Modeling Environments for Exploration of Business Dynamics within P-LEO Constellation Markets

by

Rehman S. Qureshi

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
May 6, 2023

Keywords: proliferated low Earth orbit constellation, deterministic dynamic programming, transitory dynamics, game, competition

Approved by

Davide Guzzetti, Chair, Assistant Professor of Aerospace Engineering
Daniel Tauritz, Associate Professor of Computer Science and Software Engineering
Ehsan Taheri, Assistant Professor of Aerospace Engineering

Abstract

Due to recent investments in the space industry, companies have begun to further explore the business feasibility for space-based systems. This rapidly expanding space resources economy includes proliferated low Earth orbit (P-LEO) satellite constellations which provide a variety of services such as broadband internet and telemetry to a diverse set of customer types, for multiple use cases. Although constellation operators hope this service is a financially lucrative business, uncertain market conditions coupled with limited resources create a techno-economic environment in which large capital expenditure and competition reduce the likelihood of profitability. Strategic development of P-LEO constellations is a central pillar that links orbital dynamics of constellations to downstream business performance. Since different constellation development strategies can lead to business success or failure, understanding the environment in which P-LEO constellations are developed is of paramount importance for any such business endeavors to be profitable.

When considering factors such as space debris, competition, socio-economic variations, and geographical demand, this becomes an even more challenging environment to navigate. Although comparisons of P-LEO constellations have been performed and individual constellation designs have been critiqued, modeling competitive business dynamics and complex resource management strategies involving such large space systems have not been fully explored. To study these complex interactions and understand how satellite internet business strategies evolve with environment complexity, business environments with varying levels of modeling fidelity are created.

Two different approaches are taken to model the complex P-LEO constellation environment: dynamic programming and gamification. A discrete, deterministic grid world formulation is used to develop simplified satellite communications business environments which utilize dynamic programming to generate optimal strategies. Such strategies gleaned from this environment demonstrate several concepts such as: optimal timing of investments, coupling of

design variables with actions, and optimal resource allocations. However, this environment is limited in terms of modeling elements and strategy complexity, so a gamified, multi-agent environment is also developed. Using a mixture of gamification and modeling, the environment is constructed as a multi-player, real-time-strategy (RTS) simulation game to simulate business competition between constellation operators. This expanded environment provides opportunities to explore competitive and cooperative strategies in a high dimensional environment.

As modeling fidelity increases, solving for an optimal strategy becomes increasingly infeasible. This is due to two factors: the rapid increase of the P-LEO environment's dimensionality and the increased complexity of processes modeled within the environment. Through qualitative comparisons of environments and investigations between action couplings, we can understand the dynamical, agent-environment system of P-LEO constellation markets at different levels of complexity.

Acknowledgments

First, I would like to thank to my advisor, Dr. Davide Guzzetti, for offering me a place in his lab during a time of incredible uncertainty. This position allowed me to return to the plains and be closer to my family. He has always supported my rather eccentric research pursuits, and I would not have been able to complete this research without his guidance.

I would also like to express my gratitude to my remaining thesis committee members - Dr. Daniel Tauritz and Dr. Ehsan Taheri - for their patience, guidance, and input throughout this process. Sharing their expertise requires significant demands on their time, and I am grateful to them for all they have contributed towards reviews and questions. Additionally, I would like to thank all team members of the Satellite Tycoon project who made the game possible. Specifically, I would like to give a special mention to Dr. Akhil Rao for his guidance in the areas of economics and business strategy.

I would also like to acknowledge my fellow lab-mates in the 3i Space Dynamics Lab for the passionate discussions, continuous sources of inspiration, and numerous coffee breaks. I would like to give a special mention to Manuel Indaco — for the many hours of discussion. I am also very thankful for the following items in my life: Panera Bread coffee, Moleskine journals, and Abbvie (makers of Lexapro). I couldn't have made it without them.

The warmest thanks go to my family who supported me throughout my time at Auburn. I especially want to express my gratitude to my parents: Mama and Papa, thank you for all the supportive calls, home-cooked meals, and wisdom which have made all of the difference. You're my heroes, and I'm thankful to be your son. Home wouldn't be the same without Nana phopho and chotu (Noor), and I'm very grateful for all of the love you both bestow upon me.

Lastly, my life would be nothing without the inspiration, motivation and support of Mehruba. You always pushed me to be a better version of myself and were my number one cheerleader. I will never fully be able to express the thanks I owe you for your patience and support. I will always be in your corner.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In 2015, an announcement by SpaceX lead to the renaissance of private investment in the satellite communications (SATCOM) industry, and a publication by DeepMind reinvigorated the artificial intelligence community pulling it out of an “AI winter”. On January 20, 2015, SpaceX announced their business expansion from launch vehicle manufacturing and proposed the development of a proliferated-low Earth orbit (P-LEO) constellation (also known as a mega-constellation) of telecommunications satellites intended to provide broadband services directly to retail consumers [1]. However, despite out-sized interest and investment by many governments and many telecommunications companies, it is still unclear how this will change the economic landscape of satellite internet and broadband as a whole.

Exactly one month after the SpaceX announcement, DeepMind published a novel reinforcement learning technique known as Deep Q-Learning, which revolutionized the artificial intelligence community by demonstrating advanced levels of control for various modeling environments [2]. Due to the strong results demonstrated with this class of algorithms, complex, multi-agent environments have seen a resurgence and become a popular research area which can be explored using artificial intelligence agents.

As discussed later in this work, technological advancements, increased competition, and a growing user base make the dynamics of the SATCOM industry an interesting environment to model using various degrees of fidelity — the central idea of this thesis. While certain classes of environments can leverage existing methods such as dynamic programming to identify optimal strategies, more complex, multi-agent environments employing gamification are needed to engage human players.

1.1 Motivation

The SATCOM economy has seen a recent surge in growth due to large investment by the private sector. Both private corporations and state-sponsored entities [3] are attempting to capture market share of the expanding, global high-speed internet marketplace. Such activity is causing a radical transformation in the space resource economy, rapidly transforming it into a complex territory of dispute where unknown logistical dynamics are emerging [4]. The development of cheaper, more accessible launch vehicles coupled with economies of scale has led to increased development of large satellite internet constellations to be operated in low-Earth orbit [5]. If successful, such constellations could service millions of consumers and deliver a more flexible broadband internet experience. Although it is speculated that these businesses will be profitable, such ventures come with high financial and safety risks: it is not readily apparent if constellation operators are guaranteed strategies that yield feasible (if not profitable) business cases. If such complex strategies do indeed exist, private space companies have not made them publicly available in open literature (nor are they expected to release such plans anytime soon). Constellation development requires large upfront capital expenditures to precipitate expensive research and development for a competitive market that has existing, conventional internet service providers (ISPs) [6] and long time-horizons to profitability. Additionally, due to increased launch activity and congestion of desirable orbit shells, there is an ever-increasing likelihood that constellations will interact with one another (or existing space debris); thus, increasing the likelihood of a collision event [7].

Historically, the commercial SATCOM industry has been separated into two categories: unidirectional broadcasting and bidirectional broadband internet services. In the past, companies such as Iridium [8], Teledesic [9], and Orbcomm [10] had difficulty gaining market-share of retail broadband internet services due to the poor relative performance, high customer costs, and limited use cases. Such systems were also expensive to develop and launch, thus making their services cost-prohibitive to the average retail consumer. Therefore, bidirectional satellite broadband internet systems were relegated to specific use cases involving limited availability and low throughput. However, increased internet usage and evolving content consumption of retail consumers [11] has facilitated an explosion in funding and subsequent research of broadband communications technologies. Such a shift in the retail marketplace has also changed the retail consumer demand for satellite internet [12]. Existing constellation operators have now shifted away from the shrinking broadcasting sector and are developing large constellations best suited for high-throughput bidirectional broadband internet services.

The shift towards broadband internet services has been even more expedited with the emergence of a new breed of competitors in the SATCOM marketplace. New, private corporations have begun requesting permission from the Federal Communications Commission (FCC) to expand, or begin construction of, proliferated low Earth orbit (P-LEO) constellations. Such constellation designs call for thousands of relatively inexpensive satellites to be deployed in low Earth orbit (LEO) and provide internet access directly to consumer user terminals. This strategy differs from conventional constellations in both volume and price per satellite, utilizing cheaper launch vehicles and economies of scale to drive down production costs. However, such constellation endeavors require substantial capital expenditures and offer low profitability during the initial development and adoption period. Since 2019, more than a dozen applications have been submitted to the FCC from companies requesting to launch thousands of high-throughput satellites into LEO [13]. Existing constellation operators such as ViaSat and SES are also launching additional satellites to boost their constellation performances and appeal to the growing retail customer base. As each new constellation operator attempts to gain retail market share, they not only contend with other constellations, but with traditional, regional ISPs as well. Such a hyper-competitive environment poses out-sized financial and national security risks, and warrants increased scrutiny regarding the existence of feasible business strategies. Figure 1.1 has been reproduced from [14] and shows the main players in the industry and categorizes them based on round-trip latency and system throughput.

Simulation and modeling of P-LEO constellations has been a critical topic of research in recent years. Much attention has been given to analyzing the impact of such large constellation designs, but little research has been conducted to understand the constellation operators' financial and strategic decision-making environment. While routing algorithms have been developed to address the data congestion issues associated with multi-gateway networks [15], extensive

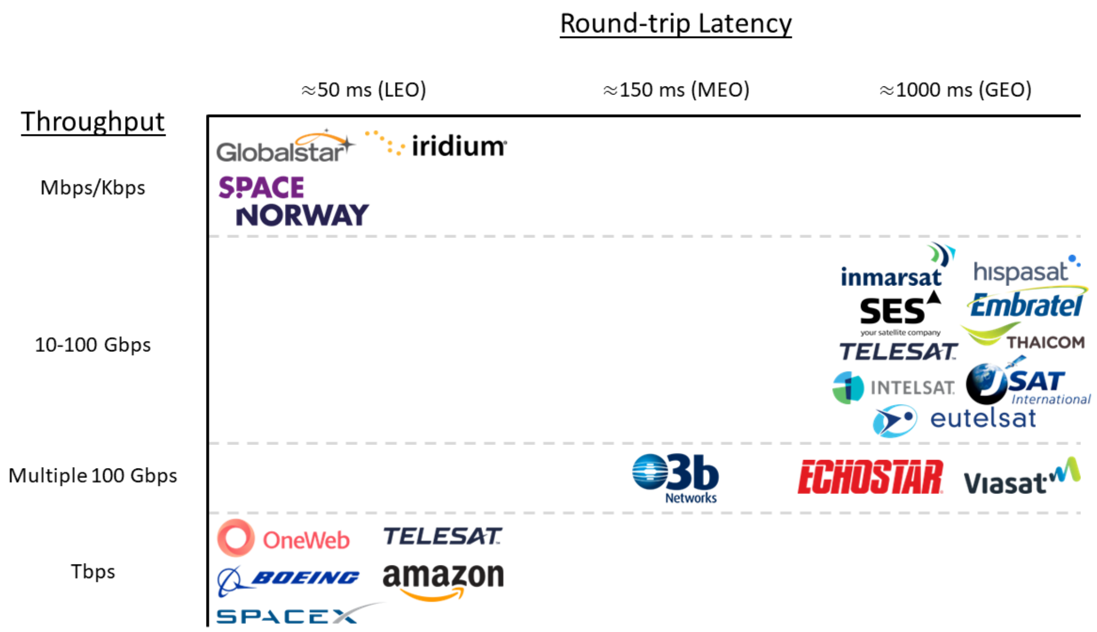

Figure 1.1: Key Players in the SATCOM Industry (Adapted from [14])

works have analyzed the relationship between P-LEO constellations and space debris [16, 17, 18]. Safety analysis [19] and augmentation of constellations for emergency observations has also been explored [20]. Financial research within the satellite telecommunications space has largely focused on specific studies on the financial feasibility of select constellations or specific areas of interest such as resource allocation [14] or customer demand estimation [21]. However, business strategies throughout a constellation's lifespan have not been modeled and explored adequately via traditional methods of analysis. Using traditional combinatorial approaches such as grid search would yield an infeasible number of strategies to be modeled and temporally propagated. Additionally, such strategies are affected by actions of all constellation operators; thus, requiring us to model all possible interactions between all competitors. Considering these factors along with the hyper-competitive business environment, it becomes clear that a more efficient method of modeling the development strategies of SATCOM businesses is required.

1.2 Problem Statement & Research Questions

The scope of this work focuses on the development of SATCOM marketplace environments which model the competitive environment of broadband internet offered by P-LEO constellations. The effects of modeling fidelity on these complex business logistics is investigated by varying the environment designs. The objectives of this thesis are as follows:

1. Understand the effects of adding certain modeling components, state variables, and action variables to a SATCOM environment.

2. Evaluate couplings of state and action variables within each environment.

3. Assess which performance metrics properly convey economic success of SATCOM businesses.

These objectives can be directly mapped to the following research questions:

RQ1. How does varying the level of modeling fidelity affect performance metrics and competition?

RQ2. What are the economic benefits and penalties of competition to dynamic constellation development strategies?

RQ3. Which variables impact economic figures of merit that measure success, failure, and risk during development of a P-LEO constellation, in a competitive environment?

1.3 Research Methodology

To answer the above research questions, we identify and investigate three distinct subjects of interest and use this information to construct different environments and strategies. The first area of interest is the overall SATCOM industry. This includes the previous generation systems, market shifts in recent years, and constellation designs by each SATCOM player. Additionally, we explore common challenges associated with development and maintenance of a P-LEO satellite internet constellation. The second area of interest is the current state-of-the-art regarding environment design and development. Due to interactivity between SATCOM operators, conventional methods are incapable of modeling cooperative or competitive behaviors; therefore, environment models based on sequential decision-making and multi-agent are investigated. The final subject to be explored is that of gamification and game design. This is neither meant to be a comprehensive review of game theory; rather, this subject is used to provide insight into the current literature of research games.

To create both environments with separate degrees of complexity, each was created with specific modeling assumptions driving design and implementation decisions. Because of this, each environment utilizes a different method of evaluating strategies (the grid world formulation utilizes dynamic programming to generate optimal strategies and the gamified multi-agent framework utilizes gamification to engage human players). Although both frameworks have been developed to a different level of modeling fidelity, strategies represented in the simpler environment may also be replicated in the more complex environment. The overall methodology of this thesis is shown in Figure 1.2.

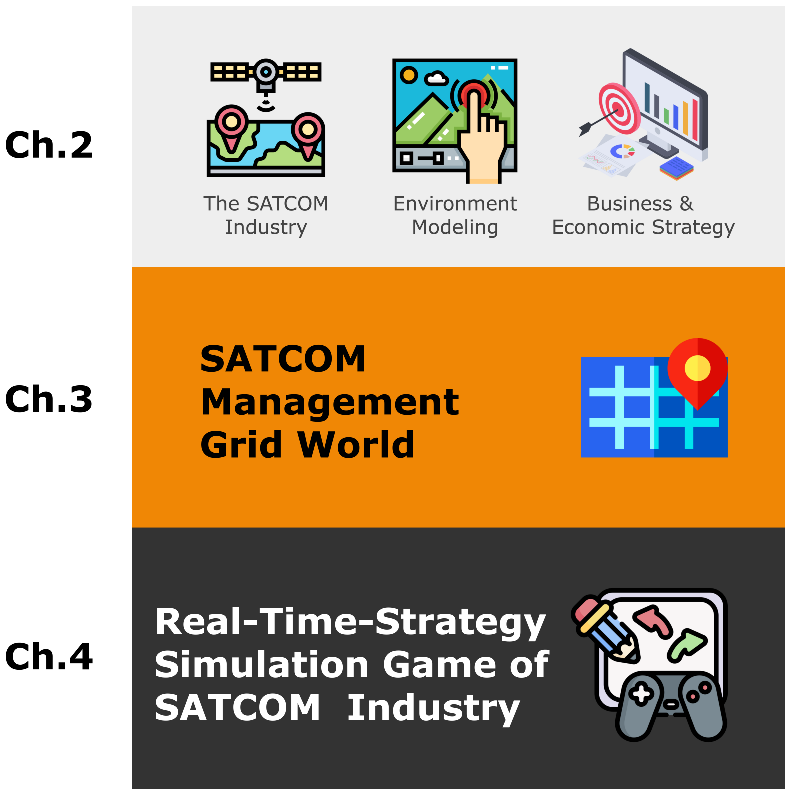

Figure 1.2: Scope & Chapter of Each Environment

1.4 Thesis Outline

The remainder of this thesis is structured as follows:

- Chapter 2 presents past literature relevant to this research and identifies gaps in the collective understanding of modeling environments specific to space mission design for P-LEO constellations and their business dynamics.
- In Chapter 3, a fundamental framework is presented in which a business with simple satellite deployments and monthly subscription price fluctuations is modeled. This environment is formulated as a discrete, deterministic dynamic program, and optimal policies are computed using the backward dynamic programming algorithm.
- Chapter 4 provides an extensible formulation of the SATCOM industry as a multi-agent simulation using gamification and user interface design. This framework includes modeling expansions of actions, resources, orbital dynamics, customer behavior, and costs/rewards. This formulation also introduces collision probabilities as a future expansion. Due to the complexity of such an environment, traditional solution methods are not capable of generating strategies for such a highly dimensional system.
- Finally, Chapter 5 offers discussions regarding the results and a conclusion of the work. Additionally, future work bridging the gap between the dynamic programming environment and the gamified multi-agent environment is proposed.

Chapter 2

Literature Review

This chapter addresses various topics and disciplines that will be discussed throughout this work. It begins with the history, technologies, and use cases of the SATCOM industry, particularly next generation proliferated low Earth orbit (P-LEO) constellations. Existing environments and their designs are discussed with comparisons made to those developed in this work. We then turn our attention toward understanding decision-making in the realm of game design and game theory. While not an extensive review of game theory, we discuss relevant frameworks and architectures required to develop gaming environments for simulation and recreation. Lastly, the existing gaps in research are examined and, in an attempt to address these gaps, hypotheses are formed from research questions posed in Chapter 1.

2.1 The SATCOM Industry

2.1.1 First Generation SATCOMs

Throughout the 1990s (and early phase of widespread internet adoption), expected revenues from SATCOM were primarily based on the notion that satellite connections will be required for broadband connectivity and cellular systems. Often, these expected revenues did not match reality. This first wave of commercial SATCOMs were predicted to usher in a era of global communications. However, due to underestimation of terrestrial competition, overestimation of the addressable market, expensive technology developments, and limited investor capital (mixed with poor financial planning), only three out of ten major systems were still in operation in 2003 [22]. For example, the Globalstar system, which came into commercial operation in 1999, reported just 84,000 subscribers 4 years later.

Although P-LEO constellations were predicted as far back as 1990, with multiple ventures trying to attract investors and develop non-GEO satellite constellations for mobile communication and broadband connectivity, the market for satellite internet did come into fruition as much of the industry expected. Even with strong technical feasibility and engineering innovations, the economic feasibility of systems such as Globalstar Iridium, and Odyssey was very uncertain [23]. This even included heavy backing from traditional telecommunications providers such as Motorola, Nokia, and Loral, and seed funding by Bill Gates or Craig McCaw. For example, Motorola proposed the 77 satellite Iridium system (aptly named since iridium is the 77th element of the periodic table). However, this constellation proved far too costly and reduced the number of satellites to 66. The name was not changed. Although the goal of Iridium was to provide global telephone services, the short-term goal of the company was commercial (economic) success and a 16dB targeted link margin. To achieve this, the final constellation design consisted of 66 satellites (with 6 spares) distributed across 11 orbit planes (equally spaced apart) with a spare in each plane. The launch vehicles of choice for the parent company (Motorola) were the U.S. Delta II rocket, the Russian Proton rocket, and the Chinese Long March IIC/SD rocket. Given the 60% reserve of consumables (primarily propellant load), the overall lifespan for these satellites was estimated to be approximately 8 years [8]. Additionally, system trades such as: complexity of payload, size and weight of satellites, altitude of satellites, individual beam size, radiation effects, receiver (telephone) size, quality of service, number of gateways, and modulation type were all considered by the Iridium team when designing this global network. Note that many of these same trades are currently being conducted for current generation system.

The largest and most expensive of these first generation SATCOM systems, Teledesic, was designed to connect users across the globe via broadband internet connectivity. Much like current SATCOM proposals, this system also proposed to operate in LEO with 840 active satellites. At the time of this system development, the estimated cost of such a system was approximately $9 billion. Given such a large cost, it is quite likely that Teledasic was expected to capture the entirety of the broadband internet market of the 1990s. However, due to low investor confidence, the original constellation design was scaled back twice (once to 288 satellites and finally to 30 satellites) greatly affect overall qualty of service and customer use cases.

Another consumer-facing satellite internet provider, Hughes Network Systems, had, at its height, 100,000 subscribers. Compared to cable broadband services, this network was not only expensive, but had limited bandwidth for single user data. As consumer demand for internet usage grew, this bandwidth cap became a critical issue.

2.1.2 Current Generation SATCOMs

Following the first generation of SATCOMs and the explosion of internet traffic, terrestrial cable broadband systems made strides in reducing latency with technology such as fiber optics. The same was not true for satellite internet services in which latency was a prominent issue with users. Additionally, when compared with terrestrial systems, satellite internet constellations were still much more costly to design, build, launch, and maintain. However, due to strides in reusable launch vehicles and new capabilities for CubeSats and smallSats to be added as secondary payloads to existing missions [5], a second generation of SATCOM was created.

Of the current SATCOM generation, three companies have made detailed plans and exectued: O3b, OneWeb, and SpaceX [24]. O3b started business in 2007, but launched its network of 20 satellites to medium Earth orbit (MEO) in 2019. While they have seen success from government entities and the United States Department of Defense, O3b does not connect users directly to satellites; rather, they utilize a system called Internet Protocol (IP) trunking. This is a system by which the company sells gateways to local ISPs, who then sell subscription services to local customers. Such a system allows for shared costs and revenues between O3b and the local ISPs.

Unlike O3b, OneWeb and Starlink both offer services directly to customers via user terminals. Additionally, both constellations have produced hundreds of thousands of terminals and hundreds (if not thousands) of low-cost satellites using mass production methods. These mass-produced satellites are currently being deployed to LEO to reduce latency concerns while offering flexible service options for users.

The technical challenges for second-generation SATCOMs must still face an old problem: To reduce latency, the average altitude of the constellation must be lower. Therefore, hundreds or thousands of satellites are needed to ensure continuous global coverage. Such an increase of active satellites not only requires increased expenses to track and maintain these large systems, but it also entails more launches to deploy the entire constellation. As launch costs continue to fall due to reusable launch vehicles and efficiency improvements, some of the economic barriers in place during the first wave of SATCOMs are now being conquered. This can further be seen in the development of cheaper, lighter satellites which are envisioned to be mass produced for less than $500,000 per satellite [24].

One of the economic challenges that is yet to be addressed is that of user terminal adoption in P-LEO networks. In the previous generation of SATCOMs, high barriers to entry for consumers made such services unpopular. However, the assumption second-generation SATCOM businesses are making is that the reduction of user terminal cost will come from mass production and network effects. Therefore, user interest in such services may also be greater than in the previous generation.

2.1.3 P-LEO Constellation Analyses

As P-LEO constellations have grown in popularity this past decade, detailed analysis has been conducted on several portions of the system. Most common areas of interest include: staged deployment effects, data routing algorithms between satellites and user terminals, the investigation of inter-satellite link paths, and general comparisons of potential constellation designs. These analyses have led to greater technical understanding of specific effects and constellation components.

Due to the scale of these large constellations, SATCOM operators must increasingly choose staged deployment options for their constellations. In [25], a trade space is developed to represent the potential architectural evolution of P-LEO constellations, and staged deployment is investigated for specific proposed constellations within the design space. It is shown that for an estimated demand function, specific cost savings exist by not under or overdeveloping ones constellation with respect to a particular demand. Although robust, many modeling assumptions are made with regard to customer adoption of services and overall life-cycle costs. Additionally, in [26], the authors investigate the effects of adding a new layer to an existing constellation via a design framework for multi-layer, staged deployment of P-LEO constellations under demand uncertainties. Cost savings for the multi-layer staged deployment are also given in life-cycle cost savings of approximately 42.8% when compared to optimized single-layer deployments. This work is also accompanied by a decision support system that allows rapid prototyping and development of different multi-layer staged deployment strategies. It should be noted that neither work considers competition and simply assume a specific customer demand throughout their simulations. Finally, in a work developed jointly by SpaceX and MIT [27], authors develop a tradespace and apply a subset of Epoch-Era Analysis called Multi-Epoch Analysis in conjuncture with the Multi-Attribute Tradespace Exploration (MATE) tool for the maximization of P-LEO value across different customer types and geographical distributions.

When considering routing and network congestion, [15] formulate a routing model in Mega-Constellation Networks with Multi-Gateway (MCNMG) system using a hybrid approach (using both satellite interlinks and terrestrial networks). Rather than implementing an IP-based algorithm for packet hopping and network traffic, a new strategy called Longer Side Priority is developed in which packet forwarding is formulated based on local network status. In this method, the MCNMG is formulated as a wireless mesh network and use Markov chains to solve the path survival problem.

Laser inter-satellite links (LISLs) are a type of free space optics (FSO) wireless communication technology [28]. More precisely, FSOs have four basic types: terrestrial, aerial, space, and deep space. The LISLs utilized in P-LEO constellations are designated as space FSO. The use of laser-based links provides the following several benefits over traditional radio-frequency (RF) based links. These include: high data rates, smaller required antenna hardware, less beam interference, and lower required power due to less beam spread. LISLs can be classified into two types of links, depending upon the orbit plane of the satellites communicating with one another. Intra-orbit plane LISLs are those which connect two satellites within the same orbit plane, while inter-orbit plane LISLs are those created between satellites of different orbit planes. Inter-orbit plane LISLs can be further classified into three types: adjacent orbit plane links (AOPLs), nearby orbit plane links (NOPLs), and crossing orbit plane links (COPLs). AOPLs are formed between satellites in adjacent orbit planes, NOPLs are formed between satellites in nearby (but not adjacent)orbit planes, and COPLs are formed between satellites which are in crossing orbit planes. In the presence of LISLs, it is also shown that gateway terrestrial gateway placement can be optimized using an integer optimization model. This is then solved by a GA-based algorithm to give optimal gateway placement [29]. A noteworthy result from this work was the characterization of preferred gateway locations due to optimization: close to coastal areas of continents, at matching latitude as the inclination of orbit planes, and near high user demand areas. Additionally, [30] discusses an estimation method for LISL hop-counts between customers in the Starlink constellation. This method attempts to understand the nature of P-LEO communications and the constellation network's design impact.

2.1.4 Impacts of Space Debris on P-LEO Constellations

As P-LEO constellations become increasingly popular for SATCOM, these proliferated systems are much more prone to interacting with space debris or other resident space objects (RSOs). Simply by the nature of collision probabilities [16, 31], the probability of a P-LEO constellation colliding with RSOs or debris is exponential compared to previous generation systems. With the increase of interactions between P-LEO constellations and space debris, assessments [32] and optimization methods [33] have begun to emerge in this congested environments. Indeed, interactions between P-LEO constellations and RSOs has already begun to happen [34, 35]. Although debris mitigation is a rich area of research right now [17], such pollution of LEO [36] with space debris and thousands of satellites may also cause an *economic* Kessler syndrome [37]. Given the large number of RSOs, space debris, and high collision probabilities, there seems to be an economic equilibrium which, when crossed, may begin to affect the financial feasibility of space resource economies.

2.2 Environment Modeling of Space Logistics Economies

Modeling of the space environment has been executed in various different way depending upon the use case being modeled and application of the space system. For instance, SpaceNet Cloud [38] is a space mission logistics planning tools which is web-based application built on the existing SpaceNet software architecture. While SpaceNet Cloud is an application that models general space mission planning, a more specialized space environment is that of satellite task planning [39] formulated as a semi-Markov decision process (SMDP). In this work, an Earth observation satellite is modeled with multiple mission objectives and must schedule these tasks. The satellites tasks include: collecting images, communicating back to Earth, recharging batteries by specifying sun-pointed periods, and data recorder management. This formulated SMDP is then solved using both forward search and Monte Carlo Tree Search. For both single objective and multi-objective cases, the SMDP solutions either led to greater rewards or faster solution time than the baseline methods (graph search and mixed-integer-linear programming). Similarly, satellite constellation replenishment and inventory control for the Global Positioning System (GPS) satellite constellation has also been modeled using a MDP formulation [40]. In this formulation, the number of satellites in the constellation is defined as the state of the system and the action is to order an appropriate amount of satellites to maintain the 24 satellite minimum for global operation. Once formulated, this MDP is solved using value iteration algorithm to produce replenishment policies.

One of the most similar works to our modeled environments is that of a static, techno-economic framework which assess LEO satellite constellations' economic feasibility [6]. This framework includes 3 major portions:

1. A capacity model for each constellation which uses very simplified connection mathematics to compute coverage area, EIRP, and SNR to get a rough approximation of overall constellation capacity

    - It accomplishes this by ignoring the actual orbital mechanics and simply using a very simplified (Pythagorean theorem) method to compute altitude.

2. A cost model which uses a combination of capital expenditures (CapEx) (launch, station acquisition, spectrum acquisition, and overall integration) and a yearly operating expenditure (OpEx) (energy, customer acquisitions, research & development, labor, and maintenance) to compute the discounted Net Present Value (NPV) of the Total Cost of Ownership (TCO) of each constellation's assets

3. A demand model which roughly approximates users per square kilometer using local population and adoption rates ranging from 0.5% to 2%

   - This user metric is then used in conjuncture with the previous two models to compute capacity per user and cost per user to understand how many active users can be sustained by these constellations without marked drops in quality.

This framework assessed Starlink, OneWeb, and Kuiper proposed constellations and showed that these are only effective with relatively low subscriber density (0.1 users per km2). However, there are many factors missing from this work including: dynamic strategies/development times, competitive markets, limited numbers of customers, inclusion of potential revenues, orbital congestion/debris, and satellite replenishment costs.

In addition to the above techno-economic framework, a series of works [41, 42, 43, 44, 45] relating to the development of a game modeling the dynamics of federated space systems very closely aligns with the overall direction of our work and the Satellite Tycoon game presented in Chapter 4. Laying the groundwork for a fully developed simulation game, [41] describes the space-based resource economy in the context of wargaming and federated systems. It also gives an example of components from the space-based resource economy used for lunar in-situ resource production applications. Finally, the abstract gaming framework is described with requisite components. In [42], federated satellite systems (FSS) are described as a system of systems (SoS), and a simulation architecture is described using high-level architecture standards. The *Orbital Federates* simulation game describing FSS logistics is ultimated developed and presented in [43]. Using the developed *Orbital Federates*, [44] represents federated systems as a Stag Hunt game and investigates risk-dominant (non-cooperative) and payoff-dominant (cooperative) strategies. Finally, [45] delved deeper into the game theoretic aspects of federated systems for the multi-actor case.

2.3 Gamification & Game Design

With over 800 studies discussing its advancements and limitations, gamification has been shown to be a viable method of simulating complex research problems with no direct or computable solution [46]. Such studies cover domains such as education [47], health [48], and social behavior [49]. While others have gamed existing engineering simulations to gain insight [50, 51, 52, 53], a gamified formulation of a P-LEO constellation marketplace has not been established. Unlike simulation gaming or serious games, gamification is able to merge the attributes of recreational games and the mathematical modeling of dynamical systems to create interactive, multi-agent frameworks [54]. Gamification particularly benefits the exploration of multi-agent systems which typically have very high dimensional, possibly continuous, action spaces and require multiple human interactions. It is difficult to gain analytical insight regarding such complex strategy sets; therefore, gamification can instead be used to explore the strategy space.

Historically, a standard method of modeling interactions between decision-makers and their environments has been to represent these interactions as a traditional game (in both the mathematical and colloquial sense). This is due to several factors: games have been shown to be an intuitive manner of simulating interactions [55], complex scenarios can be modeled and studied using games (and therefore game theory principles) [56], and games can offer a way of collecting large amounts of training data for machine learning models to train on [57]. Due to these reasons, we explore this environment through the use of gamification. As gamification becomes a more popular method of framing research and engineering tasks, it has also become more broadly used to represent many different types of processes, frameworks, and concepts.

2.4 Summary & Working Hypotheses

As we have shown, there exist significant knowledge gaps pertaining to strategic deployment of P-LEO constellations. Specifically, the interactions and effects they can have on the broadband internet marketplace remains unknown.

From this we can observe clear gaps in knowledge for the following areas:

- The financial feasibility for a SATCOM operator's business strategy
- The transitory dynamics associated with the development stage of P-LEO satellite internet constellations
- The impact competition and limited customers can have on the overall marketplace

This work attempts to address the above gaps in knowledge by introducing two environments of varying modeling fidelity to compare and contrast strategies generated at each level. In particular, there are three hypotheses regarding the impact of framework design on business strategy development that align with the proposed research questions:

1. While increasing modeling fidelity causes accuracy to increase, there may also be a detrimental impact on the overall quality of solutions produced, i.e. it will become more difficult to produce exact optimal strategies as the modeling fidelity increases (particularly if the number of competitors or complexity of the competitors' strategies increase).
2. The SATCOM business model of delivering broadband services directly to retail consumers demands a large upfront capital investment followed by years of profit uncertainty. Specifically, actions across all aspects of the environment (space, ground, and economic) should be synchronized to maximize the utility of all investments if business feasibility is to be achieved.
3. Since most P-LEO satellite constellations are business entities, economic performance metrics such as net present value and internal rate of return can be affected by constellation size, investment timing, and pricing of internet services. Sufficiently robust customer

decision-making models can be used to approximate potential customers and give insight regarding the economic performance of a particular SATCOM operator.

The remainder of this work details both environments' design and formulation.

Chapter 3

A SATCOM Grid World

To model the most basic version of a competitive SATCOM environment, we begin by introducing a simple grid world version of the resource management problem. Over a fixed time-horizon, a decision-maker must build a P-LEO constellation and adjust their offering price to acquire satellite internet service customers. The decision-maker must find an optimal strategy by choosing to launch a fixed number of satellites, into specific orbit shells, at specific epochs, and setting a competitive price for their broadband services. This all must be done within a fixed time-horizon while also maximizing profits. The following simplifying assumptions are made with regard to modeling satellite quality, customer behavior, launch availability, and computation of figures of merit for constellation performance:

A3.1 User terminal manufacturing, production, and sales are not considered in this environment. It is assumed that virtual customers already possess the relevant technologies to interact with any of the broadband services being offered, so these particular economics are ignored.

A3.2 For computational simplicity, constellation design parameters are abstracted away and focus is only given to the number of satellites in each orbit shell as our constellation design variables. To further this assumption, we assume all satellites are in circular orbit shells of varying altitude; however, detailed orbit shell dynamics are not modeled.

A3.3 A single type of satellite is modeled; therefore, the cost and performance metrics are fixed values without any upgrades or economies of scale throughout the simulation. This simplifies the available set of actions the decision-maker must consider.

A3.4 LEO space is discretized into a finite set of orbit shells that the decision-maker may launch to. This bounds the number of feasible strategies to a finite and countable number.

A3.5 Virtual customers are stateless. Therefore, their broadband service decisions are recomputed at the beginning of every virtual month. This presents the decision-maker with near-immediate feedback to their strategy alterations.

A3.6 Virtual customers are also assumed to be rational in their decisions for broadband service.

A3.7 Satellites launched in a given epoch are active constellation components on the next epoch. Therefore, there is no setup or delay time modeled.

A3.8 This environment is considered deterministic; i.e. there is no uncertainty to the models of the simulation.

As a step towards understanding strategies, a simple business environment is presented in which the decision-maker attempts to build their constellation and gain customers. The decision-maker must take actions to maximize profits from virtual customers who subscribe to their satellite internet service according to a deterministic utility function. Such a simplification avoids the computational complexity of managing the preferences of potentially hundreds of thousands of customers. Therefore, the evaluation of whether to purchase the satellite internet service from the company is modeled as a simple binary decision for the entire population of available customers. This decision is based on constellation performance (approximate values for data throughput, average coverage gap, and average latency) and price for satellite internet services. Satellite quality metrics are fixed throughout the environment for all constellations. This simplifies computation of figures of merit. Additionally, launches are available at each epoch, assuming that the decision-maker has enough funds for the purchase. This yields an instantaneous buy-and-launch mechanism that is not seen in the real world due to safety, availability, and scheduling; however, we make this simplifying assumption as a means to abstract out these concerns. Such an assumption also abstracts any potential inventory management dynamics that would inevitably occur. Figure 3.1 shows a visualization of this problem.

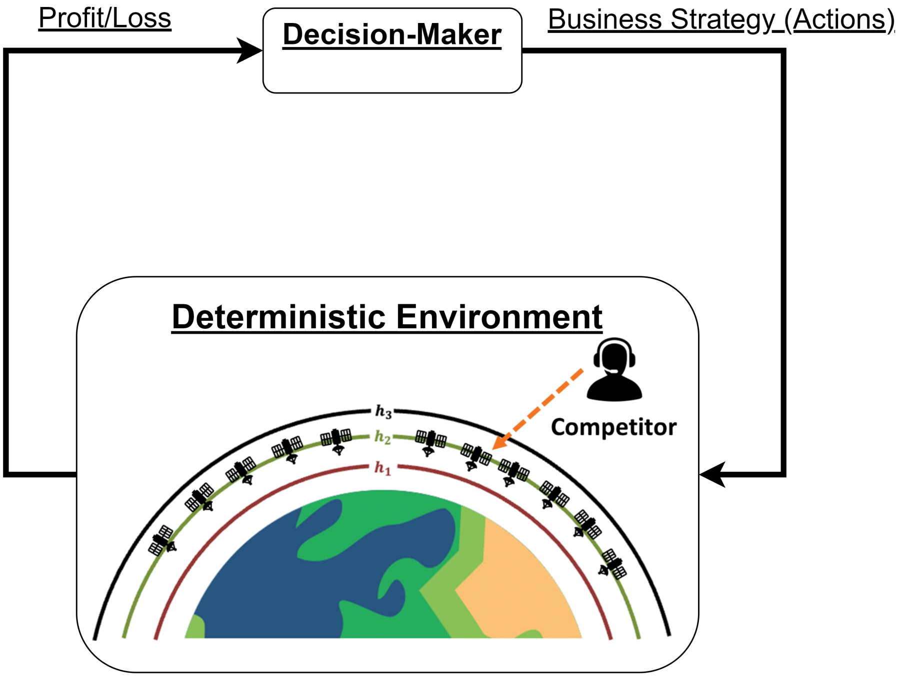

Figure 3.1: SATCOM Grid World with Deterministic Functionality

Competition within this environment is modeled as a single, static competitor with a constellation already in orbit. This competitor makes no actions or adjustments during the course of the simulation. Although impractical, modeling such a competitor gives a qualitative indication of what strategies might prove useful to new operators attempting to compete with an established aerospace business. A baseline static competitors also informs the qualitative relationship between the initial amount of funding and time horizon required before the decision-maker becomes profitable.

At the beginning of the simulation, the decision-maker has initial funds, f_{init}, which can be used to purchase and launch satellites into specific LEO orbit shells to expand their constellation network. Launching to each orbit shell has a fixed cost c_{launch_i}, where i represents the relative altitude of each orbit shell. As altitude increases, the cost to launch to that orbit shell also increases. The relative average latency of the system is directly proportional to the average altitude of the constellation, and the relative average coverage gap of the system is inversely proportional to the average altitude of the constellation. The decision-maker must also specify a price p, for their satellite internet services offered to virtual customers.

As the simulation progresses, customers recompute their decision of internet service provider at each epoch and revenues are given to the decision-maker if the customers have chosen to purchase their services. If satellites are launched during an epoch, the cost for each satellite launch is subtracted from the decision-maker's funds. Developing the SATCOM resource management problem as a sequential decision-making problem allows us to construct the problem using known techniques, albeit with some implementation limitations. A common method of solving such sequential decision-making problems may be found in optimal control theory. Specifically, recursively solving the Bellman optimality equations and the subsequent Backward Dynamic Programming Algorithm are used to model and solve for an optimal strategy.

3.1 Dynamic Programming (DP)

3.1.1 Deterministic, Dynamic Programming

The term *dynamic programming* was introduced by Bellman to study optimization problems which involve sequential decision-making [58]. A deterministic dynamic program (DP) may define any optimal control problem with the general structure of a particular dynamical environment. Specifically, this structure must have two key features: **(1)** a dynamic system with discrete epochs, and **(2)** an additive cost function over these discrete epochs [59]. A general dynamic program has several components which will be described in detail below. Note that notation in this chapter is taken from Bather Ch 2.2 [60].

Time Horizon (T):

The horizon, T, represents the number of decision epochs in the problem. For a finite-horizon DP, decision epochs are indexed by $t = 0, 1, ..., T$. These epochs are often referred to as "periods" or "stages" in different literature. Decisions in a DP are made in epochs 0 through $T - 1$ with no decision being made in the terminal epoch, T.

States and State Space (S):

At each decision epoch, the system is said to occupy a single state. We define that state at time

t as s_t and the set of all potential states as the State Space, S. The state space S is typically finite, but in other cases it may just be a countable set. However, it can also be a subset of a Borel set of a complete, metric space.

Actions and Action Space (A):

At each epoch, the decision maker must take an action, also known as a decision or control, which can be represented by a_t. The actions available to the decision-maker at epoch t and state s_t are denoted by the set of all feasible actions $A_t(s_t)$. We denote the union of all such sets as A.

Cost Function (C):

At each epoch, the system incurs a cost $c_t(s_t, a_t)$ that is a function of the current state, the action taken at this epoch, and, possibly, the epoch itself. Additionally, a terminal cost, $c_T(s_T)$, can be defined as the cost of being in a given state at the end of the horizon. Note that this cost may be zero. The collection of all costs can be denoted as C.

Law of Motion (L):

The evolution of the system from state s_t to s_{t+1} by taking actions a_t is described by a function known as the law of motion, L_t. This function is defined as:

$$s_{t+1} = L_t(s_t, a_t) \tag{3.1}$$

Such a function can either be explicitly or implicitly defined.

Decision Rules (d):

A decision rule, d_t, is defined as a function, that maps a given state at a particular epoch, s_t, to a specific action:

$$a_t = d_t(s_t) \tag{3.2}$$

Policies (π):

A policy, π, (also called a strategy or control profile) is a collection of decision rules, where there is a unique decision rule for each epoch:

$$\pi = \{d_0, d_1, ..., d_{T-1}\} \tag{3.3}$$

The set of all possible policies is denoted as Π.

Formulation of a Dynamic Program

A specific dynamic program can be defined by the quintuple $\{T, S, A, C, L\}$. The goal of dynamic programming is to find an optimal policy, π^*, which minimizes the total cost:

$$\pi^* = \arg\min_{\pi\in\Pi} \left\{ c_T\left(s_T\right) + \sum_{t=0}^{T-1} c_t\left(s_t, d_t\left(s_t\right)\right) \right\} \tag{3.4}$$

where $\pi^* = \left\{d_0^*, d_1^*, \ldots, d_{T-1}^*\right\}$, and $s_{t+1} = L_t(s_t, a_t)$. The core of dynamic programming is rooted in Bellman's Principle of Optimality, which proves the existence of such an optimal policy:

Theorem 3.1. *(Bellman Principle of Optimality) Let $\pi^* = \{d_0^*, d_1^*, ..., d_{T-1}^*\}$, be an optimal policy for a DP defined by $\{T, S, A, C, L\}$. Assume that using π^* you would arrive at state s_t at time t. Now consider the sub-problem whereby at s_t at time t, with a horizon of $T - t$ periods remaining, you wish to minimize the cost-to-go $J_t(s_t)$, over all possible policies, for the remaining horizon $\pi(t) = \{d_t, d_{t+1}, ..., d_{T-1}\}$. That is:*

$$J_t^{\pi(t)}\left(s_t\right) = c_T\left(s_T\right) + \sum_{k=t}^{T-1} c_k\left(s_k, d_k\left(s_k\right)\right) \tag{3.5}$$

Then, the truncated policy $\pi^*(t) = \{d_t^*, d_{t+1}^*, ..., d_{T-1}^*\}$ is optimal for the sub-problem.

Proof. See Appendix A. □

Given this theorem, Algorithm 1 naturally emerges and gives a manner of computing an optimal policy through the use of induction by proceeding "backwards in time". Indeed, this algorithm is often called the Backward Induction Algorithm or the Backward Dynamic Programming Algorithm:

Algorithm 1 The Backwards Dynamic Programming Algorithm

1: **procedure** DYNAMIC PROGRAMMING ALGORITHM

2: Set $J_T(s_T) = c_T(s_T)$ and set $t = T - 1$

3: For each $s \in S$, suppose $s_t = s$ and calculate:

$$J_t\left(s_t\right) = \min_{a_t \in A_t(s_t)} \left\{ c_t\left(s_t, a_t\right) + J_{t+1}\left(L_t\left(s_t, a_t\right)\right) \right\} \tag{3.6}$$

4: If $t = 0$, STOP, else set $t = t - 1$ and GO TO (3)

5: **end procedure**

Using Algorithm 1, we are able to prove the existence of an optimal policy for any correctly defined discrete, deterministic dynamic program:

Theorem 3.2. *Consider a DP defined by* $\{T, S, A, C, L\}$*. Then for every possible initial state* s_0*, the optimal cost* $J^*(s_0)$ *is equal to* $J_0(s_0)$ *given by the last step of the Backwards Dynamic Programming Algorithm, which proceeds backwards in time from period* T *to period* 0*.*

Furthermore, if you define d_t^* *in such a way that* $a_t^* = d_t^*(s_t)$ *is a minimizer in line (3) of the algorithm, for every t and* s_t*, then the policy* $\pi^*(t) = \{d_t^*, d_{t+1}^*, ..., d_{T-1}^*\}$ *is optimal for DP.*

Proof. See Appendix A. □

3.1.2 Formulation of the Grid World Environment

Using the above nomenclature and framework, we formulate a grid world version of our SATCOM management problem as a dynamic program defined by the following tuple: $\{T, S, A, C, L\}$. The rest of this section defines each of these elements for our particular resource management environment. We then use Algorithm 1 to solve for an optimal policy within the environment and further explore the solution space.

Time Horizon (T):

The environment formulation utilizes a finite-horizon in which t denotes as an index of the following set: $0, 1, ..., T$ where T is the time horizon (or length of simulation time). For the environment simulation, each epoch t represents a month of simulation time; therefore, the decision-maker is allowed to select business decisions on a monthly basis.

States and State Space (S):

The state of the system at a specific epoch, t, may be represented as the following vector:

$$\vec{s_t} = \begin{bmatrix} funds \\ price \\ n_1 \\ n_2 \\ n_3 \end{bmatrix} \tag{3.7}$$

where $funds$ is the funding available to the decision-maker for purchases, $price$ is the price of internet services set by the decision-maker, and n_i is the number of satellites in orbit shell i. Note that each element of the state vector is a discrete value; therefore, the overall state space, S, can be represented as a discrete combination of all possible values of the above state vector. Additionally, note that $\vec{s}$ grows with additional orbit shells or constellation detail. Due to this potentially long vector, the problem can suffer from the curse of dimensionality (refer to Section 3.1.3 for more information).

Actions and Action Space (A):

An action available during epoch t, at state $\vec{s_t}$, $\vec{a}(s_t)$, may be represented by the following vector:

$$\vec{a}(s_t) = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \\ p_{Change} \end{bmatrix} \tag{3.8}$$

where l_i is the amount of satellites launched at epoch t to orbit shell i, and p_{Change} is the change to the decision-maker's price of broadband service offered to virtual customers. Note that launches to multiple orbit shells are allowed in an epoch in this formulation. The rationale behind this modeling decision is that SpaceX launches multiple payloads of its Starlink constellation into different orbit shells per month [61]. The action space of this environment, A, is therefore represented by all feasible combinations of the above action vector.

Cost Function (C):

The costs generated by this deterministic environment at epoch t, by taking action $\vec{a}_t$, while in state $\vec{s}_t$, and going to state $\vec{j}$ in epoch $t+1$ may be written as:

$$c_t(\vec{s}_t, \vec{a}_t) = c_t(\vec{j}_{t+1}, \vec{s}_t, \vec{a}_t) \tag{3.9}$$

To model the financial implications of time-delay and limit the number of duplicate strategies which produce similar overall returns, the deterministic cost function is formulated as the cash flow discounted to the beginning of the time-horizon:

$$c_t(\vec{j}_{t+1}, \vec{s}_t, \vec{a}_t) = -PV_t = \frac{-CF_t}{(1+R)^t} \tag{3.10}$$

where PV_t is the present value of cash flow at epoch t, CF_t is the cash flow at epoch t, and R is a fixed interest rate throughout the simulation. In the realm of corporate finances, cash flow can be defined as simply all the revenues and costs at a particular epoch or between a specified time interval; therefore, the cash flow, at each epoch, can be described as:

$$CF_t = r_{revenue} - c_{launch} - c_{CAM} \tag{3.11}$$

where $r_{revenue}$ is defined as the revenues gained from virtual customers, c_{launch} is defined as the total cost of launches conducted during the epoch, and c_{CAM} is the total cost of collision avoidance maneuvers required during the epoch.

Monthly revenues generated from providing broadband services are modeled as:

$$r_{revenue} = \begin{cases} price \cdot n_{population} & \text{if } U(\vec{s}_t) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.12}$$

where $price$ is the decision-maker's broadband service price at epoch t, $n_{population}$ is the total number of virtual customers in the simulation, and U is a utility function that computes a binary response based on the current state of the environment (this includes the static competitor's constellation as well). The utility function is defined as:

$$U(\vec{s}_t) = \begin{cases} 1 & \text{if } (price > p') \wedge (f(\vec{s}_t) \geq f') \wedge (g(\vec{s}_t) \geq g') \wedge (h(\vec{s}_t) \geq h') \\ 1 & \text{if } (price \geq p') \wedge (f(\vec{s}_t) > f') \wedge (g(\vec{s}_t) \geq g') \wedge (h(\vec{s}_t) \geq h') \\ 1 & \text{if } (price \geq p') \wedge (f(\vec{s}_t) \geq f') \wedge (g(\vec{s}_t) > g') \wedge (h(\vec{s}_t) \geq h') \\ 1 & \text{if } (price \geq p') \wedge (f(\vec{s}_t) \geq f') \wedge (g(\vec{s}_t) \geq g') \wedge (h(\vec{s}_t) > h') \\ 0 & \text{otherwise} \end{cases} \tag{3.13}$$

where f, g, and h are functions for constellation figures of merit which approximate total system throughput, mean latency, and mean coverage gap, respectively. The competitor's broadband service price, total system throughput, mean latency, and mean coverage gap are described by p', f', g', and h', respectively. The competitor's constellation figures of merit are approximated using their static constellation design and the same functions used for the decision-maker's constellation; however, the price p' is a fixed value throughout the simulation. Note that the above utility function has been crafted such that the decision-maker only receives customers if their constellation performance and price of service are as appealing (or more appealing) than the competition's. This guarantees that the decision-maker is gaining customers from a known, explainable process.

The computation of system throughput for a P-LEO constellation, with potentially thousands of individual satellites, over a given region with potentially hundreds of thousands of user terminals, at a specific time of day, is a computationally intensive problem in itself; therefore, we approximate the total system throughput as:

$$f(\vec{s}_t) \approx R_B \sum_{i=1}^{3} n_i \tag{3.14}$$

where R_B represents the maximum data rate of the modeled satellite and n_i represents the total number of satellites within orbit shell i.

The latency of a satellite constellation is defined as the lag in time between a data request and the start of that data transfer [24]. Although latency exists in all networks, it is further exacerbated in satellite internet networks due to additional delays of data transfers between the ground and space segments. Each of these added delays is a function of the distance between terminals and satellites, and the speed of light. Having considered these limitations and the system model, we approximate a constellation's latency as a linear function:

$$g(\vec{s}_t) \approx l_{base} \cdot R_{avg} \tag{3.15}$$

where l_{base} is a baseline latency value which is multiplied by the average altitude of the constellation, R_{avg}. This average altitude can be defined using the number of satellites in each orbit shell of the decision-maker's constellation:

$$R_{avg} = \frac{\sum_{i=1}^{3} i \cdot n_i}{\sum_{i=1}^{3} n_i} \tag{3.16}$$

Coverage gaps are defined as the time in which satellites (and therefore internet services) are not available to specific ground segments [62]. The mean coverage gap can then be defined as the average length of breaks in coverage for a given user terminal. In the case of our SATCOM framework, such a figure cannot be computed accurately with the given state information, so this is also an approximation using the average constellation altitude:

$$h(\vec{s}_t) \approx \frac{1}{1 + R_{avg}} \tag{3.17}$$

In both functions g and h, the average altitude, R_{avg}, is used as a surrogate variable to approximate the *relative* performance of constellations within the environment. This is due to three factors:

1. A state vector with more detailed constellation design information would make the discrete dynamic program computationally infeasible (see Section 3.1.3 for details).

2. If detailed constellation data were available, the traditional computation of complete global coverage for LEO constellations is typically given over an average day [63] and would take an infeasible amount of time to compute for each epoch.

3. Since customer behavior in this environment can be interpreted as zero-sum, specific constellation performance is not required; only *relative* performance is needed to assign customers to the superior service provider.

In the case of relatively low altitudes, latency is minimized, but mean coverage gap is maximized. Alternatively, higher altitudes minimize mean coverage gap but maximize latency. Such a trade-off is a delicate balance in the design of P-LEO constellations.

For the sake of simplicity, the cost of manufacturing, storing, and launching satellites has been factored into a single cost function. Such a function computes the total cost at each epoch as:

$$c_{launch} = c_{L1} + c_{L2} + c_{L3} \tag{3.18}$$

where c_{Li} is defined as the cost of satellites to be launched to orbit shell i. We further define these costs as a function of the state of the system and the action executed:

$$c_{Li}(\vec{s}_t, \vec{a}_t) = \begin{cases} c_i & \text{if } (l_i > 0) \wedge (funds \geq c_i) \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in \{1, 2, 3\} \tag{3.19}$$

where c_i is a fixed cost to launch to a particular orbit shell and l_i (the amount of satellites to launch to the orbit shell i) is a non-zero launch order by the decision-maker. This formulation also guarantees the decision-maker has enough funds to pay for the total cost of launches ordered. Such conditions are required to capture the economic constraints of the environment.

Once satellites are in orbit, they must be monitored, maintained, and protected from resident space objects and other P-LEO constellations; therefore, we model our final cost as a collision avoidance maneuver (CAM) cost. For simplicity, CAM costs are simplified and modeled as:

$$c_{CAM} = c_{CAM_1} + c_{CAM_2} + c_{CAM_3} \tag{3.20}$$

where c_{CAM_i} represents the cost of performing a critical collision avoidance maneuver in orbit shell i at the current epoch. Each CAM cost can be computed based on the state of the system and the competitor's static constellation design:

$$c_{CAM_i}(\vec{s}_t) = \begin{cases} (n_i + n_i')/(n_{max}) & \text{if } (n_i > 0) \wedge (n_i' > 0) \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in \{1, 2, 3\} \tag{3.21}$$

where n_i are the satellites owned by the decision-maker and n_i' are the satellites owned by the static competitor, at epoch t, in orbit shell i. The maximum number of satellites the decision-maker can launch to any orbit shell is also defined as n_{max} and used to normalize CAM costs. Note that this formulation ensures that CAM costs are only applied when both competitors occupy the same orbit shell; otherwise, there is no need for collision avoidance and CAM costs are computed to be zero.

The final component of the cost function, known as the terminal cost, c_T, is produced upon reaching the end of the finite-horizon. Although our framework is a finite-horizon, we conclude the formulation of the dynamic program by setting the terminal cost to zero (i.e. the decision-maker does not receive any revenues or incur any costs for reaching the terminal state):

$$c_T(\vec{s}_T) = 0 \tag{3.22}$$

Law of Motion (L):

The law of motion of the environment bounds the state space from entering infeasible states and applies additional economic, business, and constellation constraints to the decision-maker's action space. Since the state of the system, s_t, is a vector, the law of motion, L, may be separated into unique laws of motion for the update of each variable within the state vector:

$$\vec{s}_{t+1} = \begin{bmatrix} funds_{t+1} = L_1(\vec{s}_t, \vec{a}_t) \\ price_{t+1} = L_2(\vec{s}_t, \vec{a}_t) \\ n_{1_{t+1}} = L_3(\vec{s}_t, \vec{a}_t) \\ n_{2_{t+1}} = L_3(\vec{s}_t, \vec{a}_t) \\ n_{3_{t+1}} = L_3(\vec{s}_t, \vec{a}_t) \end{bmatrix} \tag{3.23}$$

in which $L_i(\vec{s}_t, \vec{a}_t)$ is a specific law of motion for updating a particular variable within the state vector. The update of the decision-maker's funds is formulated as:

$$L_1(\vec{s}_t, \vec{a}_t) = funds_t + CF_t \tag{3.24}$$

where CF_t is the cash flow at time t and L_1 is an accounting update to the decision-maker's funds for the next epoch, given costs and revenues computed in the current epoch. The price of broadband service may also vary based on the decision-maker's input:

$$L_2(\vec{s}_t, \vec{a}_t) = \begin{cases} price_t + p_{Change} & \text{if } p_{min} \leq (price_t + p_{Change}) \leq p_{max} \\ price_t & \text{otherwise} \end{cases} \tag{3.25}$$

where $price$ is a variable in the state vector, p_{Change} is a variable in the action vector, and p_{min} and p_{max} are the minimum and maximum possible offering prices, respectively. Such dynamics bound the price of service to a feasible set, therefore, limiting the state space that needs to be explored. The number of satellites in orbit shell i are updated according to the remaining law of motion:

$$L_3(\vec{s}_t, \vec{a}_t) = \begin{cases} n_i + l_i & \text{if } (funds_t \geq c_{launch}) \wedge ((n_i + l_i) < n_{max}) \\ n_i & \text{otherwise} \end{cases} \quad \forall i \in \{1, 2, 3\} \tag{3.26}$$

where n_i are the satellites owned by the decision-maker at orbit shell i, l_i are the satellites requested by the decision-maker to orbit shell i, and n_{max} is a set value for the maximum number of satellites the decision-maker can launch to a particular orbit shell. Note that this update not only ensures that satellites are never removed from a constellation but also bounds satellite launches by economic and computational constraints. The economic constraint is simply the decision-maker's ability to afford more satellites. The computational constraint is one which bounds the number of satellites in any particular orbit shell to a feasible maximum to computationally limit the state space needing to be explored.

Given the above definitions of each component, the discrete, deterministic dynamic program formulation of the environment is complete. It may be noted that if the cost function and law of motion are replaced with a reward function and state transition probabilities, this formulation is very similar to that of a discrete stochastic dynamic program, also known as a Markov Decision Process (MDP) [58]. Indeed, deterministic dynamic programs are simply a class of MDPs in which state transition probabilities are either 1 or 0.

Although it has many benefits, the primary appeal of formulating an environment as an MDP is the existence of known solution methods which yield policies for infinite-horizon, stochastic environments. These algorithms further exploit the Bellman Principle of Optimality and are called Value Iteration Algorithm (VIA) and Policy Iteration Algorithm (PIA) [59]. However, the class of environments we investigate are unable to utilize these methods due to the most common issue with such problem formulations: the curse of dimensionality.

3.1.3 Curse of Dimensionality

The curse of dimensionality (also coined by Bellman [58]) refers to the increasing difficulty of solving even a simple system of linear equations when that number of equations and the associated design variables grow to be very large. Computationally, this is often the limitation of implementing MDPs; as the number of state variables in our state vector, $\vec{s}$, increase, so do the possible number of states to be explored. This also holds true for action variables within action vector $\vec{a}$. Additionally, for discrete-state MDPs, discretization of each state and action variable itself also plays a role in determining how many state-action pairs need to be explored. If a small discretization value is selected, the number of possible pairs will be very large and lead to state space explosion. If this occurs, dynamic programming-based solution methods such as VIA or PIA are no longer viable solution methods. Due to the multivariate nature of the dynamic program formulated above, discretization of each state and action variable has been carefully chosen to give meaningful, *qualitative* results while avoiding this computational hurdle.

Although optimal policy development with large state spaces has been explored using level-set functions [64], these methods often require additional constraints on the MDP, such as a partially constrained final state. Additional works have also attempted to combine constraints in the DP algorithm as a single surrogate constraint to ease and shrink the solution space [65]; however, this optimization is heavily model-dependant.

3.2 Framework Implementation

The deterministic dynamic program detailed above was implemented in MATLAB using the DynaProg [66] discrete dynamic programming solver. The dynamics presented above are modeled as a MATLAB function which is then passed into the DynaProg solver with discretized state and action spaces. The Backward Dynamic Programming Algorithm is then utilized to evaluate each possible combination of state-action pairs throughout the time-horizon of the simulation.

As with all discrete solvers, the state space and action space must both be discretized. Therefore, we present the discretization of each variable in the state and action vectors in Table 3.1 and Table 3.2, respectively. Note that although this discretization is relatively coarse, the number of possible states is the number of possible combinations of each state variable. For the given discretization, the total number of feasible states is 750,000; indeed, further discretization would exponentially increase this value. This is due to the curse of dimensionality and, therefore, the exponential increase of computations needed by the Backward Dynamic Programming Algorithm.

Coupled with Table 3.1 and Table 3.2, the parameters defined in Table 3.3 complete our numerical formulation of the dynamic program. This standard dynamic program will be further modified to investigate coupling between action variables.

Table 3.1: Discretization of state space

State Variable	Min Value	Max Value	Discretization
funds ($ m)	0	5000	Steps of 100
price ($)	0	150	Steps of 10
n_1 (satellites)	0	1000	Steps of 100
n_2 (satellites)	0	1000	Steps of 100
n_3 (satellites)	0	1000	Steps of 100

Table 3.2: Discretization of action space

Action Variable	Possible Values
l_1 (satellites)	$\{0, 100\}$
l_2 (satellites)	$\{0, 100\}$
l_3 (satellites)	$\{0, 100\}$
p_{Change} ($)	$\{-10, 0, 10\}$

Using the values given in Table 3.3 along with the assumption that a single competitor exists and has a constellation of 1000 satellites in the h_2 orbit shell, we are able to solve the above DP formulation and produce an optimal policy for the simple constellation management problem. Additionally, we assume that virtual customers will want to minimize price, pseudo-coverage gap, and latency, while maximizing total system throughput. Using the following initial conditions:

$$\vec{s_0} = \begin{bmatrix} funds = 1000 \\ price = 0 \\ n_1 = 0 \\ n_2 = 0 \\ n_3 = 0 \end{bmatrix} \tag{3.27}$$

we solve the dynamic program and visualize the control signal corresponding to the optimal policy in Figure 3.2 and the associated state evolution in Figure 3.3.

Interpreting the results of this nominal DP, a few general outcomes are noted:

- When computing the cost-to-go from Equation (3.5), using the formulation of present value cash flow from Equation (3.10), the function J_0 is equivalent to the net present

Table 3.3: Parameters defining the MDP

Parameter	Value	Units
T	300	months
p_{min}	0	\$
p_{max}	150	\$
R	7	%
n_{max}	1,000	satellites
$n_{customers}$	750,000	-
c_1	30	\$ m
c_2	60	\$ m
c_3	90	\$ m
R_B	5	Gbps
l_{base}	100	ms
n'_1	0	satellites
n'_2	1000	satellites
n'_3	0	satellites
p'	100	\$
f'	5	Tbps
g'	200	ms
h'	1/3	-

value (NPV) of executing a particular investment strategy in the SATCOM industry:

$$\pi^* = \arg\min_{\pi\in\Pi} \left\{ c_T\left(s_T\right) + \sum_{t=0}^{T-1} c_t\left(s_t, d_t\left(s_t\right)\right) \right\} = \arg\min_{\pi\in\Pi} \left\{ 0 + \sum_{t=0}^{T-1} -PV_t \right\} \quad (3.28)$$

$$\pi^* = \arg\min_{\pi\in\Pi} \left\{ -\sum_{t=0}^{T-1} PV_t \right\} = \arg\min_{\pi\in\Pi} \left\{ -\sum_{t=0}^{T-1} \frac{CF_t}{(1+R)^t} \right\} = \arg\max_{\pi\in\Pi} \left\{ NPV_\pi \right\} \quad (3.29)$$

Specifically, according to Theorem 3.2, this is an ***optimal*** development strategy that is quantifiable using either the NPV or cost-to-go.

- The optimal strategy which generates the control profile shown in Figure 3.2 indicates that launching to h_2 still remains a viable option due to the decision-maker's rapid increase of profits.

- There exists an artificial cap on the $funds$ state variable up to \$5 billion. Since the decision-maker reaches this cap via backwards induction and the optimal strategy computed includes CAM costs due to operating in orbit shell h_2, slight decreases of $funds$

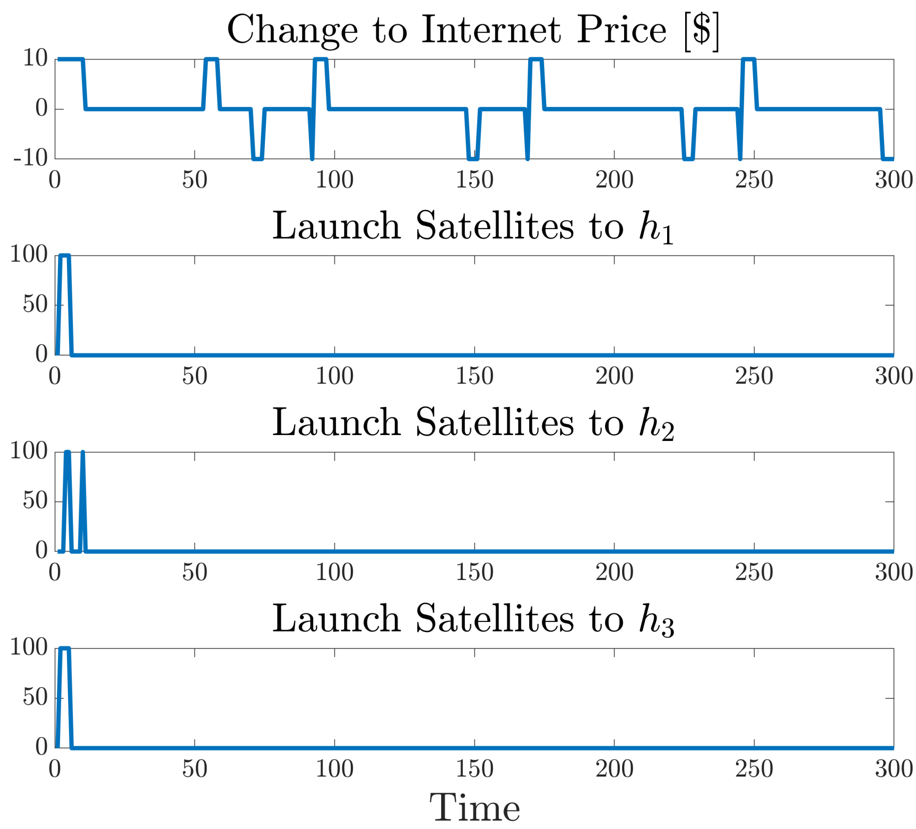

Figure 3.2: Control Profile Corresponding to an Optimal Policy obtained from DP

are shown when the maximum Funds cap is achieved. This caused an oscillatory behavior in the decision-maker's internet pricing as it attempts to regain the lost funds as quickly as possible. Therefore, we may conclude the following: the oscillatory behavior demonstrates a limitation of dynamic programming, and there exists strong coupling between CAM costs, $funds$, and the $price$ variables.

- Using the above insight, the maximum funds of the dynamic program may be interpreted as a "target" value which the decision-maker would like to achieve and maintain within the given time horizon. Note that is an artifact of a finite-horizon (which can be interpreted as a finite runway to achieve a target value); in cases of an infinite-horizon, this would not be possible under the current formulation.

Finally, given the state trajectory computed using an optimal policy obtained from DP, it is possible to evaluate traditional economic figures of merit for such a policy. Specifically, using the $funds$ state variable at each epoch, we are able to approximate the total profits over the

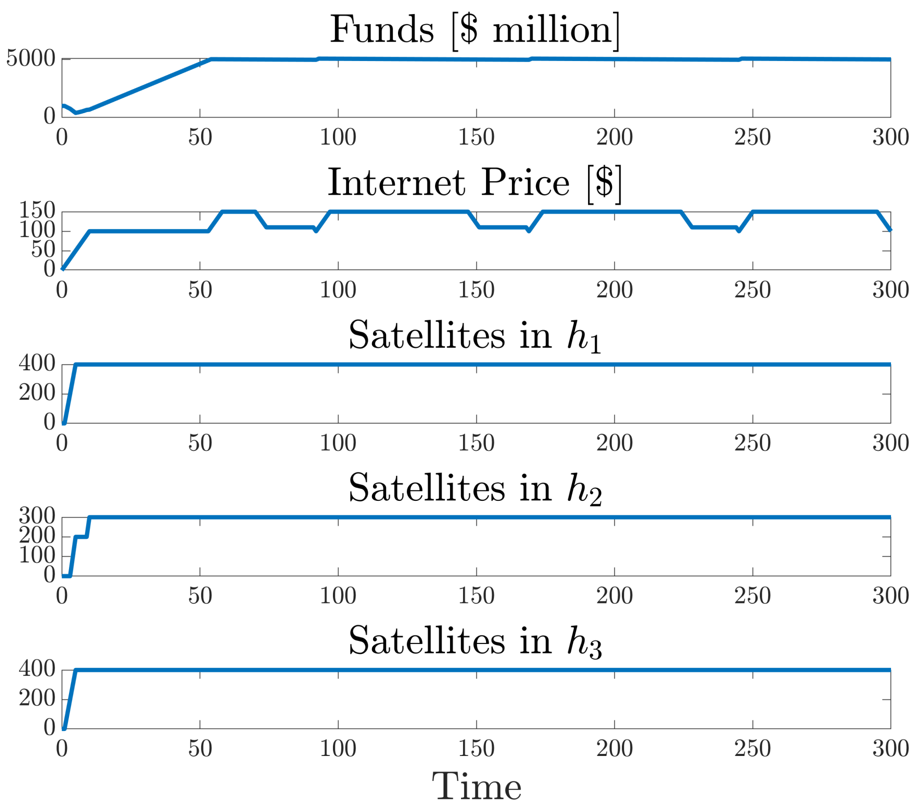

Figure 3.3: State Evolution of System Corresponding to an Optimal Policy obtained from DP

time-horizon (P_{total}), the time-to-profit (TTP), the compound monthly growth rate($CMGR$), and the compound annual growth rate($CAGR$) [67].

Total profits over the time horizon are computed as the difference between the initial funds and the NPV of investment:

$$P_{total} = NPV_T - funds_0 \tag{3.30}$$

Time-to-profit is computed simply as the epoch in which the agent's state of funds crossed above the initial state of funds:

$$TTP = \min_t \{t \mid funds_t > funds_0\} \tag{3.31}$$

If TTP returns N/A, then a profit was not successfully returned. Although economic investments are typically considered on a yearly basis, our simulation epochs are monthly. Therefore, the compound monthly growth rate, given as a percentage, is computed as:

$$CMGR = \left(\left(\frac{funds_T}{funds_0} \right)^{\frac{1}{T}} - 1 \right) \times 100 \tag{3.32}$$

To supplement the $CMGR$, we also compute the compound annual growth rate and give it as a percentage:

$$CAGR = \left(\left(\frac{funds_Y}{funds_0} \right)^{\frac{1}{Y}} - 1 \right) \times 100 \tag{3.33}$$

where

$$Y = \frac{T}{12} \tag{3.34}$$

Table 3.4 gives a consolidated review of the resulting figures of merit for the above formulation. Although these are economic figures are based on the decision-maker's trajectory, the underlying equations describe convenient performance metrics that can be used to measure the economic impact of any policy obtained from DP. Note that units of [(t_0) $ m] represent value of funds in month t_0 dollars, in millions. This is typically done in economics as a means of discounting future costs and rewards for investment strategy purposes. Additionally, we note that total profits are shown to be a negative value. This is due to the relative value computed using the net present value. We note that a policy in which no investment action is taken would result in a net present value of zero, and therefore, a total profit of -1000 due to the time value of money. While a negative value for total profits is not ideal, the Bellman optimality equation shows that this is the most optimal net present value.

Table 3.4: Consolidated Figures of Merit describing performance of an optimal policy

Figure of Merit	**Value**	**Units**
NPV_{π^*}	194.4139	(t_0) $ m
P_{total}	-805.5861	(t_0) $ m
TTP	17	months
$CMGR$	0.5388	%
$CAGR$	6.6608	%

3.3 Variation of Scenarios & Parameters

To systematically assess and understand state-action couplings associated with the economic performance metrics, the basic scenario presented in the previous section was altered according to a variation of scenarios which includes: simulation design variables (Table 3.5), data rates (Table 3.6), and competitor's constellation design (Table 3.7). Note that units for each economic figure of merit presented in these tables are identical to those presented in Table 3.4. Each row of these table corresponds to a distinct scenario with an altered parameter within the environment. Each new environment is then solved using the Backward Dynamic Programming Algorithm to obtain an optimal policy. This variation of scenarios allows us to measure and assess the economic figures of merit of different optimal policies against different scenarios to find common trends and couplings within the environment.

Table 3.5: Variation of Simulation Design Parameters

Variation	**Parameter**	NPV_{π^*}	P_{total}	TTP	$CMGR$	$CAGR$
Epochs	100 months	194.4813	-805.5187	17	1.6322	21.4444
Epochs	200 months	194.4139	-805.5861	17	0.8073	10.1294
Epochs	300 months	194.4139	-805.5861	17	0.5388	6.6608
Epochs	400 months	194.4139	-805.5861	17	0.4048	4.9677
Customers	500,000	0	-1000	N/A	0	0
Customers	750,000	194.4139	-805.5861	17	0.5388	6.6608
Customers	1,000,000	403.8492	-596.1508	14	0.5383	6.6545
Customers	1,500,000	773.0641	-226.9359	11	0.5364	6.6307
Competitor Price	\$50	0	-1000	N/A	0	0
Competitor Price	\$100	194.4139	-805.5861	17	0.5388	6.6608
Competitor Price	\$150	400.7740	-599.2260	16	0.5403	6.6804
Competitor Price	\$200	422.7023	-577.2977	15	0.5402	6.6791

We note that results in Table 3.5 show a few trends relating simulation parameters to the economic FOMs via the optimal policy. Most importantly, it is shown that if there are too few customers in the environment (500,000 in this case), the NPV_{π^*} will be zero. This corresponds to an optimal strategy of not investing in a P-LEO constellation. Although surprising, this strategy becomes intuitive when considering that the profits from customers will not be sufficient to recuperate the initial investment in constellation components (when factoring in discounting of future cash flow). This phenomenon also occurs when the competitor's offering price is too low to compete with ($50 in this case). If the decision-maker were to match this price or undercut it, the resulting profits would again not be enough to recuperate the initial constellation investment. In both cases, there seems to be an inflection point at which constellation development becomes profitable. Additionally, as the number of epochs increases, both the $CMGR$ and $CAGR$ decrease. While $funds_0$ and $funds_T$ are constant throughout the variation, the increase in simulation length causes the compounding growth rate to be distributed across a longer time-horizon. Finally, we note that as customers or the competitor's offering price increase, so does the NPV_{π^*}. In the case of customers, this is due to a larger customer base that can be charged for services. In the case of service pricing, as the competitor raises their price, the decision-maker's optimal policy becomes to match or slightly under cut this new, higher price. In both cases, the increase in revenues causes both an increase in NPV_{π^*} and P_{total}, while decreasing the TTP.

Table 3.6: Variation of Data Rates to Simulate Satellite Technology Differences

Variation	**Parameter**	NPV_{π^*}	P_{total}	TTP	$CMGR$	$CAGR$
Decision-Maker Data Rate	1 Gbps	0	-1000	N/A	0	0
Decision-Maker Data Rate	2 Gbps	0	-1000	N/A	0	0
Decision-Maker Data Rate	5 Gbps	194.4139	-805.5861	17	0.5388	6.6608
Decision-Maker Data Rate	10 Gbps	454.8921	-545.1079	12	0.5381	6.6520
Competitor Data Rate	1 Gbps	627.7524	-372.2476	7	0.5400	6.6759
Competitor Data Rate	2 Gbps	511.9247	-488.0753	9	0.5410	6.6886
Competitor Data Rate	5 Gbps	194.4139	-805.5861	17	0.5388	6.6608
Competitor Data Rate	10 Gbps	0	-1000	N/A	0	0

In Table 3.6, variations of data rate are modeled to abstractly approximate differences in satellite technology between competitors. When reviewing these results, we note that if the decision-maker's data rate is too low *or* the competitor's data rate is too high, then the decision-maker's optimal action is to not invest in the SATCOM industry due to the large disparity amongst quality of service and, therefore, potential investment costs and associated revenues. This trend can also be observed going in the opposite direction: as the decision-maker's data rate increases *or* the competitor's data rate decreases, the NPV_{π^*} increases due to the technological advantage going to the decision-maker.

Table 3.7: Variation of Competitor's Static Constellation Design

Satellite Distribution	NPV_{π^*}	P_{total}	TTP	$CMGR$	$CAGR$
$n'_1 = \mathbf{1000}$, $n'_2 = 0$, $n'_3 = 0$	267.7238	-732.2762	15	0.5402	6.6784
$n'_1 = 0$, $n'_2 = \mathbf{1000}$, $n'_3 = 0$	194.4139	-805.5861	17	0.5388	6.6608
$n'_1 = 0$, $n'_2 = 0$, $n'_3 = \mathbf{1000}$	0	-1000	N/A	0	0
$n'_1 = 0$, $n'_2 = \mathbf{500}$, $n'_3 = \mathbf{500}$	34.7818	-965.2182	21	0.5412	6.6909
$n'_1 = \mathbf{500}$, $n'_2 = 0$, $n'_3 = \mathbf{500}$	193.6613	-806.3387	17	0.5378	6.6483
$n'_1 = \mathbf{500}$, $n'_2 = \mathbf{500}$, $n'_3 = 0$	278.2267	-721.7733	15	0.5388	6.6607
$n'_1 = \mathbf{333}$, $n'_2 = \mathbf{333}$, $n'_3 = \mathbf{333}$	221.7221	-778.2779	16	0.5390	6.6629

Finally, Table 3.7 shows perhaps the most interesting coupling, which is between the competitor's satellite distribution and the decision-maker's NPV_{π^*}. Specifically, if the competitor's satellites are all distributed in the same orbit shell, the higher the altitude of the competitor's constellation, it becomes less feasible for the decision-maker to invest in a competing constellation. This is due to the modeled increase in launch costs associated with higher orbit shells. This also holds true for cases in which the competitor's constellation is distributed between two or three orbit shells. However, the maximum NPV_{π^*} is achieved when the competitor's constellation is evenly distributed between h_1 and h_2. This is due to the cheaper launch costs associated with developing a constellation or equal or greater quality.

Chapter 4

Sat-Tycoon: A Gamified Environment

Through significant modeling expansion, we introduce a multi-agent environment dubbed Satellite Tycoon (Sat-Tycoon). This environment utilizes gamification techniques and a combination of computationally inexpensive models that were abstracted in the previous environment. Specifically, this environment includes a more precise constellation coverage algorithm, an evolved customer decision model, an expansion of financial modeling, and other modeling improvements. As in the previous environments, broad modeling assumptions for the environment are stated below:

A3.1 Profits/Costs regarding user terminal manufacturing and production are not considered. It is assumed that virtual customers already possess the relevant technologies to interact with any players' broadband services.

A3.2 All orbits are circular and without any orbital perturbations.

A3.3 Virtual customers are stateless. Therefore, broadband service decisions are recomputed at the beginning of every virtual month.

A3.4 Although satellites must now be purchased prior to launch, launches themselves are instantaneous. Satellites launched in a given epoch are active constellation components in the next epoch. Therefore, there is no setup or delay time modeled.

A3.5 This environment is considered stochastic due to the inclusion of multi-player dynamics (opponents' constellation development strategies may not be readily predictable) and a proposed collision probabilities model.

While other works have focused on the economic implications of space debris [36] and simulation gaming using existing general space mission design tools [41], this framework specifically investigates the P-LEO SATCOM economy focused on providing global broadband services. Our modeling framework definitions follow an existing approach to presenting framework designs [41]; however, our framework utilizes the novel approach of framing the economic environment using elements of classical gamification. Defining this economic system as a competitive resource management game allows us to exploit common gamification techniques for research applications. Primarily, this approach allows us to model multiple decision-makers simultaneously in an open-world environment as players to a real-time-strategy (RTS) simulation game. Such an environment allows for multi-player interactions while being computationally inexpensive enough to render real-time updates of player constellation performances. Tertiary benefits of gamification relate to an enhanced user experience and increased human player engagement with the framework.

Broadly speaking, gamification can be used to intuitively model and visualize complex interactions within multi-agent systems; however, modeling fidelity is typically inversely related to gamification [68], i.e., the more detailed a simulated system becomes, the less gamified the framework (see Figure 4.1 for examples). This holds for popular RTS simulation games, which have been used as training environments for artificial agents [69]. Sat-Tycoon plans to address this by constructing simplified models which are computationally cheaper while still yielding acceptable results in a fraction of the time a detailed simulation requires. For instance, models of orbital dynamics, virtual customer behavior, and corporate finances may all be used in conjunction with a player model to actualize the environment in which multiple player agents – human or artificial – can apply strategies to gain customers. A rudimentary player resource model is also developed to manage player assets and technologies.

The objective of each player in the environment is to maximize profits by making high-level system design and business decisions for their P-LEO constellation operations while expanding their footprint across the globe via technology improvements, launch frequency, and competitive pricing strategies. To accomplish this complex objective, players are able to take actions such as: purchasing satellites, building ground stations, launching satellites into orbit,

Figure 4.1: Qualitative Comparison of Playability and Modeling Fidelity for Various Popular Games

and dynamically changing the price they charge for internet services. Unlike the previous environments, Sat-Tycoon is capable of representing geographical data in which players can choose to offer broadband services to specific geographical regions. Due to the nature of the satellite internet marketplace, competition in Sat-Tycoon is modeled as indirect interactions between players in the environment. This modeling constraint allows us to consider the environment as a *competitive* game due to the decomposed constraint of virtual customers only being allowed to pay a single player for internet services at any given epoch.

At the beginning of each game (episode), players are allocated a specific amount of funding and can set up a limited constellation network using this funding before the game clock begins. This is known as the game initialization phase. Once all players have confirmed that they are ready to begin the game, they enter the main game dashboard that has all necessary controls. The game then progresses over a fixed time-horizon with players receiving revenues from their virtual customers at specific intervals throughout the game. Due to the long time-horizon nature of infrastructure economies [70], we implement the game at an accelerated rate of time, equating a single second of real-time to two days of game-time. This allows for long-horizon actions to be modeled and their effects to be visible at a reasonable time-scale. As shown in Figure 4.2, a prototype graphical user interface (GUI) of Sat-Tycoon has been developed. The game itself lives on a university server currently able to test multi-player interactions within a proof-of-concept prototype.

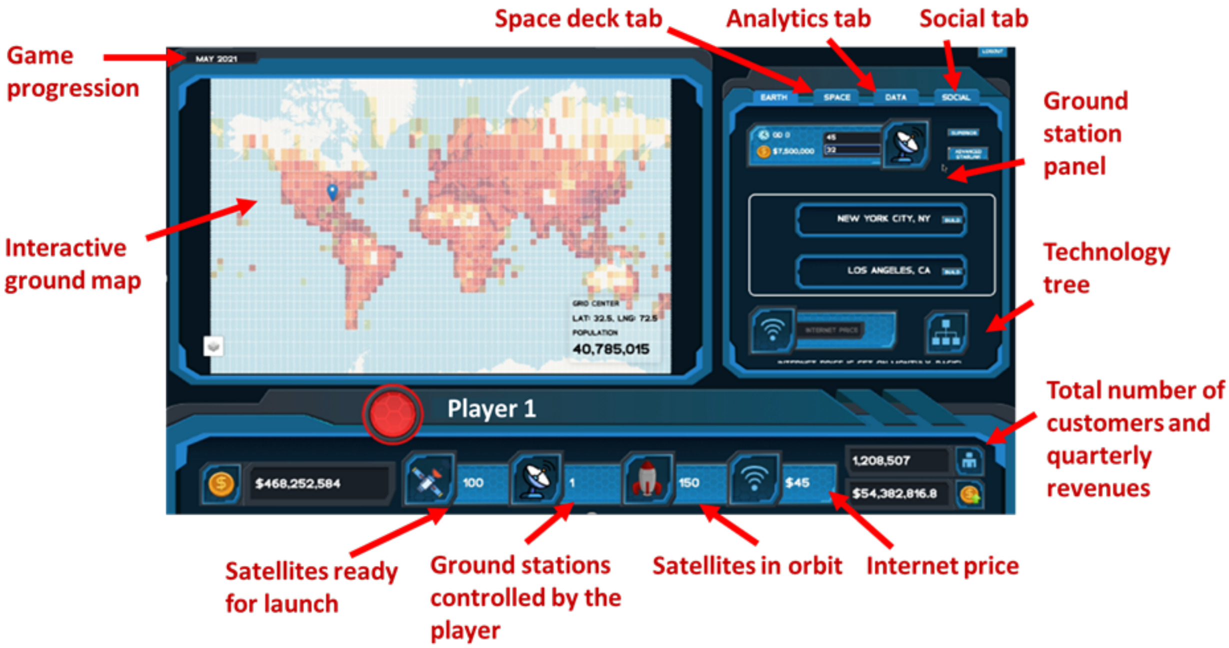

Figure 4.2: Prototype Graphical User Interface (GUI) of Sat-Tycoon

4.1 Environment Design & Modeling Process

The Sat-Tycoon framework is developed as a multi-player, strategy-simulation game in which players act as high-level constellation operators vying for virtual customers and subsequent revenues. The framework is capable of representing the strategic, temporal nature of satellite internet companies and competition associated with multi-player satellite internet marketplaces. Figure 4.3 describes the overall modeling components of Sat-Tycoon defined as an Agent-Environment system. There are six major modeling components that constitute the game mechanics of Sat-Tycoon:

1. *Player model*: defines the available actions a player can take and stores their player state at predefined epochs throughout the game

2. *Player Resource model*: defines the resources available for purchase, deployment, or upgrade

3. *Orbital Dynamics model*: defines the process by which a player's current resources translate to constellation figures of merit using both along-track and across-track coverage

4. *Collision model*: defines the process in which player constellations may be subjected to a collision event with space debris or other constellations

5. *Customer model*: defines the customer decision process for selecting ISPs by reformulating the customer decision as a multi-attribute decision-making problem

6. *Cost & Rewards model*: defines the non-recurring and recurring costs (both time and action-based), as well as a reward function based on each player's customer base and their internet service price

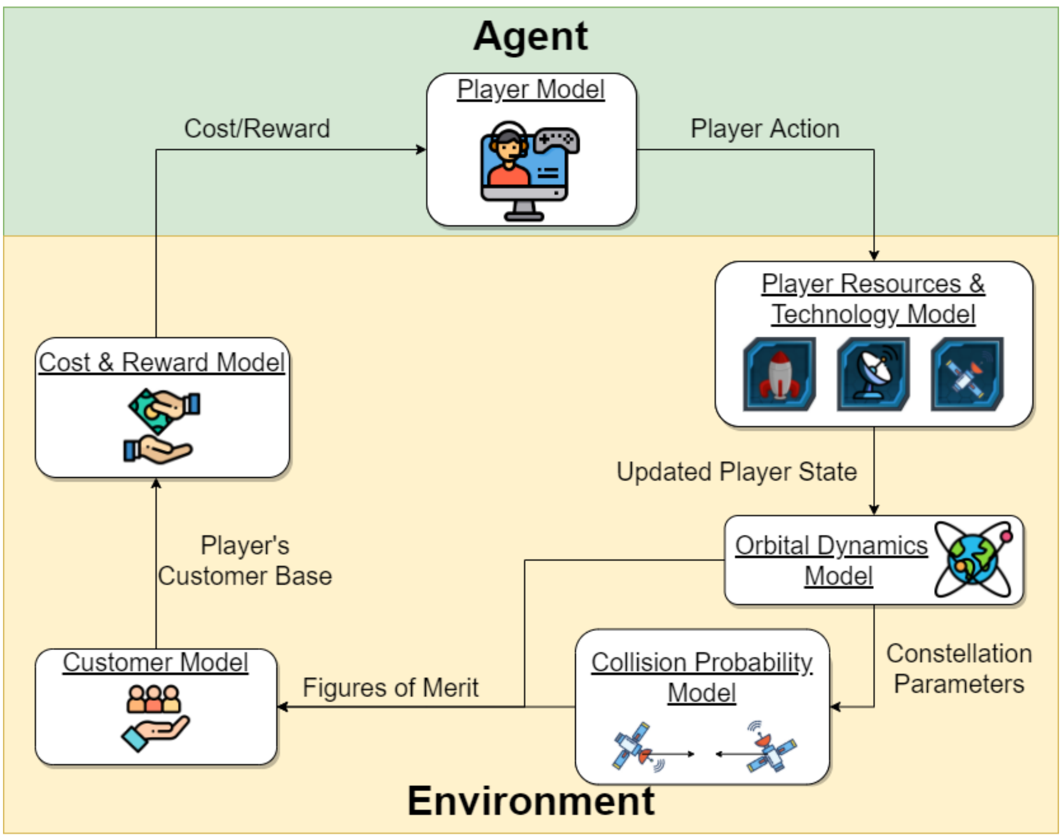

Figure 4.3: Sat-Tycoon Modeling Framework Shown as an Agent-Environment System

The following sections describe key models needed to ensure an accurate and playable game with meaningful strategy sets. These models are independently expressed and expandable with respect to their fidelity. Each model may have its fidelity improved in the future, if computationally feasible.

4.1.1 Player Model

Players must focus on becoming a competitive service provider by engaging with the satellite internet marketplace. The player state is defined as all information pertaining to the player, including (but not limited to): current funds, resources, constellation design, constellation figures of merit, and customer base. To *actively* change their state, a player must take one of the actions shown in Figure 4.4. Player states may also be changed via passive game dynamics: fluctuations of revenues from virtual customer. However, such a state change does not require any specific input from the player; rather, it is a byproduct of competitive environments and the temporal nature of the game.

Figure 4.4: Base Actions Available to Players

When building satellites, players may choose any positive integer value and that number of satellites will be manufactured for the player, provided funding availability for that amount of satellites. Each satellite has basic attributes of mass, power, and lifespan. The mass of a satellite determines how many of that type may be launched using a particular launch vehicle. The power of a satellite is used to compute the average data rate capabilities of a player's constellation network.

Players may build a ground station in any valid terrestrial location by clicking directly on the terrestrial map or by inputting a specific set of coordinates, contingent upon them having sufficient funding for a ground station. Each ground station has attributes of access area and operational personnel. The access area is used to estimate the customer base nearby each particular ground station. The number of operational personnel at each ground station directly impacts its recurring cost in the form of salaries. Note that in our model we assume that all customers have user terminals and we don't currently model the associated details such as the cost of distributing, maintaining, and switching to a competitor's user terminals.

To launch satellites into a valid orbit plane, players must: build satellites, purchase a launch vehicle, select the number of satellites to launch, and identify the orbit parameters using an inclination and a right ascension of ascending node (RAAN) map. Once successfully in orbit, the orbital dynamics model computes new constellation figures of merit. Once a player has built a valid constellation (of any size), they are able to offer a monthly internet subscription for their satellite internet service to virtual customers. This set price directly impacts not only their number of customers, but also the quality and type of customers acquired by the player. As a future expansion, unique internet subscriptions and pricing tiers based on geographical or performance differences may be introduced.

4.1.2 Player Resource Model

As in any resource economy business, players require multiple types of resources to advance their constellation performance. The resources in Sat-Tycoon are described as the following:

1. *Satellites*: players build and launch satellites into specific orbit planes to improve constellation coverage

2. *Ground Stations*: players unlock access to new customers when these are built in new areas

3. *Space on Launch Vehicles*: players must purchase these as a means of delivering their satellites into LEO

4. *Antenna Technology*: players must research these technologies to improve their constellation performance

Throughout the progression of each technology class, key attributes associated with constellation performance or overall costs are improved. As players invest in satellite technology development, transmittable power increases, satellite mass decreases, and operational cost decreases. Ground stations act similarly; however, the recurring operational costs of a ground station decreases due to automation of tasks (and therefore fewer employees to pay). Launch vehicles exhibit a similar pattern but are specifically modeled with current commercial launch vehicle performances [21].

Technology development not only unlocks a new aspect of game complexity, but also allows for more realistic modeling capabilities and scenarios. As technologies such as phased array antennae and satellite interlink technology become more standard, more competitive advantages will be afforded to those that invest strategically in select technologies. The design of Sat-Tycoon's Technology Progression allows for exactly these types of competitive studies to be conducted in the future. There are many ways of modeling technology advancement in a gamified framework: token-based attribute upgrades, experience-based resource availability, and funding-based technology upgrades. We implement this last method using a basic technology tree with fixed research costs for each resource.

4.1.3 Orbital Dynamics Model

To compare the performance of different satellite internet services, basic coverage figures of merit are computed for each player's constellation. Sat-Tycoon utilizes a first-order constellation engine to map constellation elements to relevant figures of merit. Simplifying assumptions are introduced in the construction of the constellation engine to trade modeling fidelity of the orbit motion with computational speed for game playability. For a given orbit and ground station, the constellation engine first estimates along-track coverage. Then, the constellation estimates coverage over a full right ascension range at the ground station latitude. Note that coverage over right ascension values may be mapped to coverage over longitudes as function of the simulation epoch. Using Chapter 9 of Wertz [63], we develop a formulation to estimate along-track coverage. This formulation is presented in Algorithm 2.

Algorithm 2 Along-Track Coverage Estimation

procedure ALONG-TRACK($i, \Omega, \epsilon_{min}$) ▷ Orbital elements and gateway from player state

Compute the Earth's angular radius as a function of the orbit altitude, ρ

Compute the maximum nadir angle, η_{max}, as a function of the minimum elevation and Earth's angular radius

Compute the Earth's maximum central angle: λ_{max}, as a function of the maximum nadir angle and minimum elevation

Compute the Earth's minimum central angle: λ_{min}, as a function of instantaneous orbit pole angular coordinates and ground station angular coordinates.

Compute an average Earth's central angle: $\lambda_{avg} = (\lambda_{max} - \lambda_{min})/2$

Compute the orbit period, P

Compute fraction of the orbit over which the ground station is in view, assuming non-rotating Earth (reasonable for LEO), and duration of a single transit

$$\Delta T = \frac{P}{\pi} \arccos\left(\frac{\cos\lambda_{max}}{\cos\lambda_{avg}}\right) \tag{4.1}$$

Estimate the average number of satellites visible along track at a random epoch as

$$n = \frac{\Delta T}{P} n_{sat} \tag{4.2}$$

where n_{sat} is the total number of satellites along a given orbit.

end procedure

The percentage of satellite orbits over one day that will cover a single point of Earth surface at the given latitude, C_ϕ, is then estimated. Depending on the selected orbit, one of three coverage scenarios is expected: no coverage, one coverage region, or two coverage regions. Naturally, the case with no coverage gives zero percentage of satellite orbits. For the case of a single coverage region, the percentage of satellite orbits is given as:

$$C_\phi = \frac{\Delta\lambda_1}{\pi} \cdot 100 \tag{4.3}$$

where

$$\cos\Delta\lambda_1 = \frac{-\sin\gamma_{max} + \cos i \sin\phi}{\sin i \cos\phi} \tag{4.4}$$

For the case with two coverage regions, the percentage of satellite orbits is given as

$$C_\phi = \frac{\Delta\lambda_2 - \Delta\lambda_1}{\pi} \cdot 100 \tag{4.5}$$

where

$$\cos\Delta\lambda_2 = \frac{+\sin\gamma_{max} + \cos i \sin\phi}{\sin i \cos\phi} \tag{4.6}$$

Finally, the average number of satellites visible at a random epoch, n, is registered in right ascension bands covered by the selected street of coverage at the given latitude (i.e., the across-track coverage).

For regions on Earth where constellation coverage is present, a mean data rate is estimated by averaging across all satellites servicing the given location. The data rate for an individual satellite is estimated utilizing a canonical link budget formulation. Specifically, our data rate is given as the following function:

$$R_B = \Gamma \cdot B \tag{4.7}$$

where B is the bandwidth of the given telecommunications technology and Γ is given by the fundamental limit in spectral efficiency [71]:

$$\Gamma = \log_2(1 + SNR) \tag{4.8}$$

where SNR is the signal-to-noise ratio given in dB. We compute the signal-to-noise ratio using the following formulation of the link budget (note that each value is given in dB):

$$SNR = EIRP + G_r - B - L_s - L_a - k \tag{4.9}$$

where $EIRP$ is the equivalent isotropic radiation power, G_r is the receiver gain, L_s is the free-space path loss, L_a is the atmospheric loss, k is the Boltzmann constant, and B is, once again, the bandwidth. Path losses are a function of the average altitude of the players constellation. Atmospheric, misalignment, and cable losses are not accounted for in this computation of the link budget. Each time a constellation is updated or altered, the figures of merit described above will also be recomputed.

4.1.4 Congestion & Collisions Model

The envisioned architecture of a collision model is shown in Figure 4.5. The three major components of this model will be: an analytical evolution of space debris clouds, the computation of collision probabilities from spatial densities, and the NASA Standard Breakup Model [72]. It is possible to propagate space debris, resident space objects, and competitor constellations over long time horizons via approximations of spatial density that are analytically computed [73]. The probability of a collision event occurring may be computed for each player's constellation with respect to the spatial density of space debris in their particular orbit shell. These collision events are computed each virtual month. If a collision event occurs, an analytical version of the NASA Standard Breakup Model can be used to approximate the space debris cloud generated from such an impact.

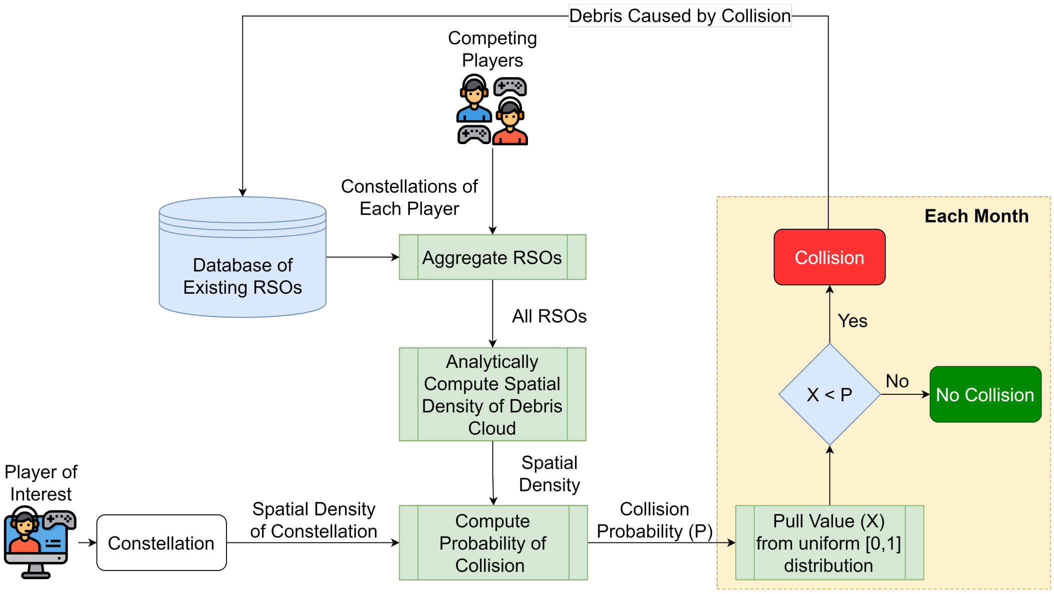

Figure 4.5: Architecture of Collision Model

As an accurate propagation model of space debris is known to be computationally expensive and time-consuming [74], alternative (first-order) methods must be employed to analytically propagate such a large number of space objects. We intend to implement an analytical evolution model for space objects and opposing constellations by framing all objects not in control of the player as a single, large space debris cloud. This will allow us to abstract more complex orbital dynamics and perturbation theory in favor of simply understanding the progression of the space debris cloud over the corresponding altitudes. Debris clouds vary along the altitude, predominantly under the effects of atmospheric drag (although additional perturbations due to J_2 will also be considered) [73]. We simplify the motion of debris by grouping the space debris cloud over altitude and area-mass ratio of individual debris objects. This allows for faster computations without loss of much accuracy. The spatial number density s, at a particular time t, and orbital height r, may be represented as the summation of all spatial number densities:

$$s(r,t)=\sum_{i=1}^{N_h}\sum_{j=1}^{N_a}s_{i,j}(r,t) \tag{4.10}$$

where $s_{i,j}(r,t)$ is the spatial number density of space debris at time t, at an altitude i, with a distinct area-mass ratio j and a specific radius r.

Once an accurate set of spatial densities is available, a simplified version of Kessler's [75] collision probability equation generates the probability of collision between a particular player constellation and the space debris cloud:

$$P_{\text{collision}}=\sigma\sum_{i=\text{ volume}}\bar{S}_{\text{satellite, }i}\bar{S}_{\text{debris },i}V\Delta U_i \tag{4.11}$$

where σ is the collision cross-section area, $\bar{S}_{\text{satellite, }i}$ is the spatial density of the satellite constellation, $\bar{S}_{\text{debris },i}$ is the spatial density of the space debris cloud, V is the collision velocity, and ΔU_i is the volume of space at the i-th height interval.

Once a collision probability can be determined, a random number will be generated to simulate if that collision has occurred or not. If the collision does not occur, the next player constellation is analyzed. If a collision does occur, the subsequent space debris is then cataloged at a particular altitude and area-mass ratio to be used in subsequent collision analysis. Note that this specific model will introduce stochasticity into the environment and is a future development item that is not implement in the "deterministic" version of the game.

4.1.5 Virtual Customer Model

Each player's customer base can be represented as the summation of all the customers allocated to them by the environment based on their constellation performance and salient business strategies. Such a formulation in which simulated customers choose suppliers of a good or service is known as a sealed-bid, reverse-auction process and is shown in Figure 4.6. We simulate customers using a population distribution and growth function that discretizes the global population by projected geographical squares. In each square, a percentage of the population requests internet access, and this subset of the population is further discretized into four different sets of customers: loyal customers, price-conscious customers, power-user customers, and nominal customers. Each customer type is modeled to have different preferences when selecting an player. This is accomplished using a different vector of weighted preferences for each customer type. The selection of a player by customers is formulated as a bidding evaluation problem. A player is then selected through the use of a behaviour model, and all of the customers of that particular type, in that particular projected square, are allocated to the chosen player. Once all of the projected squares have had their customers allocated, each player computes their revenue as a function of their total global customer base and their set internet price.

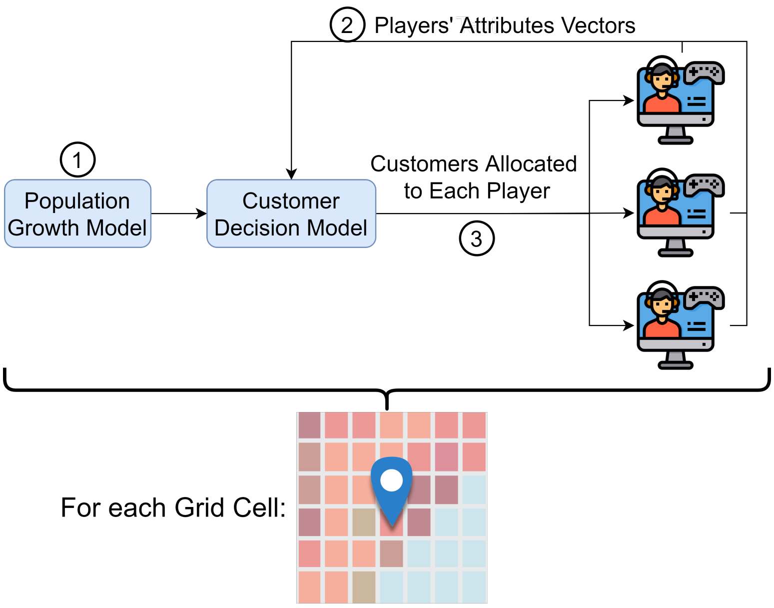

Figure 4.6: Customer Model Flowchart

The customer behaviour model is framed as a multi-attribute decision making (MADM) problem in which sets of customers, in a particular projected square, decide upon an internet service provider given a set of weighted preferences. This method of customer decision-making in the context of a reverse-bidding formulation is adapted from Section 3.2 of Cheng [76].

The general Technique for Order of Preference by Similarity to Ideal-Solution (TOPSIS) method, proposed by Tzeng and Huang [77] and outlined by Yang and Hung [78], is used to solve the MADM problem for each customer type, in each grid cell. This method is given in detail:

Step 1. *Matrix Representation*

The MADM problem can be written in matrix form with rows indicating competing alternatives (players) and columns indicating the different attributes considered in the problem. For the current version of Sat-Tycoon, we consider the following player attributes: internet price, data rate, data throughput, number of interruptions, longest interruption, and average latency between a random satellite in the constellation and a ground station. Equation (4.12) presents a MADM problem formulation with n different attributes and m different alternatives, in which each entry of the matrix, x_ij, indicates the performance (raw value) of alternative (player) i for attribute j. This matrix varies temporally and is updated once per virtual month.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{4.12}$$

Step 2. *Normalization of performance ratings*

Once a MADM matrix is formulated, we must normalize the performance ratings of each attribute such that they can be compared on an equal basis. We take the approach of [77] in which attributes are categorized into cost and benefit attributes. From the customer's perspective, cost attributes are those that should be minimized and benefit attributes are those that should be maximized. In the MADM problem considered for our project, internet price, number of interruptions, and longest interruption are all cost attributes, while data rate, data throughput, and company reputation are all benefit attributes. Equations (4.13) and (4.14) show the formulations of normalized cost and benefit attributes used in the TOPSIS method, respectively.

$$r_{ij} = \frac{\max_i \{x_{ij}\} - x_{ij}}{\max_i \{x_{ij}\} - \min_i \{x_{ij}\}} \tag{4.13}$$

$$r_{ij} = \frac{x_{ij} - \min_i \{x_{ij}\}}{\max_i \{x_{ij}\} - \min_i \{x_{ij}\}} \tag{4.14}$$

We note that, through this normalization process, each attribute is expressed as a value in the interval [0,1]. These normalized values are irrespective of attribute type since the larger the r_{ij}, the more it satisfies the j-th attribute.

Step 3. *Weighting of attributes*

Weighting of attributes is an important component of the TOPSIS method since different customer types will have different priorities regarding attribute importance during their decision-making process. We may define a weighting vector to reflect each set of customers' preferences regarding the attributes in the MADM problem. This weighting vector is applied to the normalized matrix from the previous step and computed as:

$$v_{ij} = w_j r_{ij} \tag{4.15}$$

where w_j is defined as the individual weight (or importance) applied to the j-th attribute by the customer and v_{ij} is the weight-adjusted and normalized entry within the MADM matrix.

Step 4. *Identifying ideal and negative-ideal solutions*

The ideal solution, A^*, and negative-ideal solution, A^-, are defined by Equation (4.16) and Equation (4.17), respectively.

$$A^* = \left\{ \left(\max_i v_{ij} \mid j \in J \right) \mid i = 1, \ldots, m \right\} = \{v_1^*, \ldots, v_n^*\} \tag{4.16}$$

$$A^- = \left\{ \left(\min_i v_{ij} \mid j \in J \right) \mid i = 1, \ldots, m \right\} = \left\{v_1^-, \ldots, v_n^-\right\} \tag{4.17}$$

where J is the set of all attributes. The ideal solution is therefore a theoretical vector that takes the best attribute values available from all alternatives and the negative-ideal solution is the theoretical vector of the worst attribute values available from all alternatives.

Step 5. *Distance calculation*

The distances from each alternative to the ideal solution and the negative-ideal solution are computed using the Euclidean norm and defined formally in Equation (4.18) and Equation (4.19), respectively.

$$S_i^* = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^*\right)^2}, \quad i = 1, \ldots, m \tag{4.18}$$

$$S_i^- = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^-\right)^2}, \quad i = 1, \ldots, m \tag{4.19}$$

Note that S^* and S^- are now vectors describing the ideal and negative-ideal distances for each of the m alternatives, respectively.

Step 6. *Similarity calculation*

The similarity of each alternative is then computed using the following derivation:

$$R_i^* = \frac{S_i^-}{S_i^- + S_i^*}, \quad i = 1, \ldots, m \tag{4.20}$$

Note that the similarity of each alternative will always be in the interval [0,1] due to the nature of the derived distances.

Step 7. *Ranking and selection*

Finally, we use an *argmax* function on the vector of similarity scores, R^*, to obtain the alternative with the highest similarity metric. The player chosen by the TOPSIS method is then assigned all the customers associated with that specific customer type in the evaluated projected square (winner take all model).

4.1.6 Cost & Reward Model

Each player has, at most, three cost functions at any given point during gameplay. These cost functions are: non-recurring, recurring (action-based), and recurring (time-based) costs. The player also has a revenue function which is operated per quarter (discussed below). The interaction between the cost and reward functions is summarized in Figure 4.7.

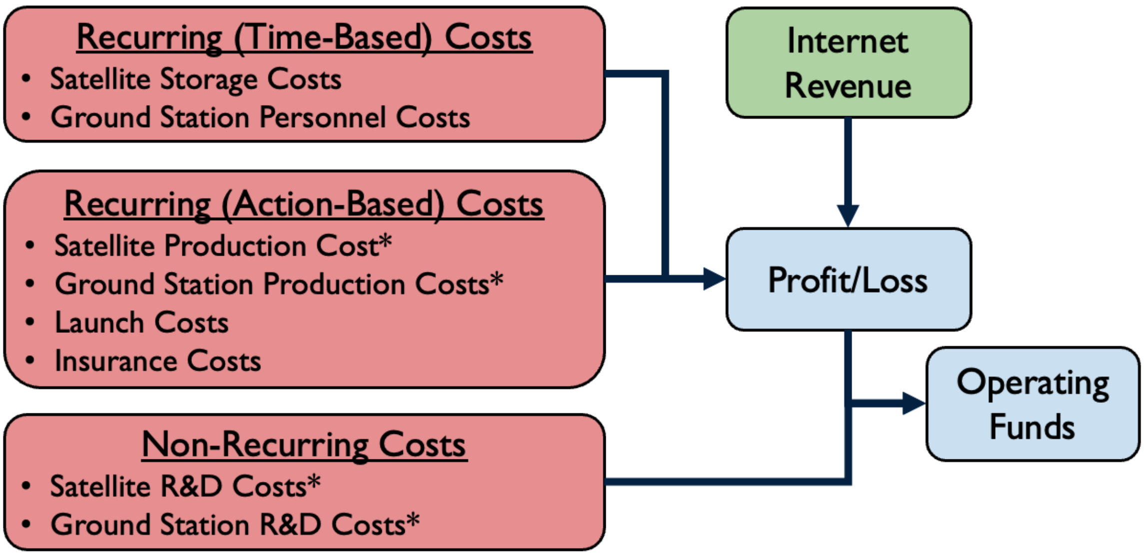

Figure 4.7: Model of Player Financial Functions

Non-Recurring Costs:

A player's non-recurring costs are defined within the game to be those costs that are only incurred once during the entire gameplay. Such costs may be thought of as capital expenditures (CapEx) of their company. Such expenditures typically take the form of: research and development, new manufacturing processes, and system validation. For the time being, we will abstract all CapEx to two simple, non-recurring costs: satellite research costs and ground station (gateway) research costs. From previous works [79], typical P-LEO satellite non-recurring costs are approximated as follows:

$$NRC_{SAT} = 7 \cdot TFU_{LEO} \tag{4.21}$$

where NRC is the non-recurring cost and TFU is the theoretical first unit cost of the product (a LEO satellite in our case). Also, from the same literature, we may take the non-recurring costs associated with researching and constructing an initial ground station:

$$NRC_{GS} = 5 \cdot TFU_{GS} \tag{4.22}$$

where these are the corresponding values to non-recurring costs and theoretical first unit costs. Such values will be incurred when the player purchases their first satellite or constructs their first ground station. Afterwards, purchases of these products will be considered recurring (action-based) costs and use a learning rate curve to simulate decreases in costs due to economies of scale.

Recurring (Action-Based) Costs:

As previously mentioned, recurring costs can be broken down into action-based and time-based functions. Action-based recurring costs are costs that are computed when the player provides input to the game. Such inputs include: buying additional satellites, buying additional ground stations, and launching satellites. Both satellite and ground station costs are classified as recurring costs due to their ongoing cost to the player; however, we may model both products with dynamic pricing using a simple cost function that leverages learning rate during production. The general cost for each product is given below:

$$C_{\text{prod,Sat}} = TFU_{LEO} \cdot L_{SAT} \tag{4.23}$$

$$C_{\text{prod,GS}} = TFU_{GS} \cdot L_{GS} \tag{4.24}$$

where TFU has already been defined as the theoretical first unit price of a given product and L is given as the learning curve factor. This factor is subsequently defined as:

$$L \equiv N^B \tag{4.25}$$

where N is the number of units ordered and B is defined with the following equation:

$$B = 1 - \frac{\ln((100\%)/S)}{\ln 2} \tag{4.26}$$

where S is defined as the learning curve slope (listed as a percentage). As a player requests the production of a resource, the overall cost of that resource will decrease from its TFU cost due to more efficient processes, manufacturing optimizations, and streamlining production lines.

Launch costs for Sat-Tycoon are a fixed quantity; however, a tiered system will be explained for further expansion. From literature [21], the base cost to launch a satellite into LEO was approximately $15,500 per kilogram. This is the average value for launch vehicles from the late-1990s up until the mid-2000s (specifically the Space Shuttle). Further expansion and game dynamics can be achieved by offering tiers of launch vehicles. Reusable launch vehicles such as the SpaceX Falcon 9 and Falcon Heavy have a lower launch cost to LEO of approximately $2,700 per kilogram and $1,400 per kilogram, respectively. In a recent NASA report [80], the next-generation SpaceX Starship can reduce the launch cost even further to $500 per kilogram. An additional layer of complexity that may be added in the future could be the variations in orbit altitude. A higher altitude would require more fuel, thus resulting in a higher launch cost. Lastly, insurance costs must also be computed for each launch. The insurance can be computed using the following equation:

$$C_{\text{ins}} = 0.2\left(C_{\text{sat}} + C_{\text{launch}}\right) \tag{4.27}$$

Recurring (Time-Based) Costs:

Finally, our last type of costs to the player are time-based recurring costs. These types of costs can be thought of as pure operating expenditure (OpEx), since they are assessed during every single epoch. Note that we will give costs over a certain sized epoch, but these costs can be divided or multiplied as needed to assess cost over varying epoch sizes. The main time-based recurring cost of a P-LEO constellation system is operating costs. Such costs can be functions of time, ground station size, telecom system complexity, and personnel required to operate the ground station. From literature [21], a typical gateway requires 4 shifts of 12 people operating the gateway. As of 2021, the yearly salary of an on-site engineer is given to be $150,000/year. We may use this information to compute the yearly (or monthly) cost of operating a single ground station and multiply this cost for ground stations a player has built. See the computation below for the yearly cost of a single ground station:

$$C_{GS,\,\text{oper}} = 4 \cdot 12 \cdot \$150,000 = \$7,200,000 \tag{4.28}$$

This may be incorporated into a broader equation to compute the total recurring cost to a player:

$$RC_{GS,\,\text{oper}} = C_{GS,\,\text{oper}} \cdot N_{GS} \tag{4.29}$$

where $RC_{GS,\,\text{oper}}$ is the yearly recurring cost of a player's ground stations and N_{GS} is the number of ground stations a player owns and operates. Note that this is a general equation and the time epoch will depend upon the epoch rate of the $C_{GS,\,\text{oper}}$ value.

System Revenue:

To compute revenue, we assume that the player has assigned a price for their internet service in each area that they operate a ground station. Furthermore, we assume that the Customer Model allocates a set of customers to this player based on their constellation figures of merit and internet service price. Note: this set of customers may be an empty set (this would correspond to a player with infeasible or uncompetitive internet service). Given these assumptions, computation of a player's revenue may be computed and assigned each yearly quarter using the following equation:

$$R_{M,N} = I_{M,N} \cdot N_{\text{Cust}} \tag{4.30}$$

where $R_{M,N}$ is the revenue generated per three month period at grid cell corresponding to M, N; $I_{M,N}$ is the internet price per month assigned to the grid cell corresponding to M, N (or a specific ground station); and N_{Cust} is the number of customers located within the grid cell corresponding to M, N.

4.1.7 The Resulting Sat-Tycoon Environment

Using a combination of first-order modeling techniques and gamification, the Sat-Tycoon environment has been developed and gone live on a private Auburn University server. Sat-Tycoon demonstrates an expandable framework that allows multiple players to develop constellations and compete against one another for virtual customers. As shown in Figure 4.8, this environment is a multi-player strategy simulation game which displays relevant information to players via a GUI. This framework has led to the development of multiple "plug-n-play" models of astrodynamics, customer decision-making, technology advancement, and player actions that can be updated to higher fidelity models for added functionality and more meaningful strategy design.

Figure 4.8: Multi-Player Functionality in Sat-Tycoon

With further development, the Sat-Tycoon framework may aid in determining the feasibility of possible business strategies and inform use cases for national security, astrodynamics education, and additional research areas, such as technology portfolio development. Figure 4.9 visualizes the possible, longer-term pathways to continue Sat-Tycoon development beyond this work. The framework demonstrated with the Sat-Tycoon prototype may also serve as a baseline for frameworks to simulate P-LEO assets servicing. A gamified framework for P-LEO assets servicing may include space logistics based on responsive launching, refueling, refurbishing, repairing and debris removal, both with a competitive or collaborative gameplay.

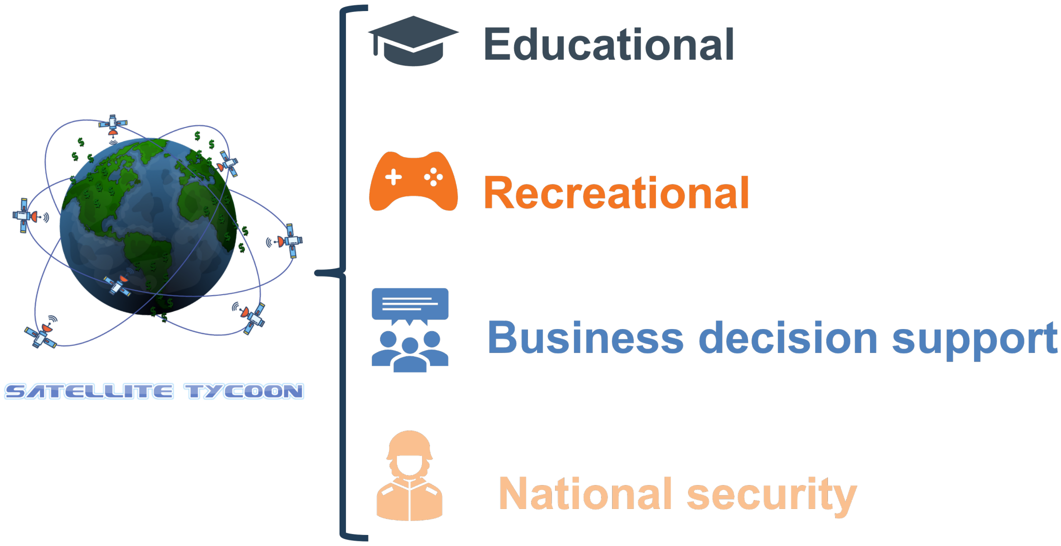

Figure 4.9: Possible Long-Term Pathways for Sat-Tycoon Development

Chapter 5

Discussion & Conclusions

The final chapter of this thesis concludes by reiterating our research questions and providing direct answers based on the findings of this work. The hypotheses introduced in Section 2.4 are also presented and we are able to comment on each according to our results. This discussion is followed by the modeling limitations of each environment and the effects they have on our results. Finally, based on the contributions, lessons learned, and limitations discovered throughout this work, future environment modeling techniques and solution methods are proposed.

5.1 Research Contributions

Through the process of developing this work, two unique modeling environments were developed: a SATCOM Grid World and Satellite Tycoon (a multi-player strategy simulation game). The key concepts differentiating these environments from previous works was the inclusion of competition, transitory dynamics, and development of business strategies relating to constellation designs. From the development of these environments (both optimal policies generated and lessons learned when designing the environment), we are able to address the initial research questions asked in Section 1.2 of this work.

The Grid World environment was developed as a dynamic program with the decision-maker competing against a static, established competitor that has an existing constellation in orbit. This environment was solved using the Backwards Dynamic Programming Algorithm and the following conclusions were noted:

1. Due to the cost function design, the cost-to-go function J_0 from Equation (3.5) is equivalent to the net present value (NPV) of executing a particular investment strategy in the SATCOM industry. Thus, according to the Bellman Optimality Principle, the optimal constellation development strategy is quantifiable as one which maximizes the NPV.

2. Due to the binary nature of the utility function design, the optimal strategy generated from DP allows for launching to an already occupied orbit shell due to the rapid recuperation of profits by customers.

3. Due to the limitations of state space exploration within dynamic programming, the decision-maker reaches an artificial "cap" on the $funds$ state variable. However, due to the existence of decision-maker satellites in an already occupied orbit shell, CAM costs are incurred to the decision-maker at each subsequent epoch. This causes the decision-maker to fluctuate the $price$ variable dictating their internet price in an attempt to regain these funds. This leads to oscillatory behavior in the decision-maker's internet pricing and available funds. The oscillatory behavior demonstrates a limitation of dynamic programming, and the existence of strong coupling between CAM costs, $funds$, and the $price$ variables.

4. Using points 2 and 3, the maximum funds of the dynamic program are interpreted as a "target" value for the decision-maker to reach and maintain across the time horizon.

Additionally, a variation of scenario analysis was conducted on the Grid World environment by adjusting simulation parameters to understand the complex couplings between environment parameters, state variables, and action variables. These variation of scenarios led to several key insights:

1. If an environment was deemed infeasible (i.e., the decision-maker is not able to recuperate their initial investments throughout the remaining epochs in the simulation), the optimal policy generated a static control profile. From a dynamic programming perspective, this "inaction" was the optimal policy in which the maximum NPV was achieved.

2. If all other simulation variables are fixed, then as total simulation length decreased, both compound annual growth rate ($CAGR$) and compound monthly growth rate ($CMGR$) increase until a certain infeasibility point is reached. This inflection point is achieved when the time-to-profit (TTP) is equal to or less than the total simulation length. In this case, the only feasible policy to take is "inaction". Therefore, these economic figures of merit are coupled to TTP and total simulation length.

3. The generated optimal policy from DP (and the subsequent NPV) of the decision-maker seems to be strongly coupled to the competitor's initial, static constellation design. This leads us to believe that there exist different criteria for barrier-to-entry in existing P-LEO markets (depending upon the level of development and/or saturation from the competition).

4. Since dynamic programming suffers from the curse of dimensionality, it becomes impractical to investigate more complex dynamics using a dynamic programming formulation of the environment. Specifically, higher dimensional systems are impractical to model and solve using dynamic programming.

To expand past the limitations of dynamic programming and model dynamic multi-agent competition, the Sat-Tycoon game was developed. This environment expands upon the orbital dynamics modeling used in the Grid World by offering an expanded set of constellation design variables for the space segment and individual orbit planes: inclination, RAAN, altitude, and satellites per plane. Given the more detailed orbital dynamics model, the figures of merit computed can also be more insightful and accurate. These figures of merit include: mean transit duration, orbit percentage coverage, total time in view of orbit, number of coverage regions, and data rate capabilities for a given area. These detailed constellation figures of merit are also due to a rich terrestrial segment modeled in Sat-Tycoon which features ground station development and customer data via a terrestrial map. Given the expanded geographical customer data as well as the constellation figures of merit, the binary utility function of the Grid World is replaced with a multi-attribute decision-making (MADM) problem which is formulated and solved at each epoch of the simulation. Finally, whereas the Grid World environment modeled a single type of satellite for computational efficiency, Sat-Tycoon includes technology expansions for satellites, launch vehicles, ground stations, and antenna arrays as a means to further model the technology gaps associated with complex competition.

While Sat-Tycoon offers a much higher dimensional state and action space for the modeling of more complex business strategies, this high dimensionality also makes computationally exploring the environment to uncover trends or economic equilibriums a difficult endeavor. However, future work in the realm of multi-agent reinforcement learning and game theory may resolve this issue (see Section 5.3 for more details). Throughout the modeling and development process of Sat-Tycoon, we have gained the following insights regarding competition and transient strategies within the SATCOM industry:

1. Although typically executed on a per launch basis, most constellation development aims to provide a minimum quality of service or cater to a specific region. However, in Sat-Tycoon, players must develop their constellations progressively. Although this gameplay favors a sequential style of development, it also highlights the ability to make adjustments based on customer population shifts or competitor actions.

2. While individual user terminals are not explicitly modeled due to computational infeasibility, ground stations (or more accurately "serviceable regions") representing the area in which a player's constellation network offers broadband services to a finite amount of customers are modeled. This allows for multiple players to compete in the same serviceable region in a more realistic manner than the Grid World's zero-sum customer model. While more robust, much of the explainability of environmental allocations of customers to players is lost as modeling fidelity of the environment increases and additional layers of complexity are added between player actions and subsequent customer response. While future stochastic processes may further erode the explainability of customer decisions, the modeling of such processes is valid as long as these environmental scenarios are reproducible and the requisite models constituting the environment are modeled with a standard set of assumptions across each modeling component.

3. The need for and resulting complexity of transient constellation development strategies seems to be inversely related to the level of competition in Sat-Tycoon. If a single player is playing Sat-Tycoon, virtually any constellation development strategy will guarantee revenues against a baseline service provider. If multiple players are competing in the same game environment, the need for a development strategy is relatively more critical to their success. This phenomenon of monopoly in and of itself is not a novel discovery; however, this highlights an implicit limit to the number of competitors the environment can support over a long time-horizon.

Given the above insights, results, and lessons learned, we are able to address each of our initial research questions with qualitative solutions and expand upon them. Note that while we address the research questions, this work is meant to introduce modeling environments to further explore these questions using the following techniques:

- Game and Economic Theory
- Underlying Satellite Motion
- Computational methods such as Reinforcement Learning

RQ1. How does varying the level of modeling fidelity affect performance metrics and competition?

- As expected, increasing the modeling fidelity of the environment (particularly the number of constellation design variables) gives players more optionality when developing constellation deployment strategies; however, this limits the usefulness of traditional strategy space exploration tools such as dynamic programming. Due to the P-LEO environment being a highly dimensional problem, the curse of dimensionality becomes painfully apparent when conducting experiments on the Grid World. The addition of a single additional state variable exponentially grows the state space to the point of infeasibility. However, the addition of constellation design variables also increases the accuracy and robustness of constellation figures of merit. Although the Grid World considers static competition, when considering dynamic competition in Sat-Tycoon, the policy space becomes even more complex. If the player's competitors have sufficiently complex strategies, the competition in Sat-Tycoon may be interpreted as a partial information game which requires far more robust solution methods.

RQ2. What are the economic benefits and penalties of competition to dynamic constellation development strategies?

- *Benefits:* For both environments, there exist non-intuitive, complex couplings between a competitor's constellation design and a resulting optimal policy. Such couplings can inform constellation designers or operators if/when development of a competing satellite constellation is feasible or not simply based on the competitor's existing P-LEO constellation.

- *Penalties:* For certain constellation performance differences that are driven by superior technology, if competitors have the superior technology, developing a constellation and recuperating initial investments in a specified time-horizon becomes infeasible due to the decision-maker's added expenses of developing a competitive service. However, in environments such as Sat-Tycoon (and in practical applications) a competitor's technological capabilities are not precisely known; rather, a competitor's quality of service and constellation design allows us to approximate their technology capabilities. It must be noted that neither environment captures a P-LEO constellation's customer capacity.

RQ3. Which variables impact economic figures of merit that measure success, failure, and risk during development of a P-LEO constellation, in a competitive environment?

- The time-horizon, competitor's constellation capabilities (technologies), and available customers all impact economic metrics such as NPV, TTP, and $CAGR$. Specifically, each of these environment variables impact the feasibility of investing in the SATCOM industry by impacting the NPV. Additionally, if a pricing competition is too aggressive, the player cannot undercut their competitors and, therefore, the economic risk of not recuperating one's investment is too high to engage in the market.

5.2 Limitations of Environments

While both environments developed throughout this work are able to model different versions of transient constellation development and competition, they also possess limitations that prevent further experimentation or analysis. It is crucial to identify these limitations as a first step towards bridging the knowledge gap between the Grid World (which is solvable, but limited) and Sat-Tycoon (which is strategically complex, but difficult to solve).

5.2.1 Limitations of Grid World

Although a powerful and motivating base environment, the SATCOM Grid World is modeled as a discrete, deterministic dynamic program. This modeling approach produces two major caveats: discrete states and actions typically have dimensionality restrictions, and the deterministic nature of the DP formulation limits the possible modeling scenarios. Such an environment is in fact the simplest case of a Markov Decision Process (MDP). Typically, MDPs include stochasticity and are solved using state transition probabilities (known as model-based methods) or direct state-action interactions with the environment (known as model-free methods). However, neither method guarantees a globally optimal policy as in the case of dynamic programming.

Regarding modeling limitations, both the orbital dynamics and customer models are very limited due to the aforementioned dimensionality constraints, need for an explainable customer allocation process, and lack of constellation information stored in the decision-maker's state. Although relative and mainly used for comparative analysis, the approximate constellation figures of merit are only based on satellites per orbit shell. In actuality, computation of a constellation's coverage over a specific region requires most of the orbital elements and information regarding coverage region of interest. Additionally, while we model the customer decision model using a zero-sum utility function, this is mainly done for simulation explainability and recreation.

5.2.2 Limitations of Sat-Tycoon

While a much more advanced environment than the Grid World, Sat-Tycoon also has a few key limitations to its functionality. The two major hurdles facing Sat-Tycoon experimentation are: a lack of solution methods for obtaining strategies or economic equilibriums, and a lack of intuitive gameplay. The first hurdle has both a model complexity and a competition design issue. While the Grid World has a *relatively* small policy search space, depending upon the number of players, Sat-Tycoon can have an exponentially larger and more complex search space. This search space cannot be explored using single agent methods due to the lack of sophisticated competitor strategies needed to properly compete against the agent. This not only hinders strategy development, it also limits systematic exploration of game and economic theories such as Nash or Walras equilibriums. To overcome this hurdle, Sat-Tycoon will need to be formulated and solved with multi-agent considerations.

The second major hurdle facing Sat-Tycoon is the user interface design and overall lack of intuitive gameplay. While most research games feature a dashboard of relevant information (such as the state of the player), by the nature of P-LEO constellations, this may not be feasible in the case of Sat-Tycoon due to the many variables and high dimensional state space. Additionally, model implementation into the Sat-Tycoon framework does not guarantee a gamified experience.

5.3 Future Works

Given the research contributions of this work and the current limitations of both the SATCOM Grid World and Sat-Tycoon, we map out three major paths forward regarding research and future development directions. The first path is meant to bridge the Grid World and Sat-Tycoon environments by introducing a combination of models from both environments into a new environment that can be solved using reinforcement learning (RL). The second path forward is the addition of stochastic models into Sat-Tycoon. These models can include phenomena such as: cyber-security attacks, satellite collision events, limited launch availability or launch failures, natural disasters, supply chain disruptions, and uncertainty of customer demand. Finally, the more long-term research path forward is the general improvement of Sat-Tycoon for both multi-agent reinforcement learning and an eventual human playability study. Such a study may be used to generated training data for multi-agent RL but can also elucidate strategies which were not considered. This long-term path forward will be in the hopes of developing a mathematically rigorous understanding of the equilibriums or trends within the P-LEO SATCOM industry.

The development of an intermediate RL environment between the Grid World and Sat-Tycoon offers several benefits from both environments while having the major drawback of training uncertainty. As the Grid World was formulated as a dynamic program, the RL environment can be formulated as a Markov decision process (MDP). This approach allows us to incorporate stochastic processes into the environment and also utilize key RL infrastructure and libraries such as OpenAI's Gym [81] for environment creation and Stable-Baselines3 Contrib [82] for RL agent algorithms. Such an environment would not only be able to use RL to search the strategy space, it would also not require the development of graphical user interface like Sat-Tycoon. If successfully implemented and solved as a RL environment, additional RL tools such as the Farama Foundation's PettingZoo [83] library can be utilized to expand the RL environment to a multi-agent environment. However, even with a proper MDP formulation and the requisite RL tools, meaningful optimal policies are not guaranteed in RL algorithms.

While neither of the environments developed throughout this work feature stochastic models, the eventual goal of Sat-Tycoon is the be able to model multiple types of events which add an additional layer of complexity to the dynamics and can improve the overall player experience. Specifically, modeling collision events as a player constellation interacting with space debris clouds in LEO can add a layer of complexity and strategy to the environment due to a potential economic Kessler syndrome disrupting competitor operations [37]. Similarly, stochastic availability of launch vehicles and launch failures can also add competition to the environment by limiting potential opponent launch actions, thereby limiting their constellation performance and subsequent internet services. Additional stochastic processes which are not player induced such as natural disasters, customer demand shifts, and supply chain disruptions can also improve game playability, environment authenticity, and policy robustness. Finally, cyber-security threats can be classified as their own branch of stochastic processes and can be modeled in a variety of different way depending upon the specific threat.

The final, and most long-term, research path for this work is the improvement of Sat-Tycoon modeling and user interface to the point where a human playability study can be conducted for data collection and play-testing. This research path will also allow us to study the different strategies humans tend to develop and receive external feedback regarding the user interface design. Once Sat-Tycoon is sufficiently developed as a multi-agent environment, RL agents can be trained to compete against one another in a style similar to OpenAI Five [84]. Finally, using collected player data, multi-agent RL training, and game and economic theory, we hope to develop new understandings of the P-LEO satellite internet market. This can include economic equilibriums, mathematical theory, and underlying environment dynamics.

Bibliography

[1] Dominic Gates. “Elon Musk touts launch of ‘SpaceX Seattle’”. In: *The Seattle Times* (2015). URL: https://web.archive.org/web/20150213044054/http://seattletimes.com/html/businesstechnology/2025480750_spacexmuskxml.html (visited on 11/04/2022).

[2] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning”. en. In: *Nature* 518.7540 (Feb. 2015), pp. 529–533. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/nature14236. URL: http://www.nature.com/articles/nature14236 (visited on 10/25/2021).

[3] Jeff Foust. “Satellite operators criticize “extreme” megaconstellation filings”. In: *Space-News* (2021). URL: https://spacenews.com/satellite-operators-criticize-extreme-megaconstellation-filings/ (visited on 03/31/2022).

[4] Nils Pachler et al. “An Updated Comparison of Four Low Earth Orbit Satellite Constellation Systems to Provide Global Broadband”. In: *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. Montreal, QC, Canada: IEEE, June 2021, pp. 1–7. ISBN: 978-1-72819-441-7. DOI: 10.1109/ICCWorkshops50388.2021.9473799. URL: https://ieeexplore.ieee.org/document/9473799/ (visited on 04/01/2022).

[5] M. Swartwout. “Cheaper by the dozen: The avalanche of rideshares in the 21st century”. In: *2013 IEEE Aerospace Conference*. Big Sky, MT: IEEE, Mar. 2013, pp. 1–12. DOI: 10.1109/AERO.2013.6497182. URL: http://ieeexplore.ieee.org/document/6497182/ (visited on 04/01/2022).

[6] Ogutu B. Osoro and Edward J. Oughton. "A Techno-Economic Framework for Satellite Networks Applied to Low Earth Orbit Constellations: Assessing Starlink, OneWeb and Kuiper". In: *IEEE Access* 9 (2021), pp. 141611–141625. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3119634. URL: https://ieeexplore.ieee.org/document/9568932/ (visited on 04/01/2022).

[7] Gian Luigi Somma, Hugh G. Lewis, and Camilla Colombo. "Sensitivity analysis of launch activities in Low Earth Orbit". en. In: *Acta Astronautica* 158 (May 2019), pp. 129–139. ISSN: 00945765. DOI: 10.1016/j.actaastro.2018.05.043. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576517311244 (visited on 01/24/2022).

[8] T. P. Garrison et al. "Systems Engineering Trades for the IRIDIUM Constellation". en. In: *Journal of Spacecraft and Rockets* 34.5 (Sept. 1997), pp. 675–680. ISSN: 0022-4650, 1533-6794. DOI: 10.2514/2.3267. URL: https://arc.aiaa.org/doi/10.2514/2.3267 (visited on 03/23/2022).

[9] M.A. Sturza. "The Teledesic satellite system". In: *Proceedings of IEEE National Telesystems Conference - NTC '94*. San Diego, CA, USA: IEEE, 1994, pp. 123–126. ISBN: 978-0-7803-1869-4. DOI: 10.1109/NTC.1994.316677. URL: http://ieeexplore.ieee.org/document/316677/ (visited on 03/23/2022).

[10] Andrew W. Lewin. "Low-Cost Operation of the ORBCOMM Satellite Constellation". In: *Journal of Reducing Space Mission Cost* 1.1 (1998), pp. 105–117. ISSN: 13857479. DOI: 10.1023/A:1009987231306. URL: http://link.springer.com/10.1023/A:1009987231306 (visited on 03/23/2022).

[11] Lorin Hitt and Prasanna Tambe. "Broadband adoption and content consumption". In: *Information Economics and Policy* 19.3 (2007). Economics of the Media, pp. 362–378. ISSN: 0167-6245. DOI: https://doi.org/10.1016/j.infoecopol.2007.04.003. URL: https://www.sciencedirect.com/science/article/pii/S0167624507000297.

[12] Jose Del Rosario. "Pricing The Satellite Markets". In: *Northern Sky Research* (2017). URL: https://www.nsr.com/pricing-the-satellite-markets/ (visited on 11/04/2022).

[13] Inigo del Portillo, Bruce G. Cameron, and Edward F. Crawley. "A technical comparison of three low earth orbit satellite constellation systems to provide global broadband". en. In: *Acta Astronautica* 159 (June 2019), pp. 123–135. ISSN: 00945765. DOI: 10.1016/j.actaastro.2019.03.040. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576518320368 (visited on 11/19/2021).

[14] Markus Guerster. "Revenue Management and Resource Allocation for Communication Satellite Operators". PhD thesis. MASSACHUSETTS INSTITUTE OF TECHNOLOGY, Sept. 2020. URL: https://dspace.mit.edu/handle/1721.1/129158.

[15] Quan Chen et al. "A distributed congestion avoidance routing algorithm in mega-constellation network with multi-gateway". en. In: *Acta Astronautica* 162 (Sept. 2019), pp. 376–387. ISSN: 00945765. DOI: 10.1016/j.actaastro.2019.05.051. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576518317387 (visited on 01/24/2022).

[16] S. Le May et al. "Space debris collision probability analysis for proposed global broadband constellations". en. In: *Acta Astronautica* 151 (Oct. 2018), pp. 445–455. ISSN: 00945765. DOI: 10.1016/j.actaastro.2018.06.036. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576518304375 (visited on 01/24/2022).

[17] Hugh G. Lewis. "Evaluation of debris mitigation options for a large constellation". en. In: *Journal of Space Safety Engineering* 7.3 (Sept. 2020), pp. 192–197. ISSN: 24688967. DOI: 10.1016/j.jsse.2020.06.007. URL: https://linkinghub.elsevier.com/retrieve/pii/S2468896720300616 (visited on 01/24/2022).

[18] C. Pardini and L. Anselmo. "Assessing the Risk of Orbital Debris Impact". In: *Space Debris* 1.1 (1999), pp. 59–80. ISSN: 13883828. DOI: 10.1023/A:1010066300520.

URL: http://link.springer.com/10.1023/A:1010066300520 (visited on 03/12/2022).

[19] A. Rossi, A. Petit, and D. McKnight. "Short-term space safety analysis of LEO constellations and clusters". en. In: *Acta Astronautica* 175 (Oct. 2020), pp. 476–483. ISSN: 00945765. DOI: 10.1016/j.actaastro.2020.06.016. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576520303830 (visited on 03/09/2022).

[20] Jiaxin Hu et al. "A multi-objective optimization framework of constellation design for emergency observation". en. In: *Advances in Space Research* 67.1 (Jan. 2021), pp. 531–545. ISSN: 02731177. DOI: 10.1016/j.asr.2020.09.031. URL: https://linkinghub.elsevier.com/retrieve/pii/S0273117720306785 (visited on 01/24/2022).

[21] Harry W. Jones. "The Recent Large Reduction in Space Launch Cost". In: *Proceedings of the 48th International Conference on Environmental Systems* (July 2018).

[22] Edward W. Ashford. "Non-Geo systems—where have all the satellites gone?" en. In: *Acta Astronautica* 55.3-9 (Aug. 2004), pp. 649–657. ISSN: 00945765. DOI: 10.1016/j.actaastro.2004.05.018. URL: https://linkinghub.elsevier.com/retrieve/pii/S009457650400181X (visited on 11/19/2021).

[23] Gary Comparetto and Neal Hulkower. "Global mobile satellite communications - A review of three contenders". en. In: *15th International Communicatons Satellite Systems Conference and Exhibit*. San Diego,CA,U.S.A.: American Institute of Aeronautics and Astronautics, Feb. 1994. DOI: 10.2514/6.1994-1138. URL: http://arc.aiaa.org/doi/10.2514/6.1994-1138 (visited on 11/19/2021).

[24] Matthew Graydon and Lisa Parks. "'Connecting the unconnected': a critical assessment of US satellite Internet services". en. In: *Media, Culture & Society* 42.2 (Mar. 2020), pp. 260–276. ISSN: 0163-4437, 1460-3675. DOI: 10.1177/0163443719861835. URL: http://journals.sagepub.com/doi/10.1177/0163443719861835 (visited on 10/30/2022).

[25] Stephen Bosomworth and Paul T. Grogan. *Effects of Staged Deployment on the Economics of Global Broadband Internet Satellite Constellations*. preprint. engrXiv, Aug. 2021. DOI: 10.31224/osf.io/svnt2. URL: https://engrxiv.org/index.php/engrxiv/preprint/view/1852 (visited on 03/08/2022).

[26] Joshua F. Anderson, Michel-Alexandre Cardin, and Paul T. Grogan. "Design and analysis of flexible multi-layer staged deployment for satellite mega-constellations under demand uncertainty". en. In: *Acta Astronautica* 198 (Sept. 2022), pp. 179–193. ISSN: 00945765. DOI: 10.1016/j.actaastro.2022.05.022. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094576522002181 (visited on 01/26/2023).

[27] Andrew Rader, Adam M. Ross, and Matthew E. Fitzgerald. "Multi-Epoch Analysis of a Satellite Constellation to Identify Value Robust Deployment across Uncertain Futures". en. In: *AIAA SPACE 2014 Conference and Exposition*. San Diego, CA: American Institute of Aeronautics and Astronautics, Aug. 2014. ISBN: 978-1-62410-257-8. DOI: 10.2514/6.2014-4269. URL: https://arc.aiaa.org/doi/10.2514/6.2014-4269 (visited on 03/13/2022).

[28] Aizaz U. Chaudhry and Halim Yanikomeroglu. "Laser Intersatellite Links in a Starlink Constellation: A Classification and Analysis". In: *IEEE Vehicular Technology Magazine* 16.2 (June 2021), pp. 48–56. ISSN: 1556-6072, 1556-6080. DOI: 10.1109/MVT.2021.3063706. URL: https://ieeexplore.ieee.org/document/9393372/ (visited on 03/23/2022).

[29] Quan Chen et al. "Multiple gateway placement in large-scale constellation networks with inter-satellite links". en. In: *International Journal of Satellite Communications and Networking* 39.1 (Jan. 2021), pp. 47–64. ISSN: 1542-0973, 1542-0981. DOI: 10.1002/sat.1353. URL: https://onlinelibrary.wiley.com/doi/10.1002/sat.1353 (visited on 03/23/2022).

[30] Quan Chen et al. "Analysis of Inter-Satellite Link Paths for LEO Mega-Constellation Networks". In: *IEEE Transactions on Vehicular Technology* 70.3 (Mar. 2021), pp. 2743–

2755. ISSN: 0018-9545, 1939-9359. DOI: 10.1109/TVT.2021.3058126. URL: https://ieeexplore.ieee.org/document/9351765/ (visited on 03/23/2022).

[31] Nathan Reiland et al. "Assessing and minimizing collisions in satellite mega-constellations". en. In: *Advances in Space Research* 67.11 (June 2021), pp. 3755–3774. ISSN: 02731177. DOI: 10.1016/j.asr.2021.01.010. URL: https://linkinghub.elsevier.com/retrieve/pii/S0273117721000326 (visited on 03/09/2022).

[32] Eduardo Maria Polli, Juan Luis Gonzalo, and Camilla Colombo. "Analytical model for collision probability assessments with large satellite constellations". en. In: *Advances in Space Research* (July 2022), S027311772200686X. ISSN: 02731177. DOI: 10.1016/j.asr.2022.07.055. URL: https://linkinghub.elsevier.com/retrieve/pii/S027311772200686X (visited on 09/13/2022).

[33] Lorenzo Olivieri and Alessandro Francesconi. "Large constellations assessment and optimization in LEO space debris environment". en. In: *Advances in Space Research* 65.1 (Jan. 2020), pp. 351–363. ISSN: 02731177. DOI: 10.1016/j.asr.2019.09.048. URL: https://linkinghub.elsevier.com/retrieve/pii/S0273117719307264 (visited on 03/09/2022).

[34] Jonas Radtke, Christopher Kebschull, and Enrico Stoll. "Interactions of the space debris environment with mega constellations—Using the example of the OneWeb constellation". en. In: *Acta Astronautica* 131 (Feb. 2017), pp. 55–68. ISSN: 00945765. DOI: 10.1016/j.actaastro.2016.11.021. URL: https://linkinghub.elsevier.com/retrieve/pii/S009457651630515X (visited on 03/23/2022).

[35] Alexis Petit, Alessandro Rossi, and Elisa Maria Alessi. "Assessment of the close approach frequency and collision probability for satellites in different configurations of large constellations". en. In: *Advances in Space Research* 67.12 (June 2021), pp. 4177–4192. ISSN: 02731177. DOI: 10.1016/j.asr.2021.02.022. URL: https://linkinghub.elsevier.com/retrieve/pii/S0273117721001514 (visited on 01/24/2022).

[36] Nodir Adilov, Peter J. Alexander, and Brendan M. Cunningham. "An Economic Analysis of Earth Orbit Pollution". en. In: *Environmental and Resource Economics* 60.1 (Jan. 2015), pp. 81–98. ISSN: 0924-6460, 1573-1502. DOI: 10.1007/s10640-013-9758-4. URL: http://link.springer.com/10.1007/s10640-013-9758-4 (visited on 02/04/2022).

[37] Nodir Adilov, Peter J. Alexander, and Brendan M. Cunningham. "An economic "Kessler Syndrome": A dynamic model of earth orbit debris". en. In: *Economics Letters* 166 (May 2018), pp. 79–82. ISSN: 01651765. DOI: 10.1016/j.econlet.2018.02.025. URL: https://linkinghub.elsevier.com/retrieve/pii/S0165176518300818 (visited on 09/26/2022).

[38] Leigha Capra et al. "SpaceNet Cloud: Web-based Modeling and Simulation Analysis for Space Exploration Logistics". en. In: *ASCEND 2021*. Las Vegas, Nevada & Virtual: American Institute of Aeronautics and Astronautics, Nov. 2021. ISBN: 978-1-62410-612-5. DOI: 10.2514/6.2021-4068. URL: https://arc.aiaa.org/doi/10.2514/6.2021-4068 (visited on 03/08/2022).

[39] Duncan Eddy and Mykel Kochenderfer. "Markov Decision Processes For Multi-Objective Satellite Task Planning". In: *2020 IEEE Aerospace Conference*. Big Sky, MT, USA: IEEE, Mar. 2020, pp. 1–12. ISBN: 978-1-72812-734-7. DOI: 10.1109/AERO47225.2020.9172258. URL: https://ieeexplore.ieee.org/document/9172258/ (visited on 06/24/2022).

[40] Richard Kim. "Stochastic Inventory Control Modeling for Satellite Constellations". en. In: *Journal of Spacecraft and Rockets* 57.3 (May 2020), pp. 612–620. ISSN: 0022-4650, 1533-6794. DOI: 10.2514/1.A34614. URL: https://arc.aiaa.org/doi/10.2514/1.A34614 (visited on 06/24/2022).

[41] P. T. Grogan and O. L. de Weck. "Federated Simulation and Gaming Framework for a Decentralized Space-Based Resource Economy". en. In: *Earth and Space 2012*. Pasadena, California, United States: American Society of Civil Engineers, Apr. 2012, pp. 1468–1477. ISBN: 978-0-7844-1219-0. DOI: 10.1061/9780784412190.156. URL:

http://ascelibrary.org/doi/10.1061/9780784412190.156 (visited on 03/10/2022).

[42] Paul T. Grogan et al. “Multi-stakeholder interactive simulation for federated satellite systems”. In: *2014 IEEE Aerospace Conference*. Big Sky, MT, USA: IEEE, Mar. 2014, pp. 1–15. ISBN: 978-1-4799-5582-4. DOI: 10.1109/AERO.2014.6836253. URL: http://ieeexplore.ieee.org/document/6836253/ (visited on 03/08/2022).

[43] Paul T. Grogan and Olivier L. de Weck. “Interactive simulation games to assess federated satellite system concepts”. In: *2015 IEEE Aerospace Conference*. Big Sky, MT: IEEE, Mar. 2015, pp. 1–13. ISBN: 978-1-4799-5380-6. DOI: 10.1109/AERO.2015.7119101. URL: http://ieeexplore.ieee.org/document/7119101/ (visited on 03/08/2022).

[44] Paul T. Grogan et al. “Bounding the value of collaboration in federated systems”. In: *2016 Annual IEEE Systems Conference (SysCon)*. Orlando, FL: IEEE, Apr. 2016, pp. 1–7. ISBN: 978-1-4673-9519-9. DOI: 10.1109/SYSCON.2016.7490657. URL: https://ieeexplore.ieee.org/document/7490657/ (visited on 03/08/2022).

[45] Paul T. Grogan et al. “Multi-Actor Value Modeling for Federated Systems”. In: *IEEE Systems Journal* 12.2 (June 2018), pp. 1193–1202. ISSN: 1932-8184, 1937-9234, 2373-7816. DOI: 10.1109/JSYST.2016.2626981. URL: https://ieeexplore.ieee.org/document/7756322/ (visited on 03/08/2022).

[46] Jonna Koivisto and Juho Hamari. “The rise of motivational information systems: A review of gamification research”. en. In: *International Journal of Information Management* 45 (Apr. 2019), pp. 191–210. ISSN: 02684012. DOI: 10.1016/j.ijinfomgt.2018.10.013. URL: https://linkinghub.elsevier.com/retrieve/pii/S0268401217305169 (visited on 11/19/2021).

[47] Katheryn R. Christy and Jesse Fox. “Leaderboards in a virtual classroom: A test of stereotype threat and social comparison explanations for women’s math performance”. en. In: *Computers & Education* 78 (Sept. 2014), pp. 66–77. ISSN: 03601315. DOI: 10.

1016/j.compedu.2014.05.005. URL: https://linkinghub.elsevier.com/retrieve/pii/S0360131514001195 (visited on 11/20/2021).

[48] Brooke A. Jones, Gregory J. Madden, and Heidi J. Wengreen. "The FIT Game: preliminary evaluation of a gamification approach to increasing fruit and vegetable consumption in school". en. In: *Preventive Medicine* 68 (Nov. 2014), pp. 76–79. ISSN: 00917435. DOI: 10.1016/j.ypmed.2014.04.015. URL: https://linkinghub.elsevier.com/retrieve/pii/S0091743514001467 (visited on 11/19/2021).

[49] Jennifer Thom, David Millen, and Joan DiMicco. "Removing gamification from an enterprise SNS". en. In: *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12*. Seattle, Washington, USA: ACM Press, 2012, p. 1067. ISBN: 978-1-4503-1086-4. DOI: 10.1145/2145204.2145362. URL: http://dl.acm.org/citation.cfm?doid=2145204.2145362 (visited on 11/19/2021).

[50] Michel-Alexandre Cardin et al. "Simulation Gaming to Study Design and Management Decision-Making in Flexible Engineering Systems". In: *2013 IEEE International Conference on Systems, Man, and Cybernetics*. Manchester: IEEE, Oct. 2013, pp. 607–614. ISBN: 978-1-4799-0652-9. DOI: 10.1109/SMC.2013.109. URL: http://ieeexplore.ieee.org/document/6721862/ (visited on 03/26/2022).

[51] Marcin Wardaszko. "Simulation Game Complexity Perception: An Approach to the Research Model". en. In: *Neo-Simulation and Gaming Toward Active Learning*. Ed. by Ryoju Hamada et al. Vol. 18. Series Title: Translational Systems Sciences. Singapore: Springer Singapore, 2019, pp. 473–484. ISBN: 9789811380396. DOI: 10.1007/978-981-13-8039-6_45. URL: http://link.springer.com/10.1007/978-981-13-8039-6_45 (visited on 03/09/2022).

[52] Maarten P.D. Schadd et al. "Single-player Monte-Carlo tree search for SameGame". en. In: *Knowledge-Based Systems* 34 (Oct. 2012), pp. 3–11. ISSN: 09507051. DOI: 10.1016/j.knosys.2011.08.008. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950705111001869 (visited on 10/26/2021).

[53] Bill Roungas and Sebastiaan Meijer. "Towards the Management and Dissemination of Knowledge from Gaming Simulations". en. In: *Serious Games*. Ed. by Minhua Ma et al. Vol. 12434. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, pp. 276–288. ISBN: 978-3-030-61814-8. DOI: 10.1007/978-3-030-61814-8_21. URL: http://link.springer.com/10.1007/978-3-030-61814-8_21 (visited on 03/09/2022).

[54] Sebastian Deterding et al. "Gamification. using game-design elements in non-gaming contexts". en. In: *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11*. Vancouver, BC, Canada: ACM Press, 2011, p. 2425. ISBN: 978-1-4503-0268-5. DOI: 10.1145/1979742.1979575. URL: http://portal.acm.org/citation.cfm?doid=1979742.1979575 (visited on 02/21/2022).

[55] Sebastian Deterding. "The Lens of Intrinsic Skill Atoms: A Method for Gameful Design". en. In: *Human–Computer Interaction* 30.3-4 (May 2015), pp. 294–335. ISSN: 0737-0024, 1532-7051. DOI: 10.1080/07370024.2014.993471. URL: http://www.tandfonline.com/doi/full/10.1080/07370024.2014.993471 (visited on 03/17/2022).

[56] Jitesh H. Panchal, Zhenghui Sha, and Karthik N. Kannan. "Understanding Design Decisions Under Competition Using Games With Information Acquisition and a Behavioral Experiment". en. In: *Journal of Mechanical Design* 139.9 (Sept. 2017), p. 091402. ISSN: 1050-0472, 1528-9001. DOI: 10.1115/1.4037253. URL: https://asmedigitalcollectio asme.org/mechanicaldesign/article/doi/10.1115/1.4037253/383838/Understanding-Design-Decisions-Under-Competition (visited on 03/08/2022).

[57] Yi Ren, Alparslan Emrah Bayrak, and Panos Y. Papalambros. "EcoRacer: Game-Based Optimal Electric Vehicle Design and Driver Control Using Human Players". en. In: *Journal of Mechanical Design* 138.6 (June 2016), p. 061407. ISSN: 1050-0472, 1528-9001. DOI: 10.1115/1.4033426. URL: https://asmedigitalcollection.

asme.org/mechanicaldesign/article/doi/10.1115/1.4033426/472540/EcoRacer-GameBased-Optimal-Electric-Vehicle-Design (visited on 03/08/2022).

[58] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Dec. 2010. ISBN: 978-1-4008-3538-6. DOI: 10.1515/9781400835386. URL: https://www.degruyter.com/document/doi/10.1515/9781400835386/html (visited on 07/22/2022).

[59] Dimitri P. Bertsekas. *Dynamic programming and optimal control. volume 1*. eng. Fourth edition. Belmont, Mass: Athena Scientific, 2017. ISBN: 978-1-886529-43-4.

[60] John Bather. *Decision theory: an introduction to dynamic programming and sequential decisions*. Wiley-Interscience series in systems and optimization. Chichester ; New York: Wiley, 2000. ISBN: 978-0-471-97649-3.

[61] *Launches*. URL: https://www.spacex.com/launches/.

[62] Lake A. Singh et al. "Low cost satellite constellations for nearly continuous global coverage". en. In: *Nature Communications* 11.1 (Dec. 2020), p. 200. ISSN: 2041-1723. DOI: 10.1038/s41467-019-13865-0. URL: http://www.nature.com/articles/s41467-019-13865-0 (visited on 10/29/2022).

[63] James Richard Wertz. *Mission geometry: orbit and constellation design and management: spacecraft orbit and attitude systems*. Space technology library 13. El Segundo, Calif. : Dorfrecht ; Boston: Microcosm Press : Kluwer Academic Publishers, 2001. ISBN: 978-0-7923-7148-9.

[64] Philipp Elbert, Soren Ebbesen, and Lino Guzzella. "Implementation of Dynamic Programming for n-Dimensional Optimal Control Problems With Final State Constraints". In: *IEEE Transactions on Control Systems Technology* 21.3 (May 2013), pp. 924–931. ISSN: 1063-6536, 1558-0865. DOI: 10.1109/TCST.2012.2190935. URL: http://ieeexplore.ieee.org/document/6178777/ (visited on 12/26/2022).

[65] D. Li et al. "Mitigation of Curse of Dimensionality in Dynamic Programming". en. In: *IFAC Proceedings Volumes* 41.2 (2008), pp. 7778–7783. ISSN: 14746670. DOI: 10.3182/20080706-5-KR-1001.01315. URL: https://linkinghub.elsevier.com/retrieve/pii/S1474667016401989 (visited on 07/22/2022).

[66] Federico Miretti, Daniela Misul, and Ezio Spessa. "DynaProg: Deterministic Dynamic Programming solver for finite horizon multi-stage decision problems". en. In: *SoftwareX* 14 (June 2021), p. 100690. ISSN: 23527110. DOI: 10.1016/j.softx.2021.100690. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352711021000352 (visited on 07/22/2022).

[67] Michiel van Genuchten and Les Hatton. "Compound Annual Growth Rate for Software". In: *IEEE Software* 29.4 (2012), pp. 19–21. DOI: 10.1109/MS.2012.79.

[68] Michael Meder and Brijnesh-Johannes Jain. "The Gamification Design Problem". In: *arXiv:1407.0843 [cs]* (July 2014). arXiv: 1407.0843. URL: http://arxiv.org/abs/1407.0843 (visited on 03/17/2022).

[69] Glen Robertson and Ian Watson. "A Review of Real-Time Strategy Game AI". In: *AI Magazine* 35.4 (Dec. 2014), pp. 75–104. ISSN: 2371-9621, 0738-4602. DOI: 10.1609/aimag.v35i4.2478. URL: https://ojs.aaai.org/index.php/aimagazine/article/view/2478 (visited on 10/20/2021).

[70] Martha Amram and Tabitha Crawford. "The Upside to Fiscal Challenges: Innovative Partnerships Between Public and Private Sector". en. In: *Journal of Applied Corporate Finance* 23.3 (Sept. 2011), pp. 53–59. ISSN: 10781196. DOI: 10.1111/j.1745-6622.2011.00341.x. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1745-6622.2011.00341.x (visited on 04/10/2023).

[71] John R. Barry, Edward A. Lee, and David G. Messerschmitt. *Digital Communication*. en. Boston, MA: Springer US, 2004. ISBN: 978-1-4615-0227-2. DOI: 10.1007/978-1-4615-0227-2. URL: http://link.springer.com/10.1007/978-1-4615-0227-2 (visited on 03/25/2022).

[72] N.L. Johnson et al. "NASA's new breakup model of evolve 4.0". In: *Advances in Space Research* 28.9 (2001), pp. 1377–1384. ISSN: 0273-1177. DOI: https://doi.org/10.1016/S0273-1177(01)00423-9. URL: https://www.sciencedirect.com/science/article/pii/S0273117701004239.

[73] Binbin Zhang, Zhaokui Wang, and Yulin Zhang. "An analytic method of space debris cloud evolution and its collision evaluation for constellation satellites". en. In: *Advances in Space Research* 58.6 (Sept. 2016), pp. 903–913. ISSN: 02731177. DOI: 10.1016/j.asr.2016.03.016. URL: https://linkinghub.elsevier.com/retrieve/pii/S0273117716300412 (visited on 01/24/2022).

[74] Shuyi Ren et al. "The Interaction between the LEO Satellite Constellation and the Space Debris Environment". en. In: *Applied Sciences* 11.20 (Oct. 2021), p. 9490. ISSN: 2076-3417. DOI: 10.3390/app11209490. URL: https://www.mdpi.com/2076-3417/11/20/9490 (visited on 01/24/2022).

[75] Donald J. Kessler. "Derivation of the collision probability between orbiting objects: the lifetimes of jupiter's outer moons". en. In: *Icarus* 48.1 (Oct. 1981), pp. 39–48. ISSN: 00191035. DOI: 10.1016/0019-1035(81)90151-2. URL: https://linkinghub.elsevier.com/retrieve/pii/0019103581901512 (visited on 03/12/2022).

[76] Chi-Bin Cheng. "Solving a sealed-bid reverse auction problem by multiple-criterion decision-making methods". en. In: *Computers & Mathematics with Applications* 56.12 (Dec. 2008), pp. 3261–3274. ISSN: 08981221. DOI: 10.1016/j.camwa.2008.09.011. URL: https://linkinghub.elsevier.com/retrieve/pii/S0898122108004896 (visited on 11/17/2021).

[77] Gwo-Hshiung Tzeng and Jih-Jeng Huang. *Multiple attribute decision making: methods and appliations*. CRC Press, Boca Raton, FL, 2012.

[78] Taho Yang and Chih-Ching Hung. "Multiple-attribute decision making methods for plant layout design problem". In: *Robotics and Computer-Integrated Manufacturing* 23.1 (2007), pp. 126–137. ISSN: 0736-5845. DOI: https://doi.org/10.1016/j.rcim.

2005.12.002. URL: https://www.sciencedirect.com/science/article/pii/S0736584506000044.

[79] Kelic Andjelka. "Assessing the technical and financial viability of broadband satellite systems using a cost per T1 minute metric". PhD thesis. Massachusetts Institute of Technology, May 1998.

[80] Thomas Colvin, John Karcz, and Grace Wusk. *Cost and Benefit Analysis of Orbital Debris Remediation*. report. NASA, Mar. 2023. URL: https://ntrs.nasa.gov/citations/20230002817 (visited on 04/18/2023).

[81] Greg Brockman et al. *OpenAI Gym*. 2016. eprint: arXiv:1606.01540.

[82] Antonin Raffin et al. *Stable Baselines3*. https://github.com/DLR-RM/stable-baselines3. 2019.

[83] J Terry et al. "Pettingzoo: Gym for multi-agent reinforcement learning". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 15032–15043.

[84] OpenAI et al. "Dota 2 with Large Scale Deep Reinforcement Learning". In: (2019). arXiv: 1912.06680. URL: https://arxiv.org/abs/1912.06680.

Appendix A

Bellman Optimality & Dynamic Programming Proofs

Theorem A.1. *(Bellman Principle of Optimality) Let* $\pi^* = \{d_0^*, d_1^*, ..., d_{T-1}^*\}$*, be an optimal policy for a DP defined by* $\{T, S, A, C, L\}$*. Assume that using* π^* *you would arrive at state* s_t *at time* t*. Now consider the sub-problem whereby at* s_t *at time* t*, with a horizon of* $T - t$ *periods remaining you wish to minimize the cost-to-go* $J_t(s_t)$*, over all possible policies, for the remaining horizon* $\pi(t) = \{d_t, d_{t+1}, ..., d_{T-1}\}$*. That is:*

$$J_t^{\pi(t)}(s_t) = c_T(s_T) + \sum_{k=t}^{T-1} c_k(s_k, d_k(s_k)) \tag{A.1}$$

Then, the truncated policy $\pi^*(t) = \{d_t^*, d_{t+1}^*, ..., d_{T-1}^*\}$ *is optimal for the sub-problem.*

Proof. We proceed by contradiction. Assume there exists another truncated policy $\pi'(t) = \left\{d_t', d_{t+1}', \ldots, d_{T-1}'\right\}$, such that $J_t^{\pi'(t)}(s_t) < J_t^{\pi^*(t)}(s_t)$. Now consider a complete policy π', which is identical to π^* until period $t-1$ and then identical to $\pi'(t)$. That is $\pi'(t) = \left\{d_0^*, d_1^*, \ldots, d_{t-1}^*, d_t', d_{t+1}', \ldots, d_{T-1}'\right\}$. Then would have for some s_0 that:

$$J_0^{\pi'}(s_0) = \sum_{k=0}^{t} c_k(s_k, d_k^*(s_k)) + J_t^{\pi'(t)}(s_t) < \sum_{k=0}^{t} c_k(s_k, d_k^*(s_k)) + J_t^{\pi^*(t)}(s_t) = J_0^{\pi^*}(s_0) \tag{A.2}$$

which contradicts the optimality of π^*. $\square$

Theorem A.2. *Consider a DP defined by* $\{T, S, A, C, L\}$*. Then for every possible initial state* s_0*, the optimal cost* $J^*(s_0)$ *is equal to* $J_0(s_0)$ *given by the last step of the Backwards Dynamic Programming Algorithm, which proceeds backwards in time from period* T *to period* 0*.*

Furthermore if you define d_t^* *in such a way that* $a_t^* = d_t^*(s_t)$ *is a minimizer in line (3) of the algorithm, for every t and* s_t*, then the policy* $\pi^*(t) = \{d_t^*, d_{t+1}^*, ..., d_{T-1}^*\}$ *is optimal for DP.*

Proof. For any feasible policy $\pi = \{d_0, d_1, \ldots, d_{T-1}\}$, and for any t, let $\pi(t) = \{d_t, d_{t+1}, \ldots, d_{T-1}\}$. Also, for each t let $J_t^*(s_t)$ be the optimal cost for the problem with $T - t$ periods, which begins at state s_t at time t, and ends at time T. That is,

$$J_t^*(s_t) = \min_{\pi(t)} \left\{ c_T(s_T) + \sum_{k=t}^{T-1} c_k(s_k, d_k(s_k)) \right\} \tag{A.3}$$

We proceed by induction. We have by definition $J_T(s_T) = c_T(s_T)$, so it is obvious that $J_T^*(s_T) = J_T(s_T) = c_T(s_T)$ for each $s_t \in \mathcal{S}$. Now consider an arbitrary time t and assume that $J_{t+1}^*(s_{t+1}) = J_{t+1}(s_{t+1})$ for every state $s_{t+1} \in \mathcal{S}$. Then we have

$$\begin{aligned} J_t^*(s_t) &= \min_{\pi(t)} \left\{ c_T(s_T) + \sum_{k=t}^{T-1} c_k(s_k, d_k(s_k)) \right\} \\ &= \min_{d_t, \pi(t+1)} \left\{ c_t(s_t, d_t(s_t)) + c_T(s_T) + \sum_{k=t+1}^{T-1} c_k(s_k, d_k(s_k)) \right\} \end{aligned} \tag{A.4}$$

Using the Principle of Optimality:

$$\begin{aligned} J_t^*(s_t) &= \min_{d_t, \pi(t+1)} \left\{ c_t(s_t, d_t(s_t)) + c_T(s_T) + \sum_{k=t+1}^{T-1} c_k(s_k, d_k(s_k)) \right\} \\ &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + \min_{\pi(t+1)} c_T(s_T) + \sum_{k=t+1}^{T-1} c_k(s_k, d_k(s_k)) \right\} \end{aligned} \tag{A.5}$$

Using the definition of J_{t+1}^*:

$$\begin{aligned} J_t^*(s_t) &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + \min_{\pi(t+1)} c_T(s_T) + \sum_{k=t+1}^{T-1} c_k(s_k, d_k(s_k)) \right\} \\ &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}^*(L_t(s_t, d_t(s_t))) \right\} \end{aligned} \tag{A.6}$$

By induction:

$$\begin{aligned} J_t^*(s_t) &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}^*(L_t(s_t, d_t(s_t))) \right\} \\ &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}(L_t(s_t, d_t(s_t))) \right\} \end{aligned} \tag{A.7}$$

which simplifies to the following:

$$\begin{aligned} J_t^*(s_t) &= \min_{d_t} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}(L_t(s_t, d_t(s_t))) \right\} \\ &= \min_{a_t \in A_t(s_t)} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}(L_t(s_t, a_t)) \right\} \end{aligned} \tag{A.8}$$

$$\begin{aligned} J_t^*(s_t) &= \min_{a_t \in A_t(s_t)} \left\{ c_t(s_t, d_t(s_t)) + J_{t+1}(L_t(s_t, a_t)) \right\} \\ &= J_t(s_t) \end{aligned} \tag{A.9}$$

Because we picked an arbitrary t and s_t, this concludes the proof. $\square$