 2012.

[69] Daniel Selva, Bruce G. Cameron, and Edward F. Crawley. Rule-Based System Architecting of Earth Observing Systems: Earth Science Decadal Survey. *Journal of Spacecraft and Rockets*, 51(5):1505–1521, 2014. Available: `https://doi.org/10.2514/1.A32656`.

[70] ACIL Allen Consulting. The Value of Earth Observations from Space to Australia, 2015. Available: `http://www.crcsi.com.au/assets/Resources/CRCSI-The-Value-of-Earth-Observations-from-Space-to-Australia-Final-web.pdf`.

[71] Hannah K. Miller, Lucian Richardson, Nick Sexton, Linda D. Koontz, Jennifer Loomis, and Steve Koontz. Users, Uses, and Value of Landsat Imagery in the United States. *2012 Socio-economic Benefits Workshop: Defining, measuring, and Communicating the Socio-economic Benefits of Geospatial Information*, pages 1–17, 2012.

[72] John Albert White, Marvin H. Agee, and Kenneth E. Case. Principles of Engineering Economic Analysis. 1977.

[73] National Snow and Ice Data Center. Available: `https://nsidc.org/cryosphere/glaciers/questions/move.html`.

[74] Michael D. King. *Our Changing Planet: The View From Space*. Cambridge University Press, 2007.

[75] Steve Cole, George Diller, and Rani Gran. NASA Launches New Earth Observation Satellite to Continue 40-Year Legacy, February 2013. Available: `https://www.nasa.gov/home/hqnews/2013/feb/HQ_13-040_LDCM_Launches.html`.

[76] Brian Markham. Landsat Data Continuity Mission: Overview and Status, March 2011.

[77] OSCAR: Observing Systems Capability Analysis and Review Tool. Instrument: Operational land imager.

[78] J. Gebbie, Mark Pollard, Haval Kadhem, Lee Boland, Alex Da Silva Curiel, Philip Davies, P. Palmer, Joe Steyn, and George Tyc. Spacecraft constellation deployment for the rapideye earth observation system. *60th International Astronautical Congress 2009, IAC 2009*, 5:3623–3634, 01 2009.

[79] George Tyc, John Tulip, Daniel Schulten, Manfred Krischke, and Michael Oxfort. The RapidEye mission design. *Acta Astronautica*, 56(1):213 – 219, 2005. 4th IAA International Symposium on Small Satellites for Earth Observation.

[80] EO Portal. RapidEye Earth Observation Constellation. Available: `https://directory.eoportal.org/web/eoportal/satellite-missions/r/rapideye`.

[81] George Tyc, John Tulip, Daniel Schulten, Manfred Krischke, and Michael Oxfort. The RapidEye Spacecraft. *Acta Astronautica*, 56(1):213 – 219, 2005. Available: `http://www.sciencedirect.com/science/article/pii/S0094576504003170`.

[82] OSCAR: Observing Systems Capability Analysis and Review Tool. Instrument: RapidEye Earth Imaging System. Available: `https://www.wmo-sat.info/oscar/instruments/view/421`.

[83] A. Siddiqi, E. Magliarditi, and O. de Weck. Small Spacecraft Earth Observing Missions for Natural Capital Assessment. In *International Astronautical Congress 2019*, 2019.

[84] Rachel Holm. Flock 2p Launches Successfully on PSLV, 2016. Available: `https://www.planet.com/pulse/flock-2p-launches-successfully-on-pslv/`.

[85] EO Portal. Planet - Flock Imaging Constellation. Available: `https://directory.eoportal.org/web/eoportal/satellite-missions/f/flock-1`.

[86] Ben Howard. What Is Agile Aerospace? Learn Planet's Approach, 2019. Available: `https://www.planet.com/pulse/what-is-agile-aerospace-learn-planets-approach/`.

[87] Incn Pro-West & Associates. Low Cost Aerial and Spatial Data, March 2018. Available: `http://dot.state.mn.us/research/TRS/2018/TRS1803.pdf`.

[88] Eric Kulu. Nanosats Database, 2020. Available: `https://www.nanosats.eu/`.

[89] James Mason. Keeping Space Clean: Responsible Satellite Fleet Operations, 2014. Available: `https://www.planet.com/pulse/keeping-space-clean-responsible-satellite-fleet-operations/`.

[90] Board of Governors of the Federal Reserve System. What are the Federal Reserve's objectives in conducting monetary policy?, 2019. Available: `https://www.federalreserve.gov/faqs/money_12848.htm`.

[91] B. Pyne, H. Saito, P. R. Akbar, J. Hirokawa, T. Tomura, and K. Tanaka. Development and Performance Evaluation of Small SAR System for 100-kg Class Satellite. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3879–3891, 2020.

[92] MIT 6.036. 6.036 Lecture Notes, 2019.

[93] Oleksii Kharkovyna. Machine Learning vs Traditional Programming, Apr 2019. Available: `https://towardsdatascience.com/machine-learning-vs-traditional-programming-c066e39b5b17`.

[94] A. Siddiqi, S. Baber, O. d. Weck, C. Durell, B. Russell, and J. Holt. Integrating Globally Dispersed Calibration in Small Satellites Mission Value. In *34th Annual Small Satellite Conference*, August 2020. Available: `https://digitalcommons.usu.edu/smallsat/2020/all2020/25/`.

[95] L. Yang, A. Siddiqi, and O. L. de Weck. Urban Roads Network Detection from High Resolution Remote Sensing. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 7431–7434, 2019.