Support of Latency-sensitive Space Exploration Applications in Future Space Communication Systems

by

Marc Sanchez Net

M.S., Massachusetts Institute of Technology (2014)
B.S., Universitat Politecnica de Catalunya (2012)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
April 2017 [June 2017]

Signature redacted

Author ....
Department of Aeronautics and Astronautics
April 24, 2017

Signature redacted

Certified by......
Prof. Edward F. Crawley
Ford Professor of Aeronautics and Astronautics
Thesis Supervisor

Signature redacted

Certified by......
Prof. Eytan H. Modiano
Professor of Aeronautics and Astronautics
Committee Member

Signature redacted

Certified by.
Dr. Bruce G. Cameron
Director of the MIT Systems Architecture Laboratory
Committee Member

Signature redacted

Certified by...
Dr. Kar-Ming Cheung
Technical Group Supervisor at NASA Jet Propulsion Laboratory
Committee Member

Signature redacted

Accepted by ....
Prof. Youssef M. Marzouk
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Support of Latency-sensitive Space Exploration Applications in Future Space Communication Systems

by

Marc Sanchez Net

Submitted to the Department of Aeronautics and Astronautics
on April 24, 2017, in partial fulfillment of the
requirements for the degree of
DOCTOR OF PHILOSOPHY

Abstract

Latency, understood as the total time it takes for data acquired by a remote platform (e.g. satellite, rover, astronaut) to be delivered to the final user in an actionable format, is a primary requirement for several near Earth and deep space exploration activities. Some applications such as real-time voice and videoconferencing can only be satisfied by providing continuous communications links to the remote platform and enforcing hard latency requirements on the system. In contrast, other space exploration applications set latency requirements because their data's scientific value is dependent on the timeliness with which it is delivered to the final user. These applications, henceforth termed latency-sensitive, are the main focus of this thesis, as they typically require large amounts of data to be returned to Earth in a timely manner.

To understand how current space communication systems induce latency, the concept of network centrality is first introduced. It provides a systematic process for quantifying the relative importance of heterogeneous latency contributors, ranking them, and rapidly identifying bottlenecks when parts of the communication infrastructure are modified. Then, a custom-designed centrality measure is integrated within the system architecture synthesis process. It serves as a heuristic function that prioritizes parts of the system for further in-depth analysis and renders the problem of analyzing end-to-end latency requirements manageable.

The thesis includes two primary case studies to demonstrate the usefulness of the proposed approach. The first one focuses on return of satellite-based observations for accurate weather forecasting, particularly how latency limits the amount of data available for assimilation at weather prediction centers. On the other hand, the second case study explores how human science operations on the surface of Mars dictate the end-to-end latency requirement that the infrastructure between Mars and Earth has to satisfy.

In the first case study, return of satellite observations for weather prediction during the 2020-2030 decade is analyzed based on future weather satellite programs. Recommendations on how to implement their ground segment are also presented as a function of cost, risk and weather prediction spatial resolution. This case study also serves as proof of concept for the proposed centrality measure, as ranking of latency contributors and network implementations can be compared to current and proposed systems such as JPSS' Common Ground Infrastructure and NPOESS' SafetyNet.

The second case study focuses on supporting human science exploration activities on the surface of Mars during the 2040's. It includes astronaut activity modeling, quantification of Mars Proximity and Mars-to-Earth link bandwidth requirements, Mars relay sizing and ground infrastructure costing as a function of latency requirements, as well as benchmarking of new technologies such as optical communications over deep space links. Results indicate that levying tight latency requirements on the network that support human exploration activities at Mars is unnecessary to conduct effective science and incurs in significant cost for the Mars Relay Network, especially when no optical technology is present in the system. When optical communications are indeed present, mass savings for the relay system are also possible, albeit trading latency vs. infrastructure costs is less effective and highly dependent on the performance of the deep space optical link.

Thesis Supervisor: Prof. Edward F. Crawley
Title: Ford Professor of Aeronautics and Astronautics

Committee Member: Prof. Eytan H. Modiano
Title: Professor of Aeronautics and Astronautics

Committee Member: Dr. Bruce G. Cameron
Title: Director of the MIT Systems Architecture Laboratory

Committee Member: Dr. Kar-Ming Cheung
Title: Technical Group Supervisor at NASA Jet Propulsion Laboratory

Acknowledgments

First, I would like to thank Ed and Bruce for their support during these five and a half years at MIT. Their leadership and advice has been an invaluable source of inspiration, guidance and support. I would like to thank the other members of my committee, Kar-Ming and Eytan, as well as my thesis readers, Paul and Sam, for the time they devoted to me, this research and this thesis. PhDs are winding roads with more curves than straight lines, but your directions have been essential for phrasing the research questions and figuring out how to tackle them.

409 is a "magical" place. Time passes, faces change, but the quality of people and their hard-working nature never changes. To the now emeritus members of 409, Dani, Alessandro, Alexander, Francisco, Jonathan, Emily, and Paul, thank you for welcoming me into the MIT community during my first years in Boston. You made the transition as smooth and easy as possible and showed me how great this place is. In that sense, a huge shout-out to Dani, my partner in crime during multiple programming nights, runs around the river and trips across the country! To the senior members of 409, Peter, Morgan, Sydney, Narek, and Demetrios, thank you for making MIT a memorable place to be in on a daily basis. From basketball games and country concerts with Peter, to nights out with Morgan and gym sessions with Narek, Sydney and Demetrios, every minute has been exciting, fun, and has contributed to making this experience unique and unforgettable. Finally, to the current members of the 409 gang, Inigo, Anne, Veronica, Andrew, Sam, John, Juanjo, and Markus, thank you for your support during the last stages of my journey. I hope I was able to inspire you the same way that you all did for me. In particular, a huge shout-out to Inigo, I couldn't have done it without you!

I also would like to take a moment to acknowledge my sources of funding and data. During the first three years, NASA's Space Communication and Navigation Program fueled my interest in space communications and allowed me to work on it almost exclusively. The baton was then handed over to JPL, and specially Kar-Ming, who allowed me to work alongside with him for two summers and has been an invaluable source of domain-specific knowledge throughout this research.

To my friends here in the US - Ana, Maria, Fernando, Ainara, Jorshua - and Barcelona - Jordi, Manel, Roser, Anna, Gerard, Gloria, Matias - thank you for your encouragement and friendship throughout this PhD. Distance and time zones have set us apart for most of the trip, but your messages and visits have been essential to its success. Similarly, a huge *thank you* to my family both in Barcelona and California. Special thanks to my mother, who encouraged me to work hard and always helped me find the best opportunities that eventually led to this day. And most of all, I would like to thank Romina, for navigating through the PhD journey with me and making everyday unique and exciting. MIT and

Boston have given me many opportunities, but none as valuable and important as the chance of meeting you.

Contents

List of Figures

List of Tables

1 INTRODUCTION

1.1 Context

Communications and navigation services are essential to the success of any space mission. Indeed, data produced on-board a spacecraft is only useful if it is returned to Earth, processed and analyzed by the scientific community. Similarly, insertion into Mars or lunar orbit is only possible if a mission operator is able to precisely determine the spacecraft's position and adjust its trajectory accordingly. Despite the critical nature of communication networks for space programs, their importance is seldom acknowledged. After all, their presence is usually "invisible" to the broad public, who is typically more interested and excited by incredible exploration feats such as landing a human on the Moon or a rover on the Mars surface.

Despite being "invisible" to the public eye, NASA spends significant resources building, maintaining and upgrading communication networks to support their missions (e.g. approximately $1.4B is spent per TDRS generation, including both the space and ground segments [16]). In that sense, the Space Communication and Navigation Program (SCaN) was originally created in 2006 as part of the Human Exploration and Operations Directorate. It oversees, funds and manages the three networks that the agency has in place for providing communications in Earth orbit, the cis-lunar space and deep space. Their objective is twofold: First, ensure that communication and navigation services are being reliably delivered to all missions currently in operations, especially during critical events or in case of unexpected contingency situations. Second, pave the way for ever more capable spacecraft that seek to gather and return larger amounts of data, contact with Earth more often, and navigate more precisely.

This dissertation fits within the second aforementioned objective. Its primary goal is to support and foster the development of effective future space communication networks that will provide services to missions with data timeliness requirements. This is essential since we live in an era where more and timelier data is being increasingly demanded, and yet resources available to design and build space networks that meet the required performance are stagnated at best. Consequently, throughout this research I expand the system boundary to include not just the networks themselves, but also the missions and scientists that interact with them. This approach provides interesting insights into which parts of the space data production and analysis system are the bottleneck, and let's me identify and focus on areas with high expected return on investment.

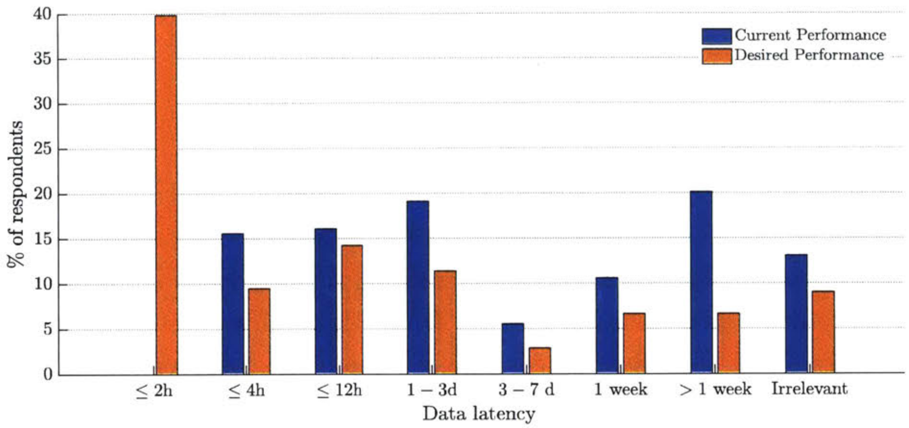


Figure 1-1: Latency Survey Results (Adapted from Reference [1])

1.2 Motivation

The SCaN Architecture Definition Document [17] defines the evolution roadmap for NASA's space communication networks, as well as the high level capacity requirements to provide across the solar system and their rationale. While this provides a first anchor point and vision on how NASA networks will evolve in the coming decades, two questions arise: Will the resulting network be able to successfully satisfy missions that produce large volumes of data and must return it in a timely manner? Second, what will be the cost of implementing these networks and, more importantly, can we afford it?

To exemplify the first problem, NASA's Earth Science Division has already been studying the need for data timeliness for Earth-related data products. In that sense, in 2013 they conducted a survey with more than 500 participants to understand how fast data is being provided to scientists, how fast they would actually like it should there be no system constraints, and what is the added value of providing satellite observations and their high-level data products with lower overall latency. Figure 1-1, adapted from Reference [1], depicts the difference between desired and current performance for the end-to-end production system that take space-based measurements, processes them and delivers the resulting products to the end scientists. Interestingly, observe the clearly expressed desire to increase data timeliness for Earth observation data products. Indeed, Molly [1] found that lower data latency could improve active fire analysis and tactical response, as well as be beneficial for ocean coastal applications, weather forecasting, park monitoring and agriculture assessment among others. She also concluded that "although current uses of low latency data are [...] limited, there is an enormous opportunity to expand these uses far beyond what is done today" [1].

Molly also indicated that "latency [...] can be minimized through investment of resources,

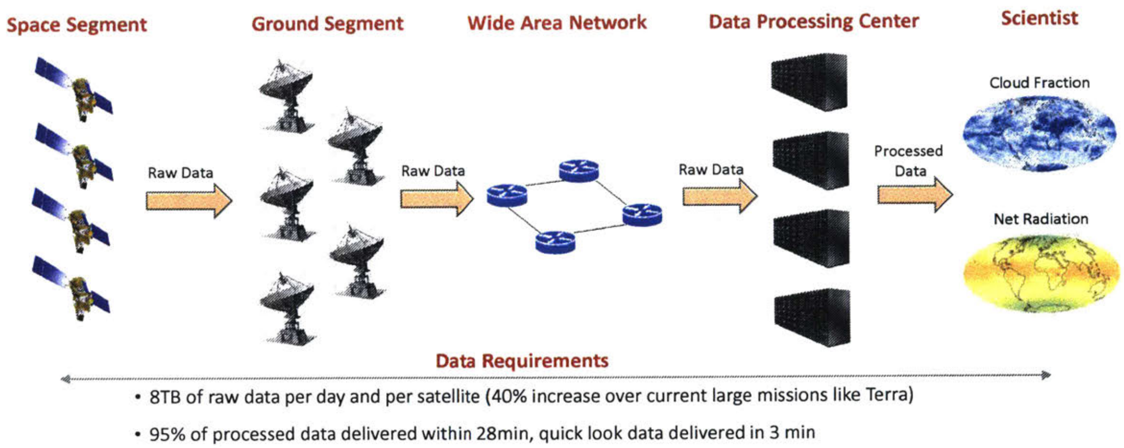

Space Segment
Ground Segment
Wide Area Network
Data Processing Center
Scientist
Cloud Fraction
Raw Data
Raw Data
Raw Data
Processed Data
Net Radiation
Data Requirements
• 8TB of raw data per day and per satellite (40% increase over current large missions like Terra)
• 95% of processed data delivered within 28min, quick look data delivered in 3 min

Figure 1-2: NPOESS Production System

but there are significant constraints on the ability to reduce time spent in some stages" [1]. This is essentially equivalent to my second original question: How much can we afford to spend in developing a more timely production system for science data products given the current budgetary constraints? For instance, consider the failed NPOESS program (see Figure 1-2). It was supposed to deploy a highly distributed space-to-ground network, with up to 14 ground stations interconnected through a WAN that would repatriate satellite observations to four state of the art processing centers. These would generate the L1, L2 and L3 data products in near real-time and disseminate them to both the end users and data archiving systems. As indicated by Dwyer [18], "in order to produce high quality data products according to the DoD's driving latency requirement, the ground system utilized a complex and costly parallel processing system" that could have been reduced "if latency requirements for non-time sensitive EDRs were loosened". While the ground segment was not the sole contributor to the cost and schedule overruns experienced by the NPOESS program, it certainly increased complexity and ultimately contributed to the entire program failure. In fact, after NPOESS was canceled, its successor JPSS reduced the number of ground sites and processing centers, and the end-to-end latency requirement was increased from 15 to 100-180 minutes depending on the data product.

1.2.1 Performance vs. Cost in Communications for Space Exploration

I previously argued that providing communications for space exploration purposes is a costly endeavor. To substantiate this claim, in this section I provide a succinct summary of how the budget and capabilities of NASA's space communication program have evolved over time. In that sense, Figure 1-3 depicts the evolution of communication capability for the DSN, expressed in channel data rate at Jupiter. Observe that between 2005 (i.e. since SCaN's inception) and today, the achievable data rate has been increased by two orders of

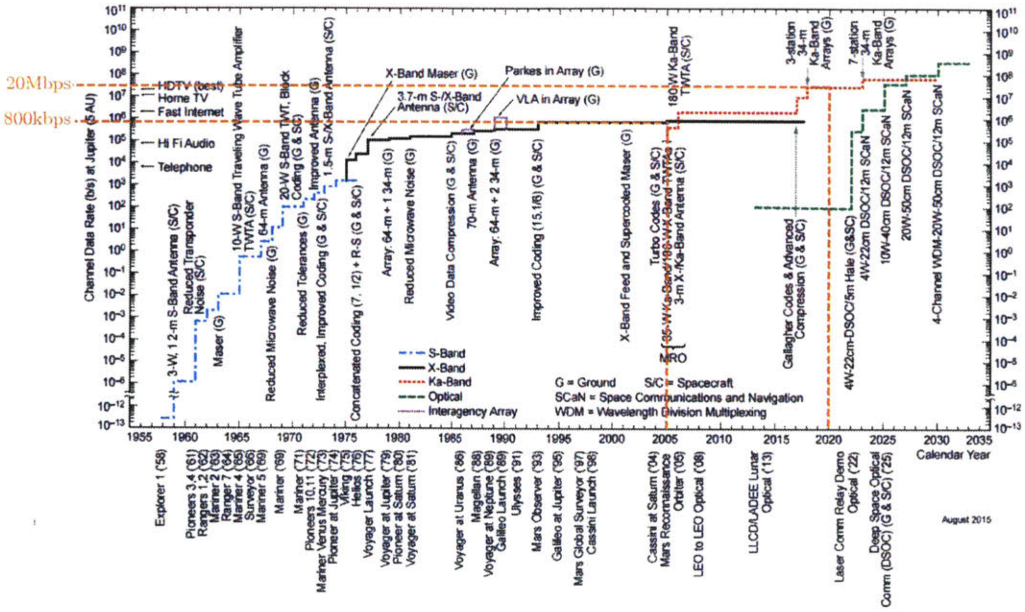


Figure 1-3: DSN Performance Evolution (Adapted from Reference [2])

magnitude, from around 800kbps to 20Mbps. In other words, if the network was supporting traditional terrestrial services, in 2005 we were able to deliver high definition audio or slow-speed Internet to a user in Jupiter, while in 2020 we will be able to provide high definition video.

On the other hand, Figure 1-4 plots the evolution of funding for the DSN, NEN and SN in the same period of time, with the budget normalized with respect to the year 2015, both in terms of fiscal year inflation and metric value. In that sense, while the amount of resources allocated by NASA to its networks has increased since 2005 (see the two spikes around the year 2010, which correspond to the development and launch of the 3rd generation TDRSS), the total funding has essentially remained constant in the last five years and is predicted to decline in the coming years. Importantly, observe the mismatch between performance and cost. While the former has improved by almost two orders of magnitude, the latter is stagnated at its current value and forecasts are not optimistic.

Given the observed constraints in resources for future space communication networks, the persistent request to generate and return more data from remote spacecraft and, as Molly indicated, the increasingly stringent latency requirements, one of the fundamental questions this research wants to address is understanding whether current and future infrastructures will be able to successfully satisfy the needs of space exploration in the future. In that sense, while previous efforts have mainly focused in the domain of data rate, i.e. how much capacity is required in the system and how to deliver it, this thesis focuses on latency and its impact in the infrastructure cost. To that end, it first introduces and categorizes space

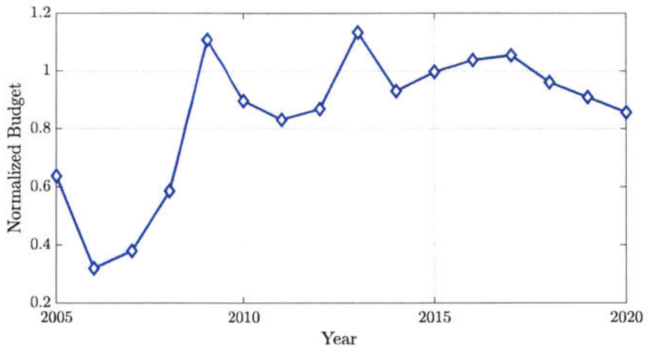


Figure 1-4: SCaN Budget Evolution

applications for which latency is an issue. Next, it explores which elements in our current infrastructures induce latency. And third, it quantifies the trades between providing low latency data products for space exploration applications, and the cost of the supporting infrastructure that returns and analyzes it.

1.3 Background

1.3.1 Latency in Space Exploration Applications

Transfer of information from an origin to a destination is never instantaneous. At a purely physical level, data sent through an electromagnetic wave that propagates inside a wire or wirelessly is delayed due to the speed of light. This is clearly the most basic form of latency since it is dictated by the laws of physics and cannot be overcome by designing better systems. Nevertheless, latency is typically also induced by other factors. Transmission delay due to limited bandwidth, inefficiencies in the communication and networking protocols, as well as restrictions in computational power are also known to be important factors to be considered when assessing the time it takes for information to reach its destination (see for instance Reference [19]).

Unfortunately, latency in communication networks has received several names. Probably the most common is delay, albeit in some cases *delay* is used to name latency caused by signal propagation. Since there is no clear agreement on which terminology to use, I now provide the basic definitions that I will consistently use throughout this document:

- **Latency**: Time it takes to deliver data acquired by a satellite to the hands of a user in an actionable format [20]. As it stands, this definition is already tailored to the space context but can easily be adapted to other applications.

- **Latency requirement**: Numerical value, in units of time, that measures the time

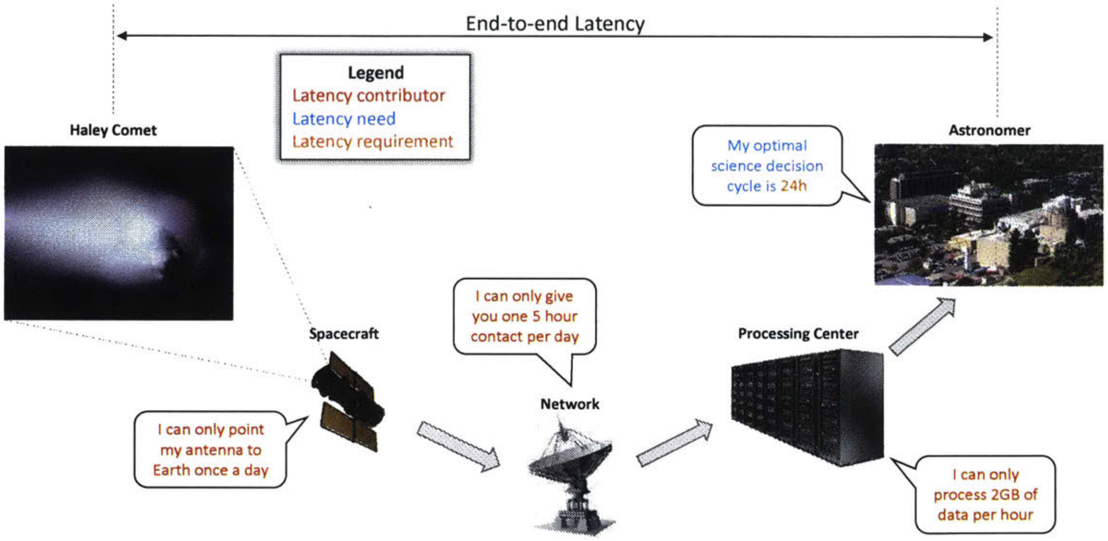

End-to-end Latency
Legend
Latency contributor
Latency need
Latency requirement
Haley Comet
Astronomer
My optimal science decision cycle is 24h
I can only give you one 5 hour contact per day
Spacecraft
Processing Center
Network
I can only point my antenna to Earth once a day
I can only process 2GB of data per hour

Figure 1-5: Example of Latency Requirement, Need and Contributor

between data acquisition and data delivery.

- **Latency need**: Scientific or operational rationale that justifies a latency requirement. It is equivalent to a stakeholder need in the system engineering literature.

- **Latency contributor**: System functionality that induces latency and ultimately retards data delivery to the end user.

Figure 1-5 provides a pictorial representation of the four aforementioned latency-related concepts in the context of a notional mission that observes the Haley comet. At the highest level, the system can be decomposed into three elements, the spacecraft, the network of antennas that provide contact opportunities to downlink the data and the processing centers that turn the spacecraft instrument's raw data into a time-tagged user friendly image. End-to-end latency is measured form the time the spacecraft instruments take a picture of Haley's comet, to the time it is fully processed and delivered to the astronomers. Examples of latency contributors include limitations on the number of times the spacecraft can point its antenna towards Earth, the frequency with which the network can provide passes to the mission, or the rate at which data can be processed. These contributors have to be managed so that the astronomers' optimal decision cycle of 24 hours is met and the spacecraft can be commanded to point its instruments to areas of high scientific value within the comet.

I will refer to latency-constrained applications as all space-related scientific activities for which the value provided by data products generated in remote assets is sensitive to, dependent, or affected by the latency incurred while transmitting it from origin to destination (see Figure 1-6). Furthermore, I will categorize latency-constraints applications into three groups that exhibit distinct properties towards latency:

- **Real-time applications**: Users at the two ends of the communication network are actively *interacting* with each other. The session's value is largely reduced if interruptions occur and interactivity is cut. Latency requirements for these types of applications are stringent and set to avoid perceiving lags while utilizing the network [14].

- **Near real-time applications**: Users at the two ends of the communication network exchange information that must be consumed in real-time upon delivery at destination. Note, however, that no interactivity between the two parties is required and therefore short term interruptions in the data stream are either unnoticeable or of minor consequence.

- **Latency-sensitive applications**: Users at the two ends of the communication network send information that must be consumed at some point before a given time horizon. This time horizon can be interpreted as a "data expiration date" and should ideally not be exceeded.

Finally, there is a wide variety of space applications for which latency is not an issue. I term them *latency-unconstrained*, because missions that perform them collect data that has no inherent latency requirement. Examples of these types of applications are climate measurements, planetary science measurements or deep space radio astronomy. Note, however, that latency-unconstrained applications is not necessarily equivalent to latency insensitive applications since scientists typically want to receive their instrument data as soon as possible. Furthermore, the mission can be sensitive to latency from an engineering perspective, since on-board storage capacity might be a limiting factor that dictates how often data should be downlinked to Earth.

1.3.2 System Architecture and System Architecting

The discipline of system's architecture was originally conceived in the late 80's in order to develop the necessary body of knowledge for engineers to successfully design and build large scale, complex systems [21]. Multiple definitions for a system architecture have been proposed in the literature. For instance, Reference [22] equates system architecture to "the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution". Similarly, Reference [23] defines system architecture as "an abstract description of the entities of a system and the relationship between those entities. In systems built by humans, this architecture can be represented by a set of decisions".

Regardless of the specific terminology, all definitions of system architecture convey the same general idea: The architecture of a system defines both its functional and physical elements and their inter-relationships, as well as the system's interfaces with its environment. In the

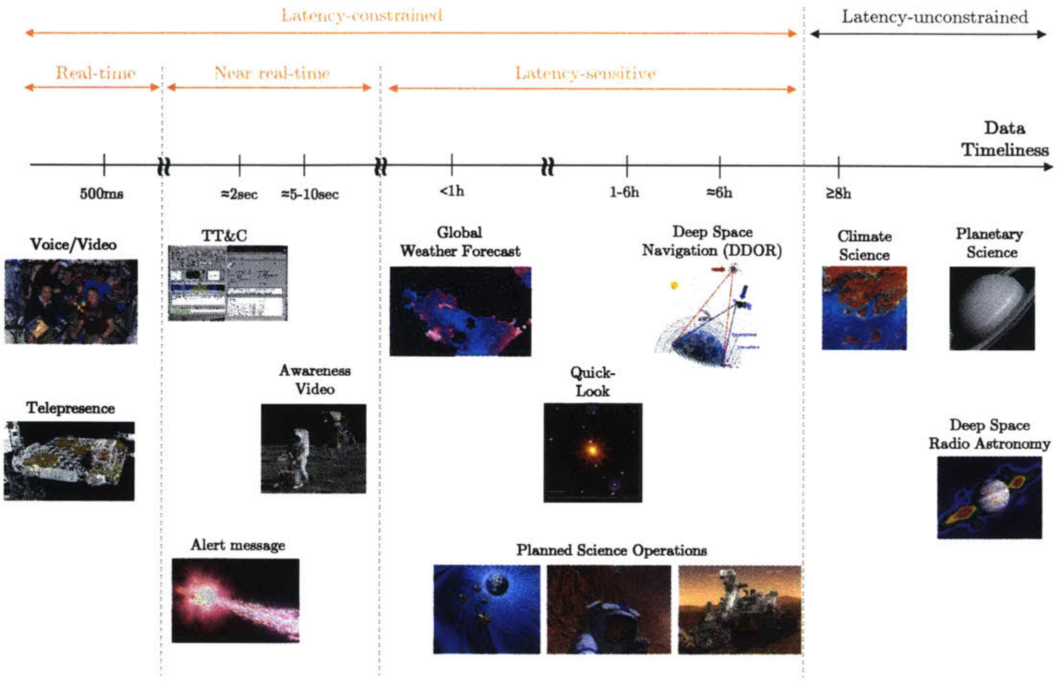


Figure 1-6: Latency Constrained vs. Latency Unconstrained Applications

words of Crawley, the system architecture is "the embodiment of a concept: The allocation of physical/informational function to elements of form, and the definitions of interfaces among them and with the surrounding context" [24].

Implicit to the previous definitions is the notion of the system's *function* and *form*. Reference [23] defines form as "the physical or informational embodiment of a system that exists [...] and is instrumental in the execution of function. Form includes the entities of form and the formal relationships among the entities". It is instrumental to the execution of function in that the system must physically exist in order to perform them. However, it is also the primary source of cost since building the system requires procuring, assembling and maintaining the different elements of form that constitute it. On the other hand, the system function can be defined as "the activity, operation or transformation that causes or contributes to performance. [...] Function is the actions for which a system exists, which ultimately lead to the delivery of value" [23]. Note the duality between function and form. The former is the primary source of benefit to the system user and therefore is the reason why the system is built. However, the system function can only be executed if the system form has been previously implemented, thus inevitably resulting in an undesired source of cost. Finally, and based on these concepts, systems architecting encompasses all practices, processes and tools that allow system architects to conceive, define, document, communicate, certify, maintain and improve complex systems throughout their life cycle [25].

1.3.3 System Architecture Synthesis, Tasks, Problems and Tools

System Architecture Synthesis

Reference [23] extensively describes the process of system architecting synthesis (SAS). Although a thorough descriptions of its steps is not relevant to this document, a broad introduction is indeed beneficial since it will be implicitly used to guide the overall structure and development of this thesis.

The first step of the SAS is to identify and characterize the system stakeholders and their needs. This includes considering both the system beneficiaries, i.e. those who will directly benefit form the system functionality, as well as the system stakeholders, that is, those that have a stake or interest on the system but do not necessarily benefit from it. Furthermore, it also includes taking into account influences from the environment during the system's life cycle. For instance, a system architecture that results in high operation costs is clearly less preferable than one easy to maintain. Similarly, a system architecture that challenges the corporate strategy or the limits imposed by external regulators might be undesirable regardless of its outstanding technical performance.

Once the system stakeholders and needs have been identified, the next step is to transform these needs into a set of goals for the system. While needs are defined by the system beneficiaries and broadly express their overall desire, goals are set by the system architect and concisely specify what the system should accomplish [23]. Goals are important for the system architect because they explicitly identify the set of solution-neutral functions that the system has to execute in order to deliver value to the beneficiaries.

The third step in the SAS is tied to the process of concept generation. The concept of a system is the "vision, idea, notion or mental image that maps function to form" [23]. The goal of the concept generation phase is twofold: First, it fosters creativity during the architecting process by transforming the goals' solution-neutral functions into alternative system functionality that can address them. Second, it clarifies the system value delivery path by highlighting the set of functions that are fundamental to successfully satisfying the system's goals.

Finally, the last step to synthesize a system architecture is to map the set of functionality identified in the previous phase to a set of physical elements, their attributes and their inter-relationships. This step entails several challenges for the system architect, most notably managing the complexity inherent to the system being architected[1]. Reference [23] suggests

[1] In our context, complexity can be generally defined as "the property of having many interrelated, interconnected or interwoven elements and interfaces" [23]

multiple methods to help manage complexity such as system decomposition, modularization and tradespace exploration.

System Architecture Tasks and Problems

Based on the description of the SAS, Selva identified in Reference [26] five canonical system architecture tasks (SATs) that the system architect has to address:

- Function-to-form mapping, or the allocation of system functionality to elements of form. This happens mostly during the 4th phase of the SAS.
- Decomposition/Aggregation of function or form, which occurs during the 4th phase of the SAS (possibly during phase 3 too) to help the architect manage complexity.
- Specialization of function and form, which happens while generating system concepts (3rd phase).
- Form attribute selection, that occurs both during the 3rd and 4th phase of the SAS and is mostly tied to defining the architecture's key attributes.
- Structural relationship and interface definition, also related to the architecture definition (4th step).
- Reactive and proactive commonality, which appear mostly during the 4th phase of the SAS.

At the same time, Simmons noted that the 4th step of SAS process can be in general structured through a set of decisions that generate the space of valid architectural options [27]. Each decision is represented by a decision variable with a finite set of mutually exclusively options that, when assigned to an architectural decision, result in a new system architecture. This definition was later expanded by Selva [28], who pointed out that even though all architectural tasks can be indeed characterized using Simmons decision-option assignment formulation, in some cases other mathematical patterns might be more intuitive. As a result, he extended Simmons' definition of an architectural decision to different types of constrained combinatorial problems and characterized their desirability with respect to each aforementioned SAT.

On the other hand, once system architectures have been systematically generated using combinatorial patterns, it is necessary discriminate and ultimately down-select the best ones. In other words, if multiple alternatives to implement a given system are feasible, then comparing them and choosing the "best" one is a critical task of the system architect. Reference [29] provides an overview of different methods accomplish this task in the context of Earth observation programs and their socioeconomic impact. The central idea conveyed by the document is that choosing the "best" architecture is essentially a project valuation exercise. In that sense, the goal of the system architect is to select the system architecture that delivers the highest value given a set of predefined metrics and constraints. In the engineering domain, this valuation exercise is typically conducted as a cost-benefit analysis [29]. Other types of projects, such as financial endeavors, favor monetized valuation approaches (e.g. net present value) by taking advantage of the fact that system benefit (or monetary revenue) is additive with system cost (or monetary investment) [29].

Additional considerations, typically referred to as *ilities*, are also of capital importance in engineering projects [30]. In de Weck's words, "*ilities* are desired properties of systems (...) that are not the primary functional requirements of a system's performance, but typically concern wider system impacts with respect to time and stakeholders than are embodied in those primary functional requirements" [30]. Reliability, flexibility, robustness, maintainability or riskiness are common examples of *ilities*. They are typically introduced during the architecting phase of an engineering system as extra dimensions of the cost-benefit analysis [31]. As a result, the cost-benefit analysis is typically formulated as a multi-objective optimization problem with decoupled orthogonal metrics [23]. Once the optimization problem has been solved, results are typically represented in a tradespace, a multi-objective plot where a large number of system architectures are benchmarked using a finite set of metrics [23]. Tradespaces are typically built around the notion of Pareto dominance, which ultimately results in the identification of a set of optimal architectures (also referred to as Pareto Front). In that sense, all architectures in the Pareto Front of a tradespace are equally optimal. Therefore, selecting one or another can only be done through additional criteria such as budget caps, minimum performance requirements, or extra *ilities* not initially captured in the original cost-benefit tradespace.

System Architecture Tools

In Section 1.3.3, I stated that system architecture is used during the high-level design of complex systems, where complexity typically arises from the large number of elements and interfaces that compose it. Moreover, it also indicated that managing this complexity can be accomplished by formulating the system architecting process as a constrained combinatorial problem over a set of architectural decisions that map its functionality to its form. Therefore, it is apparent that system architecture problems are prone to suffer from challenges inherent to combinatorial explosion [32]. In other words, the number of possible system architectures growths rapidly as new decisions are included, thus hindering the ability of humans to successfully explore and understand the resulting architectural space.

To alleviate this problem, different system architecture tools have been developed in the past. For instance, Koo developed Object-Process Network (OPN), a software-based system architecture tool that implemented an algebraic formulation of the system architecture process and provided an integrated environment for architecture enumeration, simulation and visualization. OPN was used in Reference [27] to study the architecture of the Apollo program. Nevertheless, OPN's inability to handle different SAP combinatorial formulations led to the development of enhanced tools grounded in state-of-the-art optimization techniques. Indeed, Reference [33] indicates that "space exploration for architecting purposes is an NP-hard problem [...] and therefore evolutionary and heuristic search algorithms should be utilized to identify promising design variants at a reasonable computational cost". This finding has eventually led to the the development of advanced system architecture tools such as the one presented in References [34] and [28] for architecting human space exploration campaigns and Earth observation systems respectively.

Finally, other types of tools useful during the system architecture synthesis are also available in the literature. For instance, the System Modeling Language (SysML) [35] has been widely applied in industry to represent multiple views of a given system (e.g. structural/component view vs. functional/behavioral view) and facilitate the definition of a system's architecture through multiple hierarchical and modular components [36], [37]. Other tools such as STK [38] are specifically tailored to facilitate simulation of a given architecture and thus provide a powerful and efficient engine to assess the system performance against a wide variety of metrics (e.g. revisit time, contact time, link margin, etc.).

1.4 General Problem Statement

At this point in the document, I have provided a clear definition of latency in the context of space exploration applications and proposed an initial categorization. Furthermore, I have also described the notion of system architecture and system architecting as a fundamental process by which systems can be optimized to meet their stakeholder requirements. Combining both domains leads to the formulation of this thesis' general problem statement:

> The goal of this thesis is to study the impact of latency-constrained human and robotic space exploration applications on future space network architectures by developing a set of medium fidelity system architecture tools that explore and quantify the trade-off between service provision, infrastructure cost and, whenever appropriate, risk.

Two important remarks should be clarified with respect to the proposed generic problem statement. First, at this point of the dissertation it is not possible to focus the research question towards a specific category within latency-constrained applications. This issue will be addressed in the literature review that ensues, where differences between latency requirements of real-time, near-real time and latency-sensitive applications will be explored. Similarly, the specific type of system architecture tool to be developed is still not concisely defined. This will also be addressed during the literature review once I describe the types of latency contributors to be considered for effective decision-making.

1.5 Literature Review

This section summarizes the literature relevant to further focus and specify the generic problem statement. While the present discussion is not fully exhaustive in any of the themes, it condenses the main topics and arguments relevant to this dissertation. Lessons learned from this literature review will be used in Section 1.6 to transform the generic problem statement into the thesis statement, i.e. the definition of the specific research goals addressed in the following chapters.

Four fundamental bodies of knowledge are relevant to this thesis: Architecture of space networks and their latency contributors, latency-constrained space applications, utility theory and centrality measures. They are sequentially addressed in Sections 1.5.1, 1.5.2, 1.5.3, and 1.5.4. Finally, the key findings of the literature review are summarized in Section 1.5.5.

1.5.1 Characterization of Latency Contributors

For the purposes of this dissertation, I will explicitly differentiate between a *space communication network* and *space communication system*. The former refers to the infrastructure put in place to communicate data to/from a remote spacecraft from/to the a user on Earth and is typically divided into space and ground segment [39], [40]. Alternatively, the latter includes both the space communication network as well as the spacecraft and end user. To exemplify the distinction, consider the system from Figure 1-5 once again. The *space communication network* is composed by the ground sites and processing center, while the *space communication system* includes these two elements plus the telescope and astronomer. As I will demonstrate throughout this thesis, including the end-users as part of the end-to-end system is necessary since they might the dominant source of latency.

To understand which functions induce latency in current space communication systems, I conducted a thorough literature review on current NASA networks, namely the DSN, the NEN, the SN and CSO (see Appendix A), as well as data processing systems for space-related data products such as LANCE. Their functional decomposition resulted in the identification of three core areas of functionality:

- **Service execution functions**, which enable the network to successfully implement the set of offered services to customer platforms (e.g. transmit data to and from

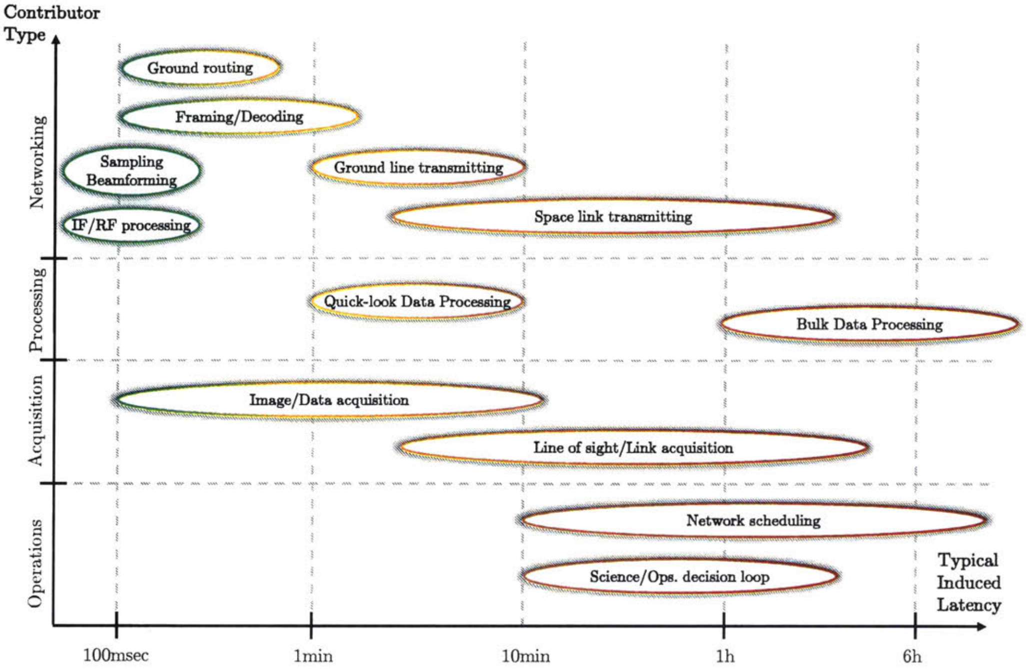


Figure 1-7: Latency Contributor in Space Exploration Applications

the MOC to the remote spacecraft, track it and provide time synchronization measurements) and end-users on Earth (e.g. ingestion of raw instrument measurements, processing of L1, L2 and L3 data products, etc.) [41]. In the case of the DSN, they also include scientific services at the measurement level (e.g. Very Long Baseline Interferometry, radar science) [42].

- **Network management functions**, which allow the network operator to successfully deploy, control, maintain and upgrade the different assets that compose the system. Inherent to these control functions is the level of redundancy provided by each network element in order to ensure that customer operations could be supported in case of catastrophic failure, and archived data records would not be lost. It also includes all functionality required in order to test different communication technologies before they become operational, as well the security elements deployed in order to prevent malicious users from interacting with the system and the system customers.

- **Service management functions**, which encompass all functionality that the network provides during the mission planning and operations phase in order to correctly negotiate and manage the communication and navigation services to be executed. It includes functions such as network scheduling, computing navigation solutions, interfacing with missions under development, accounting for provided services, scheduling science operations, as well as generating and validating commands to execute remote science activities.

Both service execution and service management functions induce latency in current space communication systems. On the service execution side, delays are typically related to how data is transported reliably from a remote location to the final user. They range from low level digital processing functionality, to transport functions such as delay tolerant networking, custodian mechanisms and file transport protocols for the space and ground environment. On the other hand, service management functions induce latency because they are responsible for appropriately configuring and scheduling the network and spacecraft assets to obtain data of high scientific value. Functions such as scheduling DSN antennas to support a mission given the network load and constraints, as well as generating new commands for the remote platform in response to previously delivered data are examples of service management functionality. Finally, I argue that network management functions can induce latency, but will intentionally be left outside the scope of this thesis. Indeed, they are mostly related to ensuring reliable operations in contingency situations and therefore will only be present in special, out of the ordinary occasions.

Table 1.1 lists the different latency contributors identified during the literature review as part of the service execution and service management functions. In turn, Figure 1-7 provides a visual representation of how much latency is introduced by each functionality in current space communication systems. Observe that, for service execution functions, quantifying the latency induced is not a simple task because all layers in the network architecture can potentially delay data. Yet, they can be considered homogeneous from the perspective of the latency contributor type. Indeed, they are all related to space communications and networking. On the other hand, factors that induce latency for service management functions are not necessarily attributable to the communication infrastructure but rather to operational constraints that are hard to define or model in advance. This realization leads to two primary conclusions: First, from an end-to-end system perspective, latency contributors are heterogeneous both in their nature and amount of delay introduced. Second, the problem of latency cannot be solely studied from a communications perspective, but rather necessitates interdisciplinary techniques that are characteristic from the domain of systems engineering.

Another interesting problem when considering latency contributors in the context of space communication systems is the fact that they are not always present, and different contributors can be the system bottleneck depending on the end-to-end architecture. Indeed, for a typical deep space mission latency is dominated by the propagation delay and possibly DSN schedule limitations [I1], [I2]. Yet, in the case of Mars rovers the primary latency contributor switches to the science decision cycle [I3]. This once again indicates that understanding latency in the context of space exploration is not only a communications problem (after all, the network that supports a mission at Mars or Saturn is exactly the same), but rather it includes operational considerations both in terms of how we command the missions and how we do science with the data we gather from them.

Table 1.1: Summary of Latency Contributor

Latency Contributor	Latency Induced	Driving Factors	Reference
LOS acquisition	Time between contact opportunities	Network topology	[43]
Image acquisition	Time to acquire entire image	Instrument design	[44], [45]
Data Transmitting	Transmission and propagation time	Link capacity, data volume	[I14], [I15]
RF/IF Functionality	Processing time	Electronics	[I11], [I12]
Sampling	Processing time	Electronics	[I11], [I12]
Beamforming	Processing time	Beamformer performance	[I11], [I12]
Receiving	Frame acquisition time	Space packet frame length, data rate	[I11], [I12]
Decoding	Decoder synchronization time, interleaver delay	FEC implementation, data rate	[I11], [I12]
Framing	Frame generation time	SLE frame length, data rate	[I11], [I12]
Transporting	Retransmissions	CFDP ARQ mechanism	[I13]
Store & Forwarding	Time stored in memory device[a]	Delay tolerant and custodian protocols	[46], [I13]
Ground delivering	FTP/TCP/IP transmission time	Service level agreement with ground provider	[47]
Data processing	Time to generate L1, L2, L3 data products	Processing capacity and algorithm complexity	[48], [49], [50]
Data distributing	FTP/TCP/IP transmission time	Service level agreement with ground provider	[51]
Network scheduling	Time between two allowed scheduled contacts	Mission priority	[I7]
Science planning	Time to generate tactical science plan[b]	Science decision loop	[I1], [I6]

[a] This could happen on the ground, in DSN site for instance, or on-board a spacecraft.

[b] Some missions differentiate between tactical and strategic science plan. The former defines day-to-day operations, while the latter specifies the long term goals that the mission should accomplish.

Modeling of Latency Contributors

Given the highly heterogeneous nature of latency contributors in space communication systems, the last part of this section's literature review was devoted to understanding how they can be modeled and quantified. In that sense, the discussion that follows is approximately organized based on Figure 1-7, from the top left corner to the bottom right one.

The first set of latency contributors considered in this thesis were specifically related to the underlying communication infrastructure. In that sense, three primary areas were considered: Ground routing and transmitting; low level analog and digital signal processing of data to and from remote spacecraft; and performance of transport protocols over space links. Focusing on the first area, end-to-end packet delay measurement in ground networks is a complex problem typically tackled from a dual perspective, analytic and experimental. The former analyzes multi-hop network properties from a purely analytic standpoint based on a limited set of assumptions (e.g. packet arrival processes, network topology, packet processing distribution). Probably the most well-known results in this domain come from the field of queuing theory (e.g. Kleinrock independence approximation [52]), and rely on simplification that make the problem computationally tractable in exchange for yielding results that are only useful for relative performance assessments and topology design [53]. On the other hand, a separate body of literature has studied the performance of ground networks from an experimental point of view. In that sense, probe packets are sent from a given origin to a given destination, and the end-to-end delay is measured as a random variable [54], [55]. While these measurements are only specific to the networks they characterize, they provide realistic values that can be used to assess the actual network performance and set service level agreements with the institutional clients they support.

Modeling of low level analog and digital signal processing in the context of space networks has already been considered in the literature. For instance, Reference [56] provides insight into how NASA network equipment processes space packets and the delay incurred as a function of the mission type. Similarly, [57] and [58] quantify the impact of delivering different types of data downlinked from satellites to network control centers as a function of the imposed latency requirements. Finally, performance of data transmission over error-prone space links has been largely studied in the context of the CFDP protocol. In that sense, latency in this part of the system has typically been related to the number of retransmissions required to deliver an entire file without errors, sometimes coupled with the specific characteristics of the underlying communication media that supports the system [59], [60], [61].

Outside the context of data transmission, acquisition and processing of data products has also been identified as a source of end-to-end latency. For the former, two main mechanisms are known to induce latency and must therefore be modeled: LOS acquisition, and full-image capturing. LOS acquisition is typically dictated, in the context of space, by orbit characteristics of the spacecraft that must be supported by the communication network. Consequently, quantification of latency can be typically performed through direct simulation (see, for instance, References [38] and [62]). Alternatively, and typically for particular orbits and constellations, figures of merit such as average or maximum revisit times can be approximated analytically [43]. On the other hand, relatively little work has been done to assess delays incurred due to image acquisition in the context of space-based instruments. In that sense, Reference [63] describes delays incurred in focal plane array readout correction and Fourier transform for a hyperspectral imager. Similarly, Reference [64] sets maximum latency requirements for the new generation of imagers on-board the GOES satellites as function of the number of pixels to acquire given a certain area of coverage and desired resolution. Finally, Reference [48] indicates the latency expected for processing the data products of the NPOESS program, albeit no indication of their drivers is included.

The last set of latency contributors identified in the literature review are related to operations, specifically network scheduling and science and operations decision loop. For the former, significant attention has been placed on the problem of network scheduling. For instance, Johnston [65], [66], [67] has tackled the problem of optimal scheduling in the context of the DSN. Similar analyses have also been provided by Cheung [68], including modeling of data volume, link characteristics and latency effects. On the other hand, the effect of latency on science operations has been reported both in the context of robotic and human operations. In either case, mission operations are essentially modified and optimized to fit the large amounts of data generated within the limited network resources available. This is exemplified, for instance, by the MMS mission, which utilizes fast and slow collection periods of data collection and has both an automated and manual data downlink prioritization scheme to ensure that resources are spent transmitting only the most relevant data [69]. Similar schemes are utilized to operate rovers at Mars while ensuring that scientists work in synchrony with Earth's daytime [70], [71].

1.5.2 Latency-constrained Space Exploration Applications

In Section 1.3.1, I proposed a categorization of space exploration applications with respect to their latency requirement. At the highest level, I suggested that missions can be divided into latency-constrained and latency-unconstrained groups. The fundamental characteristic of latency-constrained applications is that the value of data depends directly on the latency with which it is delivered to the final user. Latency-unconstrained applications exist in opposition to latency-constrained applications. They are characterized by gathering information that is fundamentally insensitive to latency from a value delivery standpoint.

A second level of categorization within the latency-constrained applications is also provided in Section 1.3.1, in which I differentiate between real-time, near real-time and latency-sensitive applications. In this section, I delve deeper into the differences between these three categories, not only from the point of view of space applications but also from the perspective of terrestrial services that might exhibit similar characteristics. To summarize my findings, I provide in Table 1.2 a data type categorization according to the type of space exploration application and latency requirement. It has been obtained through a thorough review of past, present and near future NASA missions spanning multiple domains: Earth observation proves, human spaceflight programs and deep space missions around the Sun and other planetary bodies. In that sense, the next sections provide concrete examples of missions within each space exploration application and how they are affected by latency (see Table 1.2). Finally, Section 1.5.2 demonstrates that some of the characteristics of latency-sensitive applications in the space context have a direct analogy in current terrestrial applications. This is important because it suggests that the methods and contributions from this thesis could be potentially applied outside the context of space exploration.

Real-Time Space Exploration Applications

The realm of real-time services is exclusively reserved to applications that require interactivity between two communicating users. In the context of space, there are currently only two types of services that require interactivity: Voice circuits that support dialog with ISS astronauts, and full-duplex video for teleconferencing [56]. Future missions might also require real-time services for support of telerobotic operations, most notably for repair and maintenance of geosynchronous satellites [72] and, in the distant future, telemedicine [73], [74]. Reference [75] details different latency contributors to be considered when architecting networks that deliver real-time applications. These include light time delay, ground network transport delay, voice encoding/decoding delay, packet size buffer delay, and boundary delay[2] among others. Additionally, jitter (variations in delay between packets) might also affect the service provided to real-time applications. On the other hand, Lester [14] provides a seminal work on defining latency requirements in the context of real-time applications. He notes that "the fundamental difference between telerobotics and astronauts on-site for space exploration is latency" [14]. Furthermore, he also compares latency requirements for real-time space applications against terrestrial gaming applications in order to emphasize the similarities. His findings are reproduced in Table 1.3.

Several remarks are important when considering the latency requirements provided by Lester. First, analogy with the online gaming experience suggests that a maximum round-trip latency requirement of 300-400 milliseconds should be enforced for real-time space exploration applications. In his discussion, this latency budget is mostly allocated to light-time delay. Other authors suggest, however, that this requirement should be imposed on the end-to-end latency that takes into account all communication and networking factors [75].

[2] Delay incurred at the transmitter and receiver as part of the transport protocol.

Table 1.2: Latency-constrained Applications

Space Exploration Application	Latency Categorization	Mission examples
Telepresence:		
Robotic exploration (Telepresence)	Real-time	-
Human exploration (Telemedicine)	Real-time	-
Voice:		
Human exploration (Dialog)	Real-time	ISS
Human exploration (Recorded message)	Near real-time	Apollo
Video:		
Human exploration (Full-duplex teleconferencing)	Real-time	ISS
Human and robotic exploration (HD buffered video)	Near real-time	ISS
Telemetry:		
Human and robotic exploration	Near real-time	TERRA, CAS
Commanding[a]:		
Human and robotic exploration	Near real-time	TERRA, CAS
Critical file transfer[b]:		
Human and robotic exploration	Near real-time	MER, MSL
Message alert:		
Solar weather	Near real-time	ACE, SOHO, STEREO
Space-based astronomy	Near real-time	NuSTAR,SWIFT,FERMI
Science data return:		
Space-based astronomy	Latency-sensitive	SWIFT
Solar weather	Latency sensitive	SOHO, STEREO
Weather data	Latency sensitive	EOS,GOES,JPSS,EOS
Science planning	Latency sensitive	MER, MMS, MSL
Humans at Mars	Latency sensitive	DRA5.0
Navigation:		
Doppler tracking	Near real-time	TERRA, AQUA
Az/El measurements	Near real-time	TERRA, AQUA
Delta-DOR	Latency sensitive	NPHC, CAS

[a] Includes uplink of navigation solutions if necessary for the mission

[b] Typically a software upload for the remote spacecraft or rover

Table 1.3: Real-time Applications (Adapted from Reference [14])

Application	Latency Requirement	Description
Human physiology		
	20-40 msec	Two-way neural signal transmission
	200 msec	Human eye-hand reaction time
	300-400 msec	Blink of human eye
Telephone service		
	260 msec	Recommended two-way maximum latency
Online gaming		
	60 msec	Limit of latency detection
	200 msec	Latency becomes noticeable
	500 msec	Game becomes unplayable
Space exploration		
	240 msec	Earth-to-GEO light time delay
	410 msec	Earth-Moon L1 or L2-lunar surface light time delay
	2600 msec	Earth-to-Moon light time delay

Second, real-time applications typically express latency requirements in the form of round-trip values since interactivity requires two-way communications [14]. Finally, the latency requirement for real-time applications is typically expressed as a maximum value, i.e. it imposes a hard requirement on the system that delivers data across the two end users [14].

Near Real-Time Space Exploration Applications

A significant portion of data returned through space networks from remote spacecraft falls under the category of near real-time data. Prominent examples are buffered video[3] (also referred as video streaming in the Internet literature [76]), telemetry streams, alert messages from space weather probes or alert messages from astronomical sources such as Gamma ray bursts (GRBs). Video streaming is currently a minor application in the space context, with only the ISS and certain launch vehicles returning near real-time video. Nevertheless, streaming video through terrestrial networks is an increasingly important topic as companies like Netflix or Youtube deliver their content through the Internet [77]. As Wu noted, "streaming video requires bounded end-to-end delay so that packets can arrive at the receiver in time to be decoded and displayed. If a video packet does not arrive in time, the playout process will pause, which is annoying to human eyes. A video packet that arrives beyond its delay bound (e.g. its playout time) is useless and can be regarded as lost" [76].

Near real-time applications in the context of space exploration are currently very present in the form of message alerts for space weather and astronomy purposes. A paradigmatic

[3] Situational awareness video in human or robotic exploration would fall within this category.

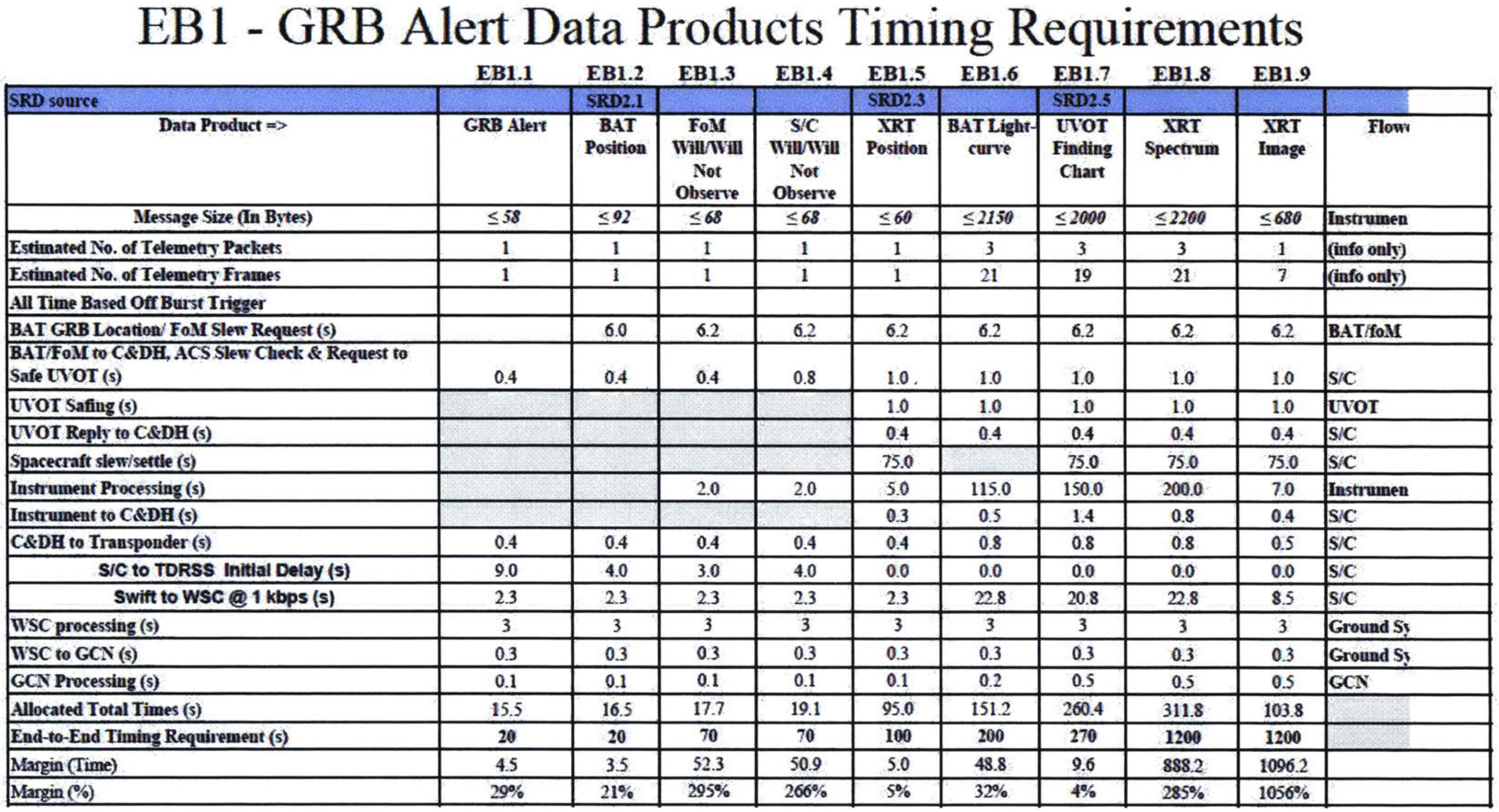
example is the SWIFT mission, which was originally launched in 2004 as an early warning system for astronomical GRB. The spacecraft is operated follows: The mission telescope continuously scans the sky searching for anomalous GRBs. Once one is detected, it quickly transmits the information to the ground so that more capable Earth-based telescopes can also point in the right direction and observe the phenomenon [3]. Figure 1-8 shows the end-to-end latency budget allocated for different near real-time data products of the SWIFT mission. It can be observed that the system is designed with a hard end-to-end latency requirement of 20 seconds with a 29% margin. A low resolution image of the GRB is also returned in near real-time with a total allocated latency of 1200 seconds. On the other hand, similar requirements are described in Reference [78] for the return of real-time space weather data from the SOHO spacecraft. It is noted that "real-time data [...], including normal scientific data, magnetogram data, and spacecraft housekeeping data, will be routed [...] with minimum processing delay following receipt at the ground station, and transferred to the Investigator workstations" [78]. Other science data (i.e. non near real-time data) will be played back "with transmission delays from DSN (approximately 3 hours) and processing delays to turn the data around (approximately 2 hours)" [78].

Latency Sensitive Space Exploration Applications

Latency-sensitive space exploration applications are primarily related to return of science data in the context of three categories: Solar and astronomy data[4], weather measurements and science planning data. Furthermore, distribution of DDOR data for deep space navigation purposes can be considered a special case of latency-sensitive data due to the large volume of data to be disseminated and the frequency with which a precise navigation solution is required (a typical DDOR pass requires each ground station to record up to 10GB of data [79]).

Solar weather and astronomy data can, in some cases, be considered latency-sensitive information. For instance, Table 1.4 summarizes the distribution of data products for the SWIFT mission according to their end-to-end latency requirement. Observe that near real-time services have latency requirements that match those provided in the latency budgets from Figure 1-8. In contrast, latency-sensitive data products are also produced in a timely manner to facilitate quick scientific engagement, yet they entail a soft latency requirement that can be expressed in the form of an hourly interval. Moreover, they are also not critical to the end-user. This allows the SWIFT technical specification handbook to safely state that "users should be aware that the quick-look data for a given observation may not be complete, especially the early dumps of an observation, and that the contents may change

[4] Note the difference explicit differentiation between *alert messages* and *data* for astronomy and solar weather probes. The former refers explicitly to a near-real time service, while the latter refers to data in the latency-sensitive domain.



EB1 - GRB Alert Data Products Timing Requirements

	EB1.1	EB1.2	EB1.3	EB1.4	EB1.5	EB1.6	EB1.7	EB1.8	EB1.9	
SRD source		SRD2.1			SRD2.3		SRD2.5			
Data Product =>	GRB Alert	BAT Position	FoM Will/Will Not Observe	S/C Will/Will Not Observe	XRT Position	BAT Light-curve	UVOT Finding Chart	XRT Spectrum	XRT Image	Flow
Message Size (In Bytes)	≤ 58	≤ 92	≤ 68	≤ 68	≤ 60	≤ 2150	≤ 2000	≤ 2200	≤ 680	Instrumen
Estimated No. of Telemetry Packets	1	1	1	1	1	3	3	3	1	(info only)
Estimated No. of Telemetry Frames	1	1	1	1	1	21	19	21	7	(info only)
All Time Based Off Burst Trigger										
BAT GRB Location/ FoM Slew Request (s)		6.0	6.2	6.2	6.2	6.2	6.2	6.2	6.2	BAT/foM
BAT/FoM to C&DH, ACS Slew Check & Request to Safe UVOT (s)	0.4	0.4	0.4	0.8	1.0	1.0	1.0	1.0	1.0	S/C
UVOT Safing (s)					1.0	1.0	1.0	1.0	1.0	UVOT
UVOT Reply to C&DH (s)					0.4	0.4	0.4	0.4	0.4	S/C
Spacecraft slew/settle (s)					75.0		75.0	75.0	75.0	S/C
Instrument Processing (s)			2.0	2.0	5.0	115.0	150.0	200.0	7.0	Instrumen
Instrument to C&DH (s)					0.3	0.5	1.4	0.8	0.4	S/C
C&DH to Transponder (s)	0.4	0.4	0.4	0.4	0.4	0.8	0.8	0.8	0.5	S/C
S/C to TDRSS Initial Delay (s)	9.0	4.0	3.0	4.0	0.0	0.0	0.0	0.0	0.0	S/C
Swift to WSC @ 1 kbps (s)	2.3	2.3	2.3	2.3	2.3	22.8	20.8	22.8	8.5	S/C
WSC processing (s)	3	3	3	3	3	3	3	3	3	Ground Sy
WSC to GCN (s)	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	Ground Sy
GCN Processing (s)	0.1	0.1	0.1	0.1	0.1	0.2	0.5	0.5	0.5	GCN
Allocated Total Times (s)	15.5	16.5	17.7	19.1	95.0	151.2	260.4	311.8	103.8	
End-to-End Timing Requirement (s)	20	20	70	70	100	200	270	1200	1200	
Margin (Time)	4.5	3.5	52.3	50.9	5.0	48.8	9.6	888.2	1096.2	
Margin (%)	29%	21%	295%	266%	5%	32%	4%	285%	1056%	

Figure 1-8: Latency Budget for SWIFT Alert Messages (Adapted from Reference [3])

Table 1.4: Latency of SWIFT Data (Adapted from Reference [15])

Data Product	Latency from Burst
Near real-time:	
Rapid BAT position and flux	20 s
XRT position	120 s
UVOT finding chart	300 s
X-ray spectrum	1212 s
Latency-sensitive:	
Quick-Look Products	2-3 hr
BAT Data	2-3 hr
XRT Data	2-3 hr
UVOT Data	2-3 hr
Refined XRT Position	2-3 hr
UVOT Position and Identification	2-3 hr
Latency-unconstrained:	
Final Telemetry Archive	1 week
BAT Products -	Daily
Afterglow Database	Daily
Calibration Database	Daily
Catalog of GRBs	Daily
GRB Redshift	Daily
Follow-Up Campaign Data	Daily
Hard X-Ray Catalog	Weekly
XRT Catalog	Weekly
UVOT Image Databases	Weekly

when and if subsequent telemetry dumps arrive" [3].

Latency in weather data is also a soft requirement. Indeed, latency does not prevent weather prediction centers from generating new forecasts. However, it does impact the amount of data they have available to feed their prediction models and ultimately reduces the quality of the forecast [80]. Current NASA missions that provide data to weather systems have already been upgraded to deliver relevant observations in under 3 hours through the LANCE system [20]. Similarly, complimentary systems such as NOAA's POES satellites and ESA's MetOp spacecraft deliver their meteorological data with latencies of over 1 hour [81]. These limitations will be overcome once the new Join Polar Satellite System (JPSS) between NASA and NOAA is deployed and a common ground infrastructure is put in place to deliver data with best-case latencies of 45 minutes approximately [82].

On the other hand, data used for spacecraft scientific and operational planning can also be categorized as latency-sensitive. Indeed, science data delivered today is used by mission planners to decide what the remote platform (e.g. spacecraft or rover) will do tomorrow. To exemplify the issue, consider the science operations planning from MSL presented in

Figure 1-9. The wide magenta arrows symbolize the flow of data between the Mars rover and the science operation team at JPL. At the start of each sol, the rover points its antenna towards Earth and receives commanding information from a DSN antenna. Throughout the martian day, the rover then conducts all science activities autonomously based on the set of received commands. Engineering and scientific data gathered during these activities is then returned to Earth through the MRN and used by mission planners at JPL to build the command sequence that will define rover operations for the next day. A similar timeline for MER is provided in Reference [71]. In both cases we can observe that latency is not a hard requirement but rather reduces the value of the operations for subsequent days by reducing the time that mission planners have to prepare the next set of observations.

A similar phenomena in the Earth context is exemplified by the MMS mission. MMS is a constellation of four small spacecraft studying the Earth's magnetosphere from a highly elliptical orbit [83]. Of particular interest for MMS are magnetic reconnection events, which are reported in a timely manner to mission scientists for further analysis on the ground (current requirements call for 1 hour latency after downlinking to a DSN site [69]). On top of that, MMS implements a dual science decision loop strategy. On the one hand, the mission has the ability to scan its stored data and select information particularly relevant for characterization of reconnection events and prioritize its return to Earth. On the other hand, once this data is analyzed by scientists on the ground, further information can be requested and specific measurements are scheduled using what is known as Science-in-the-loop decision process. This enhanced science-in-the-loop process is primarily constrained by latency induced through lack of capacity in the return link, lack of DSN availability and limited MOC/SOC personnel [I6].

Finally, a good example of navigation data in a latency limited environment is the proposed Europa-Clipper mission that will be launched around 2020 to study Jupiter's moon Europa [5]. Figure 1-10 depicts a typical science orbit for the touring spacecraft as it studies the Europa surface and composition. All science activities occur in the yellow and red part of the orbit, during which the spacecraft is pointed towards the planet and therefore has no support from the DSN. After science measurements are acquired, the spacecraft contacts its ground segment and starts downloading data. At the same time, tracking data for the spacecraft is collected and a navigation solution obtained, which is then utilized to uplink instructions for the subsequent trajectory correction maneuvers (TCMs). These TCMs are key to the success of the mission since they position the spacecraft in the right trajectory to ensure that the next Europa flyby targets areas of high scientific value. Therefore, latency of tracking data has to be monitored in order to ensure that a navigation solution is delivered with enough time to successfully build the command sequences for subsequent flybies.

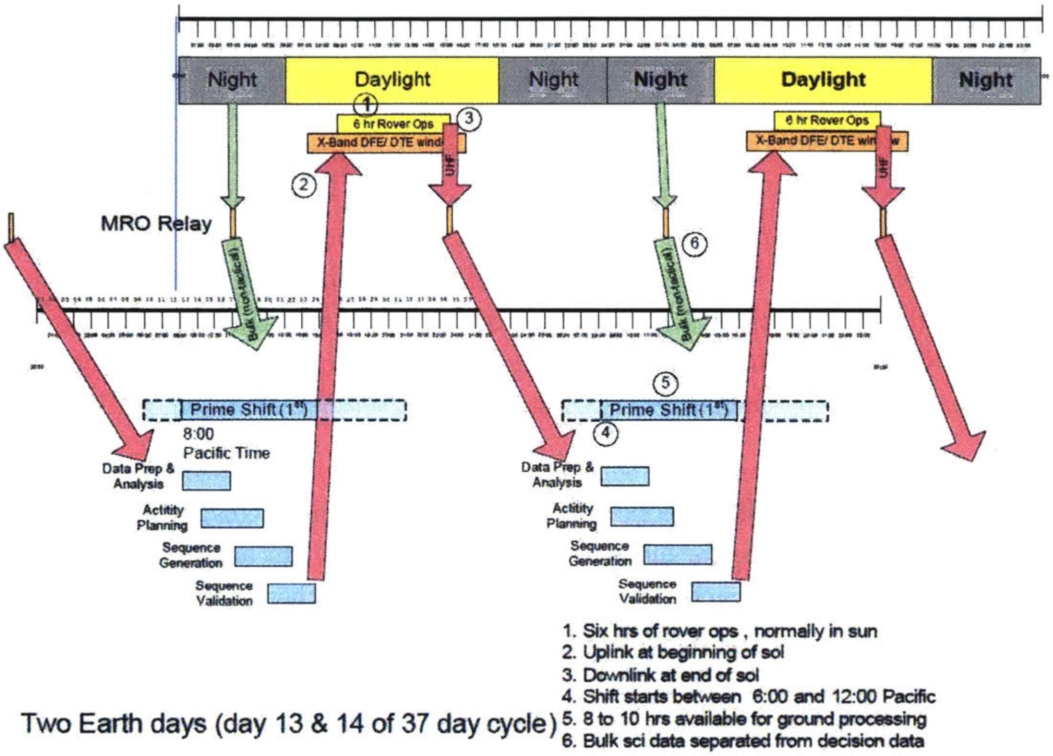

Night
Daylight
Night
Night
Daylight
Night
1
6 hr Rover Ops
X-Band DFE/ DTE window
3
2
UHF
MRO Relay
6
Bulk (non-tactical)
5
Prime Shift (1st)
4
8:00
Pacific Time
Data Prep & Analysis
Actitity Planning
Sequence Generation
Sequence Validation
1. Six hrs of rover ops , normally in sun
2. Uplink at beginning of sol
3. Downlink at end of sol
4. Shift starts between 6:00 and 12:00 Pacific
5. 8 to 10 hrs available for ground processing
6. Bulk sci data separated from decision data
Two Earth days (day 13 & 14 of 37 day cycle)

Figure 1-9: MSL Planning Cycle (Adapted from Reference [4])

Latency-Constrained Applications Outside the Space Context

Latency-constrained applications are not exclusive to the space context. As previously mentioned, real-time services in the terrestrial domain are required in applications such as online gaming, as well as voice over IP (VoIP), teleconferencing and remote desktop applications [84]. Section 1.5.2 also indicated that near real-time applications are increasingly important in terrestrial networks due to the popularity of video streaming. In fact, Cisco estimated in 20122 that by 2019 up to 80% of Internet traffic will be devoted to delivering video content over packet networks as opposed to the traditional web and email services [85].

Similarly, latency-sensitive applications are also not exclusive to the space context. For example, traditional web browsing can be considered an instance of a latency-sensitive application. Egger notes that "user quality perception in the context of interactive data services is determined by (...) waiting times to a large extent, a fact which has led to the catchy notion of WWW as *World Wide Wait*" [86]. In order to quantify this issue, he first defines a subjective quality of experience metric between 1 and 5 (5 being the highest score) and asks users to rank their network experience when performing three tasks (connect to a 3G network, download an image from the Internet and perform a Google search) under different initial delay scenarios. Then, he computes the mean opinion score (MOE) across all

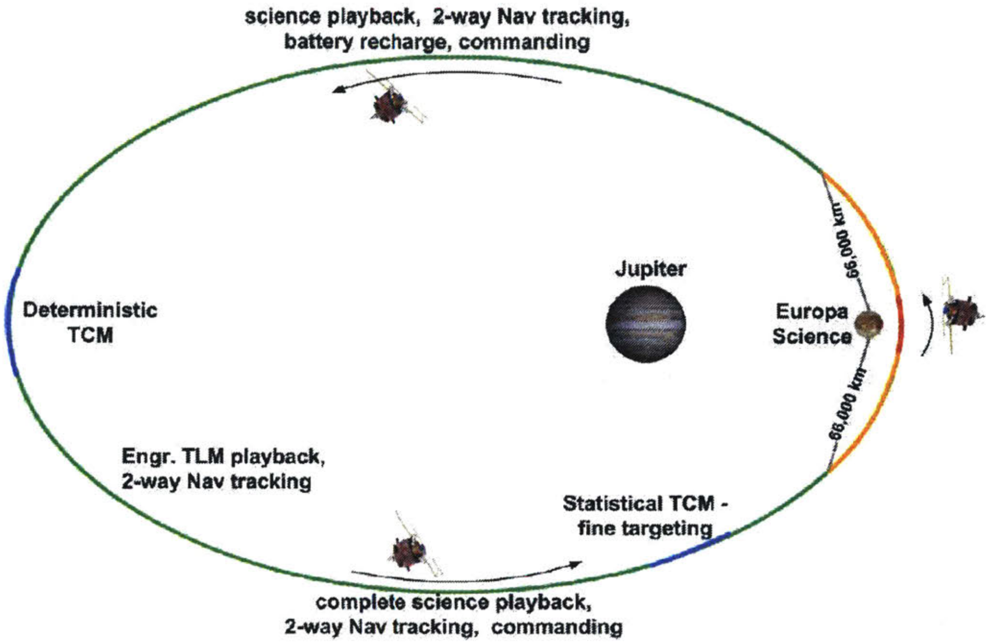


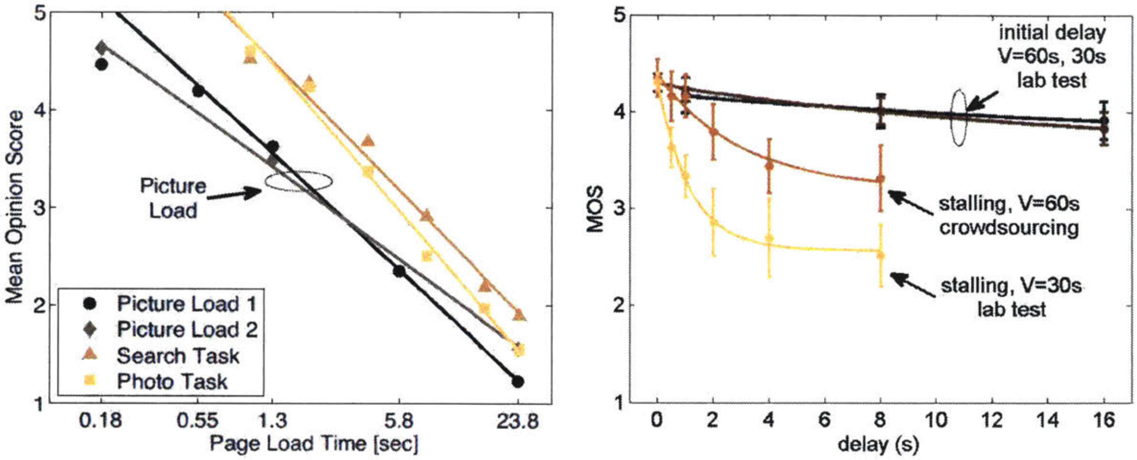
Figure 1-10: Europa-Clipper Science Orbit (Adapted from Reference [5])

respondents and builds a model that correlates quality of experience with application latency. Figure 1-11a plots the obtained results for the three aforementioned tasks. Note that the perceived quality of experience clearly decreases as the application latency increases[5].

Similar results for video streaming applications are reported in Reference [87]. In this case, the application under consideration is video streaming and latency can be experienced in two complimentary forms: initial buffering delay or interruption (stalling) during the video. Figure 1-11 presents the obtained results. Note that latency is especially deleterious if experienced as a stalling event. This result qualitatively confirms the proposed latency-based categorization of video streaming as a near real-time application during the visualization phase. It also suggests that the initial buffering time is probably better modeled as a latency-sensitive effect, thus indicating that the same application can belong to more than one category in specific circumstances.

While an in-depth analysis of how latency affects terrestrial applications is not required for this thesis, it is interesting to observe the similarity between space services and traditional Internet services. Indeed, note that in both cases the value of the data being transmitted through the network decreases as the latency introduced by the system increases. Therefore, although this dissertation will primarily focus on space-based applications, I argue that its findings could be potentially applicable to terrestrial applications.

[5] Latency is here equivalent to the time it takes for the network to execute the task.



(a) Image Download Quality of Experience [86]

(b) Youtube Video Quality of Experience [87]

Figure 1-11: Quality of Experience for Terrestrial Applications

Summary of Latency-sensitive Applications

Given the differences and similarities between space-based and terrestrial applications, it is worthwhile to briefly summarize the primary features that can be used to differentiate between real-time, near real-time or latency-sensitive applications. Table 1.5 states the main characteristics for each of them as a function of latency requirement and data stream. Observe that latency-sensitive applications are challenging due to the need to return high volumes of information in a timely manner. Similarly, they are also unique because they fundamentally impose a soft requirement on the end-to-end system, a fact that can be used by system architects to effectively trade performance vs. cost and other "ilities".

On the other hand, the provided categorization has been obtained through lessons learned and generalization from past, current and future missions. Therefore, it does not necessarily conform to the specific operational details of any given mission. For instance, I indicated in Section 1.5.2 that the quality of experience for video streaming has been found to be different if latency is applied during the buffering and viewing parts of the application. Similar problems arise in the space context. Latency sensitive navigation data during normal scientific orbits becomes near real-time data during critical events such as orbit insertion, early orbit operations and launch support.

Finally, latency specification as a requirement becomes specially important for real-time applications. Their undeniable importance is best exemplified in the history of NASA space communication networks: "During the aborted Apollo 13 lunar mission, voice contact was very important because you would have to wait sometimes 20 to 25 minutes between contacts. We had to fit as much communication as possible into that short span, whereas now, once

Table 1.5: Latency Requirement for each Application Type

Application	Req. Type	Req. Spec.	Round Trip	Typical Value	Data Stream Type	Req. Criticality
Real-time	Hard	Max.	Yes	300-400ms	Continuous	Critical
Near real-time	Hard	Max.	No	20-1200s	Periodic, low vol.	Critical
Latency sensitive	Soft	Average	No	10min - 6h	Periodic, high vol.	Desirable

the TDRSS system is fully deployed, we'll have absolute coverage for a Shuttle mission. You can call and talk just about any time you want to" [88]. In other words, levying a requirement to deliver real-time voice and video services for the Space Shuttle prompted the development of, at the time, an entirely new type of network, the SN. Such a radical solution would not have probably been needed, should the communication services for the Shuttle had been latency-sensitive instead of real-time.

1.5.3 Utility Theory in Engineering Systems

In Section 1.5.2 I argued that latency-constrained applications generate data that the end user prefers to have delivered as soon as possible. To operate with preferences, systems engineering, economics and other disciplines have extensively utilized ***utility theory*** [89] [90], [91]. At its core, utility theory defines a utility function that transforms objective metrics such as wealth or latency into a subjective scale that measures system stakeholder satisfaction. For instance, in economics it is common to use hyperbolic absolute risk aversion and constant relative risk aversion functions to specify the attitude of a person towards financial risk [91]. Their shape is specified by basic axioms of finance theory: Investors prefer more wealth than less wealth; investors are risk averse and, consequently, obtain diminishing utility from increasing levels of wealth [91].

Utility theory in the context of system architecting studies has been applied in a wide variety of fields. For instance, I utilized utility theory in the context of near Earth communication networks in order to assess the satisfaction of different types customers with respect to the total data volume returned, latency provided and imposed user burden [92]. In the space exploration domain, Golkar [93] utilized multi-attribute utility theory for the purposes of assessing the desirability of different Mars sampling campaigns as a function of the number and variety of samples collected. Similarly, Selva [28] utilized utility theory to architect Earth observation satellite systems.

Utility Theory under Uncertainty

The usefulness of utility theory in an uncertain world was originally studied in economics from the point of view of certainty equivalents and risk premia [89], [91]. The goal was to quantify how much an investor would be willing to pay in order to transform an uncertain level of utility into certain level of utility. On the other hand, utility theory under uncertainty has been recently used to obtain the value of contingent assets in incomplete markets [94]. This is of particular importance to this thesis because (1) engineering projects such as space networks are subject to uncertainties that cannot be hedged through financial replication [95], and yet (2) properly capturing the risk aversion when valuing an uncertain system can lead to contradictory conclusions when compared with the certainty case [96].

While a full discussion on the results of utility-based contingent asset pricing under uncertainty is not required for this thesis, it is worth summarizing its main results. First, the value of a system subject to uncertainty in the presence of soft requirements can be computed using expected values over risk-neutral probability measures [96]. Therefore, the value of a network that supports latency-sensitive applications such as the SN or DSN can be computed as

$$V = \mathbb{E}^{\mathbb{Q}}\left[U\left(\mathcal{L}\right)\right], \tag{1.1}$$

where $U(\cdot)$ is equal to the utility function for a given space exploration application, and $\mathcal{L}$ is a random variable that models the end-to-end latency. Importantly, observe that Equation 1.1 is only valid for applications with soft latency requirements. Therefore, it cannot be used to value systems that primarily provide service to real-time and near real-time applications.

Hugonnier et al. proved on Reference [97] that under certain conditions the change of measure that transform the real world probability space $\mathbb{P}$ into the risk-neutral probability space $\mathbb{Q}$ is unique and therefore leads to a unique system value. Additionally, Knight [96] indicated that the transformation between both probability measures is defined by

$$q\left(\theta\right) \propto p\left(\theta\right) U'\left(L\left(\theta\right)\right), \tag{1.2}$$

where $q\left(\theta\right)$ is the risk-neutral probability of being in state θ, $p\left(\theta\right)$ is the real world probability of being in the same state, and $U'\left(\cdot\right)$ is the marginal utility given the latency experienced in this state of the world. Note that Equation (1.2) allows computation of $\mathbb{Q}$ up to proportionality and therefore the values for $q\left(\theta\right)$ have to be re-scaled so that they add up to one. Note also that in the case of a linear utility function, there is in fact no difference between the real world and risk-neutral probability measure.

Finally, observe that $\mathbb{Q}$ is basically a re-scaled probability measure such that the states of the world are weighted according to the sensitivity of the stakeholder towards them. Indeed, if a given latency-constrained application is insensitive between receiving data with a latency L_1 or $L_1 \pm \delta$, then $U'\left(L_1\right) = 0$ and so does $q\left(\theta\right)$. Conversely, if $U'\left(L_1\right) \rightarrow \infty$ then $q\left(\theta\right)$ also tends to infinity. Therefore, the risk-neutral probability measure puts extra weight in states of the world where uncertainty in latency has a high impact on the value of the data delivered through the network.

1.5.4 Centrality Measures

Centrality measures are typically used to quantify the importance of a given node (or set of nodes) in a complex network [98]. In that sense, most researchers utilize them to generate rankings that identify which parts of the network are crucial to its performance given limited information on its structure and mechanisms to exchange data.

The concept of centrality has been extensively studied in the context of social networks and graph theory. Borgatti [99] provides a classification of centrality measures based on two primary dimensions, what they measure and the role of a node in the network flows. He distinguishes between length vs. volume measures in the first domain, and radial vs. medial measures in the second. Additionally, he proposes movement of information as a third criterion for categorization. In that case, centrality measures typically distinguish between networks where data moves using walks or following paths. Measures that fall under the same category based on these three criteria are known to have similar properties and typically exhibit high correlation when applied to real-life networks [99]. In some cases, however, some measures are defined for a specific purpose (e.g. Laplacian centrality measures centrality with respect to 2-step walks as opposed to walks of any length) and, consequently, their suitability for a given application or network should be carefully assessed [100].

Table 1.6 categorizes a large sample of centrality measures according to the three aforementioned criteria. Two thirds of the reviewed centrality measures measure volume as opposed to length. Volume centrality measures typically count the number of paths or walks that start, end or traverse a given node [99]. In contrast, length measures focus on the distance between a given node and the rest of the network. Differences between volume and radial measures is better understood in a simple example. Consider a node that is connected to all other nodes in the network, yet traversing these connections is highly expensive. Then this node will be central from a radial point of view, but negligible from a volume perspective. On the other hand, medial measures typically focus on assessing the centrality of a node with respect to paths or walks *traversing* the node [99]. They are complimentary to radial measures, which focus on assessing the importance of the node with respect to walks or paths that emanate or terminate in a node.

Applications of centrality measures are primarily prominent in the context of social systems. For instance, numerous authors have utilized them analyze terrorist networks [112], [113]. Similarly, social networks have also been studied from the perspective of centrality measures [114], [115]. On the other hand, in economic sciences centrality measures have been applied to understand international economic integration [116], as well as corporate interdependence and control [117].

Table 1.6: Categorization of Centrality Measures

Centrality Measure	Focus	Node Role	Data Mobility	Reference
Alpha centrality	Volume	Radial	Walk-based	[101]
Betweenness	Volume	Medial	Path-based	[102]
Closenness	Length	Radial	Path-based	[103]
Degree	Volume	Radial	Either	[103]
Eigenvector	Volume	Radial	Walk-based	[104]
Flow betweenness	Volume	Medial	Walk-based	[105]
Flow closenness	Length	Radial	Walk-based	[106]
Harmonic centrality	Length	Radial	Path-based	[107]
Information centrality	Length	Medial	Path-based	[108]
Katz status	Volume	Radial	Walk-based	[109]
k-path centrality	Volume	Medial	Path-based	[110]
Laplacian centrality	Volume	Radial	Walk-based	[111]

The use of centrality measures in the systems engineering literature is still in its initial stages. Bounova [98] used them to study the structure of US airline networks and Wikipedia, while References [118] and [119] applied them to electrical and software systems. Similarly, Sosa and Eppinger used centrality measures in the context component modularity for aircraft engines [120]. While their results cannot prove a direct relationship between system performance and component modularity, they demonstrate that centrality measures can indeed be applied in complex systems as a way to identify parts of the system that are particularly important. Note that, in that sense, all the surveyed references utilize centrality measures (or similar constructs such as Fan-In/Fan-Out visibility [121]) from the perspective of *system architecture analysis*. Indeed, given a system architecture typically described through a DSM, they identify which parts of the system are critical given their performance and the overall system structure.

Finally, observe that, in a sense, centrality measures can be considered complimentary to complexity metrics. Given a system' DSM, the goal of a complexity metric is to obtain a real-valued number that quantifies how complex the system is as a whole (see for instance Sinha [122]), i.e. it provides a quantitative measure for one of the system's overall emergent properties. In applied case studies, this has been useful because complexity tends to correlate with cost, and therefore one can be used as proxy for the other during early conceptual design[6] [18]. In contrast, centrality measures applied to an input DSM return a ranking. This ranking is characterized by a given score per system component, and intuitively assigns a numerical value to the component's importance in providing the overall system emergent property.

[6] Functions analogous to complexity metrics have been utilized to evaluate the positive synergies between scientific instrument on-board monolithic and distributed spacecraft [123]. This exemplifies how the same construct can be used to capture positive emergent properties in the system.

1.5.5 Key Findings

The following set of key findings summarize the lessons learned from the literature review:

1. The main latency contributors in space communication networks are due to both service execution and management functionality, and include space communication networks, spacecraft, mission operators and scientists. A large and highly heterogenous number of factors characterize these latency contributors. Therefore, tackling the problem of latency from the holistic, yet less accurate perspective of systems engineering (rather than traditional communications or networking engineering) is particularly interesting during early stages of the system architecture phase.

2. Proliferation of latency-constrained applications should be expected in the coming decades in the space context. Within this category, latency-sensitive applications are particularly interesting due to the need to return large volumes of data (as compared to real-time and near real-time applications) in a timely manner. They include space-based astronomy, solar weather monitoring, space-based meteorological data collection, observations for science planning (both robotic and human) and return of delta-DOR navigation data.

3. Latency-sensitive applications are unique within the realm of latency-constrained applications because they impose soft requirements on the system. Consequently, the value of the data they generate can be quantified using utility functions that express preferences in data delivery with different levels of timeliness. Furthermore, performance (or, in this case, data timeliness) can be traded against infrastructure cost and risk so that the overall end-to-end data production system is optimized.

4. Centrality measures are typically characterized by what they measure (volume vs. length), the role of a node (medial vs. radial), and the flows of information through the system (path vs. walk-based). In systems engineering, they have been used for *systems architecture analysis*, to understand which elements in a system are critical for obtaining a given emergent property. Yet, they have not been applied to *system architecture synthesis* process.

1.6 Thesis Statement

Given the primary findings from the literature review, the central hypothesis that this dissertation explores can be formulated as follows:

Significant infrastructure savings can be obtained if systems that support current and future latency-sensitive space exploration applications are architected taking into consideration and quantitatively managing end-to-end latency requirements. This can be effectively performed using system architecting tools based on the concept of network centrality.

In order to address this research question, I now present this dissertation's problem statement:

To investigate and quantify the trades between infrastructure cost and service provision for latency-sensitive applications **by**:

1. Identifying and characterizing the primary sources of latency in current space communication networks,
2. Identifying, characterizing and classifying latency-sensitive space exploration applications, both robotic and astronaut-related,
3. Developing an efficient mechanism to quantify the relative importance of all latency contributors present in the system and ultimately ranking them,
4. Proposing a latency-centric approach to architecting space communication networks to effectively manage end-to-end latency requirements,

using centrality measures.

1.7 Thesis Structure

The remainder of this thesis is structured as follows: Chapter 2 provides a systems-centric approach for studying latency in communication networks. It first derives a centrality measure suitable for latency-sensitive applications and then describes the different steps that should be followed when utilizing it as part of a system architecting exercise. Chapter 3 demonstrates the validity of the proposed approach through a simulation-based case study where I study the optimization of a terrestrial WAN that delivers latency-sensitive packets across the US. Numerous stress experiments are conducted in order to test the limits of the centrality measure and reinforce the value of proper calibration. Then, Chapter 4 applies the presented framework to the problem of communication networks for weather satellite systems. In particular, the case study focuses on the JPSS and its CGI, with specific emphasis on the trade between space and ground-based networks for returning timely space-based weather observations. A third case study related to human exploration activities at Mars is provided in Chapter 5. In this case, I analyze the trade-offs inherent to a network that provides service to astronauts at the surface of Mars and helps them conduct their scientific activities. Finally, Chapter 6 summarizes the work conducted during this dissertation, identifies its main contributions and delineates ares of future works.

THIS PAGE INTENTIONALLY LEFT BLANK

2 A Latency-centric Approach to Architecting Space Communication Networks

Given the diverse nature of latency contributors elicited in Section 1.5.1, in this chapter I present a systematic approach for architecting space communication networks that provide services to latency-sensitive applications. At the heart of this latency-centric approach is the definition of a new centrality measure that can be used to identify which parts of the network and which latency contributors are responsible for inducing more latency. In short, a system architect with limited resources and time, should focus on parts of the system that largely delay information and disregard those that would neither introduce latency, nor would they reduce latency per unit of capital expenditure. In that sense, the goal of Sections 2.1, 2.2 and 2.3 is to progressively build a centrality measure that, given a system architecture (typically captured by its DSM), can be used systematically pinpoint which elements are of primary interest. Then, Section 2.4 prescribes the set of steps that should be implemented when applying the derived centrality measure to a specific system architecting problem. As I will demonstrate, this centrality measure acts basically as a heuristic function that guides the system architecting process towards areas of high return of investment[1].

2.1 Utility Loss in a Communication Path

In Section 1.5.2 I argued that latency-sensitive applications are unique because the value of the data they generate is a function of the timeliness with which it is delivered to the final user. Consequently, I start the formulation of the centrality measure by assuming that there exists a function $U\left(L\right)$ that quantifies the level of satisfaction experienced by the stakeholder that consumes data from the space communication system, as a function of the end-to-end induced latency L. Without loss of generality, I also assume that this utility has been normalized in the $[0, 1]$ range, 0 indicating no utility and 1 indicating full satisfaction.

To guide the centrality measure definition, I first consider a single communication path between an arbitrary source and destination (see figure 2-1). Data is delivered after node N with a total delay of $L_p = \sum_{n=1}^{N} L_n$ and a corresponding utility $U\left(L_p\right)$. I define the utility loss $\bar{U}\left(L_p\right) = 1 - U\left(L_p\right)$ as the amount of utility that the data has lost due to the total delay introduced by the system. Then, the first goal is to define a metric that quantifies

[1] Return of investment is understood here from a systems engineering perspective. As a result, it should consider not only performance and cost considerations but also any other *ilities* that are relevant when assessing the value of the system.

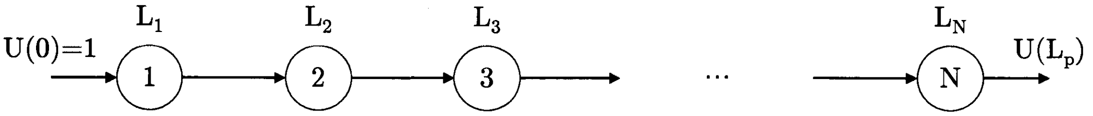


Figure 2-1: Typical Path Followed by Data Through the System

how much utility loss can be "blamed" to any given node within this path. Next, I define the set of axioms that should be satisfied by the aforementioned metric:

- **Axiom 1**: The metric must work for all utility functions as long as $U'(L) < 0$, i.e. data delivered later is less useful.
- **Axiom 2**: The utility loss attributed to a node must be ***directly correlated***[2] with the latency it introduces.
- **Axiom 3**: The metric value must allow direct comparison of all paths that traverse node n.
- **Axiom 4**: A node that introduces a certain latency L_n within a communication path must be equally blamed regardless of its position within the path.

These four axioms are used to compare five alternative formulations for the desired metric:

$$\bar{U}[n] = \frac{L_n}{L_p} \tag{2.1}$$

$$\bar{U}[n] = \frac{1 - U(L_n)}{1 - U(L_p)} \tag{2.2}$$

$$\bar{U}[n] = U(L_p - L_n) - U(L_p) \tag{2.3}$$

$$\bar{U}[n] = U\left(\sum_{i=1}^{n-1} L_n\right) - U\left(\sum_{i=1}^{n} L_n\right) \tag{2.4}$$

$$\bar{U}[n] = [1 - U(L_p)]\frac{L_n}{L_p} \tag{2.5}$$

Table 2.1 provides summary of how each of the previously defined alternatives compares against the three initial axioms. Observe that only alternative (2.5) satisfies all axioms and therefore will be used for the rest of the thesis. Nevertheless, before proceeding to its applications, I now summarize how each of the other alternatives violate the indicated axioms.

Alternative (2.1) and (2.2) are the simples conceivable. Essentially, they capture the idea that a node should be penalized proportionally to the latency they introduce (alternative (2.1)) or the utility lost (alternative (2.2)). Assuming that $L_p = \sum_{\forall i} L_i$ and that $U'(L) <$

[2] In opposition to inversely correlated.

Table 2.1: Comparison of Centrality Measure Alternatives

	Axiom	Alternative				
		(2.1)	(2.2)	(2.3)	(2.4)	(2.5)
1	Metric must work for all $U(L)$	✓	✓	✓	✓	✓
2	Metric should be directly correlated to latency	✓	✗	✗	✗	✓
3	Metric must allow path comparison	✗	✗	✗	✗	✓
4	Metric should not depend on node's position	✓	✓	✓	✗	✓

$0'$), it is immediate to see that their definition as a ration results in $\bar{U}[n] \leq 1$ for all nodes in the path. Nevertheless, it is also immediate to see that by dividing two quantities, the notion of absolute utility lost within a path is lost. Indeed, consider a node that introduces 10 seconds of latency in two paths through the system, one with 1 minute of end-to-end latency and another one with 20 seconds of end-to-end latency. In that case, both alternative (2.1) and (2.2) will flag node n as important because of the amount of delay induced in the 20 second path, without realizing that in reality you should probably focus first on the path that has a 1 minute end-to-end delay.

Alternative (2.3) is derived from the concept of synergies from Reference [28]. Indeed, the problem of latency in communication network is quite similar to that of synergies albeit with the opposite outcome: While the former reduces system value, the latter increases it by allowing scientific measurements to be synergistic with one another and therefore provide better science than if their measurements were taken independently. The key idea in alternative (2.3) is to compute the system value with and without the latency introduced by node n and then assess its difference. Note that, since I imposed a decreasing constraint for the utility function, it is immediate to see that $\bar{U}[n]$ will be strictly positive for all nodes assuming that $L_n > 0\,\forall n$.

Unfortunately, it is not difficult to find a situation in which alternative (2.3) provides a controversial attribution of utility loss. In particular, consider a hypothetic network with $N \to \infty$ and $L_n = \delta$, $\forall n$, where δ denotes an infinitesimal value. Then,

$$\bar{U}(L_n) = U(L_p - L_n) - U(L_p) = U(L_p - \delta) - U(L_p) \approx 0 \tag{2.6}$$

and consequently no node in the path is "blamed" for the utility loss experienced by the data. While this situation is unlikely, a similar problem arises for concave utility functions, specially in networks that combine long and short paths. In that case, the nodes in the long path are all penalized with a score of $\frac{1}{N}$, a significantly small number in comparison to the scores for nodes in the short path. In other words, nodes in the short path are artificially penalized because they have to share the same utility loss across less nodes than the long path.

On the other hand, alternative (2.4) was originally conceived by hypothesizing that the utility loss attributable to a node is equal to the delta in utility measured before and after data has traversed it. Even though this approach seems sensible at first, it is immediate to see that it will yield unfair results under highly non-linear utility functions. For instances, nodes at the beginning the path will be severely penalized with convex function since most utility loss occurs as the first delays are introduced. Similarly, if the utility function is concave then nodes at the end of the communication path will be unfairly penalized.

Given these limitations, I select alternative (2.5) as the baseline metric to define the utility loss attributable to node n in a given path p. Once again, its interpretation is straightforward: A node will be blamed proportionally to the total utility loss in the path and the fraction of latency it has contributed. Note that this formulation overcomes the limitations previously identified: For a path with $N \to \infty$ and $L_n = \delta$ nodes, it will blame all nodes equally with a value of $\frac{\delta}{L_p}$. In other words, it is as if all nodes in the communication path are responsible for the entire utility loss. Similarly, since the attribution of utility loss is performed proportionally to the latency introduced over the entire path, artificial asymmetric penalizations due to non-linear utility functions at the first or last nodes are also avoided.

2.2 Non-deterministic Utility Loss in a Communication Path

So far I have assumed that $\bar{U}\left(L_n\right)$ is computed using well-known latency values for all nodes within the path. In this section, I succinctly describe how the provided metric should be adapted in the general case where the latency is a parameter subject to uncertainty. To that end, let $\mathcal{L}_n$ denote the random variable that models the latency introduced by node n. Assuming that the network under consideration provides services to latency-sensitive applications, I argued in Section 1.5.2 that architecting under expected values is a valid approach due to the soft nature of the latency requirement. Furthermore, I also indicated in Section 1.5.3 that risk-neutral pricing can be used to properly quantify the total system value. With these two building blocks, I redefine alternative (2.5) as

$$\bar{U}\left[n\right] = \mathbb{E}^{\mathbb{Q}}\left[\left[1 - U\left(\mathcal{L}_p\right)\right]\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] = \left[1 - \mathbb{E}^{\mathbb{Q}}\left[U\left(\mathcal{L}_p\right)\right]\right]\mathbb{E}^{\mathbb{Q}}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] + \mathrm{Cov}^{\mathbb{Q}}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}, U\left(\mathcal{L}_p\right)\right] \qquad (2.7)$$

where the $\mathbb{Q}$ probability measure is just the real-world probability measure re-scaled such that states of high latency or high uncertainty are heavily weighted (see Equation (1.2) for the exact change of measure). Note that, unfortunately, Equation (2.7) is not easy to compute analytically even for simple networks as it requires knowing the individual probability distributions for all nodes in order to obtain the expectations and covariances.

Simplifications (or approximations) can be obtained under certain conditions such as linear (or quasi-linear) utility functions[3]. In that case, I first note that the risk-neutral $\mathbb{Q}$ and real world $\mathbb{P}$ probability measures are equivalent, and consequently there is not need to implement the change of measure. Secondly, I also observe that

$$\begin{aligned}\mathrm{Cov}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}, U\left(\mathcal{L}_p\right)\right] =& \mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\left[a\mathcal{L}_p + b\right]\right] - \mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right]\mathrm{E}\left[a\mathcal{L}_p + b\right] = \\ =& a\mathrm{E}\left[\mathcal{L}_n\right] + b\mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] - a\mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right]\mathrm{E}\left[\mathcal{L}_p\right] - b\mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] = \\ =& a\left[\mathrm{E}\left[\mathcal{L}_n\right] - \mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right]\mathrm{E}\left[\mathcal{L}_p\right]\right]\end{aligned} \tag{2.8}$$

where $a < 0$ and b are two arbitrary real valued coefficients that characterize the utility function. Therefore, the primary concern is to estimate $\mathrm{E}\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right]$ in a simple manner. Fortunately, this term is just the expected value of the division of two random variables (with no undefined values since the denominator cannot be zero by construction) and, consequently, we can use Taylor expansions of first and second order to approximate its value [124]:

- First order approximation: $\mathrm{E}_1\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] \approx \frac{\mathrm{E}\left[\mathcal{L}_n\right]}{\mathrm{E}\left[\mathcal{L}_p\right]}$
- Second order approximation: $\mathrm{E}_2\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] \approx \frac{\mathrm{E}\left[\mathcal{L}_n\right]}{\mathrm{E}\left[\mathcal{L}_p\right]} - \frac{\mathrm{Cov}\left[\mathcal{L}_n, \mathcal{L}_p\right]}{\mathrm{E}^2\left[\mathcal{L}_p\right]} + \frac{\mathrm{Var}\left[\mathcal{L}_p\right]\mathrm{E}\left[\mathcal{L}_n\right]}{\mathrm{E}^3\left[\mathcal{L}_p\right]}$

Observe that the strength of the second order correction is basically related to the variance of $\mathcal{L}_n$ and $\mathcal{L}_p$. If they are relatively small in comparison to the expected values of latency, then we can safely utilize the first order approximation, which results in Equation (2.2) being simply equal to zero. This result further simplifies Equation (2.7), which now only depends on the expected latency that each node introduces:

$$\bar{U}\left[n\right] \approx \left[1 - U\left(\mathbb{E}^{\mathbb{P}}\left[\mathcal{L}_p\right]\right)\right]\frac{\mathbb{E}^{\mathbb{P}}\left[\mathcal{L}_n\right]}{\mathbb{E}^{\mathbb{P}}\left[\mathcal{L}_p\right]} \tag{2.9}$$

To assess the magnitude of the newly introduced term in Equation 2.7, I study four scenarios from an analytic perspective (see Table 2.2). In the simplest of cases, all nodes in the communication path are equal and their random latency contributors can be modeled through 1 parameter distributions (e.g. exponential delays). On the other hand, a more realistic scenario would consider that all latency contributors in the communication path are similar except for one that dominates, and their respective random variables are characterized through at least two parameters, mean and variance.

[3] As I will demonstrate in Chapters 4 and 5, this assumption is valid for most latency-sensitive applications.

Table 2.2: Analytic Scenarios

	$\mathcal{L}_i \sim f(\boldsymbol{\theta})\ \forall i$	$\mathcal{L}_i \sim f(\boldsymbol{\theta})\ \forall i \neq n,\ \mathcal{L}_n \sim f(\boldsymbol{\theta}_n)$
$\boldsymbol{\theta} = (\mu, \mu^2)$	Scenario 1	Scenario 3
$\boldsymbol{\theta} = (\mu, \sigma^2)$	Scenario 2	Scenario 4

Given these assumptions, I assess the importance of the error committed in Equation 2.9 as

$$\epsilon = \left| \mathrm{E}_1\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] - \mathrm{E}_2\left[\frac{\mathcal{L}_n}{\mathcal{L}_p}\right] \right| = \left| \frac{\mathrm{Var}[\mathcal{L}_p]\,\mathrm{E}[\mathcal{L}_n]}{\mathrm{E}^3[\mathcal{L}_p]} - \frac{\mathrm{Cov}[\mathcal{L}_n, \mathcal{L}_p]}{\mathrm{E}^2[\mathcal{L}_p]} \right|, \tag{2.10}$$

where $\mathrm{E}_1[\cdot]$ and $\mathrm{E}_2[\cdot]$ are the first and second order Taylor expansions for the ratio between two random variables. To further simplify the problem, I assume that $\mathcal{L}_i$ are independent for all i, and I assume that the number of nodes within a path N is sufficiently large so that the Lyapunov Central Limit Theorem applies. If that is the case, the following statements are satisfied:

- The end-to-end latency $\mathcal{L}_p \sim \mathcal{N}\left(\sum_{i=1}^{N} \mu_i, \sum_{i=1}^{N} \sigma^2\right)$, with $\sigma = \mu$ in the case of one parameter distributions.
- $\mathrm{Cov}[\mathcal{L}_n, \mathcal{L}_p] = \mathrm{Var}[\mathcal{L}_n] = \sigma^2$.

Consequently, I derive the analytic expression for the error term in Scenario 4[4] as follows:

$$\epsilon = \left| \frac{\left(\sigma_n^2 + (N-1)\sigma^2\right)\mu_n}{\left(\mu_n + (N-1)\mu\right)^3} - \frac{\sigma_n^2}{\left(\mu_n + (N-1)\mu\right)^2} \right|. \tag{2.11}$$

Finally, I consider under which conditions $\epsilon \to 0$:

- Scenario 1: If $\mu_n = \mu$, $\sigma_n = \sigma$ and $\sigma = \mu$, then $\epsilon = 0$ without any further conditions.
- Scenario 2: If $\mu_n = \mu$, $\sigma_n = \sigma$, then $\epsilon = 0$ without any further conditions.
- Scenario 3: If $\mu = \sigma$ and $\mu_n = \sigma_n$, then $\epsilon \to 0$ if $(N-1)\frac{\mu}{\mu_n} \ll 1$.
- Scenario 4: $\epsilon \to 0$ if $(N-1)\frac{\mu}{\mu_n} \ll 1$ and $(N-1)\frac{\sigma}{\sigma_n} \ll 1$.

The main conclusion of this analysis can be summarized as follows: In the presence of non-deterministic latency contributors, Equation 2.9 can replace Equation 2.5 as the main building block for the centrality measure provided that (1) the system is decomposed in such a way that latency contributors can be considered independent from each other, (2) the application under consideration has linear or quasi-linear utility functions, and (3) there is one dominant latency contributor in each communication path, both in terms of expectation and variance. Importantly, note that these conditions greatly simplify the application of the

[4] Evidently all other scenarios are a simplified version of Scenario 4.

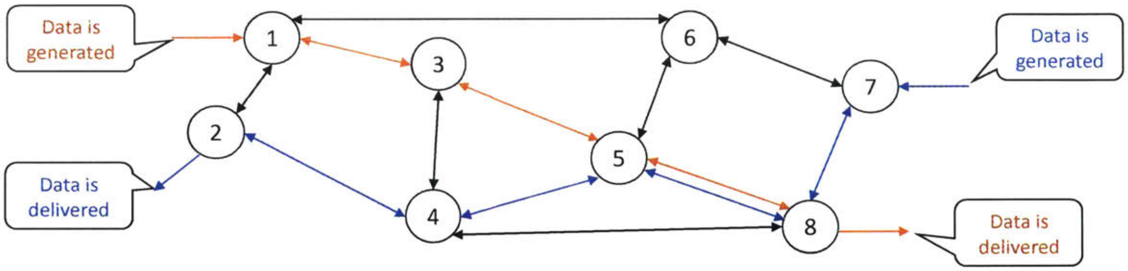


Figure 2-2: Simplified Network Example

centrality measure as it can be specified using only expected latency values. Indeed, this fact reduces the amount of information that needs to be collected from the system, since only the first moment for each latency contributor must be studied and quantified.

2.3 Utility Loss in a Communication Network

Using $\bar{U}[n]$ as the main building block, I now proceed to formulate the centrality measure that will be used as a guiding heuristic for the system architecting process. To facilitate the discussion, consider the network from Figure 2-2 with eight nodes and two independent data streams. Node 5 is unique in the system because it transmits data from both data streams. Therefore, if a large delay is induced by this node, both data flows will be significantly affected. To account for this fact, I define the system architecting centrality measure as:

$$\mathcal{H}[n] \propto \sum_{\forall p \in P_n} w_p \bar{U}_p[n] \tag{2.12}$$

where P_n defines the set of paths between any two given nodes that goes through node n, and w_p is a weighting factor that quantifies the relative importance of flow p with respect to all other flows in the system. For instance, assuming that the blue data flow in Figure 2-2 is twice as important as the orange data flow, the centrality measure would be computed as

$$\mathcal{H}[n=5] = 2\bar{U}_{blue}[n=5] + 1\bar{U}_{orange}[n=5] \tag{2.13}$$

with $\bar{U}_{blue}[n=5]$ and $\bar{U}_{orange}[n=5]$ computed using Equation (2.9) over paths $\{7,8,5,4,2\}$ $\{1,3,5,8\}$ respectively.

Several remarks about Equation (2.12) should be clarified. First, it is clearly derived from the notion of betweenness centrality (see Section 1.5.4). As such, it blames a node n for utility loss by adding up all paths that go through this node from any origin to any destination. Since this summation is, in fact, unbounded, betweenness centrality is typically normalized by $(N-1)(N-2)$ and $\frac{1}{2}(N-1)(N-2)$ for directed and undirected graphs respectively, with N equal to the total number of nodes. This normalization counts the maximum number of geodesic paths that go through a given node and therefore limits the measure domain to the $[0, 1]$ interval.

On the other hand, betweenness centrality for random walks has also been studied in the literature. It is useful when movement of elements across the system is performed using a random strategy rather than geodesic paths. Therefore, in my definition of Equation (2.12) I simply state that each node should aggregate all paths that go through the it, without prescribing how they will be computed. Similarly, I only specify the metric up to proportionally without prescribing a given normalization factor. All of them should be chosen on a case by case basis given the system architect preferences and the system under consideration. For instance, in space networks, it is often the case that geodesic paths are indeed the best way to model how data moves through the system. However, in more heterogeneous systems like the Internet, it is not uncommon for packets to not follow the geodesic path, specially if they are routed across multiple independent Autonomous Systems.

2.4 A Latency-centric Approach to Architecting Space Communication Networks

In this section, I synthesize the main steps that are necessary to apply the proposed centrality measure to a system architecting problem. When appropriate, I also provide a succinct example to clarify how it should be applied. In that sense, Figure 2-3 provides a pictorial representation of the different steps required by the latency-centric approach to architecting space communication networks, as well as the specific subsections within this document where their description can be found. Observe that, as previously hinted, the centrality measure is utilized as a heuristic function that guides the system architecture process towards areas of high performance loss.

Step 2-1. Motivation and Domain Specific Expertise

The first step of the latency-centric approach to architecting space communication networks is essentially a literature review that should answer two main questions:

- Why is latency an issue for this particular space exploration application?
- Does this application fit within the category of latency-sensitive applications?

The main purpose of the first question is to assess whether latency is, in fact, the critical problem around which the system should be architected. Indeed, communication networks face a myriad of challenges that are not necessarily related to latency. For example, return of scientific data from the New Horizons spacecraft that visited Pluto requires the DSN to

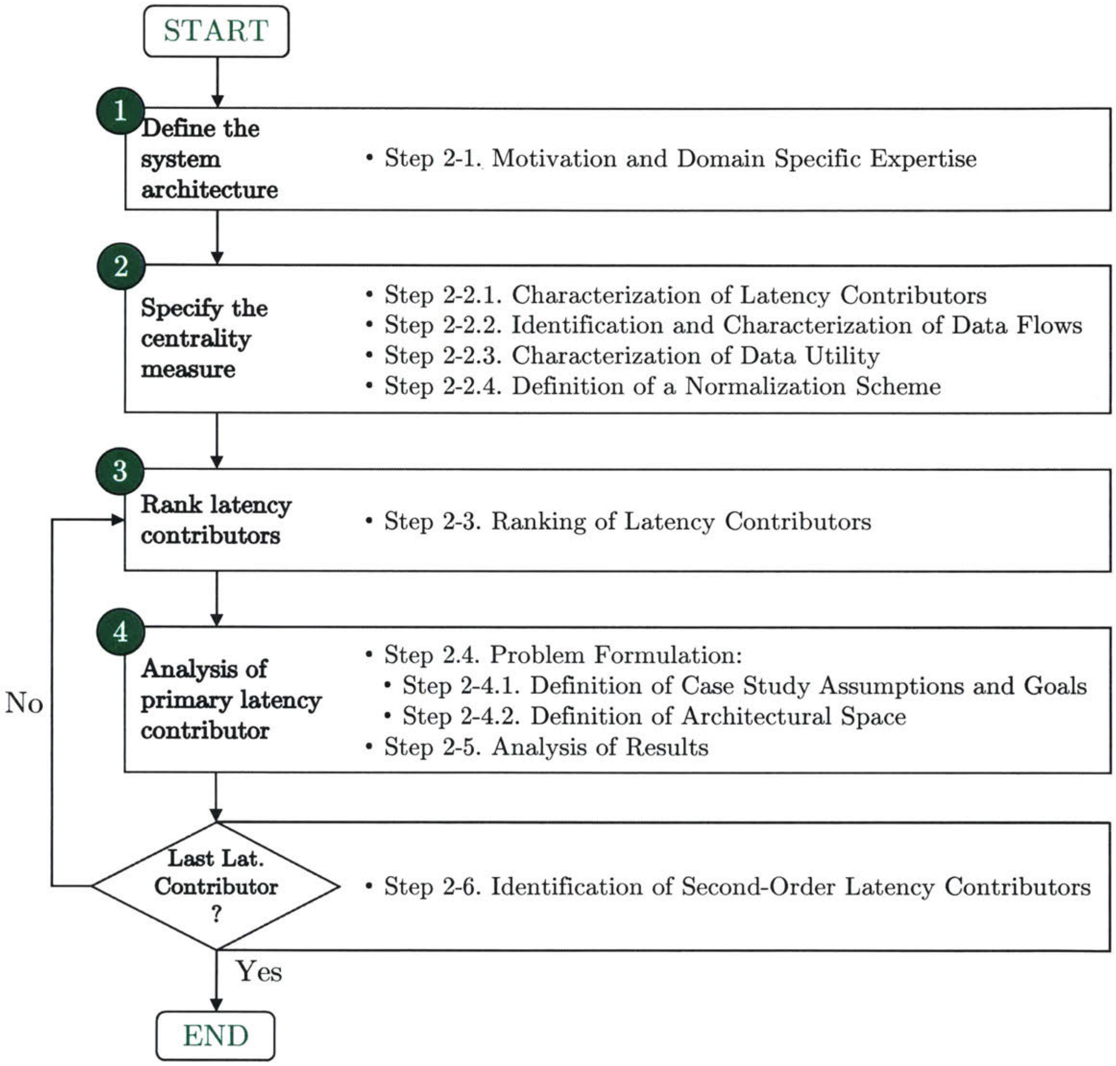

START
1
Define the system architecture
• Step 2-1. Motivation and Domain Specific Expertise
2
Specify the centrality measure
• Step 2-2.1. Characterization of Latency Contributors
• Step 2-2.2. Identification and Characterization of Data Flows
• Step 2-2.3. Characterization of Data Utility
• Step 2-2.4. Definition of a Normalization Scheme
3
Rank latency contributors
• Step 2-3. Ranking of Latency Contributors
4
Analysis of primary latency contributor
• Step 2.4. Problem Formulation:
• Step 2-4.1. Definition of Case Study Assumptions and Goals
• Step 2-4.2. Definition of Architectural Space
• Step 2-5. Analysis of Results
No
Last Lat. Contributor ?
• Step 2-6. Identification of Second-Order Latency Contributors
Yes
END

Figure 2-3: Latency-centric Approach to Architecting Space Communication Networks

push its capability limits due to large distance between the mission and the Earth. Yet, latency has never been an issue and, in fact, data from the planet fly-by will be returned to Earth over the course of an entire year.

The second question is also of utmost importance. The proposed $\mathcal{H}[n]$ hinges on the defining characteristics of latency-sensitive applications: Latency is a soft requirement. Therefore, utilizing it for studying systems that primarily serve other types of applications can easily result in misleading recommendations. To better understand this fact, consider the analogy between a space communication and a space logistic network. The latter is characterized by a set of rockets and space vehicles that deliver consumables (oxygen, food, etc.) to astronauts at Mars [125]. Assume also that this logistic network is sized based on expected values. Then, the resulting system will ensure that astronauts are alive and save on expectation. It is immediate to see that this argument is fallacious: Humans do not live in expectation, supplying them with their necessary consumables has to be guaranteed at all times and under all conditions. The same argument and fallacy applies to a communication network that is architected based on expected values and, yet, delivers services to real-time applications.

Step 2-2. Specify the Centrality Measure

The second step of the latency-centric network architecting process is related to the specification and estimation of the centrality measure $\mathcal{H}[n]$. Four elements have to be specified: The baseline system architecture and its latency contributors; the data flows and their relative importance; the utility function that links stakeholder satisfaction with data latency; and the normalization function to be used for ranking purposes. They are sequentially discussed in Step 2-2, Step 2-2, Step 2-2 and Step 2-2.

Step 2-2.1. Characterization of Latency Contributors

One of the fundamental premises of Section 2.3 is that there exists a network of nodes that relay information from origin to destination while, at the same time, introducing a certain delay L_n. The goal of this step is to construct this network given a baseline system implementation. This baseline system corresponds, typically, to a low performing, low budget alternative that is unsatisfactory from the user perspective.

A critical part of this step it to functionally decompose the system, identify the relevant latency contributors and then map them onto physical elements that are equivalent to the network nodes from Section 2.3. Note that the result of this process is not necessarily equivalent to the physical topology of the network. For instance, consider a simple WAN that carries TCP traffic. The TCP protocol can be identified as a source of the latency because it limits the transmission throughput through the sliding window mechanism. Then a possible representation of the system would be a two node network, one for the TCP protocol and another one for the underlying WAN network.

Unfortunately, there is no unique "magic" process by which the aforementioned decomposition can be performed. The systems engineering literature suggests using tools such as DSMs (see, for instance, Dwyer [18] or Sinha [122]), as well as more advanced computational techniques such as clustering algorithms (e.g. Reference [23] or Reference [126]). Regardless of the approach, the key idea is to reduce the system complexity by aggregating latency contributors in elements of form that are as independent as possible. However, as is often the case in system engineering problems, there is a trade-off between the level of aggregation and decomposition for latency contributors. Too much aggregation can lead to a single node sharing many latency contributors, thus making it impossible for the centrality measure to distinguish between them. At the same time, to much disaggregation can result in nodes of the network sharing latency contributors, thus reducing the value of applying the centrality measure in the first place.

Step 2-2.2. Identification and Characterization of Data Flows

Next, the system architect has to identify the sources and sinks of data, the paths through which data flows from a remote spacecraft to the end-user, and their relative importance. As mentioned in Section 2.3, properly characterizing data flows through the network is important because nodes should be blamed according to how many flows they disrupt with their induced latency. In social network analysis, two extreme routing strategies are typically analyzed: Geodesic paths, and random walks. While the former are clearly of interest in traditional communication networks (e.g. Reference [127]), it is worth noting that other types of network such as wireless sensor networks are better modeled using the second approach (see, for instance, Reference [128]). Nevertheless, for the purposes of modeling space communication networks, it is typically reasonable to assume that paths will be determined by minimizing some measure of distance, be it number of links traversed by data or geodesic distance.

On the other hand, characterization of the relative importance between flows is essential in order to ensure that the system is optimized to meet the requirements of its stakeholders. Importantly, observe that the weights w_p in Equation (2.12) are applied over the data utility function. Therefore, they should be estimated so that differences in relative utility are properly captured (rather than quantifying changes in communication-related metrics such as data volume). Indeed, a user that obtains the same utility from listening music or watching videos should be characterized by two data flows with the same utility, regardless of the fact that the former application requires less bandwidth than the latter. This subtle point is further explored in the third stress case of Chapter 3, where heterogeneous flows through the network are generated by varying the amount of information sent from certain nodes while keeping the utility per packet constant. Similarly, in Chapter 4 w_p is estimated by considering the importance of a certain data product for generating accurate forecasts, regardless of the data volume generated by the instruments that collect them.

Step 2-2.3. Characterization of Data Utility

The third building block of Equation (2.12) is the utility function $U(\cdot)$ that characterizes the value of data depending on the latency with which it is delivered to the end user. This utility function is unique to each application and can therefore not be generalized. That being said, I now comment on the set of characteristics that are typically assumed in utility theory. Firstly, utility functions define satisfaction in a relative scale. They are therefore dimensionless and can only be used for relative comparison analysis. Second, given that utility functions are dimensionless, their defining characteristics are specified thorough their shape. To achieve this, economics and finance impose restrictions on both the first and second derivative by virtue of two fundamental principles: Investors prefer more than less; and investors are risk averse. Similarly, and based on the discussion from Section 2.1, in latency applications it is only possible to impose a constraint on the first derivative i.e. $U'(\cdot) < L$ for all its domain. Note that if this constraint is does not exist, then there is generally no trade-off between performance and cost when architecting the system. Indeed, a network that delivers information later is better than one than delivers it immediately, as well as less expensive, thus being optimal under all metrics.

Step 2-2.4. Definition of a Normalization Scheme

The last step of the centrality metric estimation process is related to the choice of a normalization factor. Once again there is no unique solution to this problem, but rather the system architect can choose it depending on her/his preference. Typical choices that can be used are

- Sum normalization: $\mathcal{H}[n] = \dfrac{\sum\limits_{\forall p \in P_n} w_p \bar{U}_p[n]}{\sum\limits_{n=1}^{N} \mathcal{H}[n]}$
- Range normalization: $\mathcal{H}[n] = \dfrac{\sum\limits_{\forall p \in P_n} w_p \bar{U}_p[n]}{\max\limits_{\forall n} \mathcal{H}[n] - \min\limits_{\forall n} \mathcal{H}[n]}$
- Betweenness normalization: $\mathcal{H}[n] = \dfrac{\sum\limits_{\forall p \in P_n} w_p \bar{U}_p[n]}{k(N-1)(N-2)}$

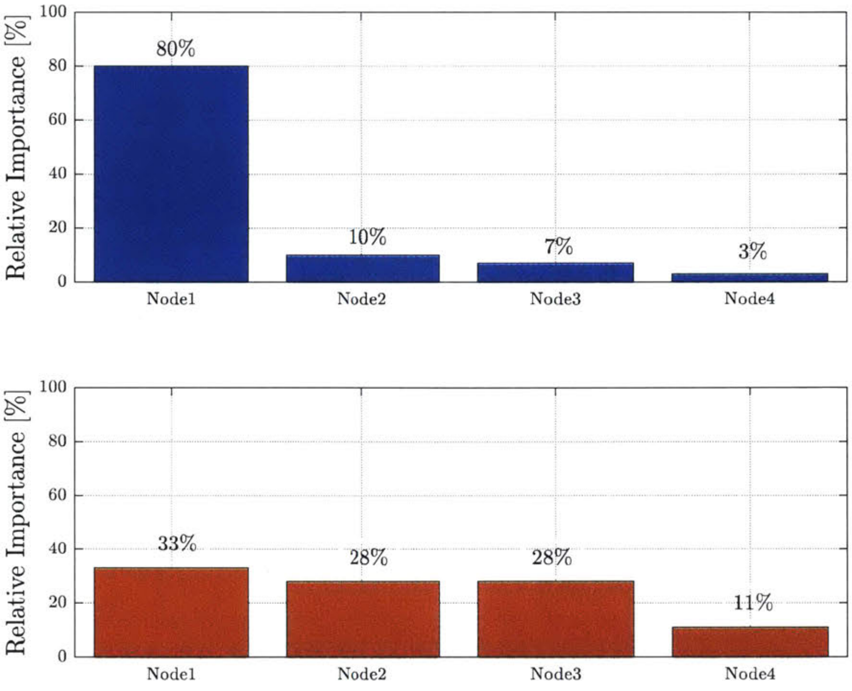


Figure 2-4: Ranking example

with $k = 1$ and $k = \frac{1}{2}$ for directed graphs and unidirected graphs respectively, and assuming that P_n are computed using geodesic paths. In general, the first or second alternative will be used during this dissertation as they are flow agnostic and are equally valid for constructing rankings latency contributors.

Step 2-3. Ranking of Latency Contributors

Once the system architecting centrality measure has been properly specified, it can now be applied to the baseline network architecture in order to assign a number that quantifies how much each node can be blamed for utility loss in the system. Two aspects of the ranking are typically of importance to the system architect: The element order and the ranking shape. To visualize the difference, Figure 2-4 presents two notional rankings with the same node ordering but with different shape, where the relative importance is equivalent to the output of the centrality measure after applying the sum normalization. Observe that making system architecting decisions based on both rankings would yield completely different results. In the upper ranking, "Node 1" is clearly the part of the system that should be improved as it contributes to 80% of the total utility loss. In contrast, in the lower ranking there is little evidence that "Node 1", "Node 2" and ,"Node 3" should in fact be improved in this order since they contribute approximately the same amount to utility loss.

In summary, ranking of system elements with respect to the percentage of utility loss they generate has the advantage of providing a clearly actionable output, i.e. which nodes in the system should be improved first. Furthermore, by virtue of the functional decomposition from Step 2-2, each node is the physical representation of a set of latency contributors. Therefore, the proposed centrality measure and the rankings it generates are an efficient approach to identifying latency bottlenecks across the entire system, i.e. which functionality is the *dominant* factor that should be improved in order to reduce utility loss. Finally, in the case of networks were data flows are well modeled by either geodesic paths or random walks, efficient implementations to compute the centrality measure and obtain the ranking for networks with thousands of nodes are available by analogy with betweenness and flow betweenness.

Step 2-4. Problem Formulation

The results of the ranking provided by the centrality measure serve as a first step to quantify which part of the system (and more importantly which latency contributors) dominates the data utility loss. In a sense, the centrality measure corresponds to a high level screening process that ultimately results in a more informed decision on where to spend time and resources conducting a more in-depth and higher fidelity system architecting exercise.

The focus of this step is precisely to formulate this higher fidelity system architecting exercise, typically with the aid of computational tools that help enumerate and explore a large space of alternatives. Importantly, this highlights another advantage of using the centrality measure as a pre-step to an in-depth system architecting analysis. Not only does it facilitate the modeling process by ensuring that only certain parts of the communication stack have to be included, but it also reduces the dimensionality of the problem and facilitates the work of optimization tools.

Step 2-4.1. Definition of Case Study Assumptions and Goals

Given that latency contributors inherent to "Node 1" have been identified as the critical part of system to address, it is now possible to clearly state the system architecting exercise goals as a set of specific research questions. Ideally these questions should be concise enough to generate a set of actionable recommendations, which will be documented in Step 2-7. Furthermore, they should also be the basis for defining of the objective metrics against which the system will be optimized. Finally, they should also help understand which assumptions were made during this part of the system architecting process.

Step 2-4.2. Definition of Architectural Space

The definition of the architectural space aims at finding the set of alternatives in the formal domain that can address the same functions (and associated latency contributors) that it currently performs. Going back to my TCP over WAN example, assume that TCP has been ranked as the first latency contributor followed by the WAN capacity. In the definition of the architectural step the system architect search alternative implementations to the TCP protocol that can address the same functions and yet overcomes the limitations of the TCP limited sliding window. Another option could be to maintain TCP and "tweak" its implementation to unbound the sliding window at the expense of worse congestion control. Finally, substituting TCP for a less capable protocol such as UDP could also be an option, albeit no reliability or congestion control functionality would be provided in that case [129].

In general, the definition of a given architectural space is a specific instance of a SAP (see Section 1.3.3). In simple cases, the different set of architectural options can be specified through a Structural Morphological Matrix (e.g. Reference [130]) where each column represents an element of form and each column contains different options for its implementation. In short, the decision-option paradigm from Selva [28] is encapsulated in a tabular format to facilitate its understandability. In other cases, other types of combinatorial problems are better suited for encoding the architectural problem. These include, for instance, partitioning problems or connecting problems among others. Both Reference [23] and [28] provide an excellent overview of the combinatorial patterns that are possible in system architecting problems. Finally, combinations of architectural decisions encoded as a combinatorial decision tree have also been applied in the literature [131]. In that case, multiple combinatorial problems are encoded together, along with their respective dependencies and constraints.

A key element to consider during the definition of the architectural space is its dimensionality. Unfortunately combinatorial problems suffer from the curse of dimensionality [132], i.e. the number of potential architectures that can be generated increases exponentially with the number of alternatives. While improvements of computational capabilities, with faster processors and multi-core CPUs have partially mitigated this problem, keeping track of the size of the tradespace still remains a critical exercise to be conducted by the system architect. Ultimately, given the current state of technology and a finite time horizon, there is a trade-off between the number of alternatives that can be evaluated and the fidelity of the models that perform the evaluation process [28]. As an example, consider a set of N candidate ground stations that can be selected to create a network that provides contact opportunities to satellites orbiting the Earth. The problem can be formulated as an *assignment problem* with two possible values $\{0, 1\}$, where 1 indicates the presence of the ground station in the network (and vice versa). Figure 2-5 plots the tradepsace dimensionality (blue line), along with the dimensionality of a partitioning problem with the same number of ground stations (orange

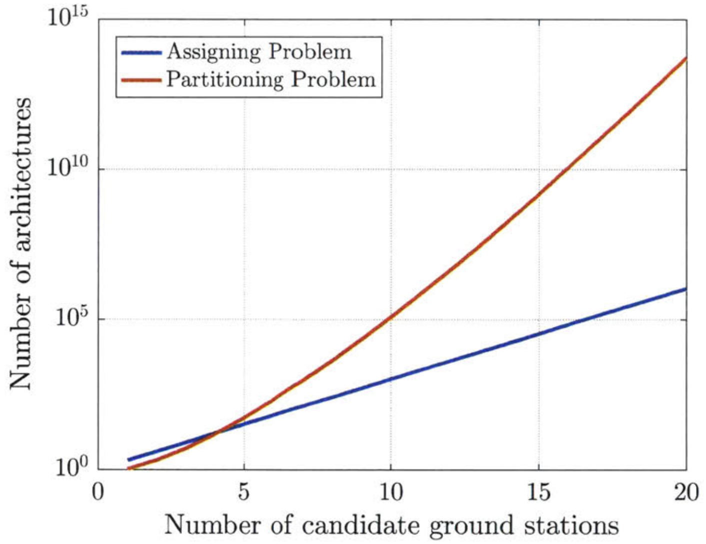


Figure 2-5: Tradespace Dimensionality

line). Observe that, even for 20 ground stations, a little over one million architectures have to be evaluated for the assigning problem, while 50,000 trillion architectures are possible in the partitioning problem. At one second per architecture, this results in 11 days of computational time for the first problem and 1.5 million years for the second.

Step 2-4.3. Model Development and Validation

At this stage, it is possible for the system architect to spend time and resources developing the necessary computational models and tools that will help her/him conduct the system architecting exercise. As indicated in Section 1.3.3, these range from low fidelity parametric models and CERs, to medium fidelity tools such as rule-based expert systems [28] or multi-commodity flow algorithms [133], to full-fledged discrete event simulation systems such as SpaceNet [125]. Selection of one versus the other is primarily related to the trade between model fidelity, complexity and computational time: High fidelity models are more accurate, but they also require more time to develop, maintain and exercise, and they also require more computational resources to run. Once again, there is no "magic" rule for selecting the right model or tool. It depends on the case study goals and the dimensionality of the tradespace, as well as the level of accuracy required. As a general rule of thumb, 10% to 15% uncertainty in the input parameters of a system architecting exercise is not uncommon. Since this uncertainty will propagate through the models and tools, a similar level of accuracy for the produced outputs is recommended.

Finally, an optional but certainly useful sub-step once the model has been developed is to check its accuracy against a known set of results. This process, also referred to as model validation, ensures that the recommendations extracted from the system architecting exercise are meaningful. That being said, model validation can be particularly difficult for architecting studies. Indeed, new concepts proposed in early stages of the system design process do not necessarily exist in reality. Similarly, validation of cost models for governmental systems (e.g. space programs, naval systems) can be challenging due to lack of historical data or restrictions to the academic community.

At least two approaches can be followed to overcome these limitations: First, use analog systems to either validate or, if necessary, calibrate the model. This is the approach used by Do [134] in his evaluation of habitation systems for human Mars surface exploration. Evidently, no real space habitats have been built or tested at Mars. But analogies with the ISS allowed him to match his model's astronaut water consumption rate with that of the Earth orbiting station. On the other hand, validation can also be done at the subsystem level. In that case, if the entire system model cannot be validated, at least ensure that each component sub-model provides reasonable results. Note that this second approach is of limited applicability due to the emergent property of complex systems. Yet, it remains a better alternative than no validation altogether.

Step 2-5. Analysis of Results

Results analysis is the fundamental step of the system architecting process. It transforms a set of inputs and models into a set of final recommendations that either specify which decisions should be implemented by the system architect, or provides insight into families of architectures or system concepts that are preferable.

To transform inputs to recommendations, the system architect must first explore the set of feasible architectures for the system under consideration. This requires enumerating and down-selecting architectures from a potentially large set of options, to a few preferred alternatives. Performing this step has been traditionally accomplished through a wide variety of combinatorial optimization algorithms - see Section 1.3.2.3 of Reference [28] for an excellent review of the topic. While some authors have compared their performance in a subset of test problems (e.g. Reference [135]), selection of an optimization algorithm largely remains an ad-hoc process that should be tailored on a case-by-case basis. As noted by Selva [28], "the common trade-off made to tackle this problem is to sacrifice exactness of the [optimal] solution to gain in computational time".

On the other hand, defining figures of merit to benchmark alternative architectures is typically well understood process, especially in combination with utility theory. Golkar notes that "in typical aerospace applications, architectures are evaluated by performance [...] metrics (such as total dry mass of an architecture), total cost, and other quantitative metrics".

They are, consequently, multi-objective optimization problems that are solved using two complimentary approaches: Scalarization and Pareto fronts [136]. In scalarization, preferences are expressed upfront through a set of weights that quantify the relative importance of each system objective. As a result, the optimization objective function typically has linear functional form

$$J(\boldsymbol{x}) = \sum_{k=1}^{K} \alpha_k J_k(\boldsymbol{x}) = \boldsymbol{\alpha} \cdot \boldsymbol{J}(\boldsymbol{x}) \tag{2.14}$$

where K denotes the number of system objectives, α_k captures the relative importance of the k-th objective, and $J_k(\boldsymbol{x})$ is the function that transforms the vector $\boldsymbol{x}$ of inputs into the k-th system metric. Alternatively, the Pareto front approach lets the system architect express preferences a posteriori by computing a family of optimal architectures for which a metric cannot be improved without worsening at least another. Figure 2-6 depicts a notional **tradespace**, i.e. a scatter plot in the metrics space

$$\boldsymbol{J}(\boldsymbol{x}^*) = \begin{bmatrix} \text{Performance}(\boldsymbol{x}^*) \\ \text{Cost}(\boldsymbol{x}^*) \end{bmatrix} \tag{2.15}$$

where each blue doc represents a given architecture. Furthermore, some of them (the red dots) are also *efficient* $\boldsymbol{x}^*$, i.e. their objective vector is *non-dominated.* In some cases, these non-dominated solutions are sub-classified into weak and strong dominance. $\boldsymbol{J}(\boldsymbol{x}_i)$ is said to weakly dominate $\boldsymbol{J}(\boldsymbol{x}_j)$ if and only if $\boldsymbol{J}(\boldsymbol{x}_i) \geq \boldsymbol{J}(\boldsymbol{x}_j) \forall i, j$ and $\boldsymbol{J}(\boldsymbol{x}_i) > \boldsymbol{J}(\boldsymbol{x}_j)$ for at lease one i. Similarly, $\boldsymbol{J}(\boldsymbol{x}_i)$ is said to strongly dominate $\boldsymbol{J}(\boldsymbol{x}_j)$ if and only if $\boldsymbol{J}(\boldsymbol{x}_i) > \boldsymbol{J}(\boldsymbol{x}_j) \forall i, j$[5]. Finally, an architecture is said to be Pareto-efficient if its metrics are at least weakly non-dominated with respect to all other architectures in the design space. In turn, the set of Pareto-efficient architectures is referred to as **Pareto front** or **Pareto frontier**.

Finally, two useful concepts to analyze tradespaces in the context of system architecture are *main effects* and *interactions.* In its simplest form, a binary decision's main effect with respect to metric $J_k(\boldsymbol{x})$ is computed as the difference between the average score obtained when the decision is on and off. As it name indicates, it quantifies the difference in system performance as a function of the decision value, which in this case is assumed to only take two values:

$$\mathrm{MF}_k(d) = \frac{1}{N} \left[J_k(\boldsymbol{x}|d) - J_k(\boldsymbol{x}|\neg d) \right], \tag{2.16}$$

where N is the number of system implementations, d is the decision "turned on" and $\neg d$ is the same decision "turned off". Note that the average is computed over all evaluated system

[5] While these concepts are defined here with more formality than is required for understanding the results of this thesis, they are fundamental to understanding the case studies and are therefore summarized for future reference.

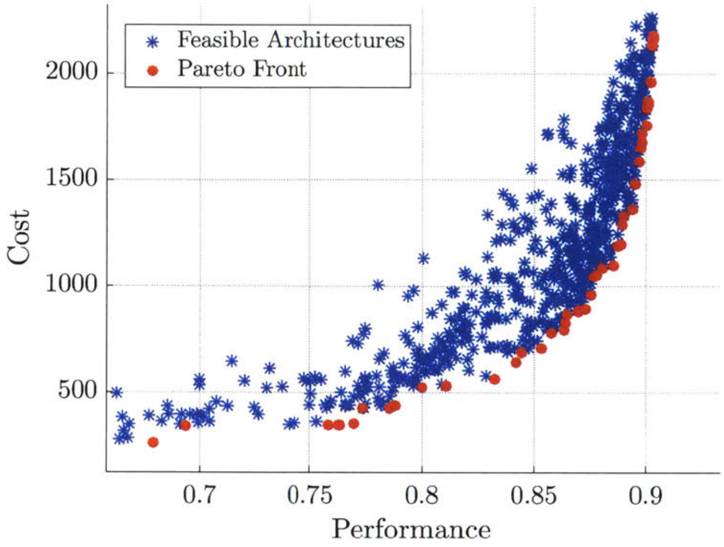


Figure 2-6: Pareto Front Example

implementations regardless of whether they lie in the Pareto front. Similarly, the interaction between decision d_1 and d_2 measures the average difference in $J_k(\boldsymbol{x})$ as a function of d_1 and d_2, as the difference between the main effect for d_1 assuming d_2 is "turned on" and "turned off". Mathematically, we say that

$$\text{Int}_k(d_1, d_2) = \frac{1}{2}\left[\text{MF}_k(d_1|d_2) - \text{MF}_k(d_1|\neg d_2)\right] \tag{2.17}$$

Step 2-6. Identification of Second-Order Latency Contributors

As indicated by Golkar, "Pareto analysis is an effective tool to facilitate the achievement of an optimal compromise between scientific ambitions, engineering requirements and program management constraints" [130]. For instance, in the case of architecting a space communication network that provides services to latency-sensitive applications, utility as a function of latency would be the proxy for performance to be traded against life cycle cost. That being said, recall here that ever since Step 2-4 the system architect has been focusing all efforts in understanding the primary latency contributor identified through the ranking provided by the centrality measure. Therefore, it is important for the system architect to understand at which points in the tradespace the relative importance of the different latency contributors changes. Otherwise, she/he could spend a significant amount of resources perfectly optimizing part of the system when, in reality, the focus should be put in re-architecting another completely different part.

To visualize this issue, Figure 2-7 plots the same tradespace as in Figure 2-6 but color

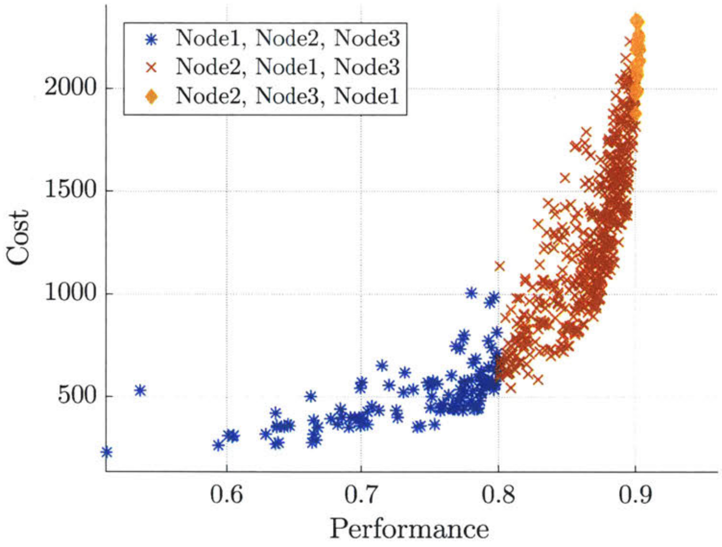


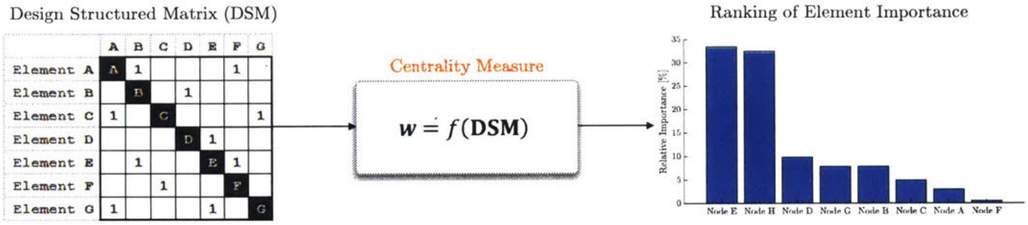
Figure 2-7: Ranking of contributors in the metrics space

codes according to the ranking of latency contributors. From Figure 2-4, we know that "Node 1" is currently the largest latency contributor and therefore create a tradespace of alternatives that improve the system performance by reducing the latency induced by "Node 1"'s functionality. Nevertheless, at approximately 0.8 in the performance scale, the system architect notes that "Node 2" has in fact become the largest contributor. Therefore, from this point onwards, she/he should be aware that a better solution for his system optimization problem could be to stop spending resources in "Node 1" and start improving "Node 2". Finally, if the system architect chooses a system with a 0.9 performance (yellow zone of Figure 2-7), then both "Node 2" and "Node 3" are larger latency contributors than "Node 1" and therefore the solution offered in this region of the tradespace is optimal from "Node 1"'s perspective, but sub-optimal from the systems perspective.

Step 2-7. Development of Recommendations

Development of recommendations provides the system architect and her/his stakeholders with clear guidance on how to implement the architecting process. From the perspective of the latency-centric approach described in this chapter, at least three types of recommendations should always be provided:

- Which are the primary latency contributors and which parts of the baseline system are currently inducing them.
- Which Pareto efficient architectures can be devised to improve the limitations outlined in the first recommendation.



	A	B	C	D	E	F	G
Element A	A	1				1	
Element B		B		1			
Element C	1		C				1
Element D				D	1		
Element E		1			E	1	
Element F			1			F	
Element G	1				1		G

Figure 2-8: Generic Centrality Measures

- Which second order latency contributors should also be taken into account by the system architect. In particular, at which point is it better to spend resources improving secondary latency contributors as opposed to the primary one.

2.5 Centrality Measures for System Architecting

In Sections 2.1 and 2.3, I defined a centrality measure to guide the system architecting synthesis process for space communication systems that deliver latency-sensitive information from space exploration applications. Importantly, observe that the centrality measure, as defined, is only applicable to the specific type of systems and the specific set of space exploration applications that I am interested in. In other words, it is built upon domain-specific knowledge that is only useful within the context of this thesis' topic.

The realization that solving system architecture problems requires a large body of domain-specific knowledge is not new. Indeed, Selva proposes in Reference [28] an approach to system architecture synthesis based on rule-based expert systems precisely because they explicitly separate the knowledge-intensive part of the problem from generic information that is always applicable regardless of the system under consideration. Being that the case, in this section I consider how centrality measures for system architecture can be defined in a generic context.

To start the discussion, I provide in Figure 2-8 a pictorial representation of how centrality measures are used to analyze generic systems. The key elements are as follows: First, the complex system is decomposed and simplified to a DSM which essentially flags interactions between elements of the system through ones in the off-diagonal. This DSM is used as an input to a given centrality measure, which outputs a vector $\boldsymbol{w}$ of weights that indicate the relative importance of a given element based on the set of interactions captured in the DSM.

To define the centrality measure $f\,(\mathbf{DSM})$, several pieces of information are required. They are essentially equivalent to building blocks that can be used to progressively refine its functional form, from a generic structure to the final equation that properly captures all

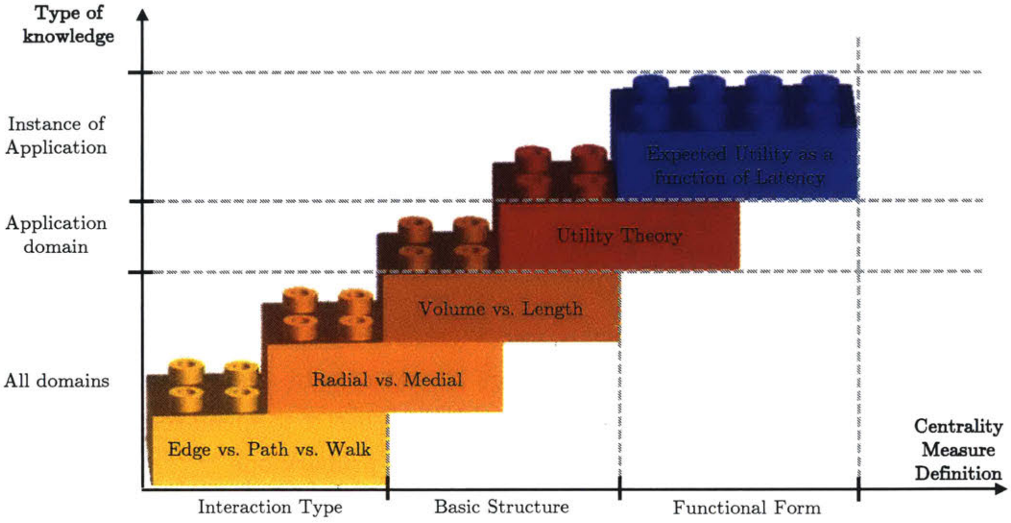


Figure 2-9: Generic Definition of Centrality Measures for System Architecting

necessary domain-specific knowledge. In that sense, Figure 2-9 shows how these building blocks were put together in Section 2.1 to obtain the proposed centrality measure. Note that in the figure, the x-axis represents the process of defining the centrality measure, while the y-axis represents the type of knowledge required, from domain-independent to application-specific.

The first three building blocks required to define a centrality measure for system architecting are categorized as domain independent since they I adapted them from Borgatti [99], who in turn obtained them by reviewing how centrality measures had been applied in a wide variety of technical and social systems. In that sense, the first piece of information required to define or select a centrality measure is related to interactions within the system, both direct and indirect. For instance, in a communication network the most important interaction is transmission of data. Since data is usually routed through shortest paths, it is sensible to assume that only centrality measures that capture indirect interactions based on the concept of paths are suitable to study these types of systems. In other disciplines such as project management, interactions are expressed as dependencies between tasks (e.g. task A cannot start until task B is completed), as well as rework cycles, which are essentially equivalent to indirect interactions. In this case, the critical task is typically defined as the one with highest project schedule overrun potential and, consequently, requires evaluating the number of rework cycles that would be affected if a given task is disrupted. To efficiently implement this notion, project managers take the original project DSM as well as its powers (DSM^n finds rework cycles of length n). Note that this is essentially equivalent to Alpha, Katz and Eigenvector centrality, with the difference that in the latter cases cycles of $n \to \infty$ are counted and divided by a weighting factor related to the DSM's largest eigenvalue to ensure convergence of the summation. In other words, in project management centrality measures are defined based on the concept of random walks, as they model the rework cycles of n-th length within the project.

Once the interactions have been modeled through either edges, paths or walks (edges model systems where only direct interactions matter), the next step is to understand weather the measure will be radial or medial. By definition, radial measures assign importance to a node because of the interactions that start or end at the given system element, while medial measures consider interactions through it. In that sense, communication networks are usually better characterized by medial measures that assign importance based on the importance of the data flows that pass through a given node. In contrast, transportation or logistic networks (see, for instance, Reference [133] in the space context) would typically prefer radial centrality measures where important nodes will identify elements in the system that can directly or indirectly reach a large number of destinations.

The last set of domain-independent information required to specify a centrality measure allows the system architect to differentiate between volume and length measures. The former assign importance based on *how many* interactions start, end, or flow through a node, while the latter consider *the length* of these direct and indirect interactions. In that sense, a communication network will typically favor volume measures that assign weight to a node because it transmits a large volume of information. Alternatively, a space logistic network will be better analyzed through length-based measures where path length is a function of the total Δv required to reach a destination from that node.

Once the domain-independent features of the centrality measure have been identified, the system architect must start incorporating domain specific knowledge to refine its basic structure. For instance, in Section 2.1 I built upon the concept of utility theory to define the importance of a node as a function of the utility loss attributable to a given element of the system. In particular, I argued that the centrality measure should prioritize areas of the system that introduce large delays since they decrease the value of the scientific data returned in latency-sensitive applications. That being said, other applications might consider multiple figures of merit or metrics to include in the definition of the centrality measure. Indeed, financial and engineering systems might also utilize return on investment as a possible candidate, while risk and other "ilities" can be incorporated through risk-adjust return on investment, or multi-attribute utility theory.

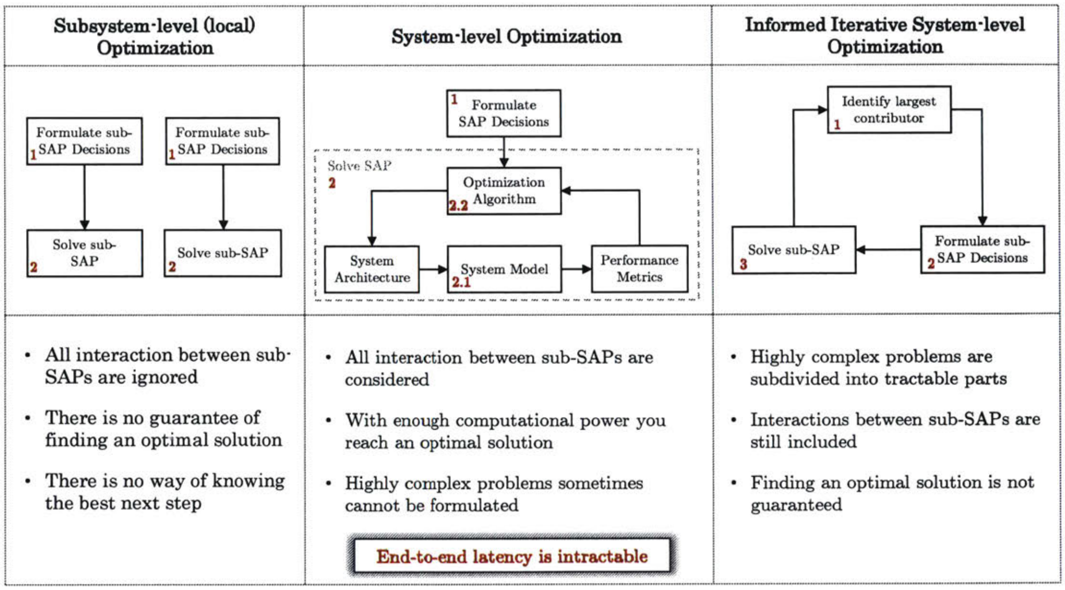
Finally, the last set of knowledge required to define a centrality measure is specific to the instance of application under consideration. To exemplify this point, consider this thesis' research questions. The application domain is clearly space communication systems, which vindicates the use of paths to model interactions, as well as the choice of a medial volume-based centrality measure that builds upon the notion of utility theory. On the other hand, the instance of application is related to tailoring the application to latency-sensitive space exploration application as opposed to generic latency-constrained applications. Indeed, focusing exclusively in latency-sensitive applications simplifies the definition as it allows the utility function to be linear or quasi-linear. This, in turn, also justifies its applicability in the presence of non-deterministic latency contributors, as long as the conditions from Section 2.2 are met.

2.6 Summary

In this chapter I have proposed a latency-centric approach to architecting space communication networks that provide services to latency-sensitive applications. Through Sections 2.1 to 2.3, I proposed a centrality measure that can be used to quantify the amount of utility loss that is incurred in a given network node due to the latency it introduces. Then, I have described a seven step system architecting procedure that utilizes this centrality measure as a pre-screening process that identifies which parts of the system would, if re-architected, yield larger performance improvements.

Figure 2-10 provides a schematic representation of how the proposed approach fits within past system architecting processes. Initially, prior to the holistic principles of systems engineering and system architecture, each part of a complex system was optimized independently (left column). For instance, Reference [137] describes an example in which a software team optimize their architecture for high performance without realizing the footprint in memory requirements. This prompts the hardware team to utilize larger memory racks, which do not fit within the optimized mechanical enclosing and overheat due to unexpected thermal requirements. Therefore, since each part of the system (software, electronic hardware and physical spacing) are optimized independently, the system ends up being more complex and costly than necessary.

To overcome these limitations, system architecture proponents consider the system holistically. Instead of seeking optimized solutions first and then worry about integration, the goal is to define a set of high level decisions that largely characterize the system performance and cost given all components and phases of its life cycle, create a simplified model for it, and optimize the architectural decisions (central column of Figure 2-10). Over the last decade, this approach has been progressively explored in multiple problems. For instance, Selva [28] applied it to Earth observation satellite programs, Alibay [123] and Golkar [130] used it for deep space planetary exploration campaigns, while Jilla [138] analyzed distribute satellite systems for communications and navigation. Yet, as the complexity and heterogeneity of problems tackled increases, so does the difficulty in obtaining reasonable holistic models to optimize, as well as the computational power and optimization techniques required to


Subsystem-level (local) Optimization
System-level Optimization
Informed Iterative System-level Optimization
1 Formulate sub-SAP Decisions
1 Formulate sub-SAP Decisions
2 Solve sub-SAP
2 Solve sub-SAP
1 Formulate SAP Decisions
Solve SAP 2
2.2 Optimization Algorithm
System Architecture
2.1 System Model
Performance Metrics
1 Identify largest contributor
3 Solve sub-SAP
2 Formulate sub-SAP Decisions

- All interaction between sub-SAPs are ignored
- There is no guarantee of finding an optimal solution
- There is no way of knowing the best next step

- All interaction between sub-SAPs are considered
- With enough computational power you reach an optimal solution
- Highly complex problems sometimes cannot be formulated

End-to-end latency is intractable

- Highly complex problems are subdivided into tractable parts
- Interactions between sub-SAPs are still included
- Finding an optimal solution is not guaranteed

Figure 2-10: Comparison of system architecting approaches

produce meaningful results [28].

The proposed latency-sensitive centric approach to architecting networks has been developed, partially, in response to the difficulty of creating holistic system level models for space communication networks with multiple heterogeneous sources of latency. In a sense, it can be thought as a step back since the system architect tries to optimize parts of the network independently. However, as opposed to the initial purely subsystem-level optimization, it retains the ability to take into account interactions across diverse network elements and functions through the use of a centrality measure that keeps track of which factors are the most important (see right column of Figure 2-10). In other words, the centrality measure acts as a heuristic function that, given a network architecture, identifies which parts of the system are most deleterious in the performance metric and lets the system architect focus its attention towards them. At the same time, it also provides the system architect with an efficient way to keep track of how much second order latency contributors affect the system. This prevents the system architect from spending all resources in a single latency contributor when in fact she/he would be better off improving other parts of the system. Note that, as indicated in Figure 2-10, this resembles heuristic optimization technique by which the system architect is performing locally optimal trades due to the impossibility of formulating overall system trades. As a result, there is, in general, no guarantee of optimality. Yet, the centrality measure informs the system architecting process and ensures that resources are not wasted in unimportant parts of system.

THIS PAGE INTENTIONALLY LEFT BLANK

3 Case Study 1: IP Wide Area Network

3.1 Introduction

In Chapter 2, I introduced a centrality measure that can be used to inform the system architecting process when a holistic model that captures all latency contributors cannot be formulated. I also indicated that this centrality measure is equivalent to a heuristic function that identifies which latency contributors are most significant in the system's performance degradation and should therefore be addressed first.

In this chapter, I study the performance of the system architecting centrality measure when optimizing a simulated terrestrial packet-based network. The rationale for this case study is better explained through the *design evaluation methods* listed by Hevner in Reference [139]. In that sense, the results herein presented intend to (1) demonstrate the usefulness of the proposed approach in a controlled environment with artificial inputs, and (2) demonstrate the bounds under which the proposed approach yields optimal or near optimal results. It is therefore complimentary to the other two case studies in this thesis, as those assert the its validity through in depth analysis of a realistic application.

To satisfy this case study stated objectives, I utilize the following process: First, define an initial idealized network architecture based on a finite and predefined set of assumptions. Following the steps defined in the latency-centric approach to architect communication networks, demonstrate that the centrality measure successfully optimizes the system by identifying the primary latency contributors. Then, revise each initial assumption to create a set of "stress cases" that exemplify the limits of the proposed approach. Finally, demonstrate that with proper calibration of the centrality measure, the limits imposed by these assumptions can be overcome.

The rest of this chapter is structured as follows: First, the case study goals are stated and the stress cases for the centrality measure are defined. Second, the benchmarking strategy is explained in detail, with particular emphasis on the metrics utilized to assess the performance of the centrality measure. Next, the latency-centric approach to architecting networks is applied in the context of a terrestrial IP-based WAN. Finally, conclusions on the validity of the proposed centrality measure are summarized.

3.2 Case Study Goals and Assumptions

The specific set of objectives to be accomplished in this chapter are addressed through 5 canonical scenarios. Next, I summarize their intent:

1. **Baseline scenario**: Given a set of cost-homogeneous and performance-heterogeneous nodes interconnected with arbitrary topology and a finite set of assumptions, demonstrate that the proposed latency-centric approach obtains the correct system Pareto front and latency-contributor ranking.

2. **Stress case 1**: Demonstrate that the effectiveness of the centrality measure does not depend on the shape of the utility function $U(L)$ as long as $U'(L) < 0$.

3. **Stress case 2**: Demonstrate that the effectiveness of the centrality measure and the network optimization process is a function of the data routing strategy.

4. **Stress case 3**: Demonstrate that misrepresenting data flow importance can lead to sub-optimal results during the network optimization process.

5. **Stress case 4**: Demonstrate the effect of cost-heterogeneity across different nodes in the network (and their corresponding latency contributors).

The initial assumptions used to define the baseline scenario from goal 1 are as follows:

1. The utility function $U(L)$ is decreasing and concave.

2. All data is sent through the network following a geodesic-based routing strategy.

3. All nodes generate the same amount of data per unit of time (on average). It is destined to any of the other nodes with equal probability.

4. All nodes are cost-homogeneous, i.e. the capital expenditure required to implement them is constant across all parts of the network.

3.3 Benchmarking Strategy

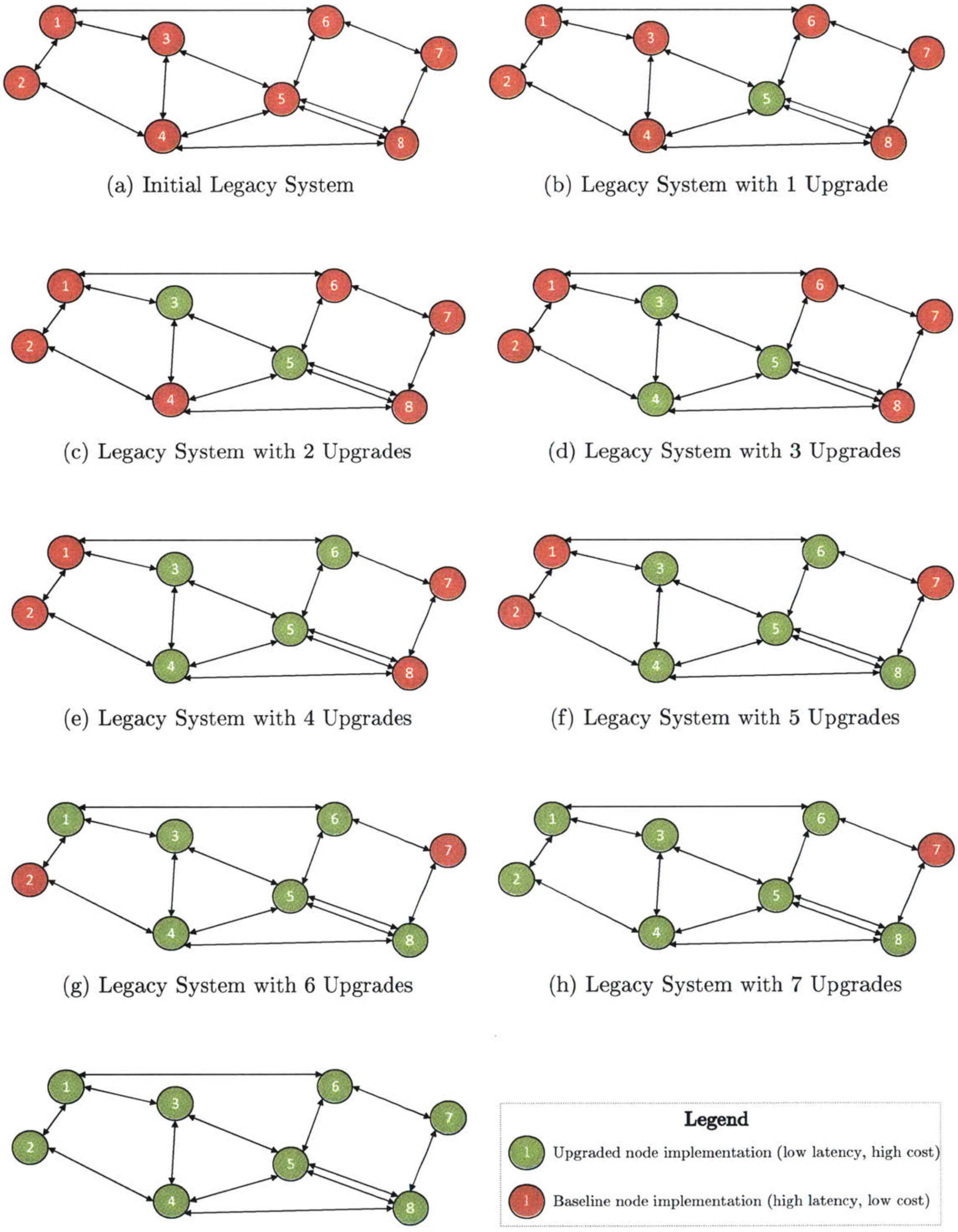
Following Section 2.3, assume that the arbitrary network from goal 1 can be modeled as a set of nodes and connections, where the defining characteristic between them is that nodes induce latency while connections do not. Assume also that each node has two potential physical implementations with different performance. Finally, assume that the network provides services to latency-sensitive applications and that we have a legacy implementation in place. Then, the key architectural question that we want to answer is, given this initial legacy system, which sequence of nodes should be upgraded so that the network performance improves (data is delivered with less latency) and yet the minimum amount of resources is spent. Importantly, recall here that each node groups one or multiple latency contributors. Therefore, in order to obtain the optimal sequence of nodes to upgrade, the system architect has to create a ranking of latency contributors, from highest to lowest, and spend resources upgrade nodes accordingly.

To exemplify the problem, Figure 3-1 presents the network from Chapter 2 with each node color coded according to the latency it introduces (red for "high" latency, green for "low" latency). Initially, the legacy under-performing system is implemented (Figure 3-1a). Then, the network is re-architected by selecting a sequence of nodes to be upgraded into a more capable, more expensive alternative implementation. Ultimately, given the available choices for any given node, the entire system is upgraded (Figure 3-1i). The same information is presented in Figure 3-2a in the form of a ranking that is analogous to the sequence presented in Step 2-3. This ranking quantifies the relative importance of a given node (and its latency contributors) in the overall system utility loss and, if ordered, indicates the optimal sequence of nodes to be upgraded. Finally, Figure 3-2b presents the network evolution path from its legacy implementation to its fully upgraded state in the metrics space. Each blue dot represents one possible system implementation from Figure 3-1 as indexed by the letter next to them. Blue dots are assumed to represent the optimal sequence of nodes to upgrade and therefore the orange arrows indicate the path closest to the tradespace's Pareto front. In contrast, the red dots and yellow-dashed arrows represent an alternative non-optimal sequence to optimize the system. Note that the end result is the same, a fully upgraded system. Yet, upgrading the network using the optimal sequence saves cost as compared to the non-optimal alternative and should, therefore, be utilized.

In Chapter 2, I claimed that the centrality measure from Equation (2.9) can, to first order approximation, be used to obtain the ranking that results in the optimal sequence of nodes to upgrade. To substantiate this claim, I conduct a simulation-based exercise in which I compare the latency contributor rankings and sequences of nodes to upgrade using two complimentary methods: Centrality measures, and dynamic programming. Ultimately, the proposed centrality measure will be validated if I demonstrate that the former can be used to obtain a "good enough" approximation of the latter[1]. Or, equivalently, it will be validated if it can estimate the correct ranking of latency contributors and their relative importance.

Traditionally, two approaches to validation have been utilized: Replicate a realistic system for which data is available; or compare the obtained results against those of a higher fidelity model that has already been validated. In this chapter, the latter approach is utilized (see Figure 3-3) using the discrete event network simulator ArchNet [140]. In particular, I first compute the performance and cost of all possible network architectures using ArchNet, and find the optimal ranking of latency contributors using dynamic programming. Then, I obtain a candidate ranking using only the centrality measure, the network DSM and the expected

[1] What defines a "good enough" sequence will be defined in Section 3.5.


1
2
3
4
5
6
7
8
Legend
🟩 Upgraded node implementation (low latency, high cost)
🟥 Baseline node implementation (high latency, low cost)

(a) Initial Legacy System

(b) Legacy System with 1 Upgrade

(c) Legacy System with 2 Upgrades

(d) Legacy System with 3 Upgrades

(e) Legacy System with 4 Upgrades

(f) Legacy System with 5 Upgrades

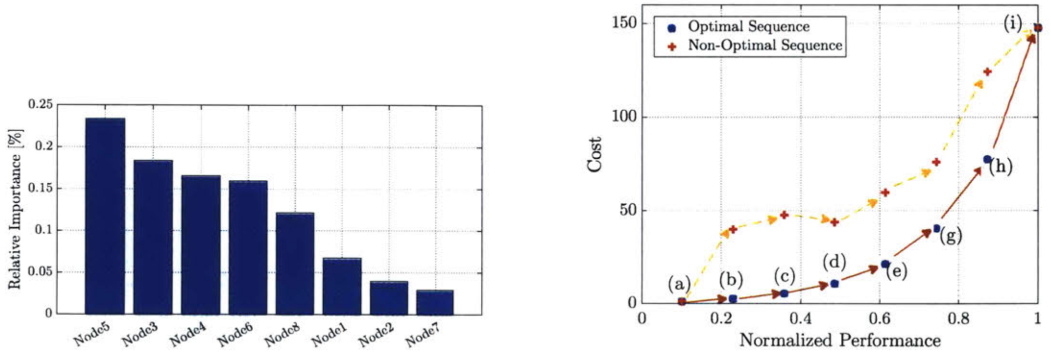
(g) Legacy System with 6 Upgrades

(h) Legacy System with 7 Upgrades

(i) Fully Upgraded System

Figure 3-1: Sequence of Network Node Upgrades

latency introduced by each element in the system. Finally, I quantify the goodness of the candidate ranking estimated with the centrality measure with respect to the optimal one using a statistical test that measures the probability of finding another candidate ranking that has higher degree of similarity (see statistical test 1 in Figure 3-3).



(a) Network Architecting Example in the Ranking Space

(b) Network Architecting Example in the Metrics Space

Figure 3-2: Network Architecting Example

Another test to quantify the performance of the proposed centrality measure can be defined in the metrics space rather than the ranking space. In particular, I utilize the concept of Pareto distance to obtain a score for each evolution path obtained from a given latency contributor ranking (see again Figures 3-2a and 3-2b for a visual representation of both a ranking and its corresponding evolution path). Then, I construct another statistical test that measures the probability with which someone could find an evolution path with lower total Pareto distance than the one found by the centrality measure (see statistical test 2 in Figure 3-3).

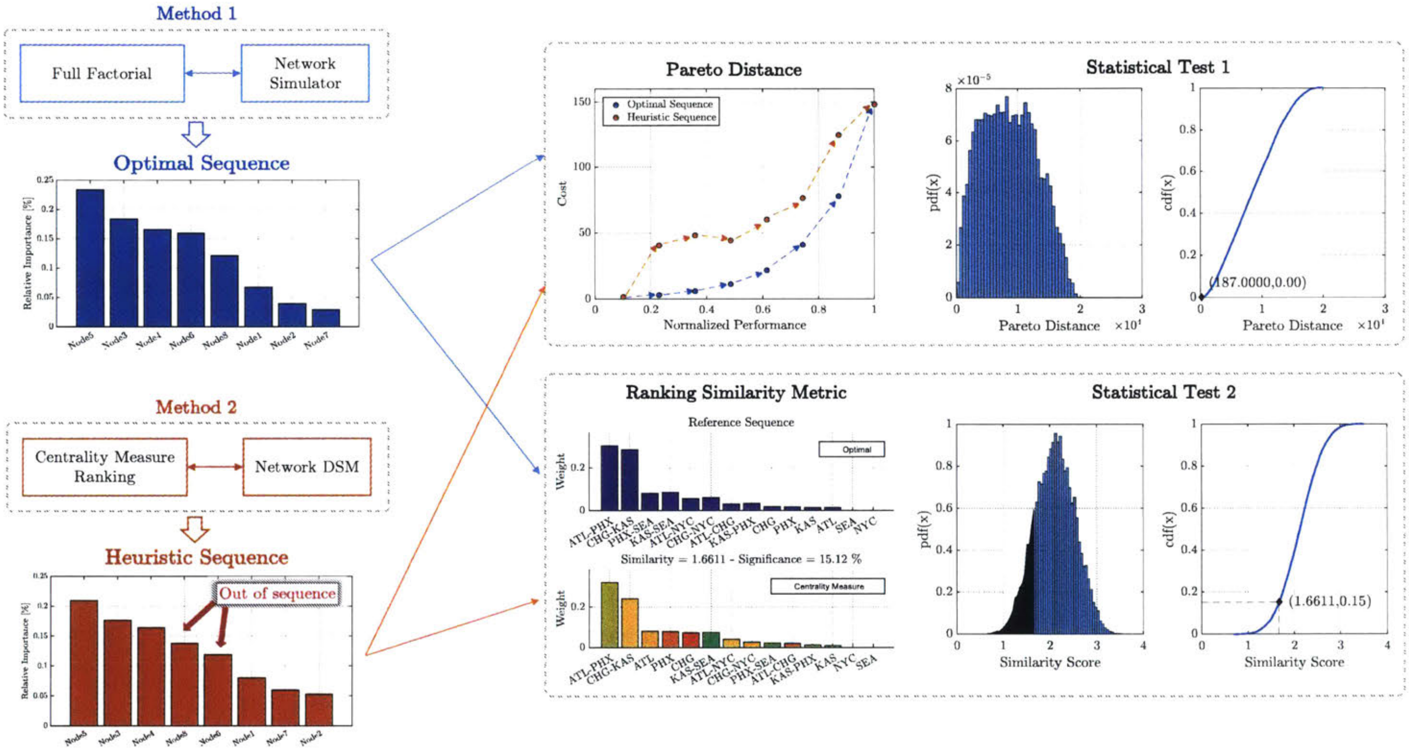


Figure 3-3: Validation Strategy

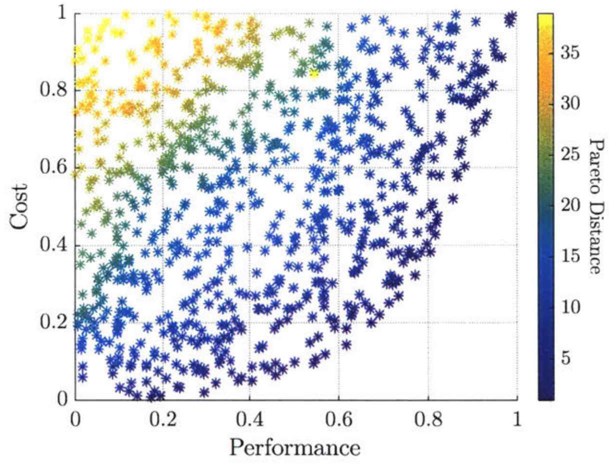


Figure 3-4: Pareto Distance Example

Finally, note that the proposed validation strategy only covers Step 2-2 and Step 2-3 of the latency-centric approach to architecting space communication networks. For Step 2-4, validation has to be performed on a case-by-case basis depending on the system under consideration and the latency contributors that dominate the system. Indeed, the performance and cost of space-based communication network like the SN has significant differences to that of the ground-based DSN. Therefore, the models utilized for tradespace exploration in either case will have to be developed and validated on an individual basis so as to ensure that results obtained are realistic given technological and programmatic limitations.

3.4 Test 1. Pareto Distance

The Statistical Test 1 from Figure 3-3 is based on the concept of Pareto distance. Therefore, in this section I first provide a succinct description of how Pareto distance is defined and computed. Then, I explain how to extract the optimal sequence of nodes as the sequence that minimizes the total Pareto distance. Finally, I describe the statistical test 1 and the obtained significance score.

Pareto distance, or distance to the Pareto front, is typically defined with an integer value that quantifies how many layers of architectures should be eliminated for a reference architecture to lie in the Pareto front [141]. To illustrate this definition, Figure 3-4 presents a notional tradespace where each architecture has been color-coded according to the Pareto distance. Architectures that lie in the system Pareto front have, by definition, a Pareto distance of 1. In turn, all other dominated architectures have Pareto distance $d > 1$, with architectures further from the Pareto front having the largest values.

Assume now that each dot in Figure 3-4 is an architecture encoded as a binary string of N

positions: $\text{Arch}_i = \{0/1, 0/1, ..., 0/1\}$. This results in a tradespace of 2^N alternatives, all of which are evaluated with respect to performance and cost. Let Arch_i be defined as child of Arch_j if the two following conditions are met:

1. $\text{Arch}_i \oplus \text{Arch}_j = 1$

2. $\sum_{b=1}^{N} \text{Arch}_i = \sum_{b=1}^{N} \text{Arch}_j + 1$

In other words, Arch_i is a child of Arch_j if it only differs by one bit, which is equal to 1 instead of 0. Further, let $\text{S} = \{\text{Arch}_0, \text{Arch}_1, ..., \text{Arch}_N\}$ be a sequence of N architectures that encodes a possible evolution path from Arch_0 to Arch_N, and let $\text{S}[i]$ denote the indexing mechanism that returns the architecture in the i-th position of S. Similarly, let $\boldsymbol{\mathcal{S}}$ denote the set that contains all sequences S to upgrade the system from its basic (Figure 3-1a) to its fully-upgraded implementation (Figure 3-1i). Then, $\boldsymbol{\mathcal{S}}$ has the following characteristics:

1. $\text{S}[0] = \text{Arch}_0 = \{0, 0, 0, ...\}\ \forall \text{S} \in \boldsymbol{\mathcal{S}}$.

2. $\text{S}[i+1]$ is a child of $\text{S}[i]\ \forall \text{S} \in \boldsymbol{\mathcal{S}}$.

3. $\text{S}[N] = \text{Arch}_N = \{1, 1, 1, ...\}\ \forall \text{S} \in \boldsymbol{\mathcal{S}}$.

As a result, obtaining a valid system evolution path is equivalent to finding a path from architecture Arch_0 to Arch_N through a graph $\mathcal{G}(\boldsymbol{\mathcal{S}})$ constructed using edges that represent parent/child relationships and are therefore encoded by consecutive architectures in all sequences $\text{S} \in \boldsymbol{\mathcal{S}}$.

On the other hand, the optimality of any path through $\mathcal{G}$ is directly related to the cost of all architectures visited. Indeed, ideally we would like to optimize the system by always improving the most cost-effective node so that at any point in time we remain as close as possible to the system Pareto front. Since closeness to the Pareto front is measured by Pareto distance (henceforth termed P-distance), it is sensible to define the total P-distance for a path $\text{S} \in \mathcal{G}$ as

$$D(\text{S}) = \sum_{\text{Arch}_i \in \text{S}} D(\text{Arch}_i). \tag{3.1}$$

Similarly, the optimal path to evolve the system from Arch_0 to Arch_N can be found by solving the optimization problem

$$\text{S}^* = \arg\min_{\text{S} \in \mathcal{G}} D(\text{S}) \tag{3.2}$$

where $D(\text{Arch}_i)$ denotes the P-distance for any given architecture. Importantly, observe that $D(\text{S})$ is additive with $D(\text{Arch}_i)\ \forall i$ and these P-distances can be pre-computed once the tradespace in performance and cost has been generated. Consequently, S^* can be efficiently

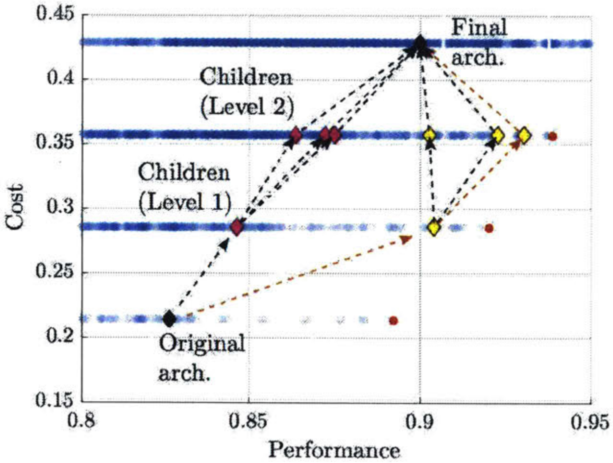


Figure 3-5: Architecture Sequence Graph $\mathcal{G}\,(\mathrm{N},\mathrm{E})$

found using dynamic programming, specifically any shortest path algorithm such as Dijkstra or Bellman-Ford. For the purposes of this thesis the latter is utilized.

To exemplify the problem, Figure 3-5 plots a notional tradespace. Blue dots represent all system architectures $\mathrm{Arch}_i\ \forall i$ evaluated both in performance and cost. Arch_0 and Arch_N are clearly marked using black diamond markers, while red dots represent the Pareto front. Similarly, the graph $\mathcal{G}$ is plotted as a collection of dotted-arrows that represent the transitions between to architectures that exhibit parent/child relationship. The original architecture Arch_0 has two children, a magenta and yellow one, which in turn have three children each. Consequently, $\mathcal{G}$ contains a total of six paths $\boldsymbol{\mathcal{S}} = \{\mathrm{S}_1, \mathrm{S}_2, \mathrm{S}_3, \mathrm{S}_4, \mathrm{S}_5, \mathrm{S}_6\}$. The optimal sequence is therefore S_6 highlighted in orange arrows, as it remains closer to the Pareto front through the entire evolution path. As previously indicated it is obtained using dynamic programming over $\mathcal{G}$ and it minimizes the cumulative P-distance.

3.4.1 Sequence P-Distance Significance

As previously mentioned, a sequence's P-distance $D(\mathrm{S})$ is real valued and strictly positive. However, if that number is 100, is that large or small value? To answer this question, I utilize a statistical approach: 100 will be a small number if, out of all possible sequences in $\mathcal{G}$, a high percentage of them have a P-distance larger than 100. Note that this argument is clearly analogous to that of a statistical test over the population mean. For instance, a 6' man is tall in Indonesia, where the mean height is 5'2.25", and just an average individual in the Netherlands, where the mean height is 6'0.5" [142].

Using this analogy, I define the P-distance significance metric as the probability of finding a path that is closer to the Pareto front:

$$\mathcal{S}_D(\mathrm{S}) = \mathcal{P}\left(D(\mathrm{S}') \leq D(\mathrm{S}) | S, S' \in \mathcal{G}\right) \tag{3.3}$$

Denote by $f_{\mathrm{D}}\left(\mathrm{D}\right)$ the probability density distribution of P-distances over all sequences in $\mathcal{G}$. Then, I estimate the P-distance significance for a reference sequence S as

$$\mathcal{S}_D(\mathrm{S}) = \int_0^{D(S)} f_{\mathrm{D(s)}}\left(\mathrm{D(s)}\right) ds \tag{3.4}$$

with $f_{\mathrm{D}}\left(\mathrm{D}\right)$ computed using a Monte Carlo Sampling approach. In other words, I generate 10^5 random sequences from $\mathcal{G}$, calculate their distance $D(S)$, and finally estimate $f_{\mathrm{D}}\left(\mathrm{D}\right)$ empirically as the relative frequency with which each value occurs. Furthermore, since I know the optimal sequence through dynamic programming, I sample the space of close-to-optimal sequences by generating all rankings that swap only two elements, henceforth termed neighbor sequences. This ensures that the left-tail of $f_{\mathrm{D}}\left(\mathrm{D}\right)$ is properly sampled.

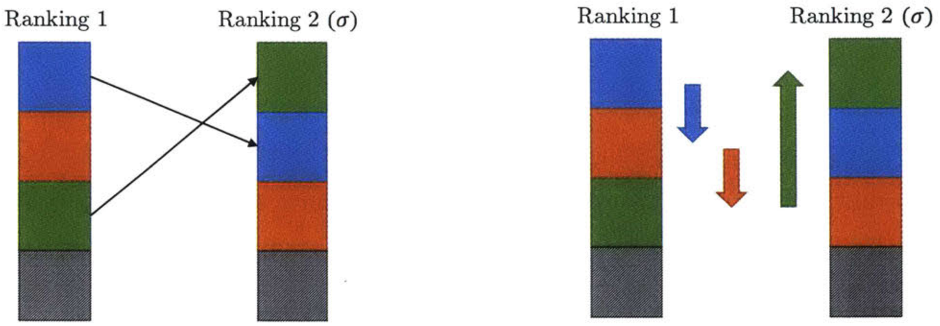
3.5 Test 2. Ranking Similarity

The Statistical Test 2 from Figure 3-3 is based on the concept of ranking similarity measures. Therefore, in this section I provide a succinct review of their definition and properties. Probably the best well-known ranking similarity metrics are Spearman's Footrule [143] and Kendall's Tau [144] criteria, although other measures of ordinal association such as Goodman and Kruskal's γ [145], Somer's D [146] or rank distance [147] have also been defined in the literature. They are all alternative formulations of rank correlation coefficients that measure the degree of similarity between two rankings.

An exhaustive categorization and review of all similarity metrics present in the literature is clearly beyond the scope of this thesis. Instead, I restrict myself to the most classic measures of ranking similarity, namely Kendall's Tau and Spearman's Footrule, which I will utilize to derive the Statistical Test 2. Note that, by virtue of the equivalence theorem from Diaconis [148] and the experimental results from Kumar [6], measuring ranking similarity through the Spearman's Footrule and the Kendall's Tau metrics generally yields equivalent results.

Kendall's Tau similarity metric is probably the most intuitive way to compare two rankings. In its simplest form, it measures the total number of inversions between any two elements i and j in the ranking:

$$K\left(\sigma\right) = \sum_{(i,j):i>j} \sigma(i) < \sigma(j) \tag{3.5}$$



(a) Kendall Tau Similarity

(b) Spearman's Footrule Similarity

Figure 3-6: Ranking Similarity Metrics (Adapted from Reference [6])

An inversion is said to occur when $i > j$ and $\sigma(i) < \sigma(j)$, where $\sigma(\cdot)$ is a function that, given the position of element i in ranking 1, returns its position in ranking 2 (for instance, in Figure 3-6a the inversion between the blue and green squares is characterized by $i = 1$, $j = 3$, $\sigma(i) = 2$ and $\sigma(j) = 1$). Note that $\sigma(\cdot)$ effectively transforms the unitary ranking $1, 2, 3, 4, 5, ..$ into another one, since Ranking 1 is arbitrary and can be set to any value. Therefore, Kendall's Tau similarity metric (as well as any other metric) is only a function of σ (instead of σ_1 and σ_2). Finally, note also that the similarity metric is a real value ≥ 0, with the equality satisfied if the two rankings are exactly the same.

On the other hand, the basic definition of the Spearman's Footrule similarity metric counts the total displacement of elements to transform Ranking 1 into Ranking 2:

$$F(\sigma) = \sum_{i} |i - \sigma(i)| \tag{3.6}$$

For instance, in the example from Figure 3-6b, the total displacement between Rankings 1 and 2 is 4 units: The displacement of the blue square (1 unit), plus the displacement of the orange square (1 unit), plus the displacement of the green square (2 units).

The main problem of both Kendall's Tau and Spearman's Footrule similarity metrics, as defined through Equations (3.5) and (3.6), is that all elements of the ranking have the same relative importance. In other words, they contribute to *one* unit of inversion or *one* unit of displacement. In Step 2-3, I argued that not only is it necessary to consider the order of latency contributors, but also their relative importance. Indeed, if one latency contributor induces 90% of the latency, it should definitely be addressed first. In contrast, if the first two contributors have a similar relative importance, then fixing them in inversed order results in a relatively small sub-optimal evolution path for the system.

To account for the notion of "relative importance", I utilize the generalized version of Spearman's Footrule similarity metric introduced by Kumar in Reference [6]. In his work, Kumar defines three common features that similarity metrics should address when applied to real-life rankings:

- **Element weights**: Each element of the ranking has an associated weight that indicates its relative importance with respect to the others. Two rankings that place highly important elements out of order should, therefore, have a higher similarity score.

- **Position weights**: Each element has a weight that is defined by its relative position within the ranking. Therefore, swapping two elements at the beginning of the ranking should be more penalized than swapping two elements at the end of the sequence.

- **Element similarities**: Any two elements i, j in the sequence have a scalar value D_{ij} that quantifies their similarity. Swapping two perfectly similar elements ($D_{ij} = 0$) does not incur in any penalty from the ranking similarity metric perspective.

Based on these three features, Kumar defines the generalized Spearman's Footrule criteria as

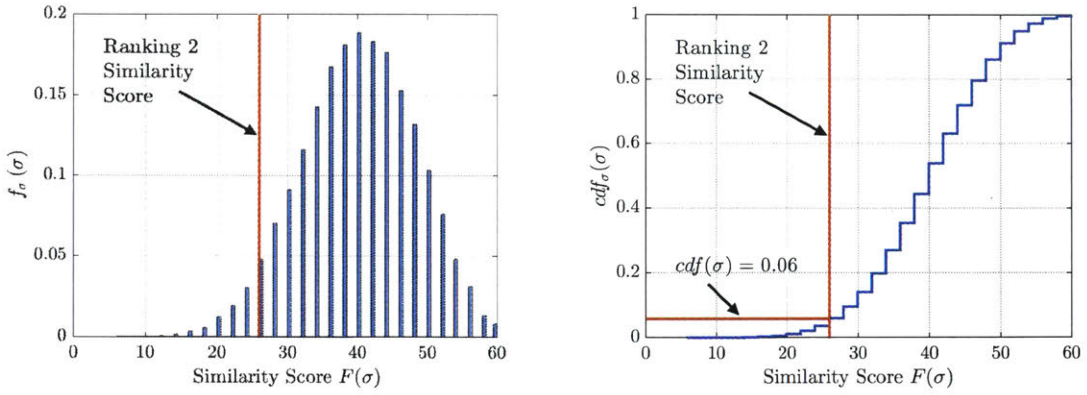
$$F_{w,p,D}(\sigma) = \sum_i w_i p_i(\sigma) \left| \sum_{j:j\leq i} w_j p_j(\sigma) D_{ij} - \sum_{j:\sigma(j)\leq\sigma(i)} w_j p_j(\sigma) D_{ij} \right|, \tag{3.7}$$

with w_i denoting the relative weight of element i, $p_i(\sigma)$ equal to the positional weight of element i given ranking σ, and D_{ij} equal to the similarity between elements i and j.

To utilize Equation (3.7) in a practical setting, it is first necessary to define w_i, $p_i(\sigma)$ and D_{ij} $\forall i, j$. In the context of this thesis there is no rationale for assigning the positional weights, so I will assume that $p_i = 1$ $\forall i$. Furthermore, I will let w_i be equal to the utility loss as measured using the high-fidelity simulator and $D_{i,j} = |w_i - w_j|$. Importantly, note that $D_{i,j}$ is specifically selected so that $D_{i,j} = 0$ if $w_i = w_j$. This ensures that two latency contributors with the same relative importance can be optimized in any order without penalty in the similarity score. In other words, since there is no evidence that one is more important than the other, they can addressed in any order.

3.5.1 Ranking Similarity Significance

The ranking similarity significance is defined analogously to the sequence P-distance significance. Let Θ denote the set of all possible rankings of length $|\sigma|$. Furthermore, let the ranking similarity significance $\mathcal{S}_F(F, \sigma)$ be defined as the probability of finding a more similar ranking, i.e. a ranking with lower similarity score. Then, if $f_{\mathcal{F}}(F)$ denotes the probability distribution function of similarity scores for rankings in Θ, I compute the ranking



(a) Probability Distribution Function

(b) Cumulative Distribution Function

Figure 3-7: Experimental Similarity Score Sampling Distribution

similarity significance as

$$\mathcal{S}_F(F,\sigma) = \mathcal{P}\left(F(\sigma' \in \Theta) \leq F(\sigma)\right) = \int\limits_0^{F(\sigma)} f_{\mathrm{F(s)}}\left(\mathrm{F(s)}\right) ds. \tag{3.8}$$

In simple cases such as the basic Spearman's Footrule similarity metric (Equation (3.6)), $f_{\mathrm{F}}(\mathrm{F})$ can be found analytically either directly or by properly normalizing it into the Spearman's rank correlation coefficient [149]. Unfortunately, the same result cannot be obtained for the generalized version of the Spearman's Footrule similarity metric. To overcome this limitation, I compute $f_{\mathrm{F}}(\mathrm{F})$ using a once again Monte Carlo Sampling. In that sense, I first generate 10^5 random permutations of N elements, as well as neighbors to the optimal sequence, and calculate their similarity score. Then, I estimate $f_{\mathrm{F}}(\mathrm{F})$ empirically by calculating the relative frequency with which each score occurs, and I finally compute $\mathcal{S}_F(F,\sigma)$ using Equation (3.8).

To exemplify the procedure, consider the following two rankings and assume that all elements have the same weight.

$$\text{Ranking 1: } \sigma_1 = \{0,1,2,3,4,5,6,7,8,9,10\}$$
$$\text{Ranking 2: } \sigma_2 = \{2,0,7,1,4,8,3,10,5,6,9\}$$

Since they have 11 elements, Θ has cardinality $|\Theta| = 11! = 39916800$. Figure 3-7 presents $f_{\mathrm{F}}(\mathrm{F})$ estimated using 10^5 of them, i.e. only 0.25% of $|\Theta|$. Observe that even for this small fraction, the Monte Carlo Sampling approach has already converged to a normal distribution. Observe, however, that unlike traditional statistical tests, this normal is not unbounded but rather has a minimum, $F(\sigma)_{min} = 0$, and a maximum $F(\sigma)_{max} \geq 60$. Furthermore, since the Monte Carlo approach cannot guarantee that the worse possible rankings have been found, only an approximation for $\mathcal{S}_F(F, \sigma)$ can be obtained. Finally, using Equation (3.8), I estimate $\mathcal{S}_F(F, \sigma_2)$ to be $\leq 6\%$, i.e. out of all possible rankings, only 6% are more similar to σ_1 than σ_2.

3.6 Baseline Scenario

In this section I start the validation process by tackling goal 1 of the validation strategy from Section 3.2. As previously mentioned, this defines a baseline scenario based on a pre-specified set of assumptions. Later, in Sections 3.7-3.10, I revise the validity of the centrality measure by modifying the baseline scenario and progressively eliminating these initial assumptions.

3.6.1 Network and System Description

The baseline network configuration assumed for this case study is based on a WAN across the US adapted from Reference [150] (see Figure 3-8). Its topology is representative of the backbone infrastructure of a major US carrier such as AT&T, Comcast or Verizon, it is completely arbitrary, and has been chosen only because it facilitates the discussion that follows. Other basic information about the system includes:

- Traffic in the network is measured in units of packets. For simplicity we assume that the size of a packet is 1500 bytes, the typical average IP packet size.
- All nodes in the network generate the same amount of traffic based on a Poisson process at a rate of $\lambda = 10\frac{packets}{s}$. The destination of each packet is chosen randomly among all other candidates so that on average all nodes send and receive the same amount of information.
- Data is routed through the network based on a shortest path algorithm, with all links having the same weight. In other words, data is routed so as to minimize the number of hops to reach destination.
- The network is operating at a stable stationary point for at least $T = 50$ seconds, so that the total number of packets sent per simulation is approximately 7000 (14 nodes generate $10\frac{packets}{s}$ simultaneously).
- Changing the performance of a node does not significantly affect the performance of other nodes in the system. Note that this is not always the case in communication networks. For instance, consider two M/M/1 queues in tandem with a feedback loop that models the re-transmission mechanisms for packets received with errors. Assume that we can improve the network by reducing the probability of error in the queues, one at a time. Then, if the we improve the 2^{nd} node we reduce the end-to-end error probability, which in turn reduces the amount of information that has to be resent and

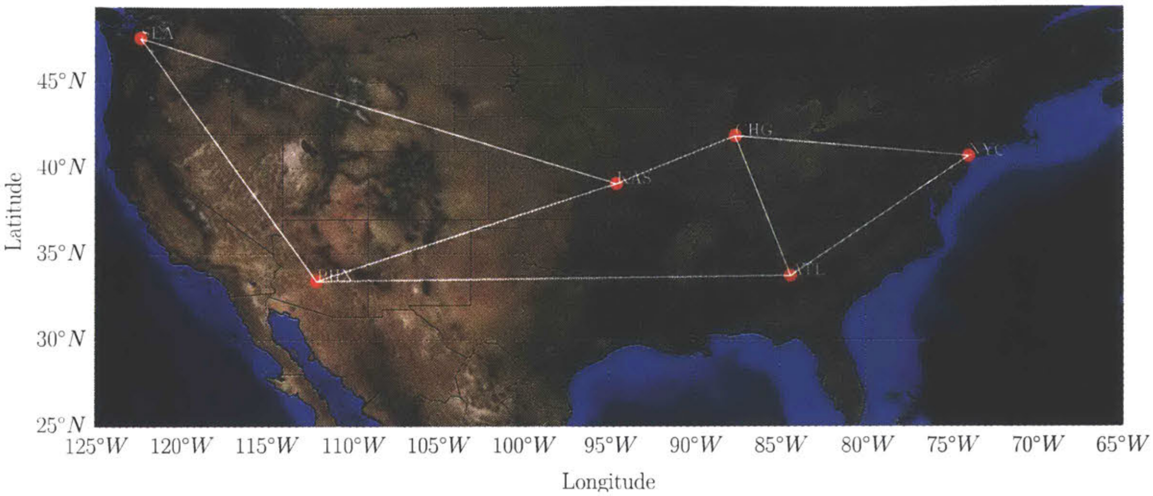

SEA
KAS
CHG
NYC
PHX
ATL
45°N
40°N
35°N
30°N
25°N
Latitude
125°W 120°W 115°W 110°W 105°W 100°W 95°W 90°W 85°W 80°W 75°W 70°W 65°W
Longitude

Figure 3-8: Baseline WAN Network Topology

therefore decreases the total rate of packets through the first queue. This, in turn, changes the operation point of the first M/M/1 queue that will now introduce less delay since the expected queue length will be smaller.

3.6.2 Specify the Centrality Measure

This section replicates Step 2-2 of the latency-centric approach to architecting space communication networks. It describes the different steps required to specify the network centrality measure before its application to guide the system architecting process.

Characterization of Latency Contributors

Two primary latency contributors are assumed to drive the delay with which packets are delivered in this network. On the one hand, nodes introduce latency due to limited processing capabilities. As packets arrive, they are queued until the router can read their final destination address, compute the next hop and deliver them to the appropriate outgoing connection. Two implementations for a node are available, a basic inexpensive and a high performance expensive router. The former is assumed to introduce a latency of 100msec, while the latter introduces a latency of 20msec.

On the other hand, connections introduce latency primarily due to the packet transition time. Once again, two implementations are available. The basic inexpensive is equivalent to a DS0 and therefore has a capacity of 64 kbps approximately. Given that an IP packet has 1500 bytes of information, this results in a transmission time of 187.5msec. Alternatively, the high performance expensive connection implements a DS1 line with 1.55Mbps of capacity, thus resulting in only 20msec of delay per packet transmission.

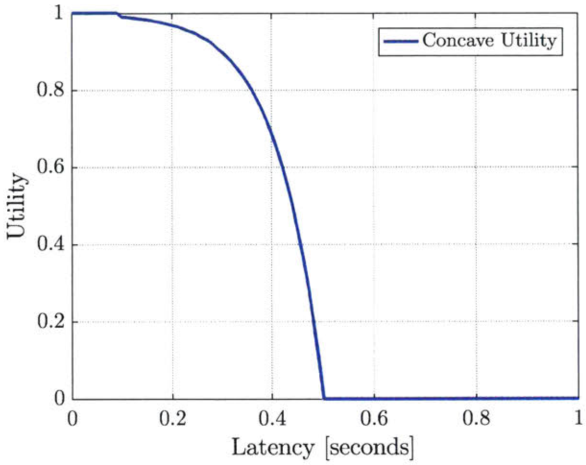


Figure 3-9: Utility Function for Baseline Scenario

Notice that in real WAN networks the user experience can be affected by other factors not included in this simplified case study. For instance, TCP connections are known to have low performance in high delay, high error environments. Since latency includes all contributors that affect the delivery of data to the final user, TCP should potentially be included in a realistic study. Similarly, packets delays caused by network congestion, intra-autonomous system data flow, or last mile connectivity problems should also be considered in a real scenario.

Identification and Characterization of Data Flows

As previously mentioned, in the baseline scenario all nodes generate packets at a constant rate. Their destination is selected at random among all nodes in the network. Therefore, all data flows are equally likely and carry the same information. To capture this fact, in this scenario I assume that the centrality measure is computed using $w_p = 1 \; \forall p \in P$, where P is the set of all geodesic paths between any origin and destination in the system.

Characterization of Data Utility

Figure 3-9 plots the concave utility function that has been assumed for packets sent through the system. Observe that any data delivered with more than half a second of delay is assumed to deliver no utility to the end user. In contrast, any packets delivered with less than one tenth of a second result in full utility.

Definition of a normalization scheme

For this case study, all computations regarding sequences and centrality measures will utilize sum normalization as defined in Step 2-2.

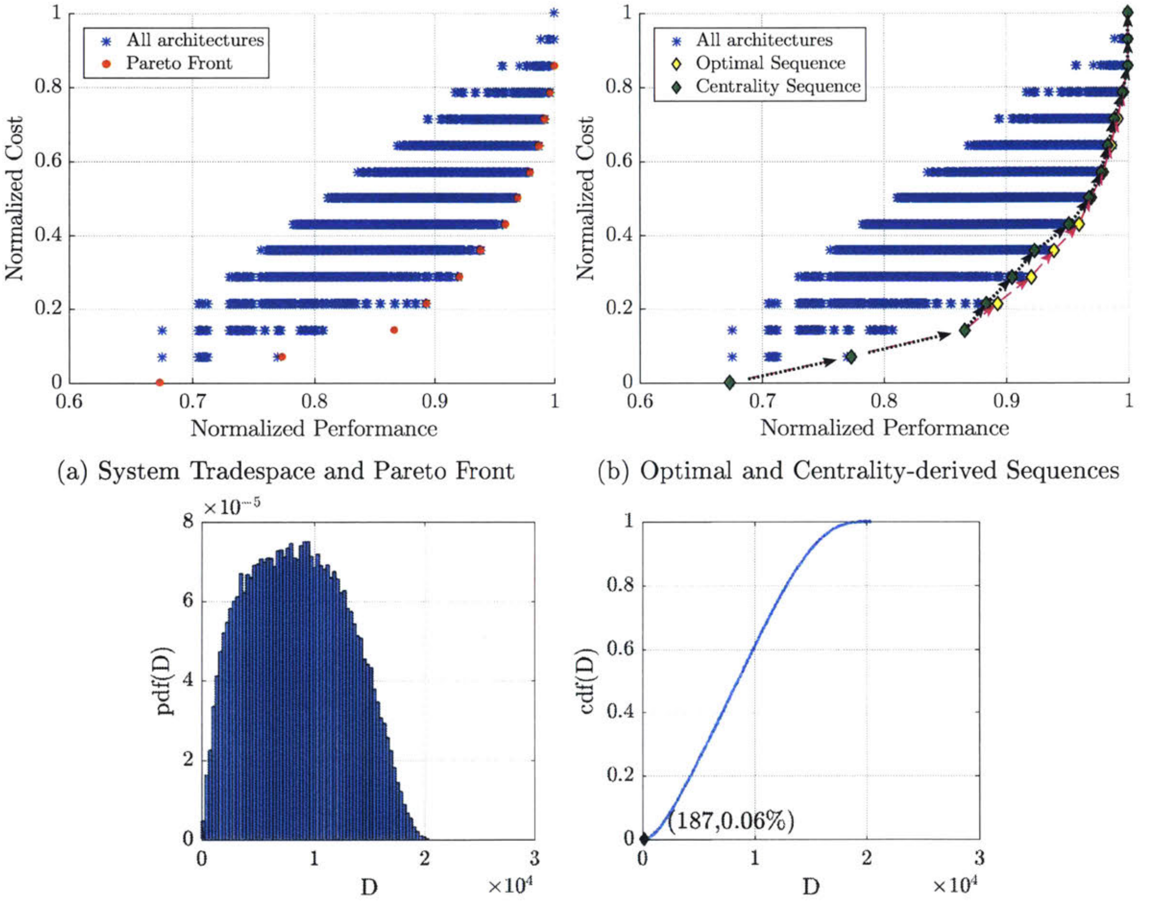
3.6.3 Test 1. Pareto Distance

Figure 3-10a presents the tradespace of all 2^{14} architectures possible in the baseline scenario along with the estimated Pareto front. Observe the stratification in the cost space as I have assumed cost-homogeneity across all nodes and connections in the system. In other words, each jump in the cost metric corresponds to upgrading one element in the network regardless of whether it is a connection or node. On the other hand, Figure 3-10b provides a visual representation of two paths that upgrade the system from its baseline implementation to a fully upgraded alternative. The optimal path, which maps onto an optimal sequence of nodes, is depicted using yellow markers and is computed using dynamic programming as defined in Section 3.4. In contrast, the centrality-derived sequence is plotted using green markers and black arrows. Observe that they are not exactly equal, albeit the resemblance in the metrics space is notorious.

To quantify this resemblance, Figure 3-10c plots the probability density function and the cumulative distribution function of the P-distance for sequences in this tradespace. The sequence computed with the centrality measure has a total P-distance of 187 units, and has a significance of 0.065%. In other words, there is only a 0.065% probability of finding a sequence better than the one obtained by the centrality measure.

3.6.4 Test 2. Ranking Similarity

Figure 3-11 plots the result of the ranking similarity test for the baseline scenario. To construct it, I first compute the optimal sequence of nodes to upgrade in this network using the method described in Section 3.4 and set is as the reference against which all other possible sequences will be benchmarked. It is represented using the top bar plot from Figure 3-11a, where the height of the bar indicates that total utility loss attributable to a given node or connection (normalized to 1) and is estimated by calculating the difference in system performance over the system with and without that part upgraded. Then, I calculate the centrality-based sequence and apply the weighted version of the Footrule similarity metric using the weights derived from the reference sequence. The result is shown in the bottom plot from Figure 3-11a. Each bar's height indicates, once again, the relative importance of a given node or connection with respect to the system utility loss, but this time the weight is computed using only the centrality measure. Additionally, each bar is color-coded from red to green using a linear function that transforms the contribution of each node/connection in the

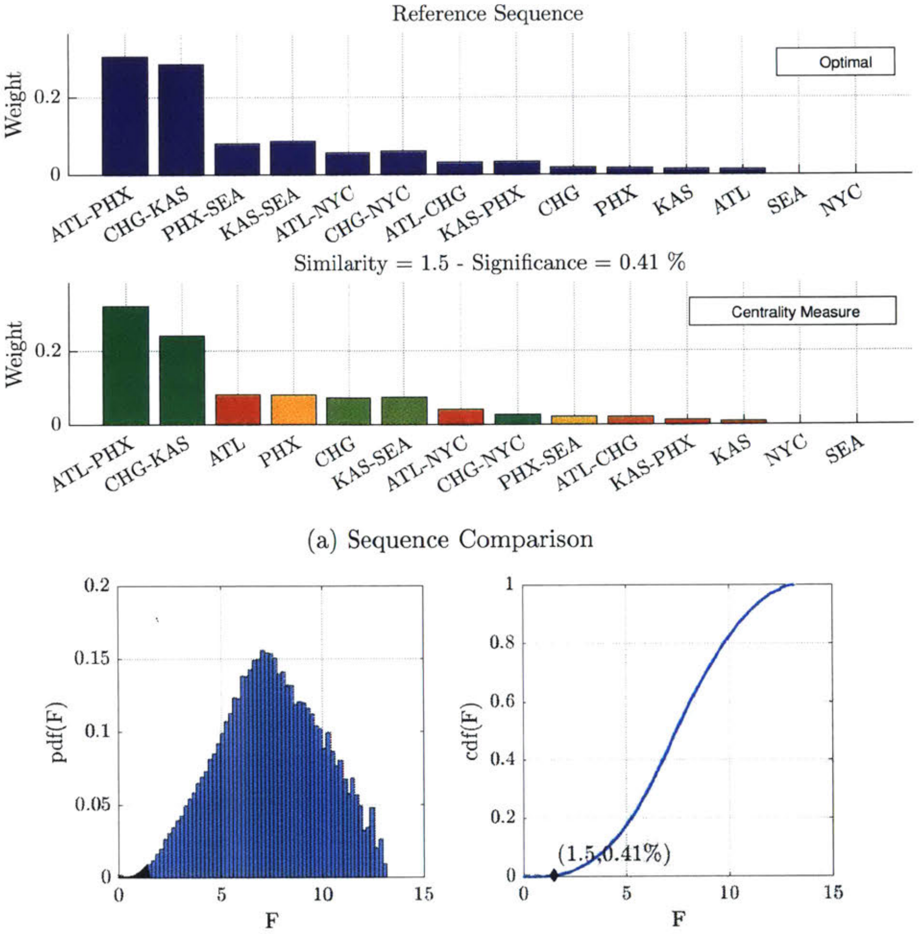


(a) System Tradespace and Pareto Front

(b) Optimal and Centrality-derived Sequences

(c) P-Distance Test

Figure 3-10: P-Distance Test Results for Baseline Scenario



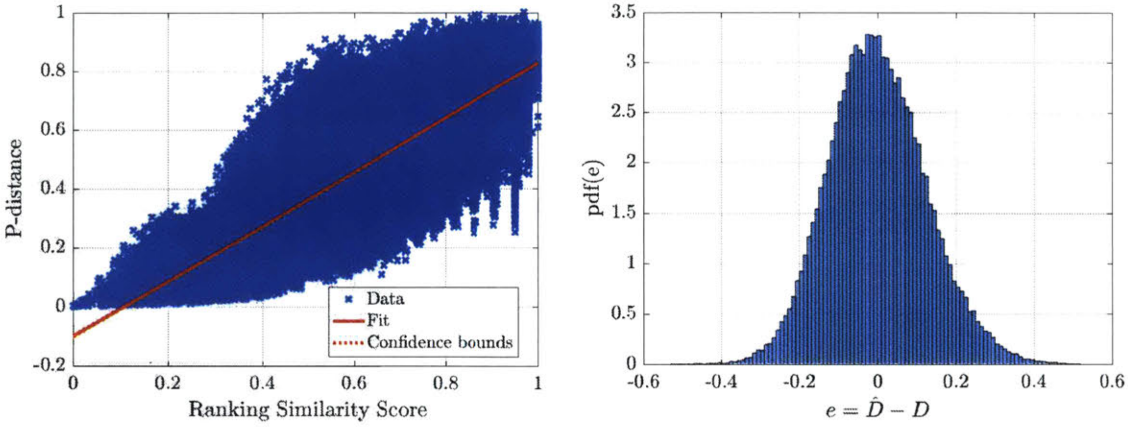
(a) Sequence Comparison

(b) Ranking Similarity Test

Figure 3-11: Ranking Similarity Test Results for Baseline Scenario

similarity score to a color. In other words, green bars are used to indicate nodes/connections in the correct order, while red bars flag items that largely contribute to the total similarity score.

On the other hand, Figure 3-11b shows the result of the ranking similarity test. In this baseline scenario, we observe that $\mathcal{S}_F(F, \sigma) = 0.41\%$, i.e. there a 0.41% probability of finding a sequence of nodes that is closer to the optimal ordering computed with dynamic programming. Therefore, I conclude that the centrality measure, as defined in Chapter 2, provides a computationally efficient way to approximate the Pareto front and sequence of nodes to update for the baseline scenario.



(a) P-distance vs. Ranking Similarity Significance

(b) Model Errors

Figure 3-12: Linear Model of P-distance vs. Ranking Similarity Significance

3.6.5 P-distance Significance vs. Ranking Similarity Significance

Once the results for the statistical tests in the metrics (P-distance) and architectural (ranking similarity) space have been completed, is worth taking some time to study the relationship between them. After all, they are both used for a common objective: Prove that the centrality measure can successfully identify the nodes/connections to upgrade in a network with multiple latency contributors. To assess this relationship, I compute the ranking similarity score and P-distance for 10^5 sequences (including the optimal sequence and its neighbors[2]). Then, I regress the P-distance scores with the ranking similarity scores[3]:

$$D(S) = \alpha + \beta F(S) \tag{3.9}$$

Figure 3-12 plots the obtained results. The same information, with the estimates for α and β are reported in Table 3.1, where the numbers in parentheses indicate the t-statistics assuming that all errors are normally distributed (see Figure 3-12b). Observe that both t-statistics are clearly beyond the 1.96 threshold required for a 95% confidence interval, and therefore we can conclude that both metrics are correlated.

To better understand this relationship, let us compute the correlation coefficient between the P-distance and ranking similarity scores. Results indicate that this coefficient is equal to 0.82, thus indicating a large degree co-movement between the two metrics. This would suggest that both test are redundant. However, a close examination to the regression coefficient of determination, or R^2, indicates that this is not the case. In fact, only 67.2% of

[2] Recall here that the neighbor of a sequence is defined as any other sequence that only swaps the position of two elements.

[3] Both variables are normalized to the $[0, 1]$ range before the regression.

Table 3.1: P-distance vs. Ranking Similarity Significance

	P-distance
Ranking Similarity Score	0.93 (453.19)
Constant	-0.10 (-82.96)
Observations	100091

the P-distance variability can be explained by the ranking similarity metric variability. The other 32.8% comes from the weighting scheme in the ranking similarity metric, which is not captured in the P-distance significance. Therefore, I conclude that, while both metrics are certainly correlated, they are also complimentary. The P-distance metric has the advantage of being directly interpretable in the metrics space, making it easy to understand and intuitive to demonstrate optimality. In contrast, the ranking similarity metric is more difficult to interpret, but captures the extra factor of relative weighting between different elements in the sequence.

3.7 Stress Case 1: Utility Function

In this stress case I violate the first assumption of the baseline scenario, a smooth concave utility function $U_b(L)$, and replace it for a steeper convex exponential function $U_1(L)$:

$$U_b(L) = \begin{cases} U(L) = 1 & L < L_{min} \\ U(L) = 0 & L < L_{max} \\ U(L) = 1 - e^{11.51L-5.76} & \text{otherwise} \end{cases} \tag{3.10}$$

$$U_1(L) = \begin{cases} U(L) = 1 & L < L_{min} \\ U(L) = 0 & L < L_{max} \\ U(L) = e^{-25.01L+2.50} & \text{otherwise} \end{cases} \tag{3.11}$$

with L_{min} = 100msec and L_{min} = 500msec (see Figure 3-13). Note the similarity between this new utility function and the MOE from Figure 1-11. Indeed, this stress case is representative of a WAN where data is mostly delivered through geodesic paths (e.g. Internet's Autonomous System) and video streaming is the primary service being provided, both in terms of data volume and user demand.

3.7.1 Test 1. Pareto Distance

Figure 3-14 presents the results of running the network optimization process after replacing $U_b(\cdot)$ by $U_1(\cdot)$. Significant differences are observed in the shape of the resulting tradespace.

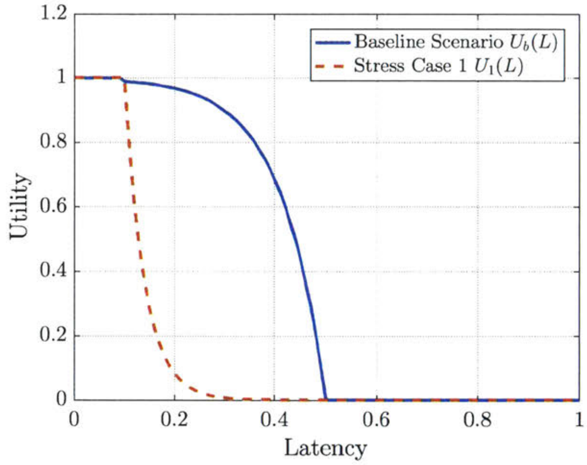
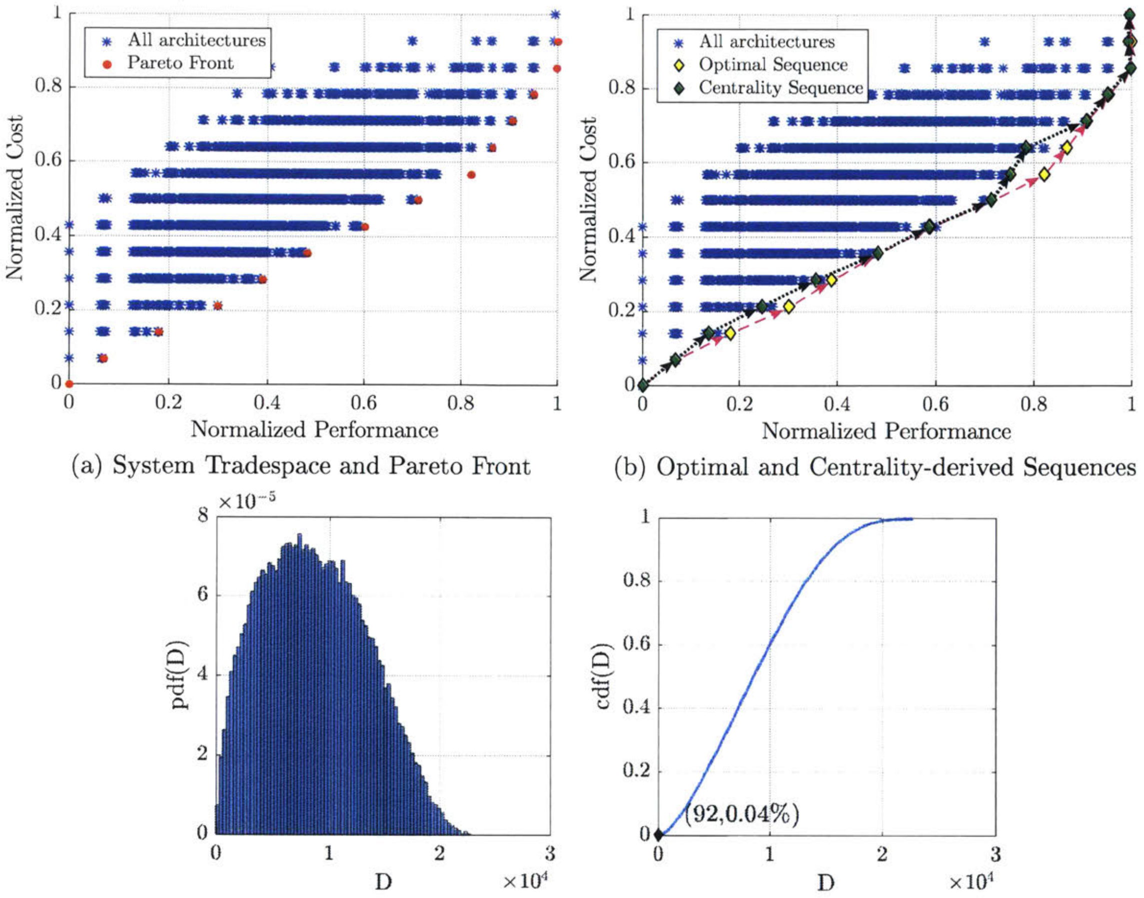


Figure 3-13: Utility Function for Stress Case 1

Most notably, observe that for a performance ≤ 0.8 approximately the Pareto front is linear. This makes sense since $U_1(\cdot)$ is approximately linear for latencies ≥ 0.2sec. In contrast, for really high performing systems ($L < 0.2$) the Pareto front is highly non-linear and eventually results in iso-performance implementations since there is no added benefit in delivering packets with latency less than 100msec. On the other hand, Figure 3-14b provides a visual comparison between the optimization paths obtained by minimizing the total P-distance and the centrality measure. Note that, similar to the baseline scenario, the centrality measure based approach is capable of finding a sequence of nodes to optimize that closely resembles the optimal sequence. In fact, when performing the P-distance test I estimate the centrality-derived sequence to have a significance of 0.04%, i.e. for all practical purposes its is indistinguishable from the optimal one.

3.7.2 Test 2. Ranking Similarity

Similar results are obtained for the ranking similarity test. In this case, the centrality-based sequence only places one node largely out of sequence, Chicago (see Figure 3-15a), causing the heuristic path to clearly separate from the optimal path (see Figure 3-14b). That being said, the results of the ranking similarity tests are also encouraging, with a significance of 0.05%. This is particularly interesting because (1) it once again vindicates that the utility function's concavity does not affect the effectiveness of the proposed centrality measure, and (2) it demonstrates that replacing the risk-neutral probability measure $\mathbb{Q}$ by the real world probability measure $\mathbb{P}$ does not yield significantly different rankings when operating over largely non-linear utility functions.



(a) System Tradespace and Pareto Front

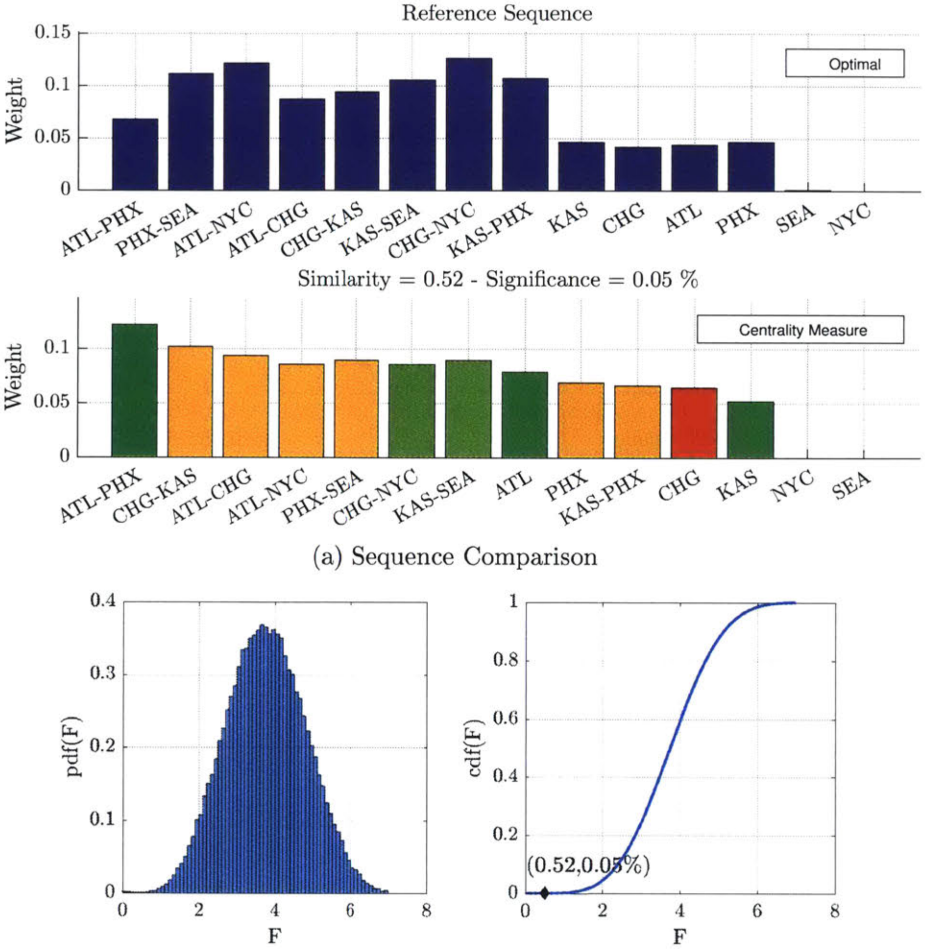
(b) Optimal and Centrality-derived Sequences

(c) P-Distance Test

Figure 3-14: P-Distance Test Results for Stress Case 1

3.8 Stress Case 2: Data Routing

In this section I investigate the effect of misrepresenting the routing strategy when applying the centrality measure to guide the system architecting approach. In general, there are two extreme approaches to routing data through a network: On the one hand, data can be directed using shortest paths over a predefined set of edges and associated weights. In the centrality measure literature, it is common to assume that all weights are equal to 1, thus transforming this criteria into one that minimizes the number of hops through the network. On the other hand, centrality measures are also interested in studying networks where data is routed following *random walks*. They are generated by letting each node select the next hop for a received packet with equal probability among all its neighbors. Note that, in real packet networks neither of these two approaches is actually implemented. Packets tend to follow the shortest path between origin and destination, but they can be diverted due to multiple reasons such as link failures, link overloads in parts of the network or load



(a) Sequence Comparison

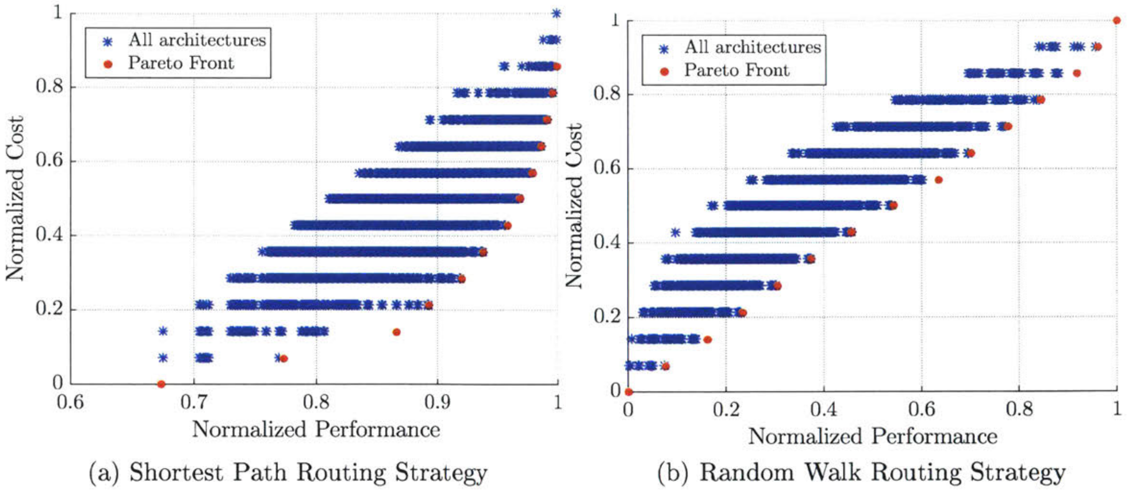
(b) Ranking Similarity Test

Figure 3-15: Ranking Similarity Test Results for Stress Case 1

sharing between multiple autonomous systems (AS). Similarly, wireless ad-hoc networks utilize clever routing strategies based on smart flooding algorithms to ensure that data is delivered to the final user despite unreliable and intermittent links between users (see, for instance, Reference [151]).

In order to model these two "extreme" routing approaches, as well as intermediate network routing strategies such as the ones encountered in real life, I let all nodes in the network utilize the following routing strategy: Choose the next hop based on a shortest path[4] approach with probability p, otherwise select the next hop randomly with probability $1 - p$. Figure 3-16 plots the tradepsace of network architectures when data is routed using a path-based ($p = 1$) and walk-based ($p = 0.33$) strategy. Observe the significant differences in the shape of the tradespace and Pareto front. Indeed, the Pareto front for the path-based routing strategy is highly non-linear due to the fact that certain nodes/connections in the

[4] If more than one shortest path is available, select one of them randomly.



(a) Shortest Path Routing Strategy

(b) Random Walk Routing Strategy

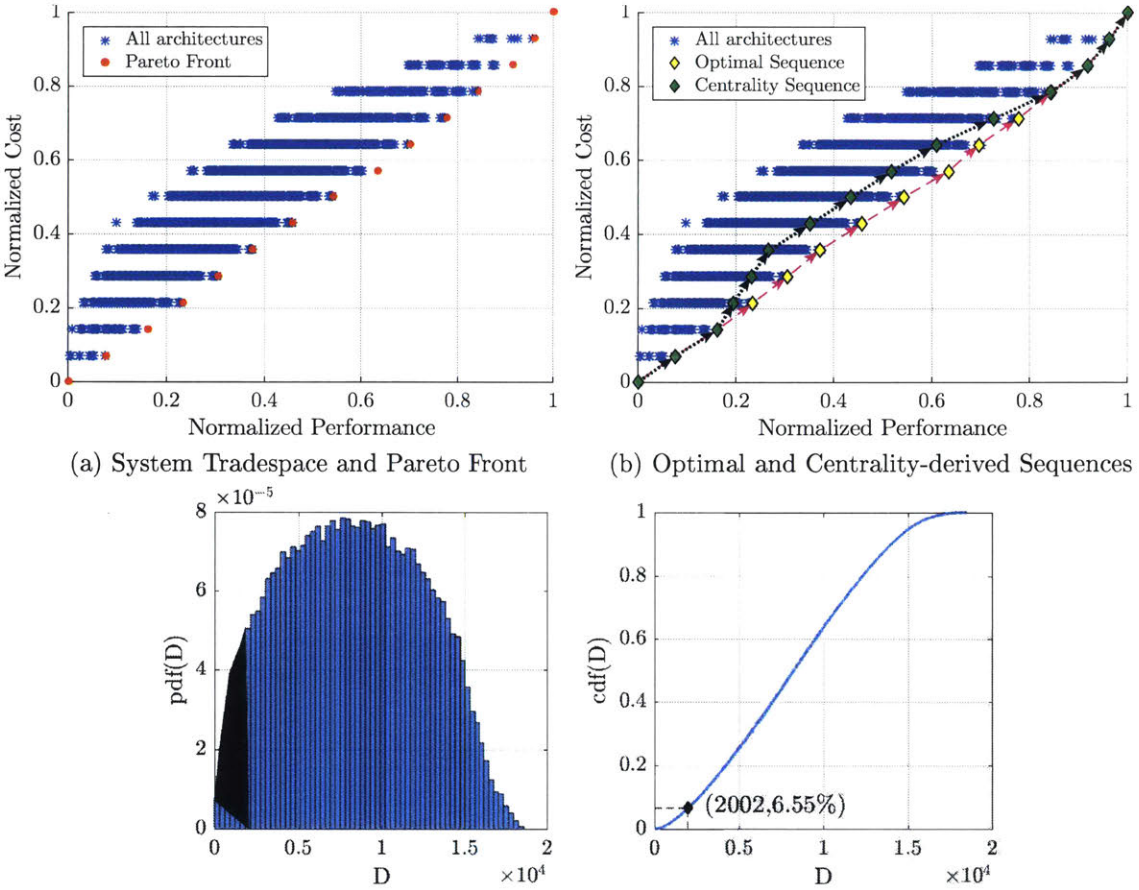
Figure 3-16: Tradespace Shape as a Function the Routing Strategy

network (those that move information from the West to the East coast and vice versa) are more important than others. Specifically, we can infer from the baseline scenario ranking (see Figure 3-11a) that the max-flow-min-cut of the network cuts connections KAS-CHG and PHX-ATL and results in two separate sub-networks: The East coast sub-network, composed of CHG, ATL and NYC; and the West coast sub-network, composed of SEA, PHX and KAS. Therefore, improving the two connections in the max-flow-min-cut results in large performance improvements, while improving nodes SEA or NYC results in marginal latency reduction.

The same reasoning is not valid when packets move through the random walk routing strategy. In this case, there is little evidence that a given node is more important than another one since packets have the same probability of being routed through any of them. This produces a linear Pareto front, where the improvement in performance per node/connection upgrade is approximately constant. While, at this point, this finding is anecdotal, it aligns with a central argument from Borgatti's work: Centrality measures, and their application to understanding the structure of a network, are dependent on the type of information that is being transmitted and how it moves through the network [99]. Consequently, this fact vindicates my original definition of the centrality measure as the sum of utility loss over all paths through a system node, without necessarily specifying how these paths are constructed. Indeed, these will have to be determined on a case-by-case basis.

3.8.1 Test 1. Pareto Distance

Figure 3-17 plots the results of the running the P-distance test on the network assuming that data moves based on random walks but without modifying the system architecting centrality



(a) System Tradespace and Pareto Front

(b) Optimal and Centrality-derived Sequences

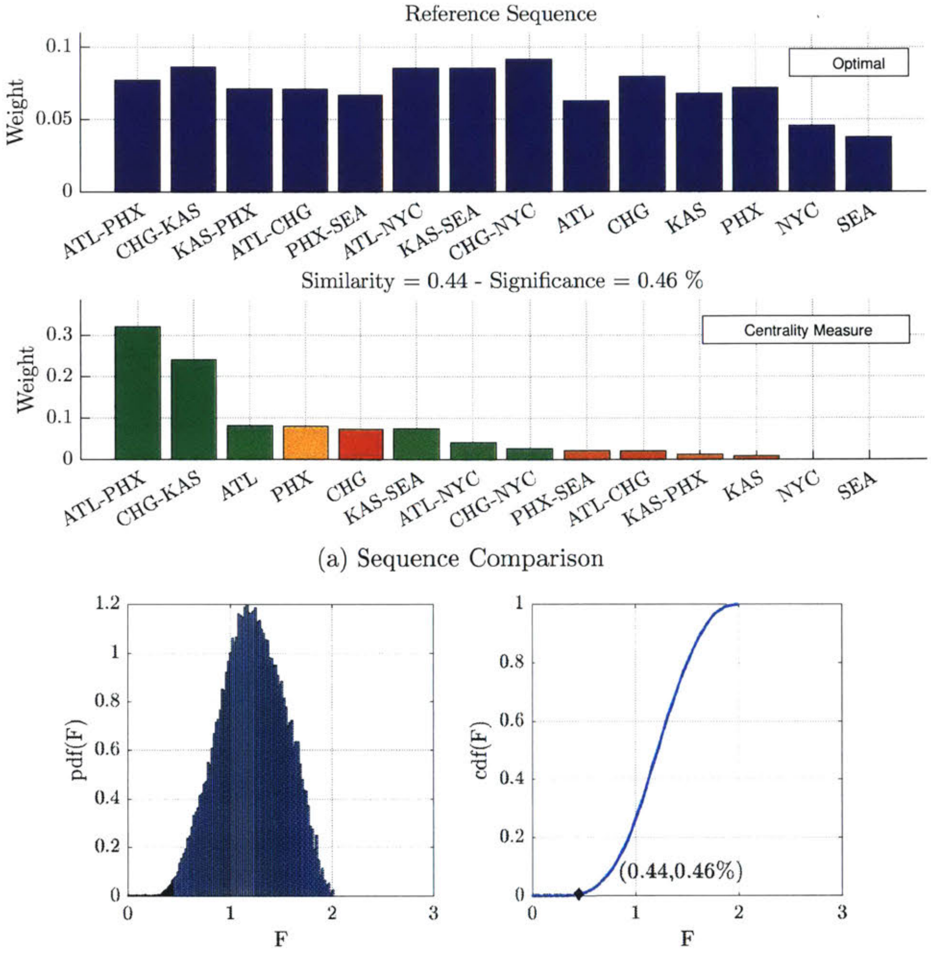
(c) P-Distance Test

Figure 3-17: P-Distance Test Results for Stress Case 2

measure accordingly. Observe that the difference between the optimal and estimated paths is highly significant, with more than 6.55% of sequences providing a better system evolution path. While this number might seem low, recall here that a well calibrated centrality measure that properly reflects how data is routed obtains a sequence with a P-distance similarity score of less than 0.1%. In other words, misrepresenting the routing strategy results, in this case, in a 6450% increase on P-distance significance.

3.8.2 Test 2. Ranking Similarity

Similar results are reported by the ranking centrality tests. Once again, the obtained similarity significance is higher than in the baseline scenario, with a relative increase of 10% approximately. More importantly, note the apparent difference in element importance between the two sequences from Figure 3-18a. While the reference sequence assigns an approximately constant utility loss contribution to all nodes (with the exception of NYC and SEA), the centrality-based approached completely misrepresents the relative importance of each node.



(a) Sequence Comparison

(b) Ranking Similarity Test

Figure 3-18: Ranking Similarity Test Results for Stress Case 2

As a result, connections ATL-CHG and CHG-KAS are attributed the majority of utility loss. This clearly leads to suboptimal decision-making from the perspective of the system architect, as well as the potential for not meeting expectations at the end of the architecting exercise (i.e. after upgrading the ALT-PHX connection the return on investment will be lower than expected).

3.9 Stress Case 3: Data Importance

A fundamental assumption of the baseline scenario is that all flows in the network are homogeneous. Let $P_{s,d}$ denote the total number of packets exchanged between origin s and destination d, and let $u_p(L_p)$ denote a normalized utility per packet as a function of the latency with which it is delivered. Then, the total utility in the flow between these two

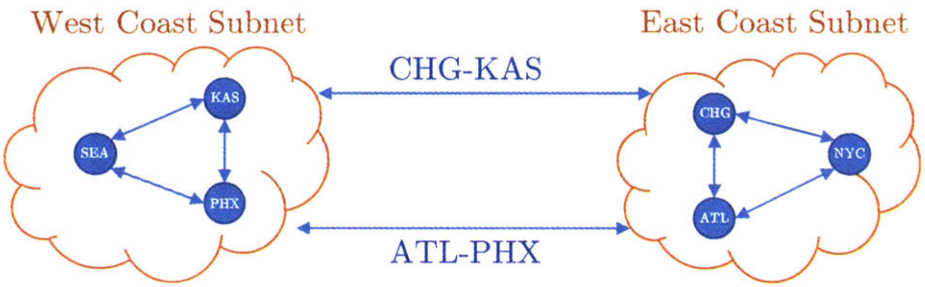


Figure 3-19: Network Division

nodes can be simply computed as

$$U_{s,d} = \sum_{p=1}^{P_{s,d}} u_{s,d}(L_p). \tag{3.12}$$

All flows in the network are said to be homogeneous if $U_{s,d} = U_{s',d'}\ \forall \{s, s', d, d'\}\quad s \neq s\ \ d \neq d'$. Importantly, observe that two flows can have the same total utility even if the amount of data sent is orders of magnitude different. Indeed, if $P_{s,d} \gg P_{s',d'}$, then we can construct a homogeneous flow by letting $u_{s,d}(L_p) \ll u_{s,d}(L_p)$. As mentioned in Chapter 2, a person that has no preference between watching a video or listening to music can generate two data streams with very different bandwidth profiles and yet have equal utility. This indicates that the unitary utility per packet in the video stream is significantly lower than that of the audio stream.

To accentuate the impact of heterogeneous data flows through the network, in this stress case I let the unitary utility per packet be constant and I vary the rate at which packets are sent depending on the origin-destination pairs. Since the network is always simulated over a fixed time horizon of 50 units of time, this results in some flows sending a larger number of packets than others and consequently generates heterogeneous data streams through the network. In particular, I divide the original network from Figure 3-8 into two sub-networks, the West Coast sub-network and the East Coast sub-network (see Figure 3-19). They are connected by the max-flow min-cut connections identified in the baseline scenario, i.e. ATL-PHX and CHG-KAS. At simulation time, I impose the following set of rules in the packet generation process:

- Nodes from the West coast send packets to other nodes on the West coast at a rate of $\lambda_{ww} = 10\frac{packet}{sec}$.
- Nodes from the West coast send packets to other nodes on the East coast at a rate of $\lambda_{we} = 1\frac{packet}{sec}$.
- Nodes from the East coast send packets to other nodes on the West coast at a rate of

Table 3.2: Relative Packet Rate Between Source-Destination Pairs

	ATL	CHG	NYC	KAS	PHX	SEA
ATL	-	30	30	1	1	1
CHG	30	-	30	1	1	1
NYC	30	30	-	1	1	1
KAS	1	1	1	-	15	15
PHX	1	1	1	15	-	15
SEA	1	1	1	15	15	-

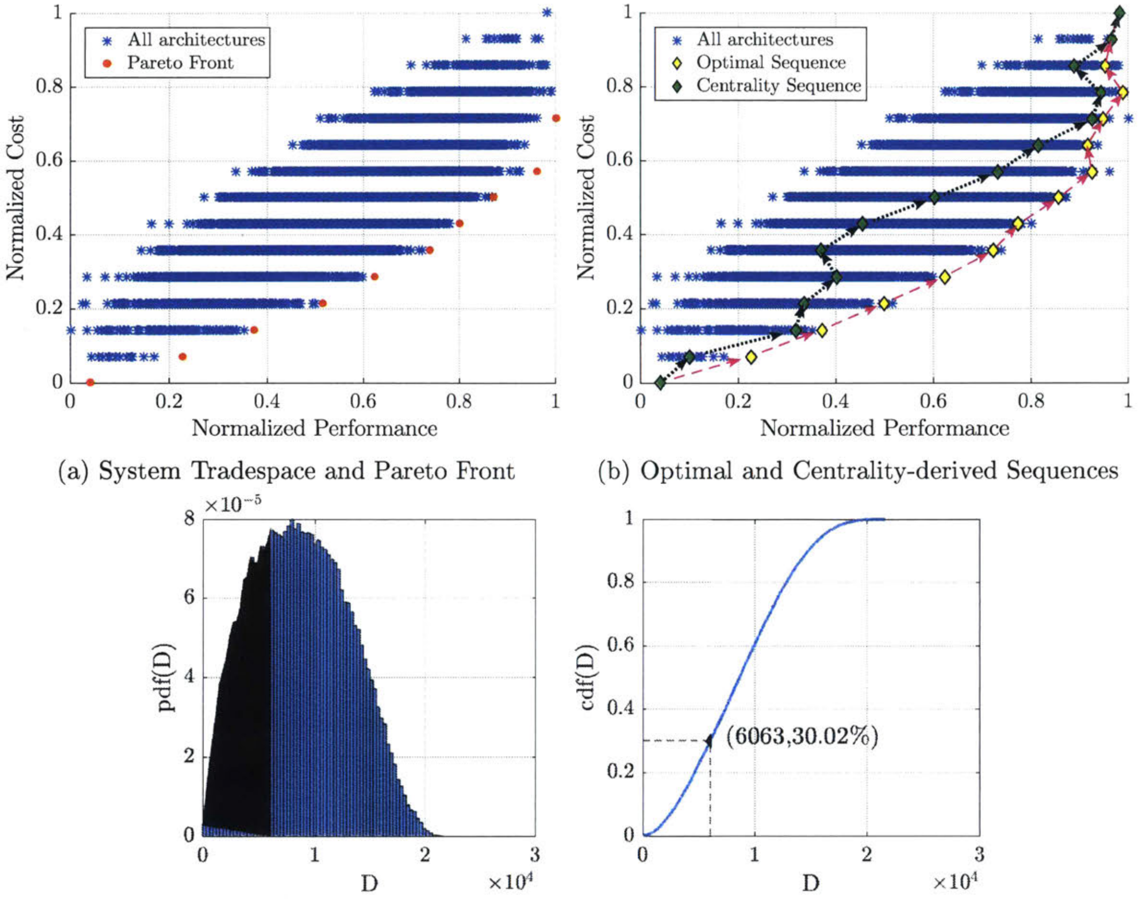
$\lambda_{ew} = 1\frac{packet}{sec}$.

- Nodes from the East coast send packets to other nodes on the East coast at a rate of $\lambda_{ee} = 20\frac{packet}{sec}$.

Given that there are only 3 nodes on each of the sub-networks, the relative data volume (normalized to 1 for the traffic between sub-networks) between any two origin-destination pairs is provided in Table 3.2. From this simplified analysis, we can hypothesize that nodes and connections in the East coast sub-network are more "important" than those in the West coast, while the previous max-flow min-cut connections are the least important. Therefore, applying an unspecified version of the centrality measure (i.e. $w_p = 1$ for all flows) should results in a largely suboptimal sequence of nodes to optimize.

3.9.1 Test 1. Pareto Distance

Figure 3-20 presents the results of stress case 3 in the metrics space, as well as the results of the P-distance test. Observe that the path followed using the centrality measure is clearly sub-optimal as it prioritizes upgrading the inter-sub-network connections even though the vast majority of utility is derived from the traffic sent/received within each of them. The effect of this sub-optimal decision making is apparent both visually in Figure 3-20b, with the optimal and centrality-based sequence largely separated from one another, as well as in Figure 3-20c, where the results of the P-distance test are reported. In this case, observe that the P-distance similarity score is as high as 30% approximately, two orders of magnitude larger than the same results for the baseline scenario. This clearly re-iterates the need of properly capturing data importance during the calibration process of the centrality measure. As a matter of fact, if the same experiment is re-run using the normalized weights from Table 3.2 to specify the centrality measure, then a P-distance similarity of 0.07% is obtained and, consequently, the performance of the proposed approach is comparable to what was observed in the baseline scenario.



(a) System Tradespace and Pareto Front

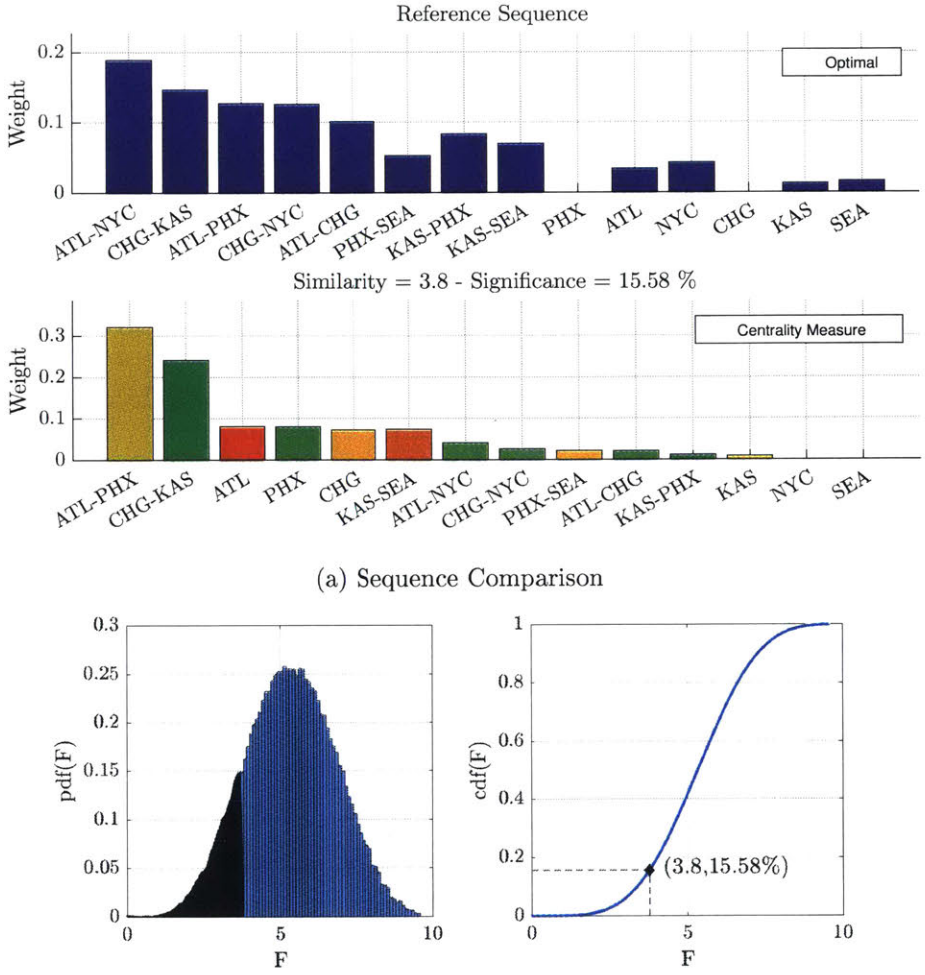
(b) Optimal and Centrality-derived Sequences

(c) P-Distance Test

Figure 3-20: P-Distance Test Results for Stress Case 3

3.9.2 Test 2. Ranking Similarity

Similarly, Figure 3-21 presents the comparison of the optimal and centrality-based sequences in the metrics space, along with the results of the ranking similarity test. In this case, results are consistent with the P-distance test, with a similarity score two orders of magnitude higher than the ranking similarity score observed in the baseline scenario. Interestingly, note how the optimal sequence as found through the graph-based approach correctly identifies that the inter-sub-network connections (ATL-PHX and CHG-KAS) are indeed the least important parts of the network. As a result, they are placed at the very end of the reference sequence, even after NYC and SEA which where the least important nodes in the baseline scenario. Similarly, the first five elements of the optimal sequence are all connections within the East sub-network, as the amount of traffic in that part of the system is twice what is generated



(a) Sequence Comparison

(b) Ranking Similarity Test

Figure 3-21: Ranking Similarity Test Results for Stress Case 3

on the other side[5]. None of these effects are captured by the original version of the centrality measure, thus resulting in a candidate sequence that has a similarity significance of almost 16%.

3.10 Stress Case 4: Cost Heterogeneity

A central assumption of the baseline scenario as defined in Section 3.3 is that all nodes and connections are cost-homogeneous. In other words, upgrading any of them results in the same amount of capital expenditure. In this stress case I break this assumption by assuming

[5] Connections of the East sub-network are prioritized over nodes in the same sub-network because in their basic implementation they generate almost twice as much latency as a node (187.5msec vs. 100msec).

the following normalized costing structure:

$$\text{Cost [node]} = \begin{cases} 1 & \text{if L=100msec} \\ 2 & \text{if L=20msec} \end{cases} \tag{3.13}$$

$$\text{Cost [connection]} = \begin{cases} 1 & \text{if L=187.5msec} \\ 1.5 & \text{if L=20msec and connection} \notin \text{\{ATL-PHX,CHG-KAS\}} \\ 15 & \text{if L=20msec and connection} \in \text{\{ATL-PHX,CHG-KAS\}} \end{cases} \tag{3.14}$$

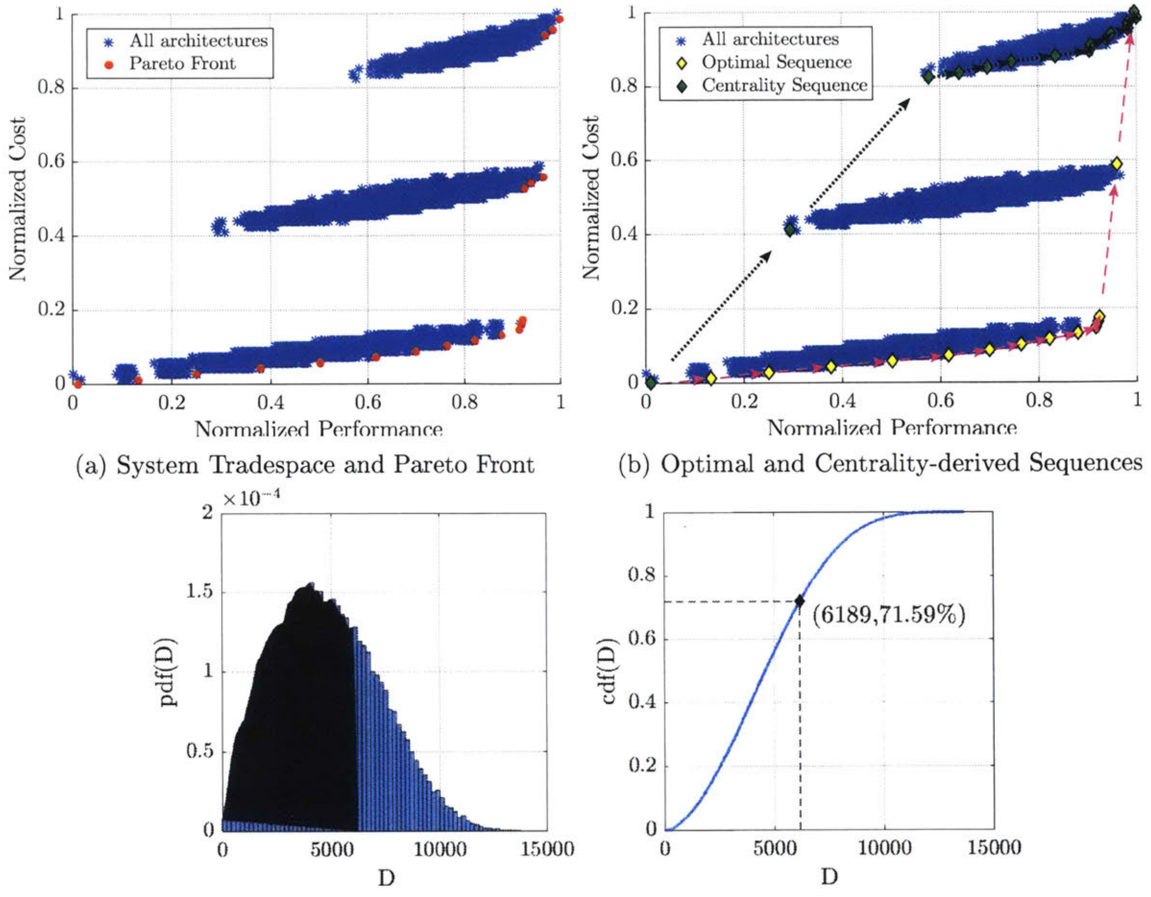
Observe that a node has only two implementations, a basic one and an upgraded one with double the cost. On the other hand, a connection has three possible associated costs. In the basic implementation, the cost is always equal to 1 normalized unit. Nevertheless, the cost of upgrading a connection depends on which one it is, hence the term cost heterogeneity. In general improving a connection is priced as 0.5 units of extra capital expenditure. However, in some special cases this capital expenditure increases by more than an order of magnitude. To accentuate this effect, I choose these "special" connections to be those of the network max-flow min-cut, i.e. ATL-PHX, CHG-KAS.

3.10.1 Test 1. Pareto Distance

Figure 3-22 plots the results of the stress case 4. First, consider Figure 3-22a. Observe that three stratified levels on the y-axis appear and, within each strata, architectures exhibit a quasi-linear cost vs. performance behavior. Both characteristics arise from the fact that the system is cost heterogeneous. Indeed, the jump before the lowest and intermediate strata occurs when either ATL-PHX or CHG-KAS are upgraded as they require an enormous capital expenditure. Similarly, the top strata contains all architectures for which both ATL-PHX and CHG-KAS have been upgraded. On the other hand, within each strata points also do not lie in perfect horizontal lines as in the previous stress cases. Indeed, in this case cost heterogeneity results in a tightly packed ensemble of points with positive slope.

Figure 3-22b plot the evolution paths from the basic network architecture to the fully upgraded system as computed through dynamic programming and the centrality measure. Once again, the latter has not been adapted to reflect the specific problems of cost heterogeneity. Observe how the centrality measure completely misinterprets the sequence of nodes to upgrade and results in a clearly sub-optimal path. Indeed, the max-flow min-cut elements of the network are once again selected as the first nodes to upgrade. Unfortunately, since these are precisely the ones that require more capital expenditure, they also lead to a large upfront investment that is unnecessary from the perspective of a gradual strategic system evolution.

All these issues are also correctly quantified by the proposed P-distance test. Interestingly,



(a) System Tradespace and Pareto Front

(b) Optimal and Centrality-derived Sequences

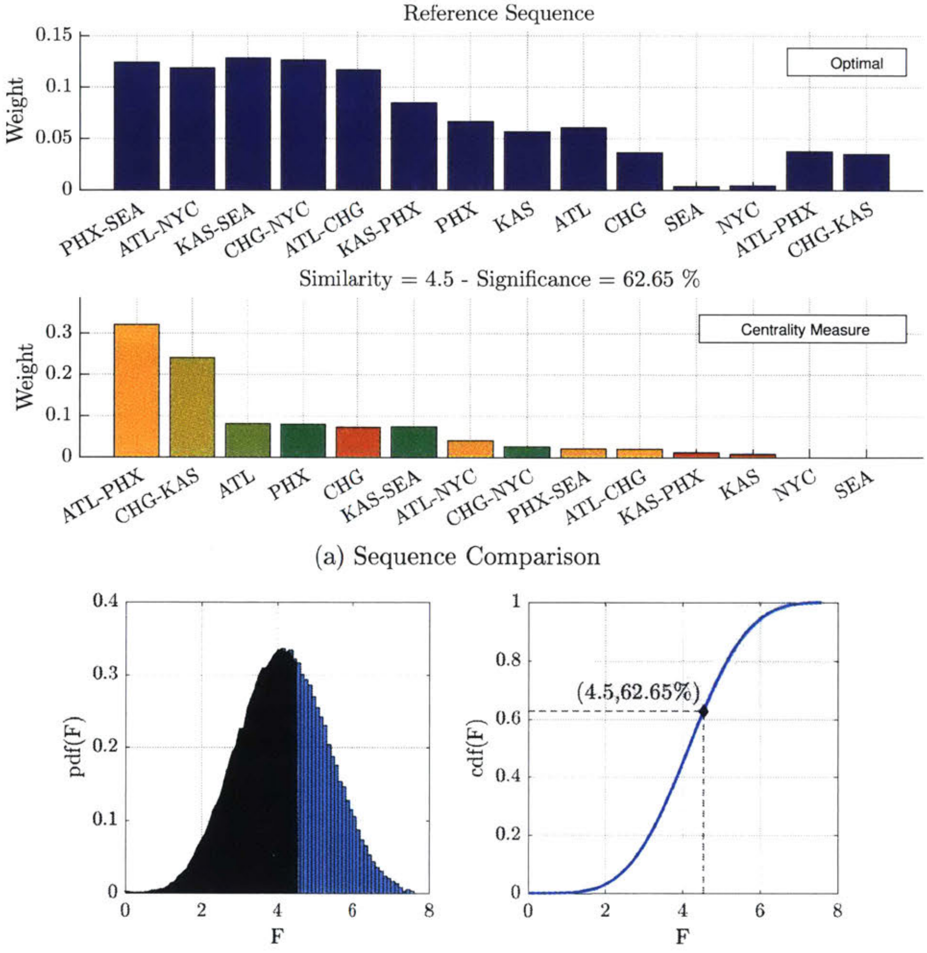
(c) P-Distance Test

Figure 3-22: P-Distance Test Results for Stress Case 4

note that the P-distance significance is as high as 70% in this case. In other words, there is a 70% chance of finding a better sequence of nodes to upgrade than the one found through the centrality-based approach. This indicates that if the system architect created a random ranking and upgraded the network accordingly, it would, on expectation, spend less resources than using the original centrality measure.

3.10.2 Test 2. Ranking Similarity

Figure 3-23a presents the two sequences that should be used to optimize the system according the optimal and heuristic method. Observe how the former method correctly identifies that the ATL-PHX and CHG-KAS connections are too expensive to be upgraded at the beginning of the optimization process and are, therefore, pushed to the 13th and 14th position respectively. Note, however, that the ATL-PHX and CHG-KAS connections are not identified as the largest contributors to the ranking similarity score. This is due to the weighting effect embedded into the metric, since these two connections are not too important in the



(a) Sequence Comparison

(b) Ranking Similarity Test

Figure 3-23: Ranking Similarity Test Results for Stress Case 4

utility loss space, misplacing them has a relatively low impact. In contrast, misplacing a more important element such as the KAS-PHX connection causes the similarity metric to be largely penalized because it generates a large utility loss and is relatively "cheap" to upgrade (i.e. spending resources in this part of the system has a high return on investment). Finally, the ranking similarity test results are consistent with those of the P-distance test. In this case the significance score is as high as 62% approximately, thus indicating that using the centrality measure in cost heterogeneous systems without calibration is not advisable.

3.11 Conclusions and Summary of Results

In this chapter, I have studied the ability of the centrality measure from Chapter 2 to optimize a WAN with multiple heterogeneous latency contributors. This benchmarking exercise has been conducted by comparing the results of optimizing a network using two

Table 3.3: Summary of Validation Results

Scenario	Calibration	P-distance Test		Ranking Similarity Test	
		Score	Significance	Score	Significance
Baseline Scenario	Yes	187	0.06%	1.50	0.41%
S.C. 1: Utility Function	No	92	0.04%	0.52	0.05%
S.C. 2: Data Routing	No	2002	6.55%	0.44	0.46%
S.C. 3a: Data Importance	No	6063	30%	3.8	15.58%
S.C. 3b: Data Importance	Yes	83	0.07%	0.82	0.56%
S.C. 4a: Cost Heterogeneity	No	6189	71.59%	4.50	62.65%
S.C. 4b: Cost Heterogeneity	Yes	156	0.05%	0.32	0.25%

complimentary approaches: A heuristic algorithm based on network centrality, and a full factorial design space exploration algorithm coupled with a high fidelity network simulator. A total of five different cases were run and evaluated, the first one defining a canonical scenario based on four main assumptions, and the other four progressively violating each of them.

Table 3.3 summarizes the results of the validation process for all the aforementioned scenarios. The score and significance of both the P-distance (test 1) and ranking similarity tests (test 2) are reported due to their complimentary nature. Furthermore, for Stress Cases 3 and 4, I report both the scores and similarities before and after calibration[6] of the centrality measure in order to demonstrate the positive effect properly characterizing how utility is lost through the system. Based on Table 3.3, the following set of conclusions can be reached:

1. The utility function convexity does not affect the validity of the proposed approach. Therefore, the only restriction I impose on the utility function is for it to be decreasing with latency.
2. Properly characterizing the data flows through the network is essential to ensuring the validity of the proposed centrality measure. This characterization should be performed as a calibration process and entails two parts: Understanding how data moves through the network; and quantifying the relative importance of different information flows with respect to the utility they deliver to the final user.
3. The centrality measure, as defined in Chapter 2, only takes into account performance or the lack thereof. Yet, most engineering project trade performance against a secondary metric such as cost. If different parts of the system are subject to different levels of capital expenditure in order to improve performance, the centrality measure should be specified so that it captures the unit of utility loss per capital expenditure. This is analogous to the approach followed by Manuse in the strategic evolution of complex

[6] The post-calibration results where not reported in the previous subsections for the sake of brevity.

systems [152], where she essentially creates rankings ordered according to return on investment, so that the system architect can identify areas of promising performance improvement at reasonable cost.

Finally, several areas of future work can be identified as part of this case study. First and foremost, the number of experiments run was necessarily constraint by the limited time and resources available to the author. For instance, each simulation exercise required approximately a half a day of computational time in high-performance 32 core Intel Xeon computer. Furthermore, a complimentary retrospective study with a real-life system would, at this point, also be beneficial (see Section 4.3). Note the significant differences between using a real-life vs. a simulation-based case study for validation. While the latter grants full control over the set of experiments to conduct and the factors that drive them, the latter asserts validity by replicating events that have happened in real-life. In a sense, this is analogous to the trade between modeling breadth and depth. Comparison with a retrospective case study asserts validity against a specific set of conditions. In contrast, comparison against higher fidelity models asserts validity by letting the user test a wide range of possible scenarios and, in my case, quantifying how the centrality measure and system optimization react to them.

All in all, and despite the limitations of the conducted benchmarking exercise, the obtained results demonstrate the proposed approach to architect networks based on centrality measures can yield to optimal results provided that the centrality measure is properly specified. Similarly, bounds on the applicability of the proposed centrality measure have been established, and factors that drive its usefulness have been identified.

4 Case Study 2: Return of Weather Satellite Observations

4.1 Introduction

The economic and societal impact of accurate weather forecasting has been a topic of interest for more than four decades. One of the first surveys to treat the topic was published in 1989 by the Midwestern Climate Center [153]. Its goal was to quantify the value of weather forecasts from a microeconomic perspective, i.e. an individual with enhanced decision-making capabilities due to *ex-ante* meteorological information. Based on their results, the study concluded that improved forecasting accuracy results in quantifiable benefits for several economic activities, including wheat and corn production, residential housing, boating and flood control among others.

More recently, the value of accurate weather forecasting has vindicated improvements in both US's data collecting and processing capabilities. For the former, the JPSS program will provide enhanced space-based measurements of the atmosphere and ensure data continuity from aging satellite programs such as POES and EOS. For the latter, assimilation of new observations provided by synergistic systems such as the Integrated Ocean Observing System, as well as data and models from private or academic institutions is recommended to increase forecasting ability both at a global and regional scale [154].

Given the societal and economic benefits of weather forecasting, this case study focuses on timely return of satellite-based observations currently used to feed numerical weather prediction (NWP) centers. In that sense, the chapter has three primary objectives: First, demonstrate how the latency-centric approach to architecting space communication networks can be applied to a real system. Second, provide recommendations on how to evolve ground and space-based networks that optimally support weather forecasting activities with different temporal and spatial resolutions. And third, further validate the proposed centrality measure by comparing the rankings of latency contributors it produces against lessons learned from currently implemented and proposed ground systems.

4.2 Framework Application

Step 4-1. Motivation and Domain Specific Expertise

NWP systems enable meteorologists to deliver timely weather forecasts by aggregating atmospheric data from different sources and predicting the state of the atmosphere. In that sense, current NWP systems use a combination of ground, airborne and space based assets to first gather atmospheric measurements, then they distribute them through space and ground communication networks, and finally they process them in a set of centralized facilities.

NWP systems are typically categorized depending on the extent of their forecasting capabilities both in terms of the temporal and spatial resolution. For instance, the National Centers for Environmental Prediction (NCEP) currently runs three main forecasting models, the Global Forecast System (GFS) which covers approximately all Earth; the North American Mesoscale Forecast System (NAM), which comprises North America including non-CONUS territories such as Alaska, Hawaii or Guam; and the Rapid Refresh (RAP), which is complemented by the High-Resolution Rapid Refresh (HRRR) system to produce short forecasts for CONUS and other US territories. A similar categorization is followed by the Japanese meteorological agency with their Global Analysis (GA), Mesoscale Analysis and Local Analysis, while other meteorological agencies such as the European Center for Medium-Range Weather Forecasts (ECMWF) are specifically interested in medium to long forecasts (1-10 days) at the global level.

In order to produce forecasts, NWP systems follow a periodic assimilation system by which they match previous forecasts with new data to produce an accurate representation of the current state of the atmosphere. This state is then propagated in time based on a set of equations that model the atmosphere dynamics and the result of this process is further post-processed and finally delivered to weather data providers such as AccuWeather [155], who make it available to the broad public.

Step 4-1.1. Data Sources for NWP Systems

As previously mentioned, both surface, airborne and satellite observations are currently used to feed NWP models. Figure 4-1 exemplifies the coverage provided by each measurement type (see Table 4.1 for a comprehensive list of observations), with ground based assets represented by buoys, airborne assets represented by airplane data and satellite data represented by scatterometer observations from MetOp satellites. Observe the complimentary nature of satellite and non-satellite observations. While the former provide great global coverage, the latter can be used to obtain specific atmospheric measures with perfectly known position and altitude. Yet, their overall coverage is significantly limited, especially in oceanic areas

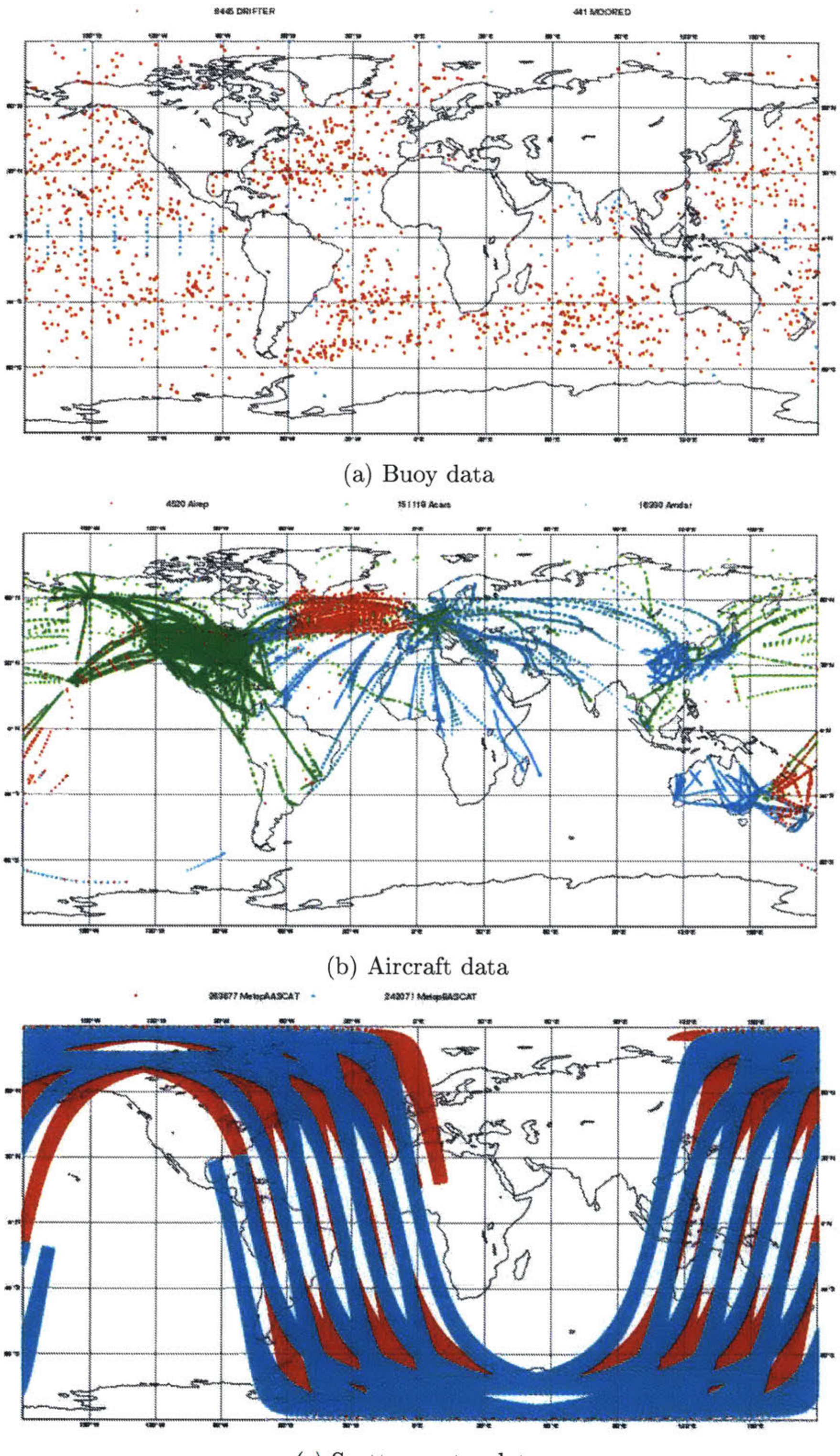
Table 4.1: Data Sources for the ECMWF NWP System

Data Product	Type	Description
Synop-ship	Surface	Ships transmit atmospheric measurement using Synop format
Buoy	Surface	Buoy measurements (see Figure 4-1a)
Ground-based GPS	Surface	GPS radio occultation measurements
Airport balloons	Airborne	Balloons launched periodically from airports
Aircraft	Airborne	Aircraft measurements of stratosphere during flight (see Figure 4-1b)
Pilot-Profiler	Airborne	Aircraft measurements of troposphere during take-off/landing
Radiances	Satellite	IR/MW radiances measured from space
GPSRO	Satellite	GPS radio occultation measurements
Ozone	Satellite	Atmospheric composition
Winds	Satellite	Wind profiles at different altitudes (see Figure 4-1c)
Could images	Satellite	Visible cloud imagery

not frequented by airline routes.

Given the complimentary nature of all observations currently used by NWP centers, it is important to determine which of them are critical for producing accurate weather forecasts. This question has been addressed in the literature using two complimentary approaches, the Observing System Experiment (OSE) methodology [156] and the adjoint method [157]. The former assesses the importance of a given data product by running a series of experiments in which a given data set is removed from the assimilation system used to determine the state of the atmosphere. Then, the performance of the forecast is compared with a control experiment in which all data is present. On the other hand, the adjoint methodology is a more advanced technique that measures the sensitivity of the assimilation system with respect to each type of data based on the influence-matrix diagnostic [157]. In short, the assimilation process is a weighted average between the new available observations and the past forecast, with weights computed based on their respective accuracy. Consequently, we can view the assimilation process as a linear regression for which the sensitivity of a given observation can be estimated from the diagonal elements of the influence or hat matrix.

Over the last two decades, the ECMWF has published several studies analyzing the sensitivity of their forecasting system to different data products. Their first studies used the conventional OSE method, while latter studies are now utilizing the adjoint method. For instance, Reference [7] studies the impact of GPSRO data in weather forecasts through the adjoint method. Figure 4-2 plots the percent forecast error reduction for each of the data products currently assimilated by the ECMWF system. It can be observed that at least 50% of the forecast error reduction is due to assimilating satellite data from instruments AMSU-A, AIRS, IASI and GPSRO. Similar results are reported in Reference [157], where it is stated that “about 25% of the observational information is currently provided by surface-based observing systems, and 75% by satellite systems. This importance is also emphasized



(a) Buoy data

(b) Aircraft data

(c) Scatterometer data

Figure 4-1: Coverage of Observations for the ECMWF 05/11/2016 00UTC Analysis

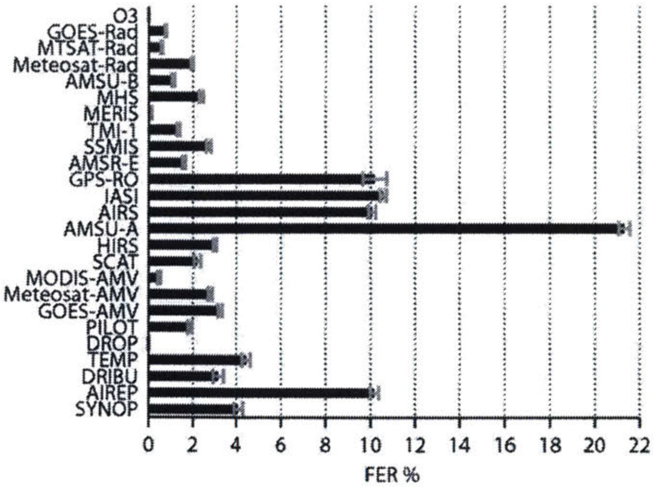


Figure 4-2: Forecast Error Reduction (Adapted from Reference [7])

by References [156] and [157], where the number of observations assimilated per run is provided. Once again, satellite-based instruments and their measurements contribute to more than 50% of data used to feed NWP systems and therefore are crucial for delivering their current weather forecasting performance.

Step 4-1.2. Satellite Data Providers for Weather Forecasting

In this section I briefly summarize the different satellite systems that are used to provide data to current NWP centers. These include the JPSS[1], a partnership between NOAA and NASA that will deploy up to three polar orbiting satellites to gather global environmental data for weather and climatological purposes. This system will be complemented, in LEO, through partnerships with other weather satellite programs such as ESA's MetOp system, JAXA's GCOM satellite and the DoD's DMSP. Additionally, observations from geosynchronous orbit through the GOES spacecraft operated by NOAA will also be considered since they also provide data to current US NWP systems. Note that weather data from geosynchronous orbit in Europe and Asia is currently provided through bilateral agreements with ESA's Meteosat program, as well as JAXA's and CMA's Himawari and Fengyun systems. However, they are not considered in this study since they are architected completely separately from the JPSS data ground infrastructure.

[1] The Suomi NPP mission is also included since it was launched 2011 and is expected to survive at least until 2025)

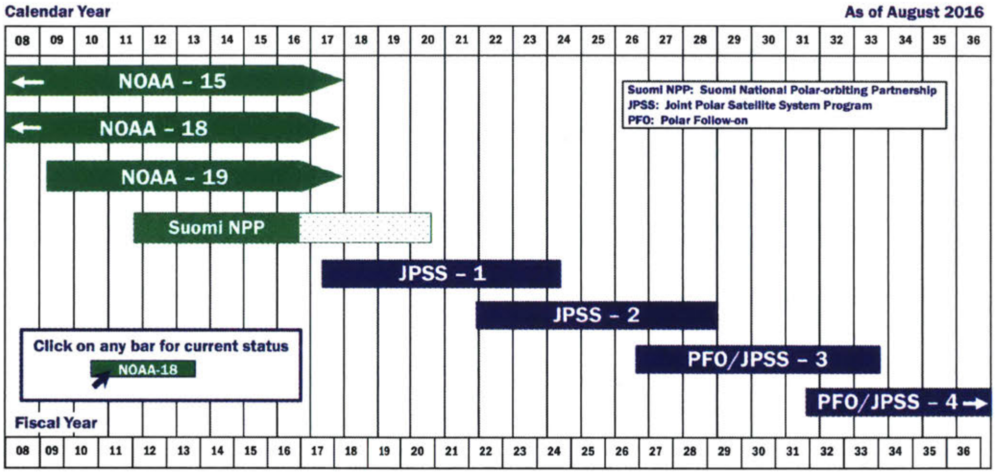


Figure 4-3: POES-JPSS Transition (Adapted from Reference [8])

The Joint Polar Satellite System

The JPSS is the latest US polar orbiting satellite program for weather and climatology purposes. It is the successor of the NPOESS, albeit in this case it is only being developed by NASA and NOAA. When deployed, JPSS satellites will replace NOAA's aging POES constellation. In that sense, the first satellite is currently scheduled for launch on the 2$^{\text{nd}}$ quarter of 2017, with three more satellites being deployed afterwards in five years intervals (see Figure 4-3).

The JPSS space segment will be supported by the JPSS Common Ground Infrastructure (CGI), which will also provide communication services to other meteorology partners within US agencies and international organizations. Table 4.2 summarizes the set of functions provided to JPSS satellites and others by the CGI [158], [159]. At the highest level, the system is currently being designed in order to support four core functions, from data acquisition to spacecraft control and monitoring. Since, the latter does not affect the delivery of weather-related data to NWP centers, I will not consider it for the rest of this case study. On the other hand, data acquisition refers to downlinking information from the space platforms to a set of ground stations where the space signal is processed to eliminate all communication artifacts such as coding bits. Similarly, data routing refers to the transmission of all data from the ground stations to each spacecraft's science facility. Then data processing and distribution functionality generate L1, L2 and L3 instrument data products and disseminates them to NWP centers for assimilation into their weather forecasting models.

The current design of the CGI is a clear improvement over current ground systems, but it is also a downsized version of the originally proposed NPOESS ground segment. The system is based on a centralized processing architecture connected to a network of ground

Table 4.2: JPSS CGI Functionality

Mission	Data	Data	Data Processing	Spacecraft
	Acquisition[2]	Routing	& Distribution	Controlling
JPSS and NPP	✓	✓	✓	✓
MetOp	✓	✓	✗	✗
NASA EOS	✓	✓	✗	✗
DMSP	✓	✓	✗	✗
GCOM	✓	✓	✓	✗
GOES	✗	✓	✗	✗
Other (e.g. Coriolis spacecraft)	✗	✓	✗	✗

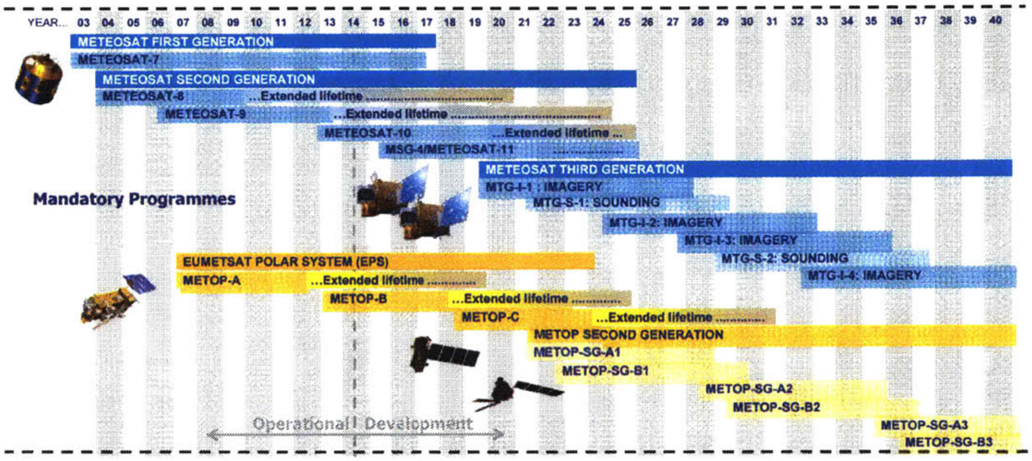


Figure 4-4: MetOp and METEOSAT Evolution (Adapted from Reference [9])

stations that offers two contacts per orbit. The selection of polar ground stations, albeit suboptimal from a costing and risk perspective, is vindicated by the need to primarily service sun synchronous satellites. Locations available for downlinking data include Svalvard and Fairbanks in the Northern Hemisphere, as well as McMurdo and Troll in the Southern Hemisphere. Furthermore, two hot processing facilities are provided for full redundancy at Suitland, Maryland and Fairmont, West Virginia. They can be swapped in under five minutes in case of emergency [160].

The MetOp Satellite System

The MetOp satellite system is developed and operated by the ESA. It includes three satellites in SSO that were originally deployed as part of the EUMETSAT/NOAA partnership, formerly known as Initial Joint Polar System (IJPS) [161]. The original MetOp satellite was lunched in 2006 and the second generation is expected to be ready by as early as 2021.

Current operations and communications for the MetOp system are provided by the EUMETSAT enterprise independently from other US meteorological spacecraft. In particular, each satellites is first supported through Svalbard (one pass per orbit), from where data is relayed to EUMETSAT's headquarters in Darmstadt, Germany for L1 and L2 processing and further distribution [162]. However, with the advent of the CGI, MetOp spacecraft will be allowed to downlink data twice per orbit using both Northern and Southern Hemisphere ground stations and data will be routed to the respective European data centers.

The NASA Earth Observation System

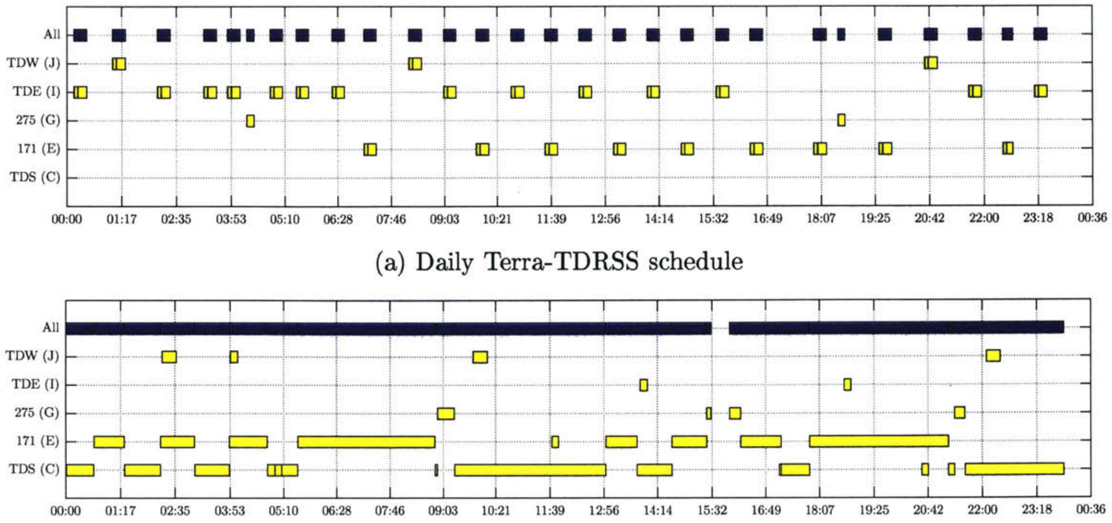
The NASA Earth Observation System (EOS) was originally conceived during the 90's and has been the key element to the agency's Earth Science program to develop a better scientific understanding of Earth as an integrated ecosystem being affected human activities, most notably greenhouse effect emissions. Not all NASA EOS missions are relevant for the purposes of NWP systems. Yet, data products from their flagship missions Aqua and Terra are routinely used to complement measurements from other dedicated systems such as NOAA's POES and GOES satellites, as well as ESA's MetOp satellites [157].

The EOS polar ground infrastructure is mainly composed of two north pole sites, Svalbard and Alaska equipped with 10-13 meter antennas [163]. Downlink of data can be performed through a Ku-band transmitter as well as an X-band transmitter, once per orbit and at a maximum data rate of 150Mbps. Data downlinked to these polar ground stations is sent to CONUS through NASA operated lines, as well as non-NASA circuits that provide a total capacity of 100 to 200Mbps [164]. Initially all data is first directed to the GSFC in Maryland and then relayed to the final destination on a case-by-case basis.

Flagship satellites Terra, Aqua and Aura are also supported by NASA's SN through multiple contacts per orbit (see Figure 4-5). In that sense, Terra is currently supported by two 20 minute contacts per orbit where data is first sent to a TDRS spacecraft using the high-rate Ku-band service. Upon reception, the relay satellite sends the information to their ground system immediately, which then repatriates and distributes it to the corresponding NWP centers through the EOS Ensight network [164]. Finally, Aqua and Aura are also supported by the SN, albeit not for return of science data. In particular, both spacecraft are granted almost continuous low data rate contacts through the TDRS S-band MA service. Therefore, this part of the system will not be considered for this case study.

The Geostationary Operational Environmental Satellite

The Geostationary Operational Environmental Satellite (GOES) is a US satellite program that has periodically deployed geosynchronous satellites for weather forecasting, meteorology and space weather since 1974. During these three decades of service, GOES satellites have been progressively updated in order to improve their remote sensing capabilities. In fact, the



(a) Daily Terra-TDRSS schedule

(b) Daily Aqua-TDRSS schedule

Figure 4-5: TDRSS support of EOS satellites circa 2012

latest addition to the fleet occurred in November 2016, when the new GOES-R spacecraft will be deployed using an Atlas 5 rocket (see Figure 4-6).

The GOES communication infrastructure is currently composed of three main assets: NOAA's Satellite Operations Facility (NSOF) in Suitland, MD, the Wallops Command and Data Acquisitions Station in Wallops, VA and the Consolidated Backup facility in Fairmont WV [165]. The first facility acts as the primary science operations center, and therefore continuously receives science data from the spacecraft and processes it to create L1 and L2 data products. Some of these data products are uplinked back to the GOES satellite, which then broadcasts them through the specially designed GOES Rebroadcast system (GBR) [166]. Note that this description is based on the upcoming GOES-R series of satellites, while past GOES utilize a similar yet less capable infrastructure.

The Global Change Observation Mission

The GCOM system is JAXA's main effort to global weather and climate monitoring in the next decades. The program is supposed to launch a total of six satellites between 2012 and 2030 approximately. Three of them, the GCOM-C series, are tasked with monitoring climate change over five year intervals by measuring Earth's carbon cycle and radiation budget. They are not considered in this study, since their data is latency-unconstrained. On the other hand, the three satellites from the GCOM-W series are tasked with observing Earth's water cycle, as well as wind velocity, sea surface temperature, snow depth among others. These data products, all relevant to weather forecasting, are latency-sensitive and

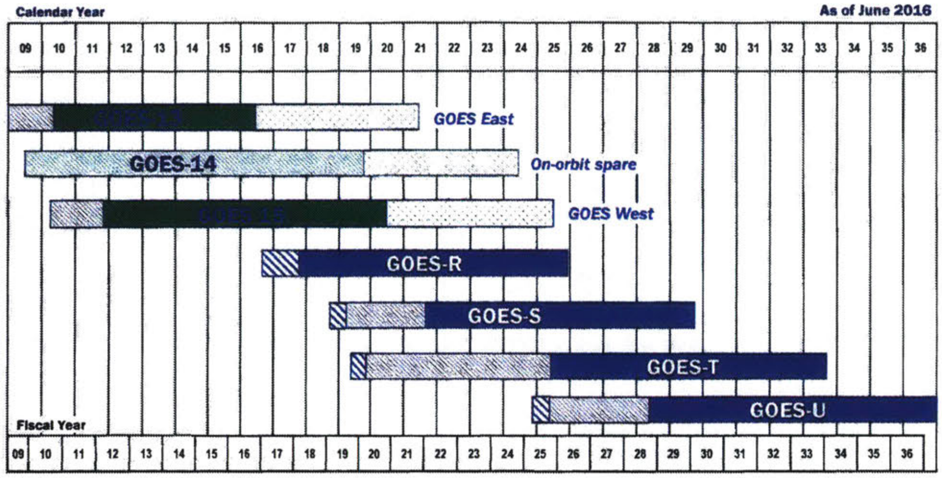


Figure 4-6: GOES Evolution (Adapted from Reference [8])

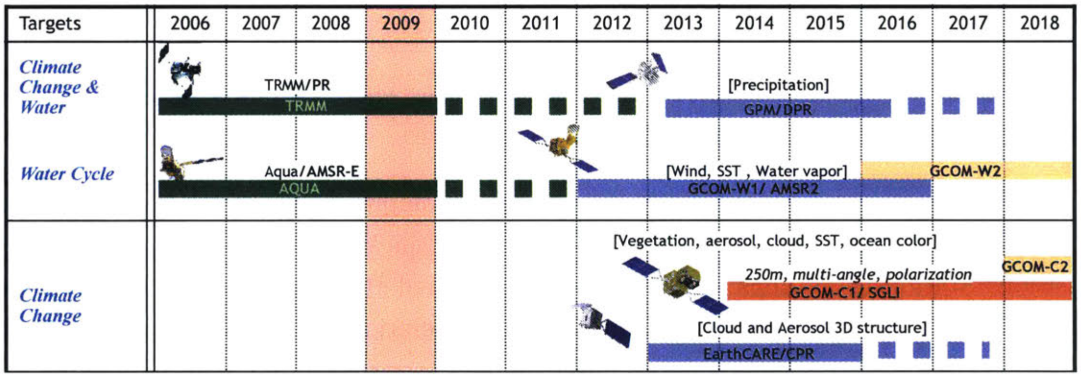


Figure 4-7: GCOM Evolution (Adapted from Reference [10])

therefore must be considered when architecting the ground infrastructure that services them.

Under current agreements, the CGI supports GCOM through two functions: First, it provides high data rate contacts with the spacecraft, one per orbit, through the Svalbard ground station. This service includes all scheduling functionality, which is centralized by JPSS given the navigation solution and operational plans developed by JAXA. Space packets received the ground site are processed at the CGI PoP, encapsulated in a return SLE service, and forwarded directly to JAXA [160]. Furthermore, packets are also sent to NOAA, specifically their ESPC and CLASS systems, where sensor and environmental data records are created from raw measurements and distributed to the final users [160].

Table 4.3: NWP End-to-End System Architecture

Physical Node	Node Functionality	Latency Contributors
Satellite (SAT)	Data acquisition and storage	Image acquisition & LOS
Ground Network (GN)	Data downlink	Packet processing, data transmission time
Wide Area Network (WAN)	Data routing	Routing of data to processing facility
Processing Facility (PF)	Data processing	Generation of L1, L2 data products
Distribution Network (DN)	Data distribution	Distribution of L1, L2 data to NWP centers

The Defense Meteorological Satellite Program

The DMSP provides weather information to all US military branches. The current infrastructure provides downlink opportunities to SSO using four locations in Fairbanks, AK, New Boston, NH, Thule Air Force Base, Greenland, and Kaena Point, HI [160]. Data is then, relayed to the FNMOC through domestic satellite systems, where it is processed and forwarded to the final users.

JPSS's CGI will enhance this infrastructure by providing contact opportunities, scheduling and data routing services to DMSP satellites at the McMurdo station in Antarctica. This will enable the DMSP system to reduce latency by as much as 40% [160]. Moreover, the CGI will also provide continuity of operations in case of contingency events in the ground infrastructure through the Consolidated Backup Facility in Fairmont, West Virginia [160].

Step 4-2. Specify the Centrality Measure

Step 4-2.1. Characterization of Latency Contributors

Given the previous description of weather satellite programs and their communication infrastructure, Table 4.3 summarizes the end-to-end system architecture for return of weather satellite data as a function of the functionality provided and the latency contributors it introduces. In that sense, latency can be induced as early as in the image acquisition process, especially in the case of geosynchronous satellites that capture full-disk images with wide area coverage and high spatial resolution. In the case of LEO satellites, data is obtained almost instantaneously thanks to the limited instrument swath, but downlink contact opportunities are constrained by line of sight visibility between the spacecraft and the ground system.

Once the data reaches a ground station, all space packets are processed and encapsulated into an SLE return service that is forwarded through a WAN. This WAN is usually ground-based, but can also be space-based as is the case for the DMSP satellites. Next, data products are received at ground processing facilities where raw instrument data products are ingested and transformed into L1, L2 and L3 data products. Of those, NWP centers usually requires L1 products for their assimilation systems, which are transported to their final destination through a separate ground distribution network.

Using information from all satellite programs described in Step 4-1, I construct Table 4.4. A total of 35 nodes are listed, mapping Table 4.3 to the six weather satellite programs considered in this case study. Latency estimates provided are expected values computed assuming a 1 contact per orbit network[3], and are based on the average instrument data rate listed for each of the spacecraft under consideration. Observe that no latency is listed for GOES WAN as data is downlinked directly to the science processing center. Finally, a 5 minute margin in data product processing is assumed due to common inter-dependencies across data products [48], and distribution latency is assumed to be on the order of 1 minute thanks to a pessimistic 150Mbps direct connection to the NWP center.

Step 4-2.2. Identification and Characterization of Data Flows

As exemplified by Figure 4-2, not all satellite data products are equally important for weather prediction purposes. Therefore, in this section I quantify the relative importance of different satellite-based data products obtained by weather spacecraft using an approach analogous to past system architecting studies related to Earth observing systems (see Figure 4-8 and References [26] and [18]). In particular, I first identify the set of sub-domains that are relevant for weather forecasting purposes. Then, I decompose each sub-domain into a set of specific measurements to be taken by space-based instruments. And finally, I elicit the ability of current and future instruments on-board the previously described satellite programs to deliver those measurements. Note that I implicitly exclude the problem of selecting a set of instruments for a given satellite program. This simplification is justified by two facts: First, by anchoring the results of this step on current planned capabilities, I ensure that the centrality measure is estimated using realistic data. Second, the problem of architecting satellite observation programs has already been studied in the literature (e.g. Reference [26]) and is ultimately related to scientific value rather than infrastructure cost.

Data Products for Numerical Weather Forecasting

Satellite measurements for NWP system are typically used to derive the state of the atmosphere at either the surface, the troposphere and tropopause and the mesosphere [167]. At

[3] This is representative of the current system implementation for all programs except for DMSP which has better ground support. No scheduling constraints are assumed since weather satellites have enough priority to ensure the assumed level of ground support.

[4] This corresponds to a maximum latency of 90 minutes, the value typically reported. The average is taken across all latitude/longitude points.

Table 4.4: Characterization of Latency Contributors

Node		Latency [min]	Rationale
JPSS:			
	SAT	56^4	1 polar ground station
	GN	7.0	44 Gbit/orbit at 300Mbps
	WAN	7.5	100Mbps dedicated line
	PF	10.0	5min margin for interdependencies
	DN	3.5	150Mbps dedicated line
MetOp:			
	SAT	56	1 polar ground station
	GN	9.5	14 Gbit/orbit at 70Mbps
	WAN	2.5	100Mbps dedicated line
	PF	10.0	5min margin for interdependencies
	DN	1.8	150Mbps dedicated line
EOS:			
	SAT	56	1 polar ground station
	GN	10.0	40 Gbit/orbit at 150Mbps
	WAN	7.0	100Mbps dedicated line
	PF	10.0	5min margin for interdependencies
	DN	3.5	150Mbps dedicated line
DMSP:			
	SAT	33	4 semi-polar ground stations
	GN	1.0	80 kbit/orbit at 3Mbps
	WAN	1.0	DOMSAT (Inmarsat GX - 50Mbps)
	PF	10.0	5min margin for interdependencies
	DN	1.0	150Mbps dedicated line
GCOM:			
	SAT	56	1 polar ground station
	GN	1.0	Downlink 1Gbit/orbit
	WAN	1.0	100Mbps dedicated line
	PF	10.0	5min margin for interdependencies
	DN	1.0	150Mbps dedicated line
GOES:			
	SAT	15.0	Full disk acquisition
	GN	0.0	Continuous downlink
	WAN	0.0	No wide-area network
	PF	10.0	5min margin for interdependencies
	DN	1.0	150Mbps dedicated line
NWP			
	NWP	0.0	Latency measured at entry of NWP

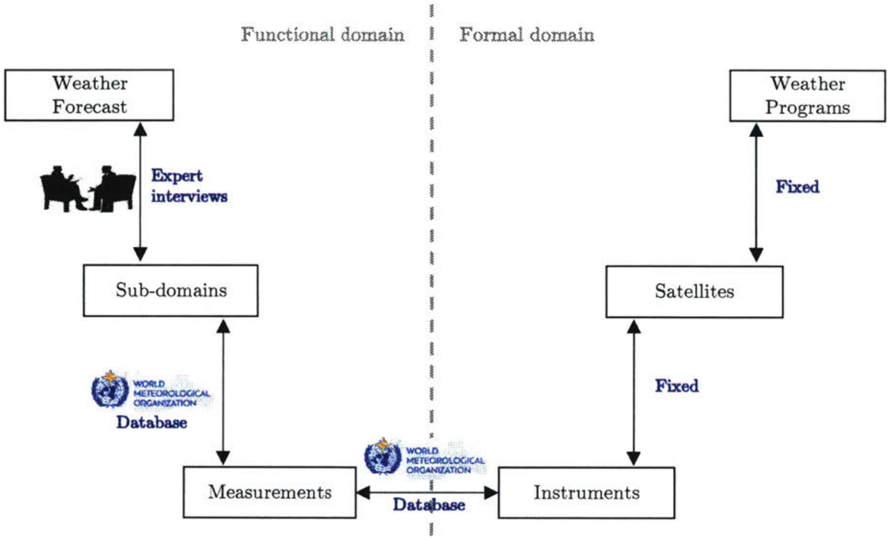


Figure 4-8: Identification of NWP Data Flows

a high level, the goal is to obtain estimates for the temperature, pressure, humidity and wind profiles at different altitudes, as well as information on the atmosphere's composition, clouds and precipitations. Additionally, certain properties from the Earth's surface (e.g. vegetation levels, sea-ice or snow coverage, etc.) are sometimes utilized to obtain boundary conditions, as well as for forecasting unique values such as minimum surface temperature [I9].

A comprehensive decomposition of the data products required for numerical weather forecasting can be obtained through the OSCAR database maintained by the World Meteorological Organization [168]. For high resolution NWP, it lists a total of 56 data products categorized in six main sub-domains[5]:

- Atmospheric characterization, including temperature, pressure, humidity and water vapor profiles.
- Atmospheric chemistry, mainly related to Ozone in the upper layers of the troposphere and lower stratosphere.
- Atmospheric winds, both at high altitudes and the surface.
- Cloud characterization (e.g. cloud cover, cloud type) and composition (e.g. cloud liquid water, cloud ice).
- Aerosols and radiation measurements, which includes measurements of the Earth albedo or aerosol mass mixing ratio.

[5] Note that these sub-domains have an almost one-to-one mapping with the the of objectives for the weather panel in Reference [26].

- Surface characterization, which includes measurements of the vegetation index, precipitation intensity for nowcasting, sea-ice cover or snow cover.

Interestingly, the database also provides a timeliness requirement for each of these data products, with three values provided: An optimal value, a desired target goal and a threshold after which the data has no value. It can be observed that the optimal values are always in the 15 minutes range while desired goals are set at 30 minutes to 1 hour approximately. There is, however, one exception: Most measurements related to surface characterization have relatively large latencies of hours or even days, which seems to suggest that they are not constrained by latency problems. This fact was corroborated by a NWP expert, who indicated that surface measurements such as sea ice cover do not vary significantly even on a daily basis and therefore do not need to be updated regularly [I9].

Unfortunately the OSCAR database does not indicate how important each of these sub-domains is for generating accurate weather forecasts. To overcome this limitation, expert elicitation from three American and European leading NWP institutions was conducted. Each expert was asked to rank and comment on the relative importance of each weather sub-domains. Results are reported in Table 4.5, where a score of 5 indicates maximum importance and a score of 0 indicates that this type of data is not used. Several conclusions can be reached:

- Atmospheric characterization measurements are clearly the most important data product since they can be used to directly estimate the temperature and humidity profiles at different atmospheric levels. Furthermore, these measurements are currently used to derive wind profiles using temperature gradients (also known as termal winds), as well as first-order estimates for clouds structure and composition using atmospheric humidity estimates [I9].

- Atmospheric wind characterization is consistently ranked as the second most important type of data product since it helps characterize the dynamics of the atmosphere.

- Cloud characterization is currently considered a secondary product for global forecasting. In fact, some centers do not use clouds at all for global weather forecasting purposes [I8]. Other centers are currently experimenting with assimilating cloud imagery, albeit concerns were raised about the ability to obtain meaningful 3D cloud information from 2D images taken from geosynchronous orbit [I7].

- Atmosphere composition (both ozone and aerosol composition) is typically considered an optional data product for weather forecasting purposes. In fact, the NCEP does not always utilize them [I7], while Meteo-France does not use them at all [I8].

- Finally, surface measurements are not typically used for weather forecasting except for the prediction of specific variables such as minimal temperature in a region [I9]. They

Table 4.5: Relative Importance of NWP Sub-domains

NWP center	**Atmospheric charact.**	**Atmospheric chemistry**	**Atmospheric winds**	**Cloud charact.**	**Aerosols& radiation**	**Surface charact.**
ECMWF	5	1	4	2	1	0
ECMWF	5	1	3	4	2	0
Meteo-France	5	0	3	0	0	0
NCEP	5	2	3	4	2	0
Relative Importance	38%	7%	25%	19%	9%	0%

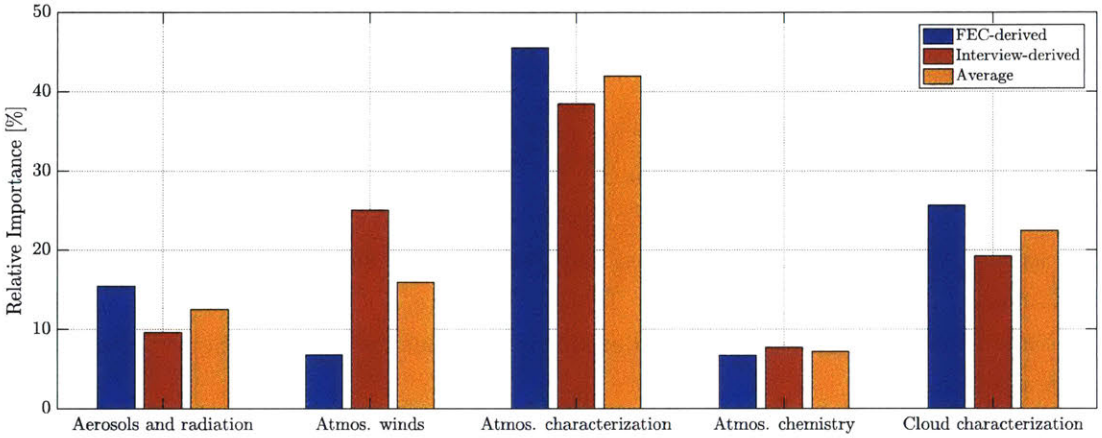


Figure 4-9: Relative importance of NWP Sub-domains

are also used for computing climatological statistics off-line and are consequently not affected by latency [I8].

Since the interview sample size for this case study was limited, other sources of information were used to ensure that the weights from Table 4.5 are accurate. In particular, I used Reference [157] and Figure 4-2 to estimate the relative importance of different sub-domains by translating the instrument-related error to a sub-domain related error using the measurement to instrument mapping obtained through the OSCAR database. In that sense, Figure 4-9 plots the obtained weight using both methods, along with their average. Note that in general there is good agreement between the expert-elicitated weights and the weights derived from the current error forecast sensitivity. The only significant difference is the importance of atmospheric winds, which seems to be overestimated by experts. In fact, the weights estimated through the FEC method do not contain any measurements from scatterometers other than those flown on-board the MetOp satellites. These are known to have much lower resolution than other dedicated scatterometers flown on-board dedicated spacecraft (e.g. QuickScat) [I9]. Therefore, I expect this method to underestimate the importance of wind characterization for accurate weather prediction.

Measurement to Instrument Mapping

Once the measurements that are important for NWP have been identified, the next step is to assess which satellite instruments are and will be used to generate them. To that end, the OSCAR database is again utilized to qualitatively assess the ability of a given instrument to generate a certain type of measurement. Note that this qualitative assessment is based on expert opinion captured in the form of a 0 to 5 scale, where a 0 indicates that a given instrument does not provide any relevant measurements to derive a certain data product (and vice versa). For instance, a generic spectro-radiometer such as MODIS is especially important for obtaining atmospheric temperature, cloud cover, Earth surface albedo or sea ice cover. It also provides secondary information for other data products such as horizontal wind profiles at surface or atmosphere specific humidity. In contrast, a highly specialized instrument such a scatterometter is basically designed and optimized to provide one type of information, wind profile at surface.

Twenty-two instruments have been analyzed using the OSCAR database in order to quantify their importance when generating measurements related to the six sub-domains identified in Section Step 4-2 (see Table 4.7 for a summary of their characteristics, as well as other instruments that provided or will provide the same information in past and future space programs). These include current instrument flown on-board GOES, POES, MetOp and Meteosat satellites, as well as future instruments that will be deployed with the new generation of JPSS satellites and their European counterparts. In general, four types of instruments are used to gather NWP data: MW and IR sounders, optical imagers, radiometers and GPSRO devices. Table 4.6 details the result of this mapping by providing a weight that indicates the importance of a given instrument when obtaining measurements of a given NWP sub-domain. The provided score for instrument i and sub-domain d is computed as

$$w_{i,d} = \frac{\sum\limits_{\forall m \in d} SC_{i,m}}{\sum\limits_{\forall j} \sum\limits_{\forall m \in d} SC_{j,m}} \tag{4.1}$$

where $SC_{i,m}$ is the 0-5 qualitative score in the OSCAR database for measurement m from sub-domain d. Note that this process is analogous to the value decomposition process in Reference [26], where measurements are assumed to all have the same relative importance. Therefore, $w_{i,d}$ quantifies the relative importance of instrument i in obtaining measurements for NWP sub-domain d. Observe, for instance, that the GPSRO devices are basically used to derive atmospheric characteristics, while scatterometers are used to derive atmospheric winds. On the other hand, more capable instruments like MODIS or VIIRS are able to gather measurements that influence multiple sub-domains at the same time.

Table 4.6: Instrument to Sub-domain Mapping. Each entry defines $w_{i,d}$.

		NWP Sub-Domain				
Instrument	**Type**	**Atmos. charact.**	**Atmos. chemistry**	**Atmos. winds**	**Cloud charact.**	**Aerosols& radiation**
ABI	Optical imager	17%	30%	1%	1%	10%
AIRS	IR sounder	6%	0%	11%	17%	2%
AMSRE	MW imaging radiometer	0%	11%	1%	0%	4%
AMSUA	MW sounder	2%	0%	5%	0%	9%
AMSUB	MW sounder	0%	0%	4%	0%	7%
ASCAT	Scatterometer	0%	30%	0%	0%	0%
ATMS	MW sounder	2%	0%	8%	0%	11%
AVHRR	Optical imager	6%	0%	1%	0%	5%
CrIS	IR sounder	8%	0%	11%	17%	5%
GRASS	GNSS radio-occultation	0%	0%	11%	0%	0%
HIRS	IR sounder	3%	0%	7%	3%	5%
IASI	IR sounder	8%	0%	14%	24%	5%
MHS	MW sounder	0%	0%	4%	0%	7%
MODIS	Multi-purpose imager	22%	15%	10%	1%	10%
OMPS	Ozone sounder	7%	0%	2%	36%	0%
SSMI	MW imaging sounder	1%	7%	6%	0%	11%
VIIRS	Optical imager	16%	7%	1%	0%	9%
CERES	Radiometer	0%	0%	4%	0%	0%

Instrument to Satellite Mapping

As previously mentioned, I will assume that the instrument to satellite mapping is outside the scope of this study and has already been performed. In other words, given the six Earth observation satellite programs from Table 4.2, I assume that their instrument allocation is pre-defined and fixed. Therefore, the aim of this section is to compute the importance of a data flow from a given satellite s given the set of measurements that can be derived from the observations it takes. In that sense, I compute the importance of a satellite program as the normalized sum of all instruments carried by the satellite:

$$w_s = \sum_{\forall i \in s} \underbrace{\sum_{\forall d} w_d w_{i,d}}_{\text{Instrument relative importance}} . \tag{4.2}$$

Subscript i is used here to denote instrument, while subscript d is related to the NWP sub-domain. Consequently, w_d indicates the relative importance of each NWP sub-domain as provided in Table 4.5. Note that since all data destination is, in this case, the NWP center, w_s is equivalent to w_p from Equation 2.12.

The obtained results for Equation 4.2 are reported in Table 4.8. As expected, the three

Table 4.7: Summary of NWP-related Instruments

	Instrument	Type	Satellites	Mass [kg]	Power [W]	Data Rate	Horz. Res.	Vert. Res.
Legacy	AMSR-E	MW imaging radiometer, conical scan.	Aqua	314	350	87.4kbps	5.4-56km	-
	AMSU-A	MW sounding radiometer, x-track scan.	MetOp-A/B, Aqua, NOAA-18/19	50	24	1.1kbps	48km	-
	AMSU-B	MW sounding radiometer, x-track scan.	NOAA-17 and before	50	90	60kbps	16km	-
	HIRS /4	Cross-nadir scanning IR sounder	MetOp-A/B/C	35	24	2.88kbps	26km	-
	IGOR	GNSS radio-occultation sounder	COSMIC	4.6	16	17kbps	300km	0.5km
	IMAGER	Moderate-resolution optical imager	GOES-12-15	140	130	2.62Mbps	4km	-
	AIRS	Cross-nadir scanning IR sounder	Aqua	177	220	1.27Mbps	13.5km	1km
	CERES	Broad-band radiometer	Terra, Aqua, TRMM	57	50	10kbps	30km	-
	OMI	Nadir near-UV/Vis. spectrometer	Aura	65	66	0.8Mbps	12km	-
	MODIS	Moderate-resolution Imaging Spectro-rad.	Terra, Aqua	250	225	6.2Mbps	1km	-
	SOUNDER	Cross-nadir scanning IR sounder	GOES-12/15	152	93	40kbps	8km	-
Current/Near Future	ABI	Moderate-resolution optical imager	GOES-R	338	450	66Mbps	2km	-
	ASCAT	Radar scatterometer	MetOp-A/B/C	260	215	42kbps	12.5km	-
	ATMS	MW sounding radiometer, x-track scan.	NPP, JPSS-1/2/3/4	75	130	20kbps	16km	-
	AVHRR/3	Moderate-resolution optical imager	NOAA-18/19, MetOp-A,B,C	33	27	621.3kbps	1km	-
	CrIS	Cross-nadir scanning IR sounder	JPSS-1/2	175	245	1.9Mbps	14km	1km
	GRASS	GNSS radio-occultation sounder	MetOp-A/B/C	30	30	27kbps	300km	0.5km
	IASI	Cross-nadir scanning IR sounder	MetOp-A/B/C	236	210	1.5Mbps	18km	1km
	MHS	Microwave Humidity Sounding	NOAA-18/19, MetOp-A/B/C	63	93	3.9kbps	16km	-
	OMPS	Limb-scanning sounder	JPSS-1/2	68	108	165kbps	300km	2.2km
	CERES-FO	Broad-band radiometer	JPSS-1/2	54	55	10kbps	20km	-
	WindSat	MW imaging radiometer, conical scan.	Coriolis	307	311	256kbps	12.5km	-
	SSM/I	MW Imager & Sounder	DMSP	96	135	14.2kbps	12.5km	-
	VIIRS	Moderate-resolution optical imager	NPP, JPSS-1/2/3/4	275	240	5.9Mbps	750m	-
Future	3MI	Moderate-resolution optical imager	MetOp-SG-A1/2/3	60	80	6.5Mbps	4km	-
	IASI-NG	Cross-nadir scanning IR sounder	MetOp-SG-A1/2/3	360	500	6Mbps	25km	-
	Sentinel-5	UV, Visible and Near-IR Sounder	MetOp-SG-A1/2/3	250	220	20Mbps	7km	-
	MetImage	Moderate-resolution optical imager	MetOp-SG-A1/2/3	262	150	20Mbps	1km	-
	MWS	MW sounding radiometer, x-track scan.	MetOp-SG-A1/2/3	132	137	30kbps	17km	-
	RO	GNSS radio-occultation sounder	MetOp-SG-A1/2/3	22	30	1Mbps	300km	0.5km

Table 4.8: Program Relative Importance w_s

Program	Instruments	Weight
JPSS[6]	ATMS, CERES, CrIS, OMPS, VIIRS	22%
MetOp	AMSU-A, ASCAT, AVHRR/3, GOME-2, GRASS, HIRS/4, IASI, MHS	34%
EOS	AIRS, AMSR-R, AMSU-A, CERES, MODIS	25%
DMSP	SSMI	6%
GCOM	AMSR-2	4%
GOES	ABI	10%

primary LEO weather programs (JPSS, MetOp and NASA EOS) contribute to 80% of all satellite data collection for NWP purposes. Note that this relative importance is not necessarily representative or correlated with the total amount of data the mission is returning. Indeed, GOES ABI imager is the most data intensive instrument in the analyzed set. Yet, its importance for weather forecasting is moderate when compared to data products from sounders on-board MetOp and JPSS satellites.

Finally, observe that the MetOp program has approximately 10% more importance than similar US programs. The primary reason for this finding is the inclusion of GRASS on-board MetOp satellites, a radio-occultation device that determines the state of the atmosphere by measuring changes in GNSS radio signals that propagate through it. In that sense, radio-occultation has the ability to provide exceptional vertical resolution (e.g. 0.5km for GRASS), which complements good horizontal resolution by traditional sounders. Therefore, by combining both types of measurements from a single platform, it is possible to obtain an improved estimate of the atmosphere's 3D structure, which enhances the NWP assimilation process and ultimately results in better forecasting ability.

Step 4-2.3. Characterization of Data Utility

Given that satellite data has been proven to be the most important data source for weather forecasting systems, I now detail how late delivery of these data products hiders the ability of NWP to successfully assimilate them. In that sense, it is necessary to have generic understanding of how operational forecasts are structured. Figure 4-10 provides a schematic representation of the assimilation time line for both a global and a local forecast system [169]. The global assimilation system is run 4 times per day (00, 06, 12, 18UTC), while the local assimilation and forecast systems are run every 3 hours[7]. Data from previous runs, referred to as *background*, is used to initialize the assimilation system and, if necessary, provide the atmosphere boundary conditions. The assimilation process matches this background information with the new atmospheric observations received before the system cut-off time.

[6] Includes Suomi-NPP spacecraft.

[7] Current operational local forecast systems are typically run on an hourly basis.

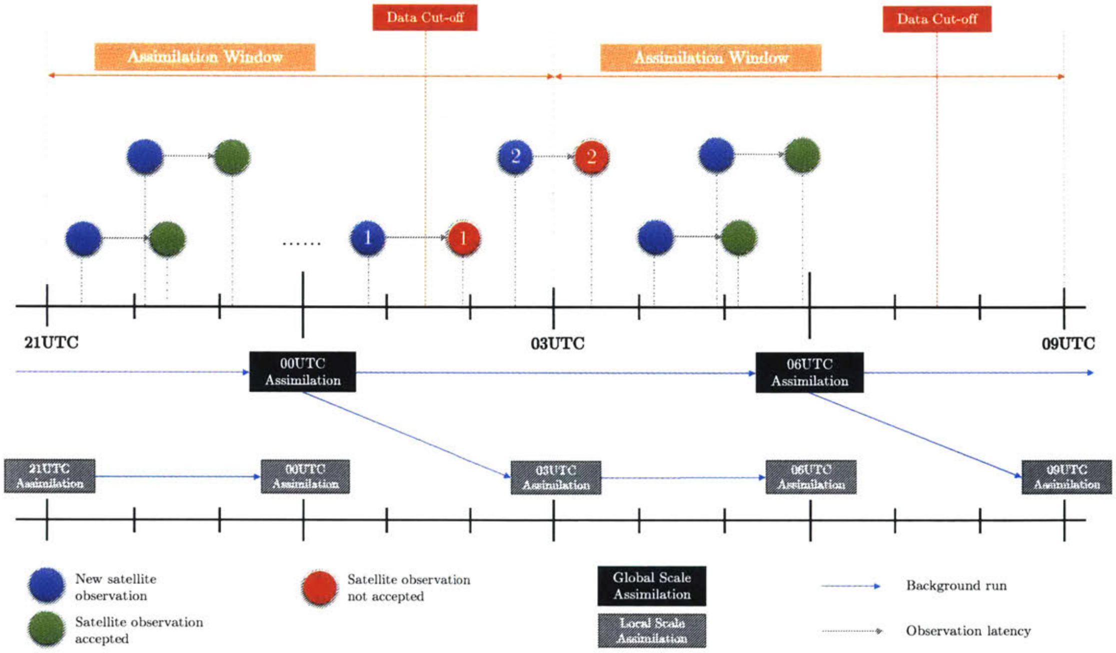


Figure 4-10: Notional Assimilation Timeline

The derived product is then used to feed the forecasting system, which essentially propagates the state of the atmosphere forward in time in order to obtain the final forecast.

Theoretically it would be optimal to run forecast models at all scales continuously so that new atmospheric observations are assimilated as they arrive. Nevertheless, in reality this is not possible due to high computational costs in the assimilation process, as well as latency with which satellite observations are received. As a result, currentNWP systems have complex time lines that balance the trade-off between data and forecast availability. Indeed, on one hand NWP centers want to wait as much as possible before running the assimilation process to ensure that as much data as possible on the state of the atmosphere has been gathered. On the other hand, if they wait too much then the forecast is no longer useful to the final user and therefore the entire production system is useless.

A balance is currently reached by which assimilation systems at different time scales are run with different periodicity and their data products are optimally intertwined. This complex scheme is difficult to understand, especially since extra tweaks such as early-decision runs are currently added to the system to meet specific customer demands. To simplify the problem, two parameters are typically used to define the time line of an assimilation system: assimilation window and data cut-off (see References [167] and [169]). The assimilation window defines the periodicity or cadence with which an assimilation system is run. In turn, the data cut-off time defines the maximum amount of time a NWP center waits for delayed observations to arrive. Figure 4-10 depicts the assimilation window and data cut-off time for a fictional assimilation system where a global and local forecast systems are intertwined. Note that observations 1 and 2 are effectively discarded in the assimilation system since they are delivered outside the data cut-off time for the 00UTC assimilation run and they are outside the 06UTC assimilation window.

The combination of data assimilation window and data cut-off time is used in Reference [169] to quantify what percentage of data generated by satellite systems is available at a NWP when the assimilation system is triggered. I use this notion of *percentage of data available* as the utility function for this case study, calibrated assuming a 6 hour assimilation window and a cut-off time of 1 hour and 15 minutes. These values are representative of NOAA's medium-range GFS during the JPSS era [I7], [169], as well as ECMWF mid-range forecasting [I9] and Meteo France's assimilation process [I8]. Note that, as previously mentioned, most forecasting centers have a complex dual scheme for assimilating observations where some runs are performed with longer ($\geq 6h$) data assimilation windows, while the others limit the data cut-off time to implement "early warning systems". This dual approach has been adapted to mitigate the impact of latency and would not be necessary should all the observations be delivered in a timely manner [I8], [170]. Therefore, we utilize the cut-off times for these early warning systems as they represent the aspirational goal that NWP centers set.

Figure 4-11 plots the resulting utility function $U(L)$ for a global or medium-range forecasting system. Perfect utility indicates that all data gathered prior to the assimilation's window cut-off time is delivered to the NWP processing facility prior to starting the assimilation process. Two main takeaways can be obtained from it: First, perfect utility is theoretically impossible since it would require a network with infinite bandwidth and perfect coverage. However, a system that delivers data with $\approx$ 15 minutes latency (such as the NPOESS program) would have a utility of 95% approximately, while a system with a latency requirement of 90 minutes (such as JPSS) would have a utility just over 60%. Second, the obtained utility function is linear, which vindicates the use of Equation 2.9 to express the system architecting centrality measure.

Step 4-2.4. Definition of Normalization Scheme

All rankings in this case study will use sum normalization so that they indicate the relative importance of a given latency contributor in the overall end-to-end system.

Step 4-3. Ranking of Latency Contributors

Once the centrality measure has been specified, computation of the ranking of latency contributors using Equations 2.9 and 2.12 is immediate. Table 4.9 provides the DSM that defines which elements of the system interact with each other and induce latency. Observe

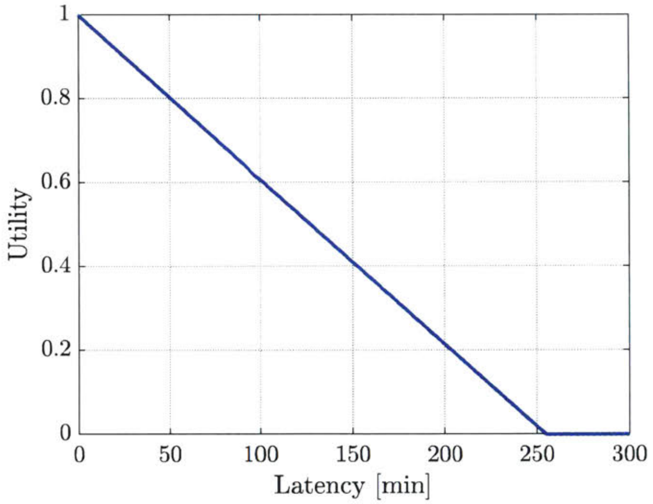


Figure 4-11: Utility Function for Weather Data

Table 4.9: Adjacency Matrix for the NWP End-to-End System

	JPSS	MetOp	EOS	DMSP	GCOM	GOES	GN	WAN	PF	DN
JPSS	▨	0	0	0	0	0	1	0	0	0
MetOp	0	▨	0	0	0	0	1	0	0	0
EOS	0	0	▨	0	0	0	1	0	0	0
DMSP	0	0	0	▨	0	0	1	0	0	0
GCOM	0	0	0	0	▨	0	1	0	0	0
GOES	0	0	0	0	0	▨	0	0	1	0
GN	0	0	0	0	0	0	▨	1	0	0
WAN	0	0	0	0	0	0	0	▨	1	0
PF	0	0	0	0	0	0	0	0	▨	1
DN	0	0	0	0	0	0	0	0	0	▨

that most of the complexity has already been abstracted by defining four canonical nodes (GN, WAN, PF and DN) that aggregate latency contributors that are intricately related.

Figure 4-12 ranks the different latency contributors for the end-to-end NWP system (see Table 4.10 for the exact numbers). Each bar height indicates the estimated relative importance for a given latency contributor, while color separations within a bar quantify the relative weight of a given satellite program within that latency contributor. Importantly, observe that the satellite system is in this case responsible for almost 70% of the end-to-end system latency. This includes both the time it takes for a satellite to be in view from a ground station, as well as the time to acquire the image. Furthermore, since JPSS, MetOp and EOS satellites have similar capabilities and carry the primary instruments that provide observations for weather forecasting, they combine to cause more than 60% of the total utility loss in the system. Finally, observe also that capacity of neither the downlink nor the ground lines is in fact a major latency contributor for current NWP systems. In fact, results indicate that resources would be better spent upgrading the processing facilities that

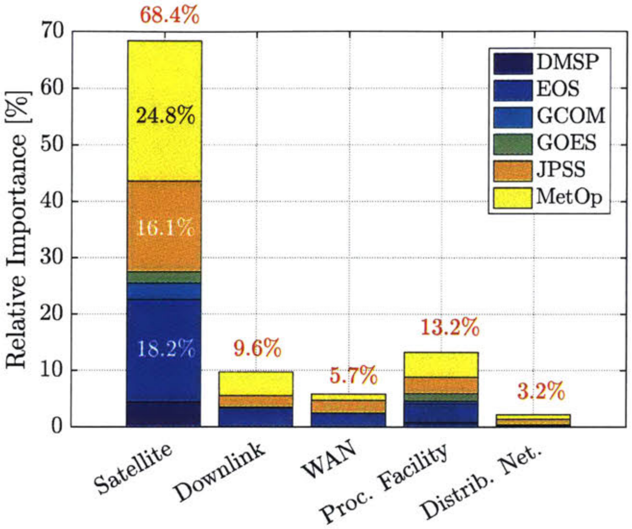


Figure 4-12: Ranking of Latency Contributors

Table 4.10: Relative Importance of Latency Contributors

	DMSP	EOS	GCOM	GOES	JPSS	MetOp	TOTAL
Satellite	4.4%	18.2%	2.9%	2.0%	16.1%	24.8%	68.4%
Downlink	0.1%	3.3%	0.1%	0.0%	2.0%	4.2%	9.6%
WAN	0.1%	2.3%	0.1%	0.0%	2.2%	1.1%	5.7%
Proc. Facility	0.8%	3.3%	0.5%	1.3%	2.9%	4.4%	13.2%
Distrib. Net.	0.1%	1.1%	0.1%	0.1%	1.0%	0.8%	3.2%
TOTAL	5.4%	28.2%	3.6%	3.4%	24.1%	35.3%	100.0%

generate the required L1 data products before investing in new technologies to increase the downlink data rate between satellites and ground stations.

Step 4-4. Problem Formulation

Step 4-4.1. Definition of Case Study Assumptions and Goals

Given that line of sight visibility has been identified as the primary latency contributor in the system, I now conduct a detailed analysis of the trades between infrastructure cost, risk and line of sight latency. In particular, the specific set of goals addressed by this part of the latency-centric approach to architecting space communication networks include:

1. Quantify the trade-off between performance, cost for ground communication networks and compare them with alternative space-based networks, as well as current and future proposed systems.

2. Demonstrate how to quantify immaterial risk factors such as political instability or anti-US sentiment using a risk-adjusted cost of capital derived from expert elicitation.

3. Quantify the trade-off between performance and risk in ground networks and compare them against current and planned infrastructures.

During the development of this analysis the following assumptions will be utilized:

- All analyzes will be based on the 2020-2030 era and the predicted space-based capabilities for that decade.
- I only consider global and mesoscale forecast systems that produce forecasts for up to 10 days. I explicitly exclude local forecast systems even though they also have stringent latency requirements because I assume that their needs can be met by the already implemented direct-broadcast systems on-board weather satellites such as GOES and MetOp. Similarly, forecast systems for time-scales greater than 10 days are similar to climatology studies for which latency is not considered a significant issue.
- The NWP assimilation windows and cut-off times are fixed as indicated in Step 4-2.

Step 4-4.2. Definition of Architectural Space

The primary architectural decision that defines the space of possible network configurations is a selection problem over a pre-defined set of ground stations. Since this is a retrospective case study, this set of ground stations has been selected based on the NPOESS SafetyNet implementation, as well as JPSS's CGI. In that sense, Figure 4-13 exemplifies the line of sight latency contributor for the SafetyNet infrastructure. In this case, latency is computed by assuming that a sun-synchronous satellite takes measurements continuously and downlinks them as soon as visibility with a ground station is acquired. Observe, for instance, that the SafetyNet system had line of sight latency values in the order of 15 to 20 minutes for most parts of the globe, except for the Indian Ocean where latency is as high as 60 minutes in some cases.

Table 4.11 summarizes the set of candidate ground sites assumed for this case study, as well as their location. The architectural problem is mathematically formulated as a selection problem over a predefined set of N sites, where the architecture is encoded using an N-element binary vector $\mathcal{A} = \{0,1\}^N$. A total of 65536 network architectures are possible, which I evaluate both in performance, cost and risk and optimize using a multi-objective genetic algorithm. This allows me to identify the Pareto Front without exploring the architectural space exhaustively, which would be computationally intractable.

Finally, four primary architectures will be used as reference points when analyzing the results:

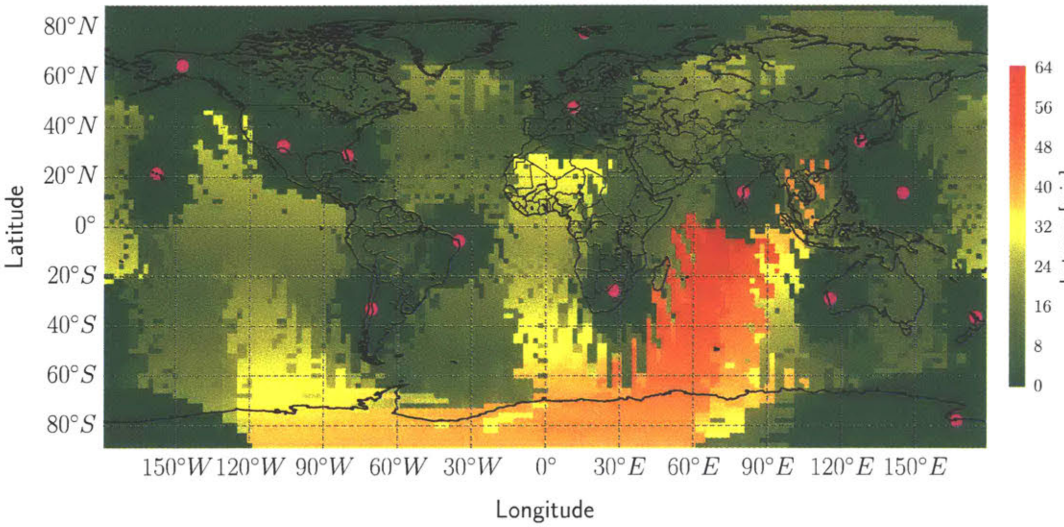


Figure 4-13: SafetyNet LOS-induced Latency

Table 4.11: NPOESS and JPSS Candidate Ground Sites

Site	Latitude	Longitude	Country	Polar	CONUS
McMurdo	-77.84	166.67	Antarctica	1	0
Troll	-72.02	2.53	Antarctica	1	0
Guam	13.62	144.86	USA	0	0
Fairbanks	64.97	-147.52	USA	1	0
Hawaii	21.32	-157.89	USA	0	0
Hartebeesthoek	-25.88	27.70	South Africa	0	0
Svalbard	78.23	15.40	Norway	1	0
Dongara	-29.05	115.35	Australia	0	0
Warkworth	-36.43	174.66	New Zeland	0	0
Naro	34.43	127.54	South Korea	0	0
Weilheim	47.88	11.09	Germany	0	0
White Sands	32.50	-106.61	USA	0	1
Santiago	-33.15	-70.67	Chile	0	0
Sriharikota	13.67	80.20	India	0	0
Kennedy	28.52	-80.65	USA	0	1
Barreira	-5.93	-35.16	Brazil	0	0

- Baseline architecture: It is representative of today's ground infrastructure and is modeled as a single polar ground stations in the Northern hemisphere, namely Svalbard.
- JPSS architecture: It is composed of four main polar ground stations, two in each hemisphere.
- SafetyNet architecture: It is composed of 15 ground stations across all continents plus two northern polar sites and one in Antarctica.
- The TDRSS system: A set of geosynchrounous satellites that provide communications on a continuous basis to LEO spacecraft. It is used as a reference network architecture for a space-based system rather than a ground-based system in order to compare both alternatives.

Figures of Merit

Several FOMs are of interest when considering communication networks that support NWP systems. Table 4.12 lists them and indicates where in this chapter they are analyzed. Furthermore, each FOM is classified as primary or secondary depending on whether is was optimized during the system architecting exercise. Next, a brief description for each of them is presented:

- Performance: As explained in Step 4-2, the value of data in latency-constrained applications depends on the timeliness with which it is delivered to the end user. Therefore, the performance of the system will be measured with respect to the overall utility, which in turn captures the percentage of satellite data available at the NWP center prior to the start of the assimilation process.
- Cost: It will be measured based on the total life cycle cost of a ground (or space) network during 15 years. Costs included in the study will include the procurement of facilities and antennas as well as their operations and maintenance.
- Risk: It will be measured through a calibrated cost of capital that quantifies exposure to intangible risks such as political and social uprising, antenna expropriation or sensitivity to natural disasters.
- Scalability: It will be measured as cost per unit of capacity, where capacity is expressed as the ability to provide a 10 minute pass for any given mission.
- User burden: It will be measured by the network sensitivity. This FOM quantifies the facility with which a return link between any given user and the network can be closed. It is relevant in the context of latency-constrained applications since the critical data path is always on the return channels that downlink information from the spacecraft to the ground.

Table 4.12: Figures of Merit for the NWP System

Objective	Metric	FOM Type
Performance	Data utility	Primary
Cost	Life cycle cost	Primary
Political, social, financial risk	Cost of capital	Primary
Scalability	Cost per pass	Secondary
User burden	Network sensitivity	Secondary

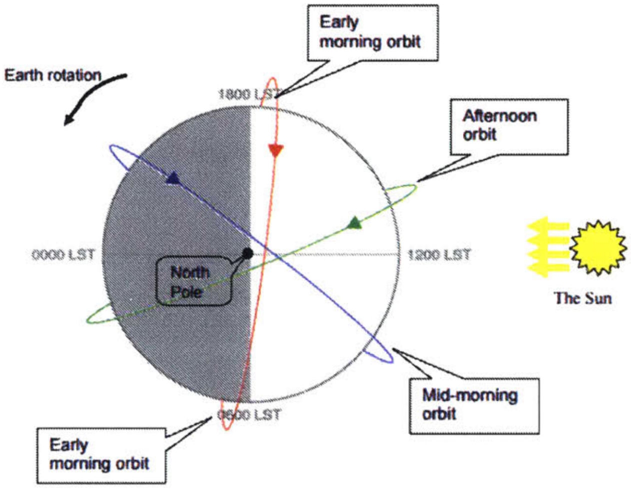


Figure 4-14: Polar Orbits for NWP Satellites

Step 4-4.3. Model Development and Validation

Performance Model

To estimate the latency induced by lack of line of sight, I simulate a SSO satellite for a period of 32 days. The satellite carries a sensor with rectangular field of view that images the Earth in a 16 day ground repeating cycle. Furthermore, the satellite is placed on an early-morning orbit (i.e. 6am Local Time of the Descending Node). Note that both the JPSS program and its partners typically place their satellites in two other orbits, mid-morning (9:30am LTDN) and afternoon (13:30pm LTAN), albeit no significant differences in latency estimation where observed when considering them instead of the morning orbit (see Figure 4-14).

Once the satellite's orbit has been propagated, I determine the instants in time at which the sensor images a particular point of the Earth surface based on a spherical grid with 4 degrees of resolution at the equator. Next, I compute the access time between the satellite and the network of ground stations assuming that no scheduling constraints restrict the spacecraft passes. To ensure that this constraint is not significantly violated, it will be assumed that all ground sites have at least two antennas ready for operation with the space segment.

Finally, I quantitatively assess the latency due to lack of visibility for a given latitude and longitude as the time between its imaging by the spacecraft's sensor and the next pass over any ground station of the network[8]. The result of this process can be graphically visualized through latency maps such as Figure 4-13.

The performance model is implemented using a combination of software technologies. First, coverage and access information is obtained using STK [38]. Given the large number of possible network architectures, it is clear that commanding STK manually is not a feasible option. Therefore, I developed a simplified STK-Matlab interface module using STK's COM technology. It allows the user to set up scenarios, add satellites, ground stations, sensors, and any other STK object required to estimate latency. Similarly, the interface also provides functions that obtain the simulation results from STK and process them directly in Matlab. Finally, once the latency maps for a given network architecture had been numerically estimated, Python's Basemap library [171] was utilized to visualize the results.

Finally, the performance score for a given architecture is evaluated by direct substitution of computed latency into the utility function from Figure 4-11. Indeed, since the other parts in the end-to-end system implementation remain constant and $U(\cdot)$ is linear, the performance score obtained through this approach will be correct for relative comparisons. In other words, a utility score of 0.5 should not be interpreted as 50% of the data is delivered in time for assimilation at the NWP center as in reality other latency contributors will further delay the data. Yet, the difference in performance between a system architecture that provides 50% or 30% utility score remains meaningful.

Cost Model

Following the guidelines of References [43] and [172] I model the cost of a space-to-ground RF network using a top-down approach based on a WBS. For any given ground station, this WBS contains the following items:

1. Mechanical antenna: Includes the cost of the stationary and moving parts of the antenna, including the control mechanisms that allow it to track a spacecraft.

2. Antenna electronics: Includes all electronics required to support RF, IF and baseband signal processing, as well as generation of frequency and timing signals. The high-power transmitter and low-noise amplifiers are included in this category.

3. Antenna supporting equipment: Includes installations of power, water, surveillance systems, heating and ventilation, as well construction of roads and trenches for cables from the antenna to the complex signal processing center.

[8] Given the 32 day time period used in the simulation, the number of images for a given latitude/longitude has a median of 10 observations and a minimum of 2.

4. Signal Processing Facility: Includes the building in which the signal processing electronics for demodulation, decoding and packetization are encloses. It typically also contains a control room to monitor the real-time status of all the site antennas.

5. Site programmatics: Includes management and systems engineering, as well as assembly and testing.

6. Site Operations: Includes operations and maintenance of the antenna mechanical parts, as well as the antenna and signal processing center electronics.

7. Wide area network (WAN): It will be assumed that the JPSS program contracts a cloud-like WAN service to a commercial entity like AT&T with a baseline bandwidth of 150Mbps.

Note that elements 1-5 are non-recurring costs that will only be incurred once during the construction of the antenna. In contrast, elements 6-7 are recurring and will be incurred every year of operations. Consequently, its discounted present value is utilized in order to obtain the total system life cycle cost. Next, describe the specific parts of this cost model WBS.

Mechanical Antenna

The primary factor that drives the cost of a mechanical antenna is its diameter. Reference [173] provides evidence of the non-linear relationship between antenna mechanical costs and its diameter by fitting an exponential model based on historical data. Similar results have been reported in studies that model ground antenna arrays for communication and astronomy (e.g References [174] and [175]). Based on these empirical findings, the cost estimating relationship for a mechanical antenna has been defined as

$$C_A = kD^{\gamma} \tag{4.3}$$

where k and γ are constants to be determined empirically. Based on the aforementioned references, γ's value is estimated between 2 and 2.7, with a typical value of 2.4. Observe that this indicates that the antenna cost scales faster than its diameter, or equivalently, the cost per unit of signal collective area increases faster than the antenna size. Indeed, the pedestal for NASA's 70m antenna is more massive and complex than that of a new BWG 34m antennas (especially the bearings that allow the antenna to rotate in the azimuth axis), thus vindicating the empirical finding $\gamma > 2$. On the other hand, k is a parameter that changes over time and translates the antenna cost diameter dependency to a given dollar value. For the purposes of this thesis, k has been estimated based on a \$23M FY2012 construction cost for DSN's DSS-35 antenna. Figure 4-15 provides a visual representation of the cost of building a ground antenna as a function of diameter, normalized to the reference

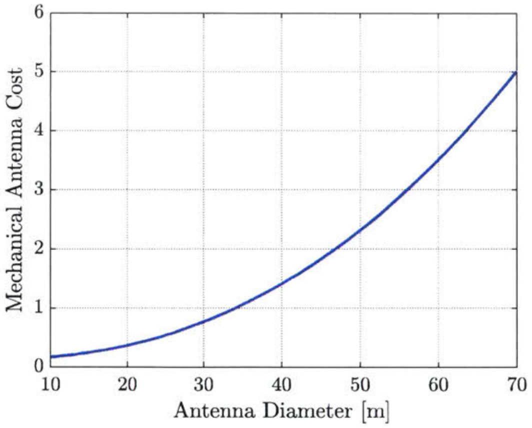


Figure 4-15: Antenna Mechanical Cost

34 meter dish. Note that typical antenna diameters for antennas that support near Earth spacecraft are on the order of 10 to 13 meters and therefore are priced at 15 to 20% the cost of a 34 meter antenna.

Antenna Electronics

The antenna electronics comprise four areas: Frequency and timing, antenna feed and low noise amplifier, transmitter high power amplifier and arraying equipment. Their costs have been estimated based on Reference [175] and transforming the obtained value to FY2010 dollars. In particular, the cost of all electronics except for the HPA is constant and assumed to be approximately $240k per antenna. This estimate does not include the cost of correlators for antenna arraying since this functionality is not typically implemented in near-Earth communications and is therefore outside the scope of this case study. On the other hand, the cost of the high power amplifier is estimated using equation

$$C_{HPA} = 0.1 + 0.3\sqrt{P} \tag{4.4}$$

where, C_{HPA} is expressed in FY2010 million dollars when P is in kilowatt units. Based on the specification of the current Near Earth Network [176], I assume that P is 200W for the type of antennas under consideration.

Antenna Supporting Equipment

The antenna supporting equipment comprises all utilities that need to be provided for the antenna to deliver its function. They have been estimated using Reference [175] and are constant regardless of the type of antenna. Combined, they are estimated at $480k per antenna in FY2010 dollars.

Signal Processing Facility

The signal processing facility represents the cost of building an on-site operations room where electronics can be securely installed. Larger antennas such as DSN's 34 meter and 70 meter antennas place part of the receiving front-ends directly under the antenna in order to maximize the antenna sensitivity. However, once the signal has been discretized it still has to undergo large processing (demodulation, decoding, packetization) before being deliverable to the end-user. The electronics for these functions are placed inside the signal processing facility. Finally, in some cases such as the DSN, the signal processing facility might also contain an on-site control room to monitor the real-time status of the antennas and rapidly resolve operational problems.

The cost of the signal processing facility has been estimated using the DoD Facility Cost model [177]. It estimates both the non-recurring and recurring costs of a facility as a function of a unitary cost, a total construction area, and a set of factors that correct the estimates based on the site location.

Site Programmatics

Site programmatics include management and systems engineering, as well as integration and testing of the antennas and the signal processing center. Based on Reference [43], they have been estimated as a fraction of the site non-recurring cost. Since the procurement of ground sites is less risky and uncertain than the procurement of space assets, we will assume that these fractions are in the low-to-moderate range, with a value of 5% and 10% respectively.

Site Operations

Site operations are decomposed in three categories: Maintenance and operations of the mechanical parts of the antenna; maintenance and operations of the antenna electronics; and maintenance of the signal processing facility. The first two categories are estimated based on Reference [175] assuming 17 and 16 full-time employees (FTEs) respectively. The cost of an FTE has been calibrated to $75k per year approximately in order to match the operations of the DSN. Furthermore, it has been assumed that smaller antennas require less FTE to operate since they are easier to automate. In that sense, we assume that the 17 and 16 FTEs are representative of a 34m antenna, while a 10m antenna only requires 5 FTEs. Finally, the signal processing sustainment cost has been quantified using the DoD Facilities cost model, assuming a unitary cost of $13 per square feet approximately.

Wide Area Network

The cost of the WAN has been estimated based on 40 data points from NASA's SCaN project. The functional form of the cost estimation relationship is a log-log linear model

$$\ln C_n = \alpha + \beta \ln R_b \tag{4.5}$$

where C_n is the location-normalized cost of the line, α and β are the parameters calibrated based on the historical data, and R_b is the capacity of the ground line. Finally, the line cost C_l is equal to

$$C_l = \gamma C_n \tag{4.6}$$

where γ is a parameter that depends on the location of the data origin and destination and has three possible values: $\gamma = 1$ for CONUS lines, $\gamma = 3$ for OCONUS lines (e.g. line between White Sands and Guam or Alaska and GSFC), and $\gamma = 10$ for international lines.

The exact values for the WAN cost model cannot be provided due to data privacy limitations. However, the resulting fitted equation has an $R^2 = 0.75$ after adjusting for location and is therefore assumed accurate enough for system architecting purposes. That being said, it must be noted that the proposed WAN model was estimated using data from 2010 and assuming Optical Carrier technology. Since then, most carriers have switched to Gigabit-Ethernet for their high capacity lines (GbE), a change that will most likely affect the coefficients of the regression.

Life Cycle Cost Estimation

Given the set of cost WBS presented in the previous sections, I now provide the life cycle cost model for the entire network. To start, I divide the costs of building and operating a site in:

- Non-recurring costs (C_{NREC,S_i}): They include the cost the mechanical antenna, antenna electronics, antenna supporting equipment, signal processing facility and site programmatics.

- First unit cost (C_{TFU,S_i}): It is equal to the non-recurring cost minus the cost of the HVAC and signal processing facility. These two elements are shared across all antennas in a ground site and therefore should only be accounted once.

- Recurring costs (C_{REC,S_i}): Cost of operating and maintaining the site, including the WAN that connects it to the system processing facility in CONUS.

Assuming that each site has N antennas, its construction cost is computed as

$$C_{cons,S_i} = \gamma_i \sum_{n=1}^{N} C_n \tag{4.7}$$

$$C_1 = C_{NREC,S_i} \tag{4.8}$$

$$C_n = C_{TFU,S_i}(1-L)^{\log_2 n} \quad 1 < n \leq N \tag{4.9}$$

where L is a learning factor assumed to be 5% and γ_i is a construction cost multiplier that depends on the site location. Finally, the cost of building M sites across the world is estimated to be the sum of construction costs for all sites without any learning factor since (1) construction of facilities is typically conducted by different local companies and (2) learning factors would interact with the construction cost factors which would obviously lead to nonsensical results.

On the other hand, the recurring costs for the network are simply estimated as the sum of recurring costs per site. For site S_i, the recurring cost is the sum of operations and maintenance for the site, plus the cost of contracting a 100Mbps line to continental US:

$$C_{O\&M,S_i} = \nu_i \left(C_{REC,S_i} + C_{WAN,S_i}\right) \tag{4.10}$$

Finally, in order to fully specify the proposed model I set the construction and sustainment cost factors γ_i and ν_i for the different candidate sites. Table 4.13 details them for the 16 ground sites under consideration and orders them from largest to smallest. As expected, placing ground stations in Antarctica is up to three times more expensive than in continental US or South America. Note that some assumptions were made when compiling these cost factors. For instance, Antarctica is not directly referenced in the DoD Facility Cost Model and therefore we used Greenland as a plausible replacement. Similar analogies where made for New Zeland and South Africa.

Cost Model Validation

Three data points where available for assessing the accurateness of the proposed cost model, two of them related to antenna construction and only one related to ground site operations. Table 4.14 summarizes them, along with the estimated values calculated with our cost model. Results indicate that, in general, there is a good agreement between the two of them, at least from the perspective of obtaining order of magnitude estimates. In that sense, observe that the price of a KSAT 10-12 meter mechanical antenna is estimated to be double the value reported in the literature. This can be due to the numbers provided in the reference not being accurate due to marketing purposes, as well as possible differences in the costing

Table 4.13: Construction and Sustainment Factors

Site	Country	Construction Factor	Sustainment Factor
McMurdo	Antarctica	2.90	2.83
Troll	Antarctica	2.90	2.83
Svalbard	Norway	2.90	2.83
Guam	USA	2.54	2.41
Fairbanks	USA	2.34	2.32
Hawaii	USA	2.36	2.30
Hartebeesthoek	South Africa	1.86	1.64
Dongara	Australia	1.45	1.35
Warkworth	New Zeland	1.45	1.35
Naro	South Korea	1.12	1.03
Weilheim	Germany	1.09	1.02
White Sands	USA	0.99	0.98
Santiago	Chile	0.95	0.89
Sriharikota	India	0.93	0.87
Kennedy	USA	0.89	0.87
Barreira	Brazil	0.91	0.82

Table 4.14: Ground Station Cost validation

Cost Type	Site/Antenna	Truth	Model	Reference
Non-recurring	DSN 34-m antenna	$23M	$23.5M	[I20]
Non-recurring	KSAT mechanical 10-12m antenna	$1.15M	$3.5M	[178]
Recurring	Green Bank Telescope	$10M	$7M	[179]

WBS. In any case, since I will assume that all antennas have the same diameter and costs will be normalized to its unitary cost, this source of error will be eliminated and the provided results will be valid from a relative costing perspective.

Risk in Ground Networks

The need for including risk in this case study was originally motivated during private conversations with a SCaN management representative. In particular, he mentioned that due to the indispensable nature of the communication services provided by NASA's networks, it is fundamental to consider which risk factors can affect the operations of a given ground station [I20]. References [88] and [180] provide great historical examples to demonstrate the dangers of placing ground stations in politically unstable countries. Based on past experiences from NASA's Minitrack and Spacecraft Tracking and Data Acquisition Network (STDN), they summarize risk factors that have historically affected the deployment and use of remote ground stations. For instance, Reference [180] notes that "putting a station in South Africa, where political and human rights policies did not align with those of the

United States, required much diplomacy and perseverance. Conversely, other places such as Australia, where as many as 10 sites operated during Gemini and Apollo, enthusiastically embraced the opportunity to participate in this new frontier and to share in its exploits adopting the American space program as its own". Similarly, Reference [88] indicates that "during the pioneering flights of Project Mercury, the Guaymas tracking station in Mexico often had to be surrounded by troops to protect it against unruly mobs espousing anti US sentiment".

While political instability or anti-US sentiment are clearly intangible risks factors, they can have significant repercussions in the system's performance and cost. Indeed, not only critical event operations might be disrupted, but also the entire station might be abandoned as was the case for the 85-foot dish in Johannesburg, South Africa due to political pressures in 1975 [I20], [88]. Similar problems where faced in the Tananarive ground station in 1975, where a coup-d'etat by a Marxist regime abruptly ended 20 years of cooperation between the US and the Madagascar government. This resulted in significant operational risk for the Apollo-Soyuz program, as well as increased network operational costs due to the unexpected deployment of tracking ships [88]. The natural question is, therefore, which risks should be taken into account when valuating ground network, and how to quantify their effect for tradespace exploration purposes. Both points will be addressed in the following sections.

Risk Factors

Several risk factors should be considered for the purposes of valuing a space communications ground site. They were assessed based on References [181] and [182] and include:

- National security risk: For the purposes of this thesis, it refers to risks associated with unintended technology transfer and/or possible destruction of assets due to conflicting national interests/policy between the network owner and the host country.

- Political risk: They are associated with the political stability of the country where antennas will be placed. These include government stability, socioeconomic conditions, law and order, democratic accountability or corruption and bureaucracy quality.

- Expropriation risk: It refers to the risk of loosing a facility due to forceful seizure by the host country government. It is sometimes bundled with the financial risk category but has been explicitly separated due to the importance of this factor for ground network infrastructures.

- Project sensitivity towards wars, strikes or terrorism: It bundles religious tensions, as well as internal or external conflicts.

- Project sensitivity towards natural disasters.

- International cooperation risk: It refers to risks associated with undertaking projects with international partners and multilateral agencies.
- Technological and resource risks: Risks associated with the technology under consideration and the lack of resources to keep its operations.
- Financial and economic risks: Includes risks associated with load default, delayed payment by suppliers, losses from exchange controls, as well as risks associated with the country macroeconomic status (e.g. currency strength, country liquidity rations, etc.).

Note that in the case of national space communication networks some of the aforementioned risk categories are not necessarily applicable. For instance, since the system purpose is not meant to generate revenue it is unlikely to be affected by changes in the host country financial and macroeconomic situation. At the same time, they cannot be fully neglected since countries with with weak economies and higher social disparities are also more prone to social and political stability.

Risk Model

The problem of international project valuation with intangible risks is not typically addressed in the system engineering literature. For instance, NASA's Systems Engineering Handbook considers risk as a main factor to consider, but almost exclusively focuses on technical risks [183]. Indeed, non-technical risks such as political and expropriation are hard to quantify since no direct mathematical models can be easily built for them. Furthermore, our aim is to be able to quantify performance-cost-risk trade-offs for a system that will potentially extend across all Earth continents. Therefore, they should be readily applicable for a large body of countries and regions.

On the other hand, project valuation for international ventures has been a field of interest in economic and financial research for at least the last two decades. Indeed, as companies become increasingly global it has become more important to be able to compare projects undertaken in different parts of the world while making sure that risks inherent to them are properly quantified. In particular, assume that a commercial project has a constant stream of positive cash flows over time and we wish to compute their present value. Traditional project valuation indicates that the cash flows should be discounted at a rate r that captures both the time value of money and the opportunity cost of undertaking this project instead of another one. Mathematically, we would denote

$$r = r_f + r_p \tag{4.11}$$

where r_f and r_p quantify the two aforementioned effects respectively. It is widely accepted

Table 4.15: International Country Credit Rating

Risk Category	ICCR Range	Countries
Very high risk	0.0-49.0	Venezuela, Nigeria, Cuba
High risk	50.0-59.5	South Africa, Morocco, Brazil, Russia
Moderate risk	60.0-69.5	Saudi Arabia, Poland, Slovakia
Low risk	70.0-84.5	Spain, Chile, China, Japan
Very low risk	85.0-100	UK, France, Switzerland, USA

that r_f is equal to the risk-free rate for the given project duration. Furthermore, by applying the Capital Asset Pricing Model (CAPM) it is also widely accepted that $r_p = \beta\left(r_m - r_f\right)$, where r_p denotes the risk premium, β is selected by analogy from enterprises with similar risk profiles and r_m is the capital market return rate [184]. Note that the capital asset model inherently quantifies the trade-off between risk and return for the project. Indeed, an investor should be compensated with a return r in order to accept the project's riskiness and deferred payment profile. Otherwise, she/he would be better off investing in capital markets.

Unfortunately, evidence of CAPM failures are especially ubiquitous for emerging markets and countries with underdeveloped financial markets [185]. Also unfortunate is the fact that ground station selection is largely driven by coverage considerations and has, from a performance perspective, little or nothing to do with economic or financial factors, thus suggesting that the CAPM-based argument might no be necessarily applicable. How, then, should we quantify the risk premium r_p in Equation 4.11 Harvey provides in Reference [186] a solution based on expert opinion, more precisely the International Investor's Country Credit Rating (ICCR). The rating assigns a 0-100 score (100 meaning no risk) to virtually any country through macroeconomic data gathering and expert opinion elicitation. Table 4.15 summarizes the resulting risk categories and lists a subset of countries representative from them. The available list has currently a total of 146 countries, and is bounded by Switzerland with a 94.94 average score over the last 15 years, and North Korea with a score of 7.57. These scores are obtained through aggregation of three broad risk categories: political, financial and economic with a weight of 2/3, 1/3 and 1/3 respectively. Each category is further subdivided into finer risk factors, each one with a given weight. Then, experts are asked to score all of them and results are averaged across all participants [187].

Once the ICCR for a given country has been elicitated, the next step is to quantify the risk-adjusted discount factor by regressing historical market return data on the ICCR for different countries. This task is described in Reference [188] and the resulting model is periodically updated by one of the authors. In particular, the proposed functional form of the risk-adjusted discount rate for a given country is

$$r_c = r_f + r_{US} + r_{ICCR} + r_I + r_F \tag{4.12}$$

where

- r_f denotes the long-term US risk-free rate and is assumed to be 2.2% based on the 20 year US treasure bond.

- r_{US} denotes the long-term US capital market risk premium. Since the network under consideration would be undertaken by governmental agencies such as NASA or NOAA, we will let $r_{US} = 0$. In other words, the US government does value a project based on the opportunity cost of investing in the US stock market as a commercial entity would.

- $r_{ICCR} = \beta \ln \frac{\text{ICCR}}{\text{ICCR}_{US}}$ is the risk premium associated to a baseline exposure to risk factors summarized in Step 4-4 and quantified through the ICCR score. Note that the national security risk factor is not included in the ICCR, but it is assumed that countries with national security concerns would never be suitable candidates for placing ground stations.

- r_I denotes the risk premium attributable to a given industry sector. Since the network under consideration would be undertaken by a governmental agency, once again we assume that $r_I = 0$.

- r_F is an additional country risk premium that at worst would be equal to r_{ICCR} and can be used to tune the expected risk exposure the the individual risk factors defined in Step 4-4. Table 4.16 summarizes the weights utilized for each of the risk factors for all countries except Antarctica and the Svalbard station[9]. Note that financial and economic risk that would typically dominate in commercial project valuation are replaced by risks of expropriation. Note also that a score of -10 is awarded for risks that are in control of the network owner, while a score of 0 is awarded to other risks that have not been significant in the history of NASA networks. Both the weights and the scores where qualitatively assessed by reviewing the history of NASA's minitrack and STDN, but can be easily tuned should other recommendations arise. Finally, the impact on r_{ICCR} reported in Table 4.16 is normalized with respect to r_{ICCR}. In other words, if a given country has $r_{ICCR} = 1\%$, then the political risk would add an extra 5% to its value.

Table 4.17 provides the discount rates estimated using Equation 4.12, ordered from largest

[9] For them, all weights are kept constant but the scores are adjusted to reflect the special needs of a continent without government and limited logistics capabilities. In that sense, risk exposures to international cooperation, natural disasters and resource and technological risk are maximized, while exposure to expropriation and political risk were set to 0.

Table 4.16: Risk Factor Weights

Risk Category	Weights	Score	Impact on r_{ICCR}
National security	NA	NA	NA
Political risk	13%	-10	5%
Expropriation risk	25%	-10	10%
Project sensitivity to wars, etc.	13%	-10	5%
Project sensitivity towards natural disasters	13%	-10	5%
International cooperation risk	26%	-5	0
Technological and resource risk	10%	0	0
Financial and economic risk	0%	0	0

Table 4.17: Discount rates for the NPOESS and JPSS Ground Sites

Site	Country	ICCR	r_{ICCR}	r_F	r
McMurdo	Antarctica	25.00	23.08%	14.08%	39.36%
Troll	Antarctica	25.00	23.08%	14.08%	39.36%
Barreira	Brazil	56.10	8.77%	6.09%	17.06%
Sriharikota	India	57.20	8.43%	5.85%	16.48%
Hartebeesthoek	South Africa	59.00	7.28%	6.07%	15.55%
Santiago	Chile	73.80	3.92%	2.72%	8.84%
Naro	South Korea	73.90	3.90%	2.70%	8.80%
Dongara	Australia	86.90	1.03%	0.71%	3.94%
Warkworth	New Zeland	86.90	1.03%	0.71%	3.94%
Guam	USA	92.10	0.00%	0.00%	2.20%
Fairbanks	USA	92.10	0.00%	0.00%	2.20%
Hawaii	USA	92.10	0.00%	0.00%	2.20%
White Sands	USA	92.10	0.00%	0.00%	2.20%
Kennedy	USA	92.10	0.00%	0.00%	2.20%
Weilheim	Germany	92.80	-0.13%	-0.10%	1.97%
Svalbard	Norway	93.90	-0.34%	-0.21%	1.65%

to smallest, as well as its ICCR component and the risk factor component. They have been computed assuming a 2.20% long-term risk free rate. Observe that the obtained results are consistent with our intuition, i.e. ground sites located in riskier parts of the world have a high discount rate while ground sites in US or European territories are basically discounted at risk-free rate. Furthermore, we can also observe the effect of extra exposure to the aforementioned risk factors. Indeed, stations and Antarctica are largely sensitive to natural disasters, international cooperation and resource and technological risk. This is modeled as an extra 14.08% discount rate for that ground station.

Finally, note that the Svalbard station has a risk-adjusted discount factor of just 1.65%. Yet, our intuition suggests that building a ground station in such an isolated area should be highly expensive. This is precisely a key advantage of the proposed method, it explicitly separates cost and risk-related concerns. Recall from Table 4.13 that construction and sustainment costs in Svalbard are indeed almost 300% more expensive than average. However, because the island belongs to Norway there is little to no political risks and is supported by a well-established logistic operation. All of these factors contribute to its low risk-adjusted discount benefit and guarantee that this location will be far more attractive than other places in the north or south poles.

Risk-Adjusted Discounted Benefit and Hurdle Rates

Discounted benefit has been successfully used in concept studies for space-based Earth observation systems [26], [189], [190]. In these cases, discounted benefit was proposed as an architecturally distinguishing metric to compare different schedules for launching satellites into orbit. In a nutshell, given a set of instruments allocated to a set of spacecraft, the system benefit can be statically computed as a function of the scientific data products obtainable from the remote observations. However, this static valuation is insensitive to the order and time line with which satellites are launched. This contradicts intuition since Earth observation programs are launched with overlap from one another to avoid data continuity gaps and allow for cross-validation between old and new instruments.

Selva argues in Reference [26] that the main idea behind discounted benefit "is that science today is better today than science tomorrow". In other words, in his definition discounting is equivalent to finance's time value of money. A similar argument for space communication networks is also possible: Ensuring today's operations has more value than supporting tomorrow's operations. Indeed, if the network fails today then a mission has to initiate safety procedures and immediately switch to a back-up alternative, a problem typically referred to as continuity of operations. In contrast, if the network is expected to fail in a year, then both the mission and network programs have time and resources to implement alternatives without risking the safety of the mission. Furthermore, this interpretation also clarifies that riskier networks should be discounted at a higher rate. Indeed, if a ground station is prone to failures and operations disruptions due to limited maintainability, then tomorrow's utility should be adjusted at a higher discount rate than another one placed in a secure and well-maintained facility. This realization leads to the definition of the risk-adjusted discounted benefit, a well-known technique for valuing financial projects and quantifying the risk-return trade-off.

In order to visualize the effect of risk-adjusting benefit, consider a fictional project that delivers a utility of 1 unit each year while the system is in operation. Next, we compare the present value of the cumulative benefit over time, risk-adjusted and time discounted, assuming 15 years of operation and two opposed and extreme risk profiles: No risk with $r = 2.2\%$ and full risk with $r = 40\%$. Figure 4-16 plots the present value of the project's utility between now and time t for the two aforementioned alternatives. We can observe that

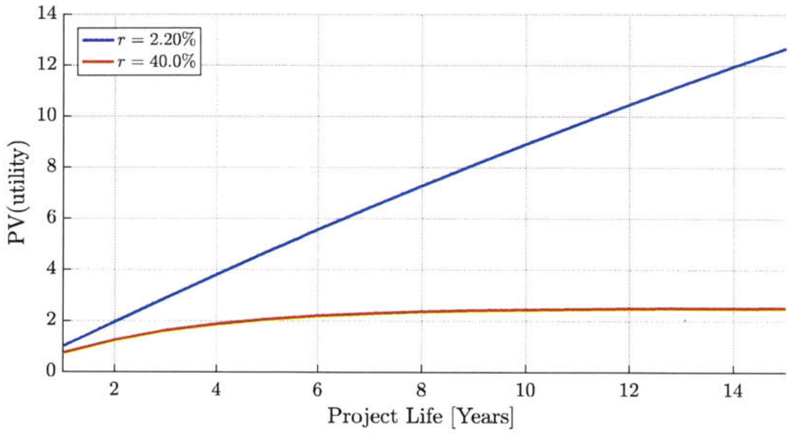


Figure 4-16: Risk-adjusted Discounted Benefit

the risk-less project accumulates benefit over time at an almost linear rate. Alternatively, consider now the project with a 40% discount rate. Due to uncertainty associated with the risk factors listed in Step 4-4 there is virtually no benefit from operating the system between years 6 or 7 onward. Indeed, the risks associated with the project are so high that, at present time, we cannot confirm nor deny the fact that the system will still perform as originally intended. Therefore, the risk-adjusted discounted benefit valuation captures this effect by assigning an almost null benefit increase in the latter years of the project.

Having notionally visualized the effect of risk-adjusting the network discounted benefit, we now detail several important considerations that must be taken into account when utilizing the proposed approach:

- Selecting the risk-free rate is not obvious for the types of systems under consideration. Previous authors that utilized discounted benefit as a metric have used a discount rates of 5% to 15% approximately (see for instance References [191], [189]). Basically, since the metric is used solely for relative comparison of architectures choosing a baseline discount rate will have the same effect across all of them and therefore will only shift the Pareto Front. The same argument applies to our method, except that now r_f has to be meaningful with respect to r_{ICCR} and r_F. In order to ensure that, we must set r_f equal to the US long-term risk-free rate based on the model from Reference [186].

- The risk premium for a given architecture r is estimated as the average across all ground sites. This is valid in this case because we assume that risk factors for the different sites are uncorrelated with each other. Otherwise, the problem would be similar to the well-known Markowitz portfolio optimization problem [192]. Furthermore, by using the average we implicitly capture the effect of diversification: Networks with more ground stations diversify risk by providing increase contact opportunity redundancy.

- The discount rate r assessed from the ICCR index can be for system optimization purposes by either discounting benefit or having it as a separate risk metric. In that sense, the results presented in Step 4-5 will assume that the network risk-adjusted cost of capital (or hurdle rate) is a separate system metric.

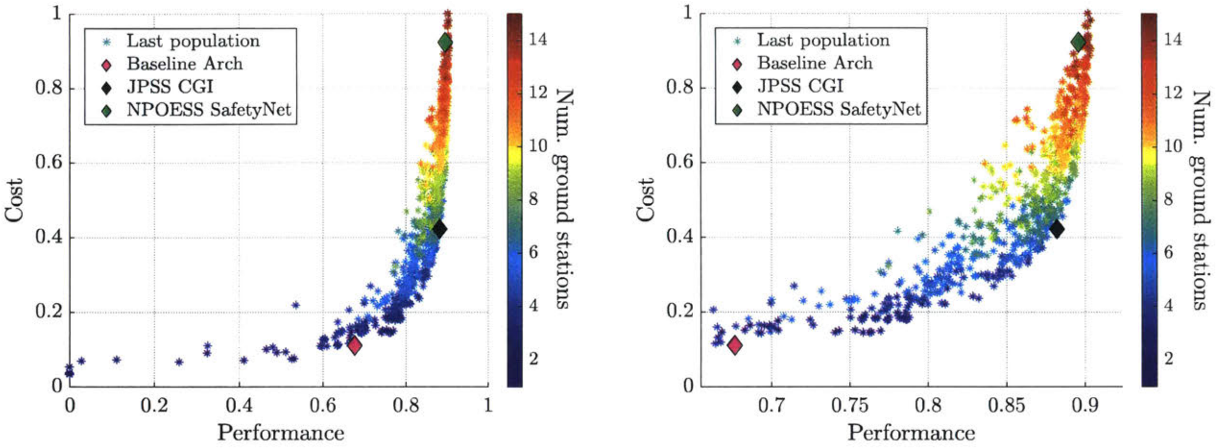
- Other alternative valuation methods for risky endeavors are present in the literature and have been proven to yield better results. For instance, discounting certainty equivalents in the presence of non-linear utility functions is known to be a better decision rule than risk-adjusting the discount rate. However, the choice of this alternative for quantifying risk was not due to its optimality but to the very limited data on intangible sources of risk. Therefore, we argue that the proposed metrics, albeit sub-optimal, are the best ones possible given the types of risk factors under consideration.

Step 4-5. Analysis of Results

Step 4-5.1. Introduction

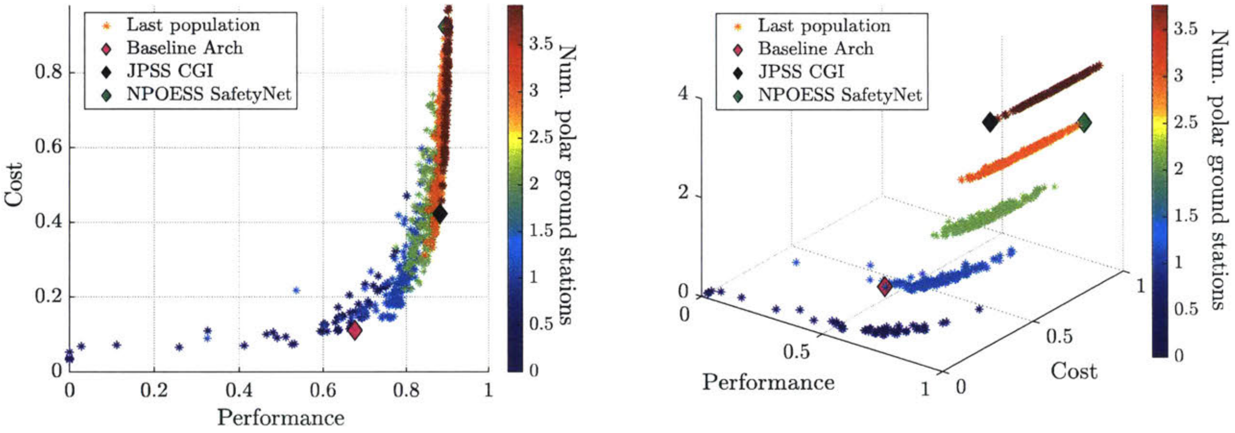
In this section I report the results obtained when optimizing the JPSS ground network. This optimization has been conducted using a multiobjective genetic algorithm configured with 750 architectures per population and 10 populations. Since 16 ground stations are available, a total of $2^{16} = 65536$ network architectures must be considered for evaluation. However, by utilizing a genetic algorithm only 10-11% of this space needs to be evaluated to successfully approximate the Pareto front. The results are presented in Figure 4-17, where I plot the performance-cost architectural space color-coded based on the total number of ground stations deployed. In the same figure we also overlay the three reference architectures that originally motivated this case study: The baseline architecture with only one northern ground station, the current JPSS CGI and the canceled NPOESS SafetyNet. The obtained results agree with our intuition: As more ground stations are added to the system, the performance progressively improves albeit with diminishing returns, causing the non-linear asymptotic structure of the Pareto front. This asymptotic behavior is in fact explained by two facts: First, since all Weather satellites but GOES are flying in Sun-synchronous orbits, securing contact opportunities at high latitudes results in one approximately one contact per spacecraft revolution. In contrast, an equatorial ground station provides at best three contact opportunities in the same time span, thus resulting in a non-linear relationship between performance and number of ground stations. On the other hand, the relationship between cost and number of ground stations is not perfectly linear, thus further contributing to the asymptotic behavior.

Given the importance of polar ground stations in weather satellite systems, Figure 4-18 plots the same tradespace color coded based on the number of polar ground stations in the



(a) Entire Architectural Space (b) Architectural Space after Downselection

Figure 4-17: Performance-cost Space with Number of Ground Stations



(a) Entire Architectural Space (b) 3D Performance vs. Cost Tradespace

Figure 4-18: Performance-cost Space with Number of Polar Ground Stations

system. Once again the result is as expected, with higher performance and cost for networks that utilize increasing numbers of polar ground stations that require large infrastructure to maintain them. Interestingly, Figure 4-18b exposes one of the main differences between the SafetyNet and JPSS. While the first one selected only 3 polar ground stations, and obtained extra performance by creating a large network of mid-latitude and equatorial ground stations, the second one reduced the total number of ground station at the cost of having an extra polar site. In section Step 4-5, I investigate how this switch affects the network riskiness.

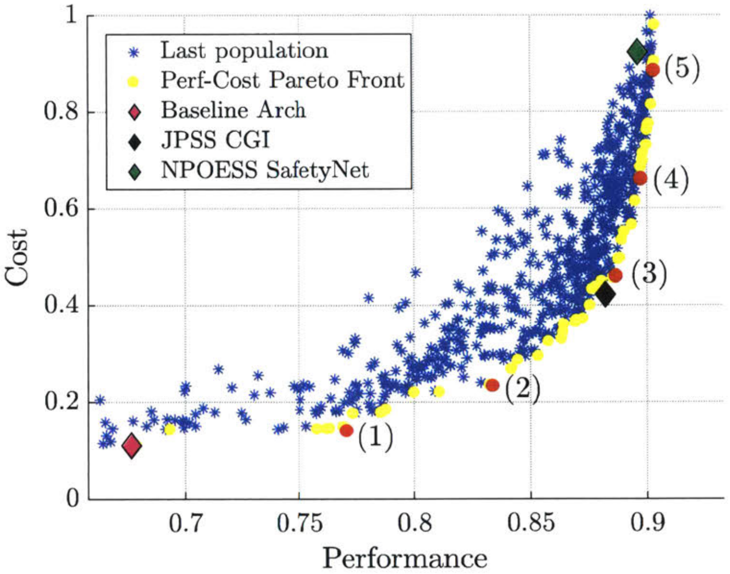


Figure 4-19: Pareto Front in the Performance-Cost Space

Step 4-5.2. Results for the performance-cost tradespace

Once basic intuition and confidence on the performance-cost tradespace has been obtained, I now focus on the Pareto front structure by sampling different individual architectures and analyzing their features. Furthermore, I downselect the total number of architectures by assuming that the baseline architecture imposes a lower bound on performance (i.e. we always want to improve the current system performance) and the NPOESS SafetyNet imposes an upper bound on cost (since it was deemed too expensive to successfully implement it).

Figure 4-19 plots the performance-cost Pareto Front, after down-selecting almost all dominated architectures. First, note that the performance of the system stagnates at a normalized utility of 0.9 approximately. This fact is explained by two factors: First, the performance metric is computed over all latency contributors in the system. Therefore, even if we were able to reduce the LOS latency contributor to zero we would still not achieve perfect utility. Second, the set of candidate sites proposed for this case study does cannot provide zero average latency due to coverage gaps in the Atlantic, Pacific and Indian oceans (see Figure 4-13).

On the other hand, observe that 5 individual architectures approximately evenly spaced in the Pareto Front have been highlighted for further exploration (see also Table 4.18). Architecture (1) is composed of only two ground stations, one in the mid-to-low latitude area while the another one is polar and delivers most of the value by providing one contact per spacecraft orbit. In that sense, I observe that this configuration is driven primarily by

Table 4.18: Summary of Pareto Efficient Architectures

Arch	Utility	Latency [min]	Cost	Num. Sites	Sites
(1)	77.0%	36.7	0.149	2	McMurdo, White Sands
(2)	83.2%	20.8	0.236	3	Fairbanks, McMurdo, Santiago
(3)	88.6%	7.4	0.457	5	Fairbanks, McMurdo, Sriharikota, Svalbard, Troll
(4)	89.7%	4.44	0.660	10	Fairbanks, Kennedy, McMurdo, Naro, Santiago, Sriharikota, Svalbard, Troll, Warkworth, Weilheim
(5)	90.4%	2.86	0.905	14	Barreira, Fairbanks, Hawaii, Hartebeesthoek, Kennedy, McMurdo, Naro, Santiago, Sriharikota, Svalbard, Troll, Warkworth, Weilheim, White Sands

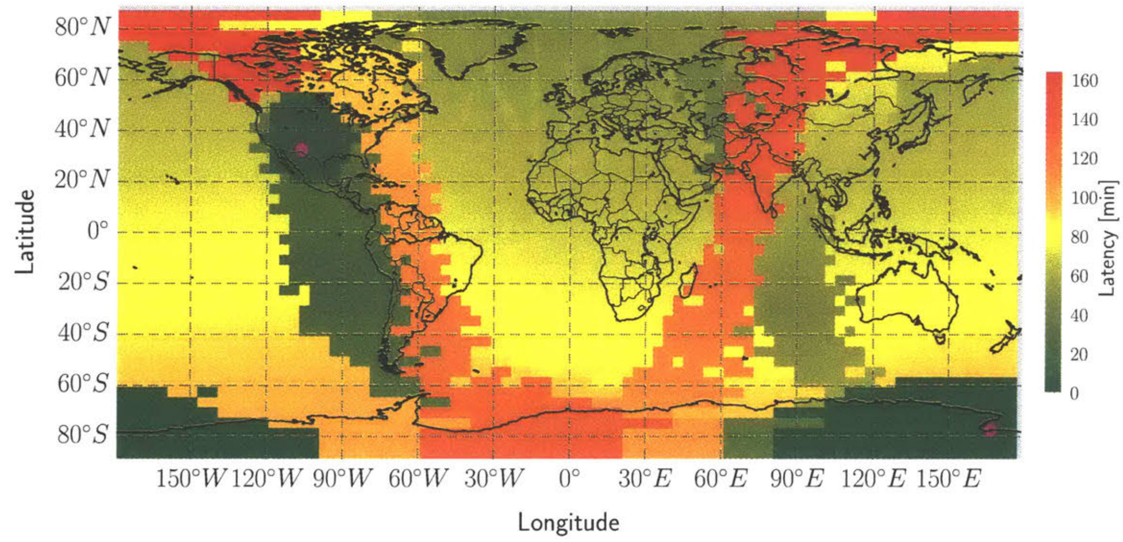


Figure 4-20: Latency Map for Architecture (1)

the construction and sustainment factors from the cost model. Indeed, since there is no difference between northern and southern hemisphere costs, the latter alternative is selected since McMurdo has the antenna can be placed closer to the geographical pole. Alternatively, the secondary ground station is placed within CONUS in order to avoid moving part of the infrastructure outside the country.

Architecture (2) is also interesting for two reasons: First, it utilizes two polar ground stations as opposed to one at the expense of a 60% cost increase. Second, by placing one single mid-latitude ground station in the American longitude region it reduces latency for North and South America, as well as Europe to just 30 minutes (see Figure 4-21), thus resulting in a large improvement over the baseline architecture. Note that this approach is very attractive for mesoscale models such as the NAM, where satellite data products of CONUS, Alaska,

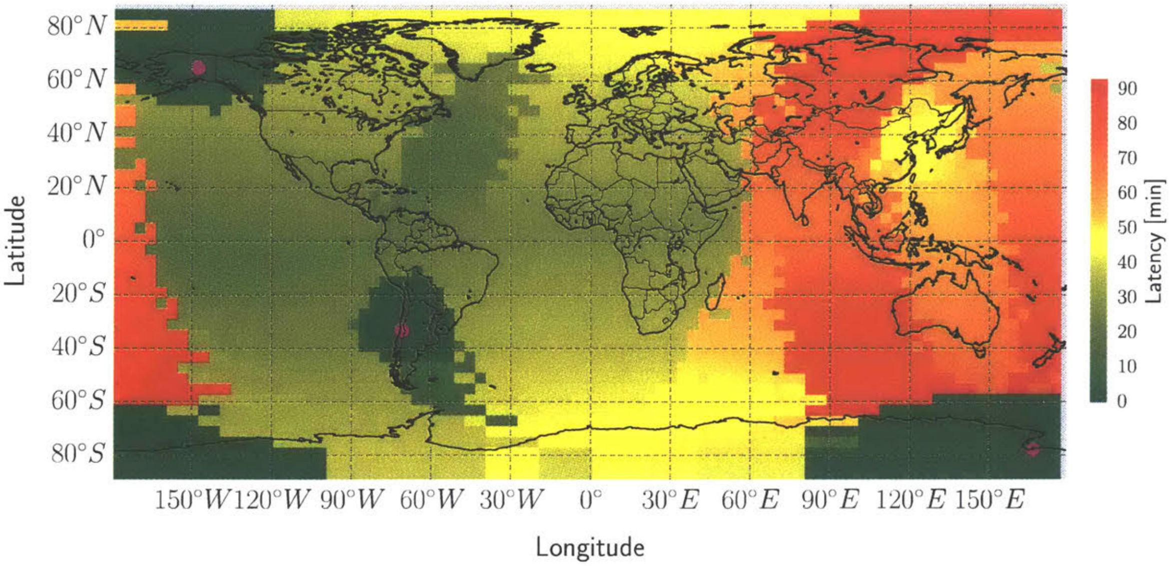


Figure 4-21: Latency Map for Architecture (2)

Hawaii and the Caribbean Ocean could be provided with almost optimal latency at much lower cost.

Architecture (3) is the first one found by the optimization algorithm that includes four polar ground stations. It is slightly superior than the current JPSS alternative by adding a ground station in India. This choice is vindicated both by performance and cost considerations. On the one hand, Figure 4-22 demonstrates that utilizing all polar ground stations reduces the maximum experienced latency to just 40 minutes for all latitudes and longitudes. The only polar area with large latency is placed north of South Korea where neither Svalbard nor Fairbanks provide coverage. This would suggest that it would be optimal to select Naro as a semi-polar ground station to service that area. Instead, India is selected due to cost considerations. Indeed, observe in Table 4.13 that India is up more than 10% cheaper on average than South Korea, thus resulting in better performance per unit of capital investment.

Finally, architectures (4) and (5) are increasingly similar to SafetyNet in that they span all continents and require more than ten ground stations. Furthermore, they are built by keeping the four polar ground stations of the JPSS network and progressively adding new mid-latitude and near-equatorial ground stations. The result is unfortunately very non-linear, indicating that the improvement in performance per unit of capital expenditure is increasingly marginal: 1.7% latency reduction requires a 36% increase in system cost for architecture (4), while the 2.3% improvement of architecture (5) is achieved at the expense of a 52% increase in life cycle cost. These findings are summarized in Figure 4-25, where

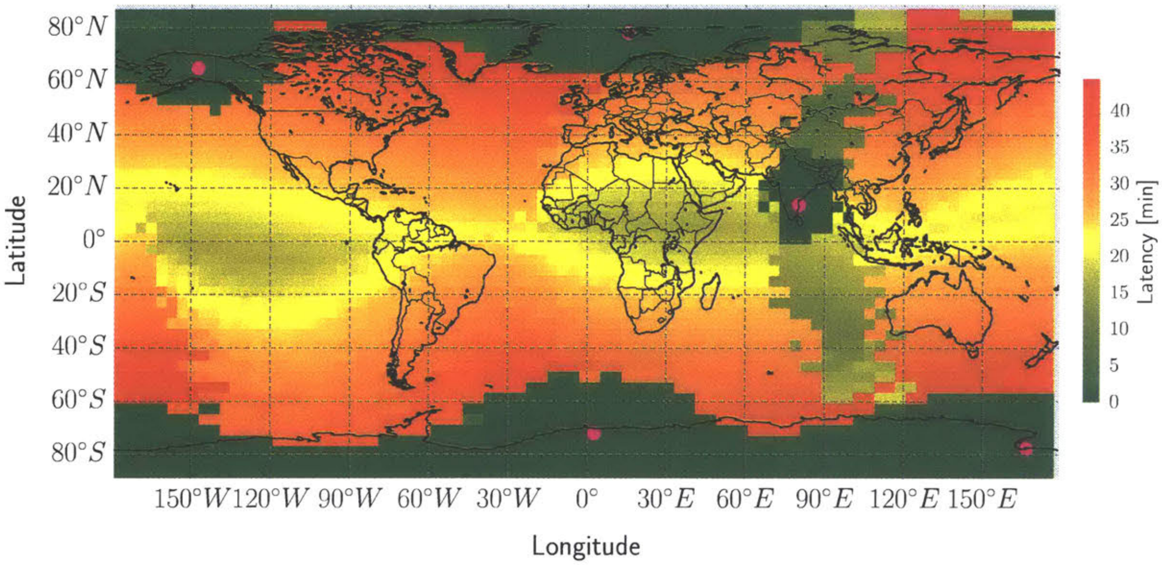


Figure 4-22: Latency Map for Architecture (3)

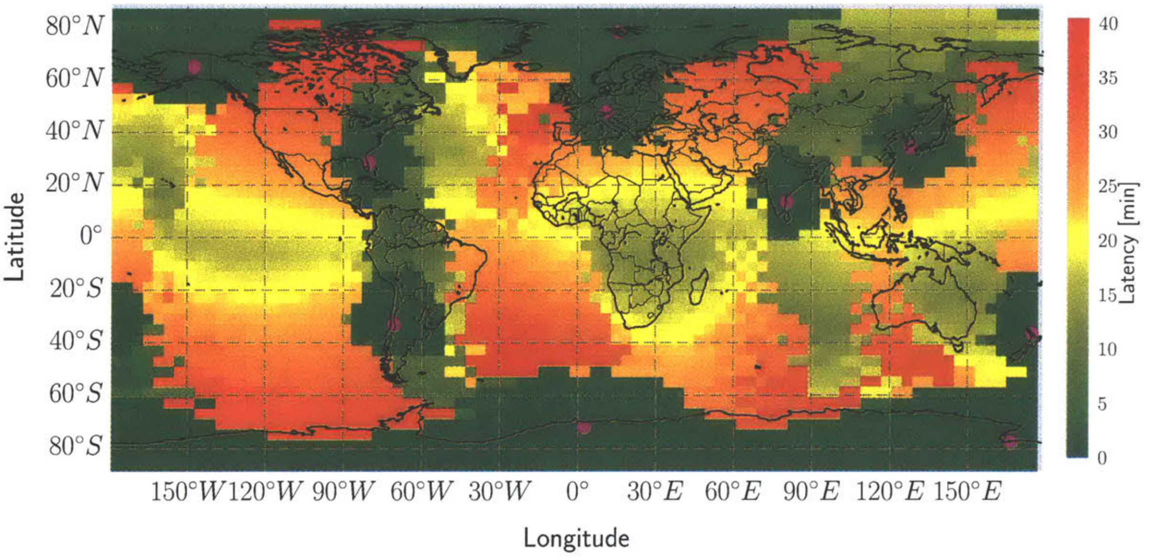


Figure 4-23: Latency Map for Architecture (4)

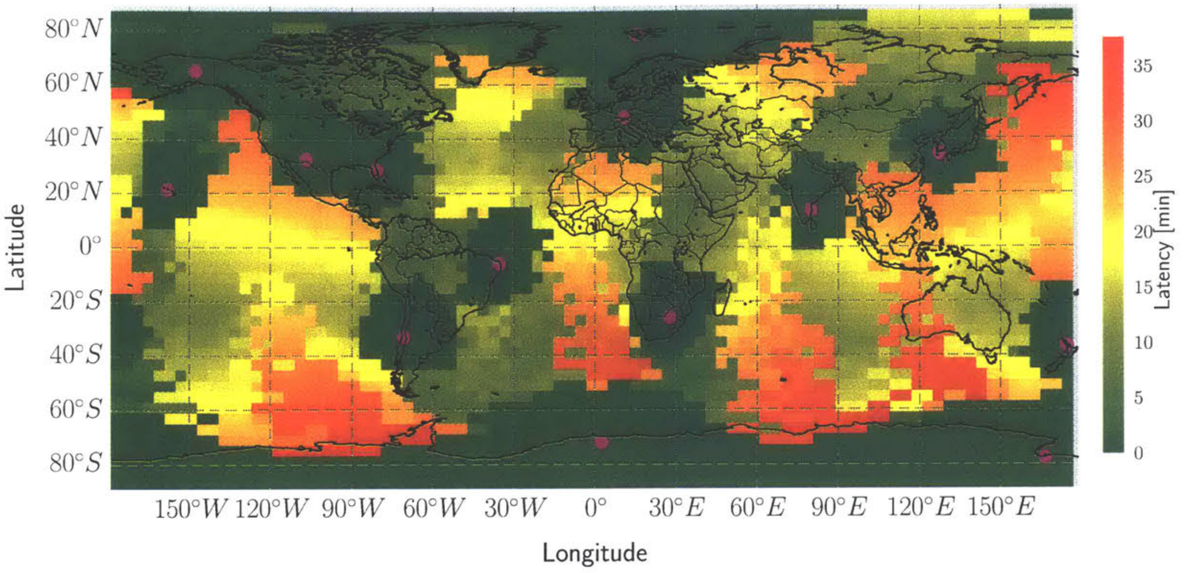


Figure 4-24: Latency Map for Architecture (5)

the different zones of the cost-performance Pareto Front are notionally depicted.

Step 4-5.3. Results for the performance-risk tradespace

Having analyzed the architectural space in the performance-cost dimensions, we now turn our attention to system programmatic risk. Figure 4-26 plots both the performance-cost and performance-risk spaces and highlights the resulting Pareto Fronts. As usual, the baseline architecture, JPSS CGIand NPOESS SafetyNet are provided for reference. The risk axis has been normalized to the 0-1 scale since it is a purely relative metric. Interestingly, we observe that there is a large disagreement between the performance-cost Pareto front and the performance-risk Pareto Front. In other words, a network designed solely based on performance-cost considerations would unfortunately typically have a poor score in the risk category and vice versa.

To understand this phenomena without having to query individual architectures one at a time, I collect statistics on the frequency with which a given ground station appears in a Pareto-optimal architecture both in the performance-cost and performance-risk tradespaces. Results are reported in Figure 4-27, where bars are normalized in percentage such that their sum is equal to 100% and they have been sorted such that ground stations with increased popularity in the performance-risk Pareto front appear first (see the specific popularity values on top of each bar). Several conclusions can be reached:

- Hawaii and Guam are suboptimal sites from a cost perspective. However, they become

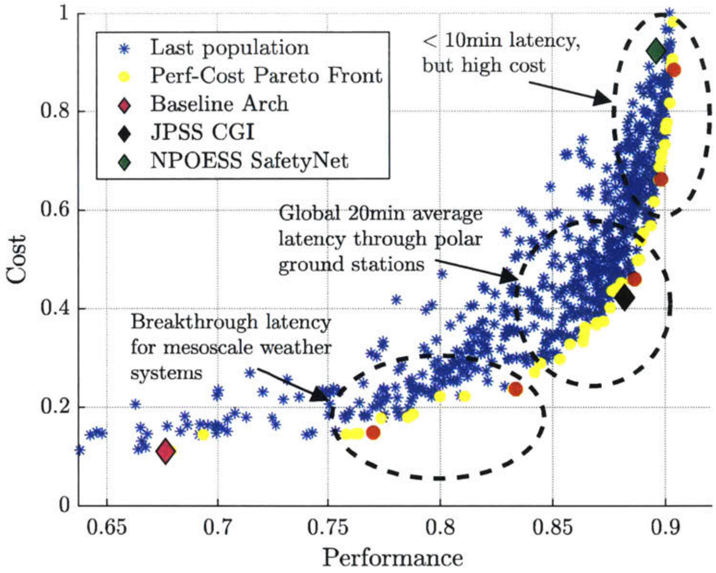


Figure 4-25: Pareto Front in the Performance-Cost Space

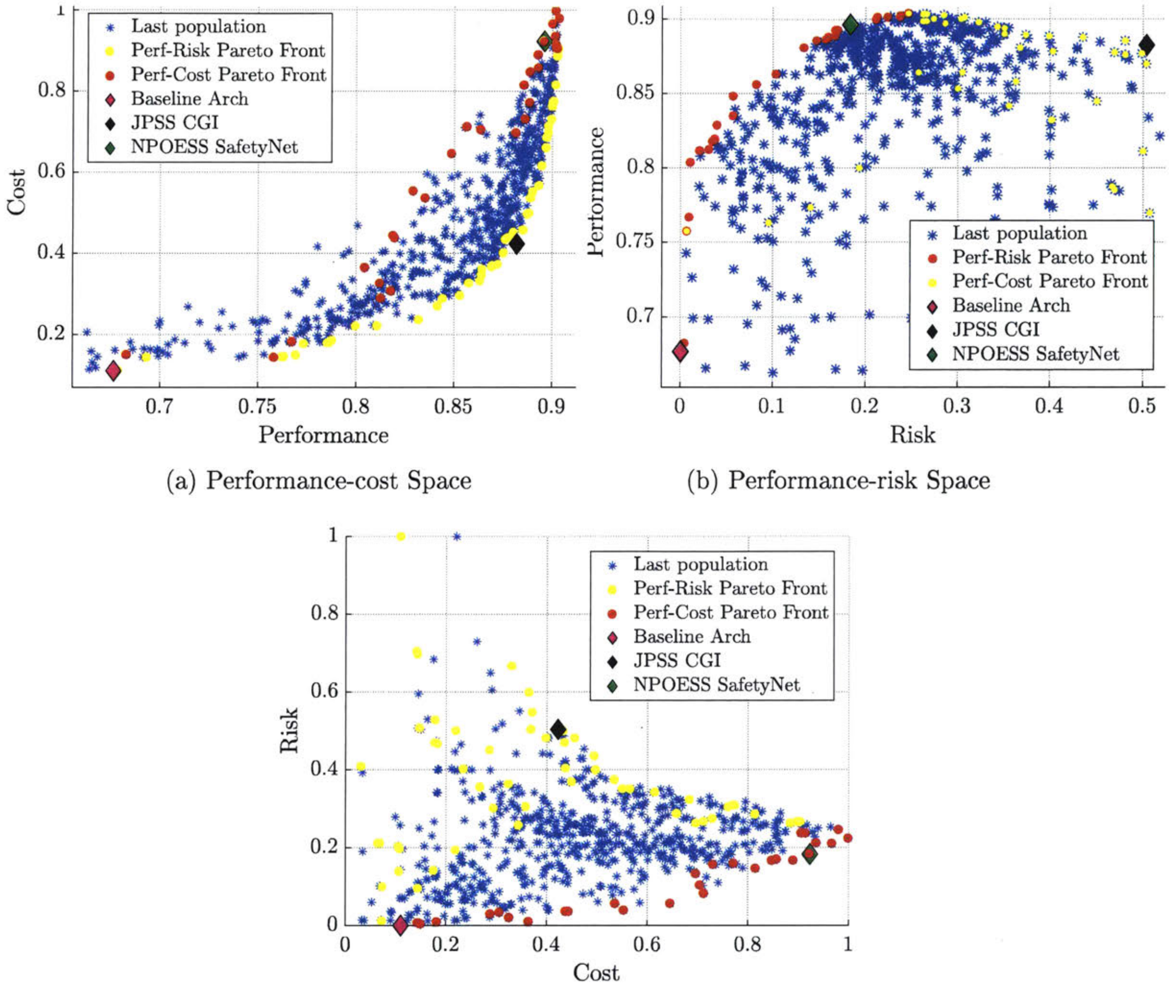
highly attractive when risk is considered since both of them are US territories sheltered from political and expropriation risks.

- The two Southern hemisphere polar ground stations are the most heavily penalized in the risk-performance space. This is mostly due to the high technological and resource risks associated with placing communication infrastructure in a continent without a fully developed supply chain.

- Ground stations in Brazil or India are significantly attractive from the cost-performance perspective since they take advantage of minimal construction and sustainment multipliers as well as advantageous currency exchange rates. However, their political and societal risks result in high cost of capital that renders them inadequate for placing ground sites.

- Ground stations in CONUS, Europe or Australia are highly favored due to the minimal political, technological or resource risks. As a result they tend to be favored in the performance-risk Pareto front. This is specially true for mid-to-high latitude sites such as Dongara and Warkworth, that take advantage of increased coverage to SSO and thus deliver extra performance at reduced levels of risk and moderate levels of cost.

On the other hand, Figure 4-26c plots the obtained tradespace in the risk-cost space. It can be observed that only two architectures are optimal in that space, a conclusion that was expected since the optimal solution with respect to those metrics would be to simply create a network with one ground station in either the least risky country possible or the least expensive country. More interesting is the fact that the performance-cost Pareto front almost perfectly aligns with the lower right envelope of the tradespace. In other words, given



(a) Performance-cost Space

(b) Performance-risk Space

(c) Risk-Cost Space

Figure 4-26: Performance-cost vs. Performance-risk Pareto Fronts

a desired level of performance, the optimal solution in terms of risk is to maximize the cost of the system. This finding can in fact be explained through the benefits of diversification. Indeed, given a level of performance, the network architect has a choice between affordability and risk diversification: More ground sites imply that failure of one of them has less impact in the overall network performance. Or equivalently, the percentage of capital investment lost per ground site failure decreases as the total number of sites increases. In contrast, the performance-cost Pareto front almost entirely lies in the upper part of the tradespace indicating that for a given level of performance the most cost-efficient architecture is the most risky one. Finally, note that the differences between the two Pareto fronts become decreasingly significant as the system becomes more expensive. Once again, the argument of diversification can be used to explain this phenomenon: Networks with more than 10 ground stations have limited variability in the risk score since this FOM is computed as the average across the riskiness score for all ground sites. Note that this argument is valid because

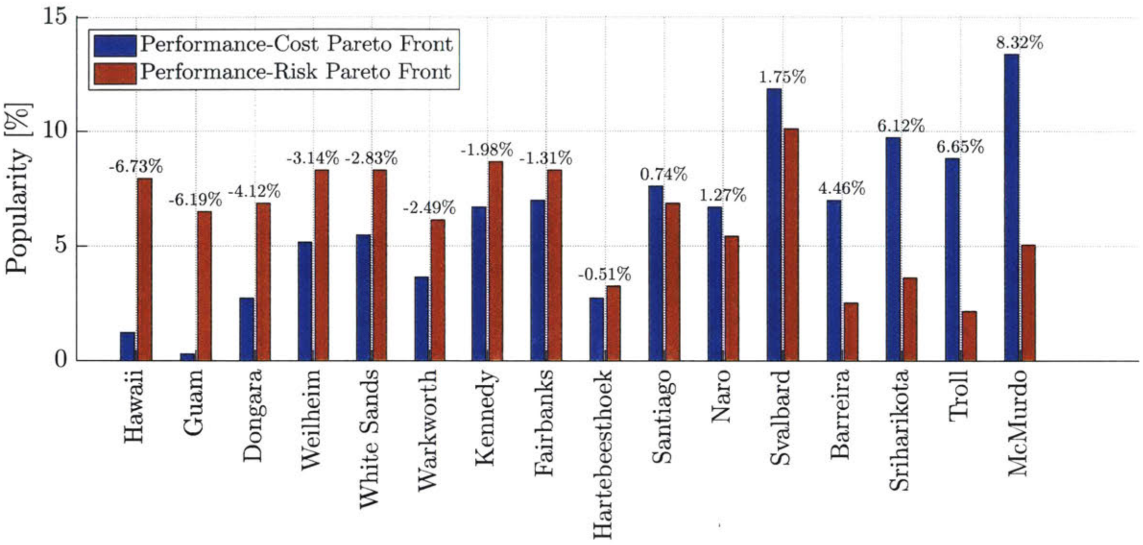


Figure 4-27: Ground Station Popularity

I assume that ground stations will always be located in geographically and geopolitically uncorrelated locations, thus being effectively independent in terms of risk factors[10].

Step 4-5.4. Space-based vs. ground-based Networks

Up until this point, I have only focused on returning NWP data in a timely manner using ground-based networks. In reality, a completely different yet valid architecture would be to develop a space-based relay system that ensures continuous visibility and therefore reduces the line-of-sight latency contributor to virtually zero. To benchmark this alternative against ground-based infrastructures, I will compare the obtained performance-cost tradespace from Section Step 4-5 against the TDRSS 3rd generation. The choice of this system as a reference point is justified by two facts: First, part of the current satellite observations (e.g. the EOS satellites) are currently supported by the TDRSS system. More importantly, the JPSS program specifies a level-1 requirement to "provide command, real-time and stored mission and telemetry data transmission to TDRSS" [193].

Figure 4-28 provides a comparison of the different ground-based network architectures and the TDRSS system. Performance is expressed in units of overall system utility as in all previous plots. In contrast, cost is provided normalized with respect to the cost of the 3rd generation TDRSS: $1B for acquiring and launching 3 spacecraft (with a lifetime of 15 years each), plus $400M in ground system upgrades[11], as well as recurring cost of $40M

[10] A simplification was made so as to assume all sites within USA to also be independent. In reality political and economical risk factors might be shared among them.

[11] The Space Network Ground Segment Sustainment (SGSS) Project was baselined at $862M and is expected to have cost overruns of $329M [194]. The assumed $400M is assumed based on the fact that ground segment modernization projects like SGSS do not occur with every new TDRSS generation.

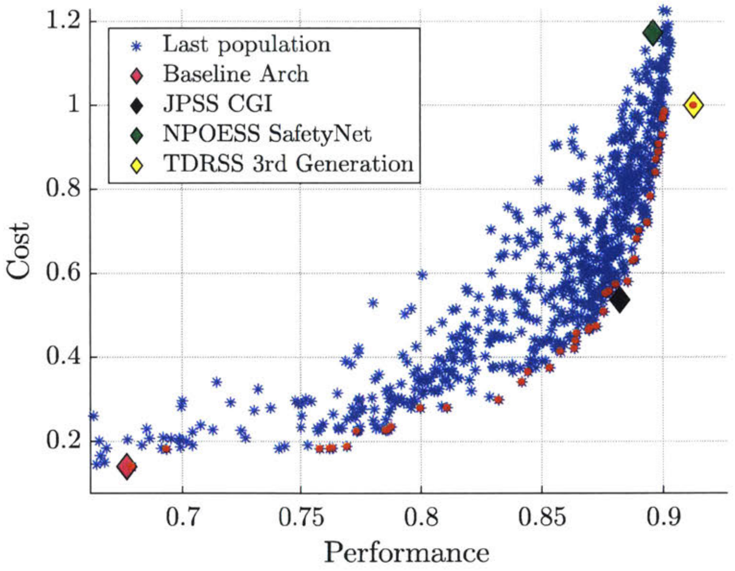


Figure 4-28: Pareto Front in the Performance-Cost Space with TDRSS

a year per ground station[12]. Results indicate that a space-based network is indeed non-dominated in the performance-cost tradespace and serves as a maximum limit in the level of distributedness for the ground network. In other words, ground networks with more than ten ground stations should be carefully evaluated during the system design phase since space-based networks like TDRSS might in fact be more cost-effective.

That being said, trade-offs between ground and space-based networks cannot be uniquely based on latency and cost considerations as they typically provide infrastructure for other missions. In that sense, two other important metrics to consider are user burden and network scalability. A quantitative measure for the former is receiver sensitivity at a given frequency band. Table 4.19 summarizes state-of-the-art receiver sensitivities for space networks, near Earth ground based networks and deep space networks. As expected, there is really no trade-off between space and ground networks with respect to receiver sensitivity. Not only can we place larger dishes on the ground, but the sky noise temperature observed by an space-facing antenna is significantly lower than that observed by an antenna that has the Earth in the background. Consequently,ground-based networks always dominate space-based networks in the user burden FOM.

On the other hand, network scalability relates to the ability of the network to effectively provision capacity in order to adapt to changes in demand. For the networks under consideration, capacity will not be measured in terms of data rate or data volume achievable as is typical in terrestrial WAN. Instead, capacity will be defined in terms of antenna time

[12] Note that the operations costs per ground site are estimated at less than $15M per year since operating a ground network is significantly simpler than a space network.

[13] No frequency band allocation available.

Table 4.19: DSN, NEN and SN Receiver Sensitivity

Band	**Sensitivity [dB/K]**		
	SN (5m)	**NEN (11m)**	**DSN (34m)**
S	9.50	23.00	41.04
X	N/A[13]	35.00	68.24
Ku	24.40	N/A	N/A
Ka	26.50	45.00	67.20

since it is typically the constraining factor that dictates how often data can be returned to Earth. In that sense, Figure 4-29 provides a tradespace that quantifies network scalability vs. performance, where scalability is measured by the cost of procuring a 10 minute pass for any given mission[14]. Several interesting conclusions can be reached:

- Ground-based networks are more scalable than space-based networks since they procure passes at a cost of $10 to $20 per minute of service. In contrast, the space-based networks require a capital investment (both procurement and operations) that at least doubles that of a ground network.
- Procuring capacity through polar ground network is significantly more expensive than creating highly distributed and diversified network. As a result, both the JPSS CGI and baseline architectures are highly dominated.
- Ground networks with more than 10 sites provide capacity at an approximate constant cost of $100 per pass, thus indicating that should be a reference level for cost effective capacity provisioning in space networks.

Step 4-6. Identification of Second-Order Latency Contributors

So far I have investigated the performance, cost and risk trade-offs inherent to a ground network that provides services to satellites that collect data for NWP purposes. Analyzing this part of the system was originally justified in Step 4-3, particularly Figure 4-12, where line of sight between the satellite and current ground systems was found to be the largest latency contributors for all satellite programs except GOES. However, as we transition from the current baseline architecture to the future JPSS system or even support from the TDRSS system, other latency contributors might become the dominating factor to be addressed. In that sense, Figure 4-30a visualizes the relative importance of the LOS latency contributors as a function the chosen ground architecture. As expected, the importance of the LOS progressively diminishes and is almost zero for SafetyNet-type architectures. Similarly, Figure 4-30b plots the relative ranking of LOS within all possible latency contributors in

[14] The cost per pass does not include the construction costs of facilities shared across antennas in a site since the scalability cost.

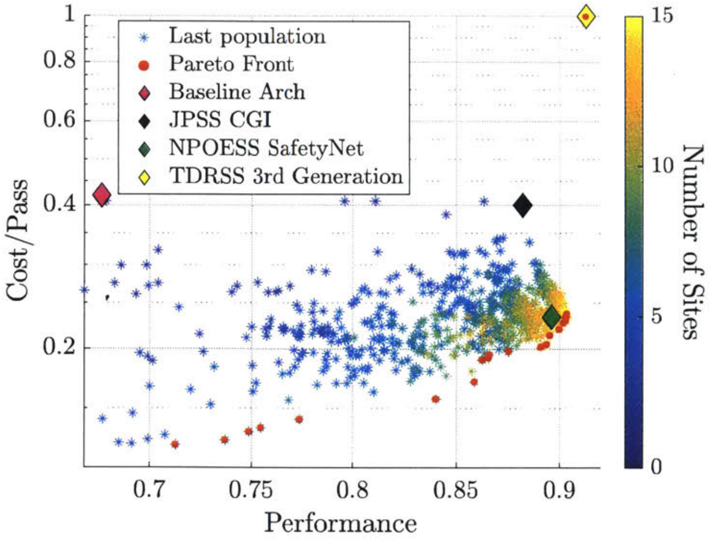


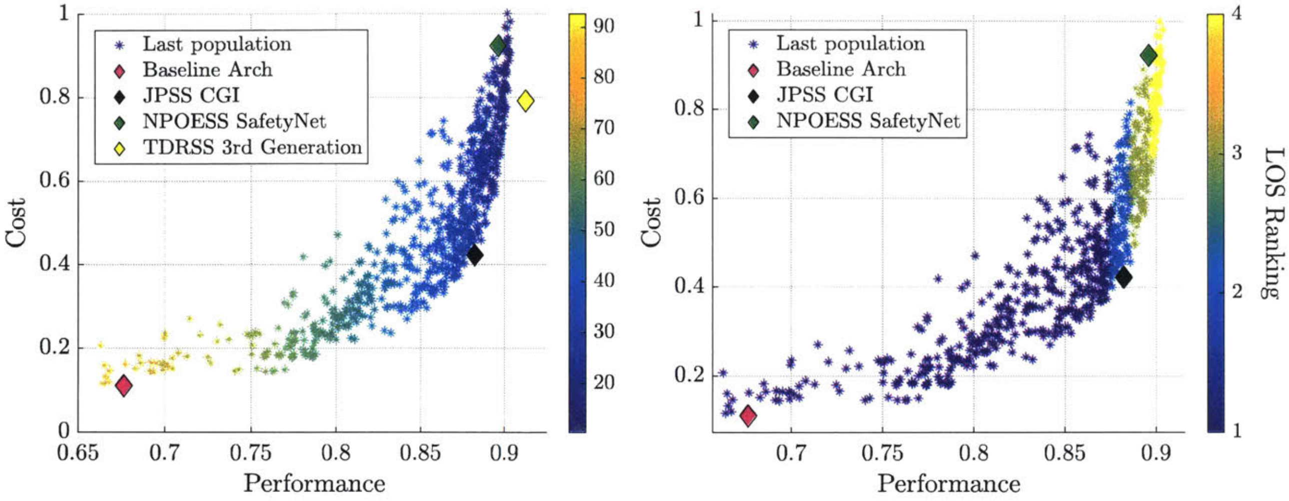
Figure 4-29: Ground vs. Space-based Network Scalability

the system. Interestingly, it can be observed that even for a JPSS-like architecture line of sight is no longer the largest latency contributor and therefore further upgrading the ground system should probably not be the primary concern. Indeed, further analysis through the proposed centrality measure reveals that, in fact, the data processing center rapidly becomes a bottleneck, specially for MetOp satellites (see Figure 4-31b).

Similar results are observed as we move up in the Pareto front and the line of sight ranking decreases. In order, the next latency contributors that progressively dominate the end-to-end system performance include: Processing facilities, downlink capacity and finally WAN capacity. Note that the distribution network never becomes a dominating factor since dedicated transfer speeds of 150Mbps within CONUS are already available. Therefore, we conclude that evolving the architecture from the baseline system to a JPSS-like architecture can be conducted without necessarily considering the rest of the latency contributors as they are in fact a second order effect. However, if the selected architecture is more capable than JPSS's CGI, then other factors need to considered and potential investments in upgrades on the processing infrastructure could be more cost effective.

Step 4-7. Development of Recommendations

Based on the results presented in Step 4-5 and Step 4-6, the following recommendations emerge:



(a) Relative importance of LOS Contributor

(b) Ranking of LOS Contributor

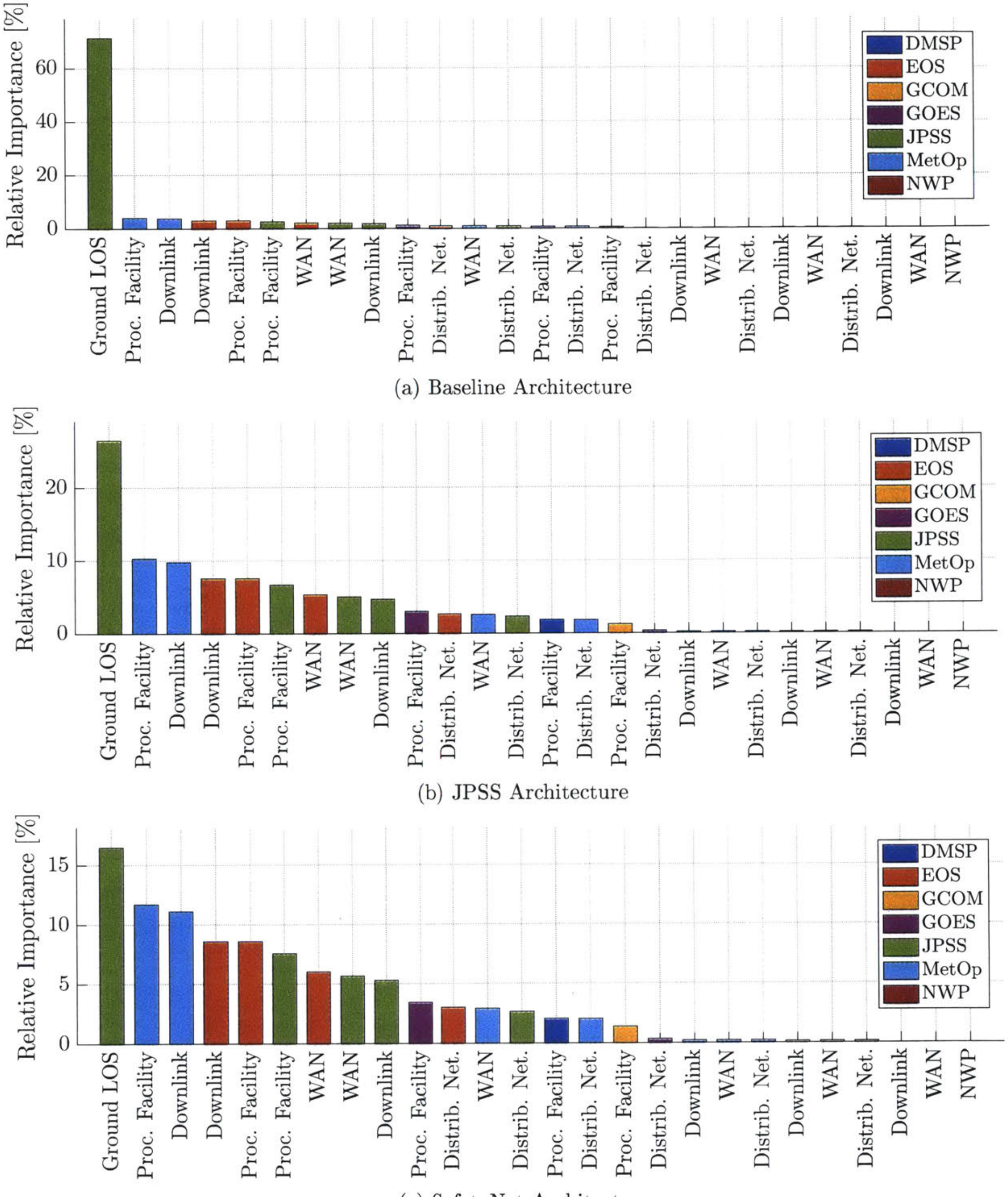
Figure 4-30: LOS Contributor Importance

Question 1: Which data products are more important for producing adequate global weather forecasts?

Atmospheric radiances and wind profiles are the most importance data products for generating accurate global forecasts. Cloud characterization is becoming increasingly important, albeit some NWP centers do not assimilate it due to problems in integrating 2D cloud images with 3D atmospheric profiles. Finally, other data products such as atmospheric composition, aerosols and radiation budget, as well as surface characterization have marginal importance for global forecasting, albeit they become essential for local and short term weather forecasting.

Question 2: What is/are the most influential latency contributors in the current baseline architecture?

Current LEO systems utilize only one or at best two northern polar ground stations. As a result, the lack of line-of-sight between a satellite and a ground stations results in an average latency of 55 minutes approximately (almost 90 minutes in the worst case).



(a) Baseline Architecture

(b) JPSS Architecture

(c) SafetyNet Architecture

Figure 4-31: Latency Contributor Ranking for Reference Architectures

Question 3: Does the current JPSS Common Ground Architecture provide a good compromise between performance and cost? How does it compare to the previously proposed SafetyNet system?

The current JPSS ground architecture does provide a good compromise between performance and cost. The system will be able to deliver 28% more data per NWP assimilation cycle at approximately twice the cost of the current system. Extending the network to 15 ground stations only delivers an extra 5% of data per assimilation cycle at four times the cost of the current system.

Question 4: What other Pareto efficient architectures can be devised at different levels of costs?

Low cost systems can easily deliver breakthrough latency (e.g. <15-20 minutes) for mesoscale, medium range weather prediction systems. For global prediction systems, breakthrough latency can be guaranteed on average with multiple polar ground stations, but data from certain high latitude areas might be delayed as much as 40 minutes.

Question 5: Are space-based networks a cost-effective solution for delivering near real-time (<15 minutes) services to NWP satellites?

Satellite systems like TDRSS become cost-effective for networks that require more than 10 ground stations distributed across all continents, since they can be built with approximately the same life cycle cost and yet deliver continuous coverage. However, they are less scalable than ground-based networks (a 10 minute pass is at least four times more expensive) and therefore their desirability should be carefully evaluated if data acquisition functionality for all LEO weather satellite programs is consolidated.

Question 6: How does programmatic risk impact the selection of ground stations for a network that supports latency-constrained applications?

There is an inherent tension between performance and risk when architecting ground-based networks. In that sense, to obtain a good compromise between performance, cost and risk, consider utilizing mid/high-latitude sites in politically stable countries such as Germany or the United States complemented with a one or two polar sites. For fully distributed networks such as SafetyNet, globalize the network sites as much as possible to increase coverage and diversify risks, and nationalize ground assets in order to foster collaboration between the network owner and the host country.

Question 7: How do other latency contributors affect the selection of a ground network architecture?

Line of sight visibility is the dominant latency contributor for networks with less than four polar ground stations. More capable networks will also be heavily constrained by lack of computational resources, insufficient downlink capacity and finally insufficient bandwidth to repatriate data to continental US.

4.3 Lessons Learned from JPSS's CGI and NPOESS SafetyNet

Given the results of this case study, I now take a step back to review the history of JPSS' and NPOESS' ground infrastructures. In a sense, this is the final step of a validation exercise: Compare the results of "the model" (i.e. the centrality measure and its subsequent analysis) against realistic data. To initiate the discussion, I focus first on the NPOESS program as it chronologically predates JPSS.

The NPOESS was originally conceived as a unified system that would replace and merge all US weather satellite programs. From the beginning, the DoD levied a tight 15 minute end-to-end latency requirement for all EDRs, including those related to climatology that were generated on a daily, weekly and even monthly scale [18]. To achieve this unprecedented level of performance and be able to return the large amounts of data collected by the NPOESS spacecraft, the program created SafetyNet, a highly distributed network architecture with up to 14 sites interconnected with a highly capable fiber optic WAN to feed data directly into four state-of-the-art processing centers.

Using References [18] and [48], I traced the specific set of elements in the system that where upgraded or built during SafetyNet's development. They included:

1. Building 15 unmanned 3.6 meter antennas operating at Ka-band to provide an average of 5 contacts per orbit at high data rate (150Mbps).

2. Implementing significant hardware improvements on the ground processing side, most notably four fully redundant centers using large parallel computing capabilities and large caches to minimize I/O delays.

3. Upgrading the legacy S-band communication payload of previous POES and DMSP satellites to a Ka-band system able to accommodate the required data volumes for the mission.

4. Leading the commission of a dedicated submarine cable to provide high data rate communications to and from Svalbard, the only polar station in the system that was not directly supported by high capacity lines at that point in time.

Interestingly, note that the four actions undertaken by the NPOESS program map one-to-one to the set of latency contributors flagged by the centrality measure as important in Figure 4-30b. Indeed, when the line of sight contributor is reduced to minimal levels as is the case with NPOESS, its contribution to overall utility loss is so small that its rank switches from first to fourth position. Furthermore, the three additional contributors to consider include data processing, data downlinking and data repatriating.

A similar historical exercise can be conducted for the JPSS. In this case, I used References [18], [82], [195] and [196] to assess the primary set of upgrades implemented by the CGI as compared to previous ground systems to successfully satisfy the data product end-to-end latency requirements. These included:

1. Transition from a single polar ground station in the Northern hemisphere to two polar ground stations per hemisphere. The second site in each pole is provided mainly for redundancy purposes after the Suomi-NPP mission experienced performance problems when the undersea cable that connects Svalbard and US was disrupted.

2. Upgrade the data processing center at NOAA to accommodate the mission latency requirements. This endeavor partially motivate the transition of NESDIS to an Enterprise Architecture that minimize function duplication across missions and reduced costs.

3. Replace the heritage Suomi-NPP X-band communication system for a newly developed Ka-band system. This change was triggered by a requirement to provide backup science downlink capabilities through the SN, which cannot operate at X-band.

Note that no improvements for the WAN and distribution networks were needed. In fact, the current implementation returns data from the south pole stations using commercial satellite systems (e.g. Optus D1) with very limited data rate (10Mbps), and yet no upgrades to the NSF infrastructure at the McMurdo station will be implemented to satisfy the end-to-end latency requirement. These findings once again align with the rankings obtained through the centrality measure. Indeed, in Figure 4-30b the CGI system is in the region of the tradespace where line of sight is second only to processing, but upgrading the WAN networks that repatriate and distribute the data to the NWP centers is not necessarily required. Finally, observe that the JPSS system also implemented upgrades in satellite downlinking functionality. However, these were mainly due to continuity of operations through TDRS and therefore are not accounted for in the proposed approach as it explicitly excludes valuation of the system under contingency situations (see Section 1.5.1).

4.4 Summary

This case study has focused on return of atmospheric satellite measurements for accurate weather forecasting. In particular, I have concentrated on the problem of latency, i.e. how delays in the communication infrastructure that supports weather spacecraft reduces the amount of data available for each periodic assimilation run. To initiate the analysis, I first studied which parts of current weather forecasting systems induce latency as function of the program that delivers the observations. Using this information, as well as the relative importance of different measurements, I identified line of sight as the primary contributor to consider when delivering timely satellite-based observations for weather purposes.

Given these findings, the rest of the case study concentrates on the problem of optimal ground station placement to support periodic downlink of information from Earth orbiting spacecraft. In that sense, optimality has been analyzed both from the perspective of performance and cost, as well as performance and risk. For the latter, a quantitative metric assessing programmatic risk in ground networks has been adapted from the well-known concept of risk-adjusted cost of capital. Results indicate that selection of sites only on performance and costs considerations yields sub-optimal decisions from the risk perspective, most notably problems related to lack of logistic support, political instability and expropriation. To mitigate them, I also studied the possibility of supporting weather satellites from space-base relay system and compared them with respect to ground-based infrastructures.

On the other hand, this case study has also demonstrated the usefulness of the utility measure presented in Step 4-2, and how to utilize it as guide for the system architecting process. Not only has it been useful in quantifying the network performance, but it also has allowed us to explicitly quantify at which point certain contributors become the leading concern to be addressed by the system architects. In particular, comparison between the evolution of two reference systems, namely the NPOESS and the JPSS, has demonstrated that during the actual system implementation both programs took into consideration similar latency contributors to invest their resources in.

5 Case Study 3 - Support of Human Exploration Activities at Mars through the Mars Relay Network

5.1 Introduction

The exploration and colonization of Mars has been a topic of interest and study for more than four decades. Ever since the first pictures of the Red Planet were returned in 1965, NASA and its partners have sent numerous spacecraft and robotic rovers to better understand its history, composition, atmosphere, and ultimately answer the question of whether there is or ever was life on Mars [11]. To support this endeavor, a large communication infrastructure has been put in place both here on Earth with the DSN, and on the Red Planet with the MRN. Together, they ensure timely transmission of data to all orbiting spacecraft, as well as rovers and landers on the planet surface [197], [198].

The goal of this case study is to analyze the evolution of this interplanetary network to support more demanding exploration activities and, ultimately, the Red Planet's colonization. To that end, I first and foremost focus on understanding how communication and navigation services should be provided to astronauts on the Mars surface to enable effective science activities. That being said, the set of customers assumed for this case study spans beyond astronauts and includes other notional orbiters and rovers, as well as supporting equipment such as ISRU modules.

Similar to Chapter 4, the rest of this chapter is structured based on the latency-centric

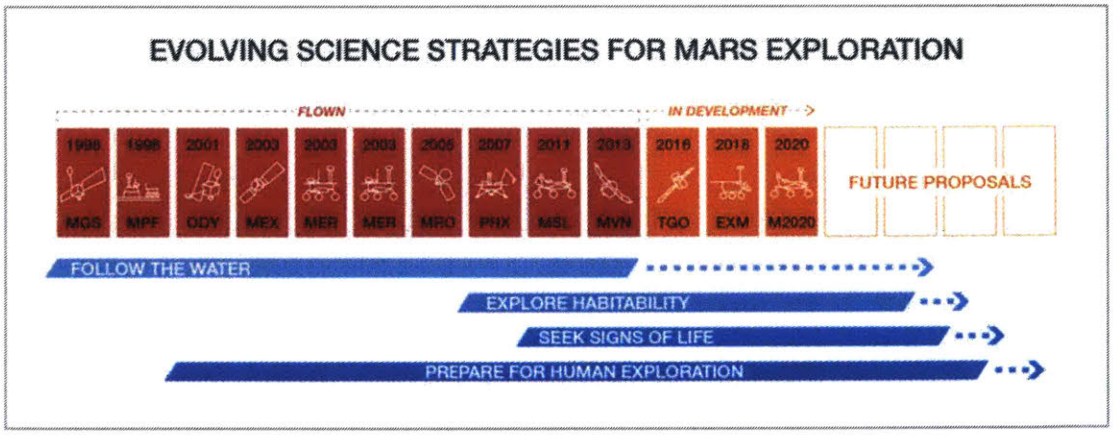


Figure 5-1: Mars Mission Time Line in Last 20 Years (Adapted from Reference [11])

approach to architect space communication networks. It starts by further motivating the problem at hand, as well as calibrating the centrality measure that guides the system architecting process. Then, the contribution of multiple latency contributors in the current implementation of the system is analyzed in order to understand primary latency contributors. Finally, recommendations on how to implement the network under different science operational modes is analyzed, compared and discussed.

5.2 Framework Application

Step 5-1. Motivation and Domain Specific Expertise

Communication and navigation services at Mars are currently provided as a collaborative effort between different entities within NASA. On the ground side, the DSN is responsible for providing all infrastructure that allows data to be routed from deep space to the MOC and vice versa [42]. The DSN is complemented, on the space side, by the MRN, which is composed of multiple Mars orbiters that relay communications to and from the planet surface back to Earth [197]. Standardization of protocols and technologies has already implemented through CCSDS standards, most notably the Proximity Link Protocol (see, for instance, References [199] and [200]), and the Electra payload that is carried by both American and European spacecraft orbiting the Red Planet [201].

A key characteristic of the MRN is its opportunistic nature. Instead of building an expensive, dedicated, high performing relay network, NASA and its partners opted for a more conservative approach: All spacecraft sent to Mars carry a standardized communication payload that effectively turns the satellite into a hybrid science-communication system [202]. During the primary phase of the mission, spacecraft operate only in scientific mode. Then, the Electra payload is turned on and the satellite is used to relay information from the surface rovers to Earth [197]. This approach has clear advantages in terms of programmatic cost, as no dedicated relay systems have to be designed, built and launched. At the same time, it also results in significant compromises in terms of communication performance, as hybrid spacecraft are always placed in orbits optimized for science collection rather than communication coverage.

While the current paradigm for the MRN has been successfully addressed the needs of robotic exploration endeavors for the last two decades, its ability to meet the stringent requirements of human activities in the Red Planet is unclear. In fact, the baseline architecture for NASA's Mars human exploration, known as Design Reference Architecture 5.0 (DRA5.0), assumes the pre-deployment of dedicated high capacity communication satellites [203]. Yet, no indication of the size and costs of such relay satellites is specified. Therefore, the primary goal of this case study is to quantify the infrastructure cost, both on the ground and around the Red Planet, required to successfully support human exploration activities. In particular, I utilize the latency-centric approach to architecting space communication networks to first identify the primary bottlenecks of the current Mars-Earth network, and then quantify the savings obtained if science activities are not conducted in real-time[1].

In summary, the specific set of research questions that motivate this case study are:

1. Is it necessary to provide continuous communications to astronauts at the surface of Mars to perform science activities efficiently and effectively?
2. What are the infrastructure savings that result from not providing continuous communications to perform the aforementioned science activities?

Step 5-1.1. The Mars Ecosystem

Before proceeding to the calibration of the centrality measure, it is worth spending some time clarifying the Earth-Mars ecosystem assumed in this 2040's case study. Both robotic and human exploration activities are conducted at the surface of Mars through a combination of crewed and uncrewed landers, mobility rovers and astronauts. On orbit, Mars and its moon Phobos are orbited by scientific spacecraft, ascent vehicles, Mars insertion orbiters, as well as a Deep Space Habitat (DSH) stationed at a highly elliptical orbit. All of them are supported by a communication infrastructure evolved from the current MRN, together with the DSN, and mission/science control centers assumed to be at JPL (see Figure 5-2 for a schematic representation of the system). Four primary types of links are required:

- Surface-to-surface links: Astronauts communicating to other ground elements utilizing a WiFi-like network [204].
- Surface-to-orbit and orbit-to-orbit links: All communication links within the vicinity of the Red Planet will be termed *proximity links* in reference to the Proximity-1 CCSDS protocol that is currently used in the MRN [199], [197].
- Mars-to-Earth links: All direct links between Mars and Earth will be termed *direct-to/from-Earth* (DTE/DFE), regardless of their origin. Two types of DTE/DFE links will be present in the system: Critical DTE/DFE, transmitted a lower frequency band and carrying all critical communications; and non-critical DTE/DFE, equivalent to a trunk line between Mars and Earth that multiplexes all large volume data streams.

Four types of network customers are present in the system:

[1] Throughout this chapter, the term *real-time* will be used to refer to communications with no latency allowance without taking the light-time delay into consideration.

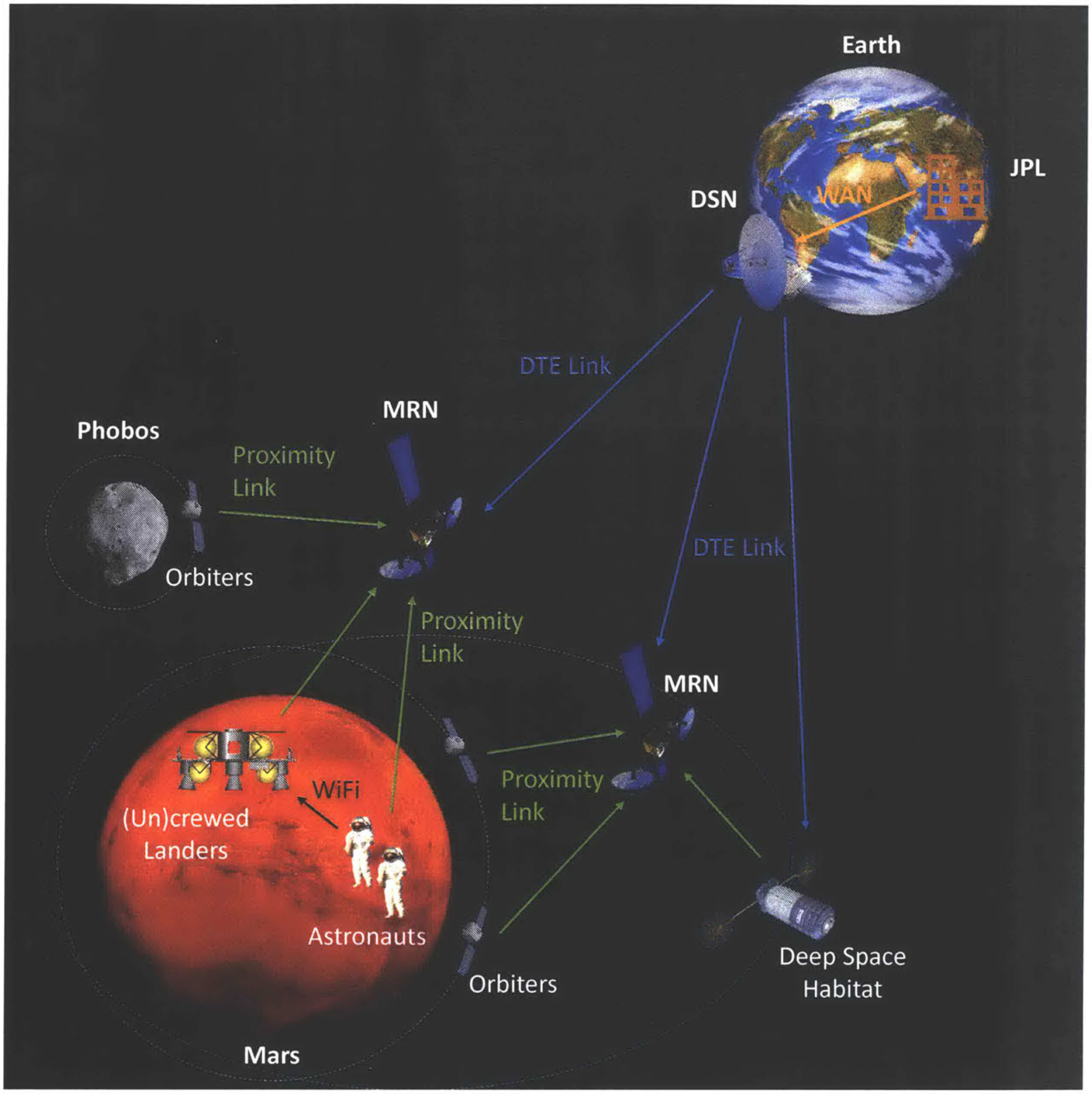


Figure 5-2: Mars Ecosystem

- Mars surface robotic elements: Any robotic surface assets that conduct scientific activities or provide operational support to astronauts on the Mars surface. These include non-crewed landers, portable utility pallets, or oxygen and water processing systems from ISRU technology.
- Mars surface human elements: Any surface assets that are used by astronauts to either live or conduct their scientific activities. It includes crewed landers, surface habitats, pressurized rovers and EVA suits.
- Mars orbit robotic elements: Any spacecraft orbiting Mars or its moons (e.g. Phobos), as well as Mars ascent vehicles and possible EDL test vehicles.
- Mars orbit human elements: Any orbiting elements that carry astronauts, most notably a DSH stationed at a highly elliptical orbit or human habitat at Phobos.

Differentiation between these four categories of customers helps reduce the complexity of the problem by grouping them based on their communication needs, most notably the services required and their criticality. This will be advantageous in latter steps of the latency-centric approach to architect space communication networks since it will facilitate the definition of heuristic rules for how data is routed and assigned to frequency bands.

Step 5-2. Specify the Centrality Measure

Step 5-2.1. Characterization of Latency Contributors

The first step to specify the system architecting centrality measure is to understand how the current system is implemented, both in the functional and formal domain, as well as understanding the primary latency contributors to be considered. To that end, Figure 5-3 decomposes the current Mars-Earth system in its constituent elements, i.e. it basically provides a level 2 decomposition in the formal domain. Elements present in the network include (1) the DSN centers with their antennas, (2) the MRN in its current implementation considering both American and European spacecraft, (3) the expected customers at Mars during the human era, and (4) the offices that perform service management functionality such as network and operations scheduling. Additionally, the numbers in each cell indicate the presence of a given system functionality implemented during the interaction of two nodes. Functions listed in the top triangular part of the DSM are representative from data uplinks (from Earth to Mars), while the bottom triangular part details the downlink portion of the system. For the latter, I assume that all communications are routed through the MRN while uplinks are directly established with the Mars surface rovers.

Based on the system decomposition from Figure 5-3, five canonical nodes are identified: The DSN is represented by the large blue box and performs all communication functionality to

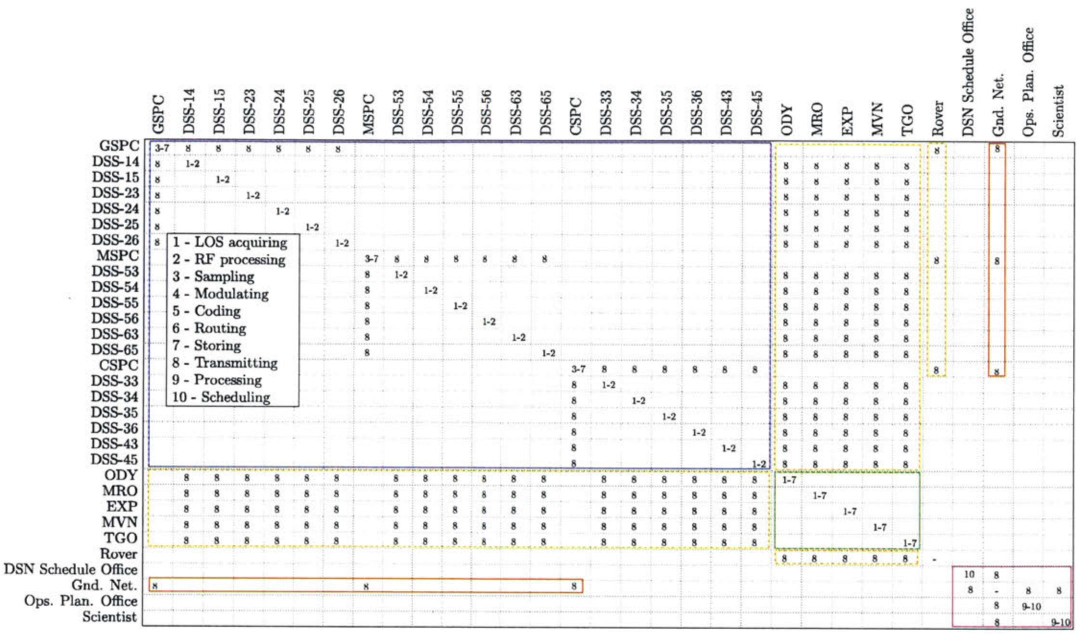


Figure 5-3: Current DSN and MRN Architecture in DSM Format

ensure reliable reception of data from Mars. Buffering at the site is sometimes performed to save bandwidth in the ground network. On the other hand, the yellow boxes represent all transmission functionality to and from the Red Planet, both over the deep space and proximity links. Next, the green box is representative of the MRN and includes functionality such as LOS acquisition, data demodulation and storage for retransmission. Finally, the red and magenta boxes are representative of all functions performed on Earth to maintain the daily science decision cycle. It includes the WAN used for repatriating information from all DSN sites, as well as the scheduling and planning offices that process data returned from the remote laboratories.

Based on this initial system decomposition, Table 5.1 summarizes the different physical nodes assumed in the system, their primary functionality and latency contributors. Next, I provide a brief discussion for each of them with specific emphasis on how latency is induced and which drivers should be considered when assessing their importance with respect to the end-to-end performance.

As previously mentioned, the current MRN provides proximity links through the standardized Electra payload [197]. Its specifications and capabilities are limited due to three primary considerations: First, it only has to support rovers with limited data generation capabilities, several orders of magnitude lower than what humans would require (MSL has a daily "allowance" of 200Mbit/sol [205], [I21]. In comparison, a HD video stream generates this amount of data in less than a minute); second, it operates at UHF, with a total bandwidth allocation of just 15MHz based on SFCG recommendations [I21]; and third, it is designed to minimize SWaP requirements on the science orbiters that carry it. The consequence of

Table 5.1: Summary of Latency Contributors in the Mars-Earth Network

Physical Node	Node Functionality	Latency Contributors
Proximity Link (PRX)	Mars data transmitting	Prop. & Tx. time, error correcting mechanism
Mars Relay Network (MRN)	Data storage and relay	LOS acquisition, data storage
DTE link (DTE)	Deep space transmitting	Prop. & Tx. time, error correcting mechanism
Deep Space Network (DSN)	Data Earth receiving	Signal proc.
DSN Schedule	Service scheduling	Schedule constraints
Wide Area Network (WAN)	Data repatriating	Ground proc. and tx. time
Operations plan	Command generation	Sequence generation and validation
Scientists (JPL)	Data processing	Science and instrument planning

all these restrictions is a design with limited data rates (1.024Mbps), that induce latency by returning data collected by rovers over one or two 10 minute pass durations despite limited data volume allocations. Note that other mechanisms such as retransmissions of the CFDP protocols or propagation time should also be considered in the proximity link. In that sense, it is easy to demonstrate that they have minimal contributions for this part of the system given that ARQ mechanisms can run efficiently in the Mars vicinity due to limited frame error probability and propagation times [206].

The current MRN also induces latency by providing limited contact opportunities with the Mars rovers. Due to its hybrid nature, all relay orbiters have primary scientific requirements that can only be satisfied if the spacecraft is positioned at low Mars orbit. This approach maximizes science value delivery from the orbiter instruments perspective, but is sub-optimal for communication purposes as it only provides intermittent coverage to Mars rovers. Therefore, an important latency contributor of the current system is LOS acquisition as dictated by orbital mechanics. Furthermore, even if LOS is available at a given point in time, other programmatic factors can also induce latency. For instance, relay operations are only enabled during the extended phased of the orbiter's mission to ensure that science goals are first addressed. Similarly, data uplinked to an orbiter cannot be typically relayed to the ground directly unless DSN contact time has been properly granted. In that sense, orbiters such as MRO currently have high priority in the DSN scheduling process and therefore coverage gaps after a rover-orbiter pass are unusual. Yet, they occur in some special instances when either the DSN resources are tied up in especial events or cross-support with other agencies, or the rover-orbiter pass occurs during planet occultation [42].

The DTE link between the current MRN orbiters and the DSN is also significantly constrained and induces significant latency. In particular, two primary contributors must be considered: On the one hand, one-way light time propagation delay between Earth and Mars varies between 4 and 20 minutes approximately depending on the two planets' synodic cycle. On the other hand, latency is also induced by the "custodian" mechanism implemented in all MRN orbiters (only Odyssey has the ability to transmit data back to Earth in bent-pipe mode [I21]). In particular, data uploaded to an orbiter is stored until the end of the proximity pass has ended. Then, it is relayed back to Earth according to a predefined priority order: First, rover critical data; second, rover science data; third, orbiter telemetry; and finally, rover science data [I21]. On top of that, all critical data is sent to twice to Earth to avoid errors upon reception. In other words, the "custodian" mechanism currently implements a repetition scheme that complements FEC codes to ensure error-free data delivery. Yet, sometimes scientists have to wait for a delayed second copy of the data before they can start processing it.

Within each DSN site latency is introduced due to two primary mechanisms: First, low level DSP functionality such as synchronization of demodulators and decoders is required and necessitates multiple frames to properly configure. These delays, though minimal if compared to other latency contributors, can be significant for services that have low data rates and can only tolerate seconds of end-to-end delay. On the other hand, DSN sites are also responsible for generating and forwarding data to the MOC through protocols such as the CCSDS's SLE. This forwarding process can be performed immediately, or data can be buffered at the site and sent back to its final destination at lower data rates to save ground bandwidth [I21]. Research has shown that the latter approach can significantly reduce the capacity requirements for the international ground network that connects the DSN sites [58]. Therefore, if this approach is utilized, latency can be induced by the ground network that repatriates the data to its final destination.

Finally, the last set of latency contributors are incurred while data is being processed on Earth by ground operators and scientists. Reference [207] provides a high-level overview of all activities conducted on Earth once data from the rover is received from the previous sol (see also Figure 5-4). Two to three hours are currently utilized for data preparation and analysis prior to activity planning. Typical activities performed include site recognition, target approach assessment or instrument status analysis. Following the activity planning tasks, commanding sequences for the next day of rover operations are built and validated. This is a lengthy process that takes 3 to 4 hours, and concludes with a command approval meeting where the next uplink to the rover is finalized. Finally, commands are transfered to the DSN antennas that have been properly scheduled to directly contact the rover using an X-band DFE link.

Step 5-2.2. Identification and Characterization of Data Flows

Four primary data flows are required to operate rovers on the Mars surface:

1. Uplink of commands at the beginning of each sol.

2. Downlink of critical and quick-look data at the end of each sol

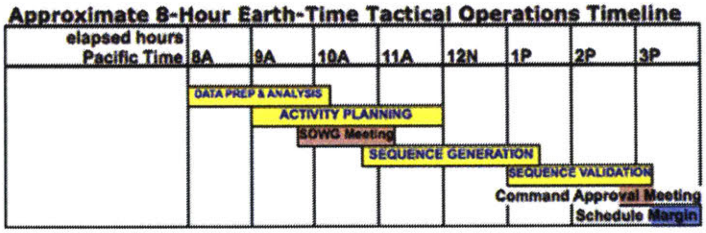


Figure 5-4: MSL Science Planning and Operations Cycle

3. Downlink of bulk data during Martian night pass.

4. Schedule of DSN and MRN assets to effectively support the surface rovers.

Command uplink from Earth to Mars is scheduled each day so that information reaches the rover at the beginning of the Martian day. In a sense, they are the robotic equivalent of an astronaut morning briefing, where the rover is told what to do and where to go during the next six hours. Thanks to its inherent autonomy, the rover then executes the set of commands provided in the morning uplink without supervision or feedback from Earth. Note that there is a tight requirement in terms of data delivery time for command uplink. Indeed, rovers are only allowed to operate during a limited set of daylight hours in order to ensure that energy constraints are always met and enough power is left for transmission of data back to Earth. Therefore, the command uplink is a critical data flow for maintaining the once-a-day science feedback loop for MER and MSL.

Flows 2 and 3 are representative of the network return direction, i.e. from Mars to Earth. Interestingly, each rover gets two different passes to return data, one at the end of the Martian day and the other one during the Martian night. The former is used to return all critical engineering and quick-look data that will then be used by the rover operators to make tactical decisions on which tasks and activities should be performed the next day. In contrast, the mid-night data flow typically carries bulk science data that can be considered latency-unconstrained and is only used for strategic decisions, i.e. long-term planning of where the rover should drive to.

Flow 4 represents all coordination activities required to properly schedule the DSN, MRN and rover assets to support science operations. On the DSN side, scheduling is typically performed two weeks in advance based on predicted pass times as estimated from available navigation data. Yet, changes in real-time could be implemented should contingency situations arise. On the other hand, the MRN and surface rovers require coordination to ensure that proximity passes are scheduled at the appropriate instants of time without significant impacts on the science activities being conducted by the orbiters. In that sense, pass opportunities for both MRO and ODY are typically not utilized in the middle of the Martian day to allow the rovers to complete all scheduled activities autonomously. Similarly, RF interference problems between MER and MRO have resulted in special coordination activities between the two projects and the DSN to ensure that MRO uplinks are properly turned-off when MER uplinks are required [208]. Finally, scheduling activities for rover assets are related to building and validating commands to have the platform instruments take measurements and move from its current location to a desired sampling spot. They are critical for the tactical decision process since they ensure the remote platform's healthy status and, therefore, its continued operations.

Since flows 1, 2 and 4 are all necessary to successfully perform science activities on the surface of Mars based on a tactical decision process, they must all be considered when ranking latency contributors. In that sense, I assume they all have the same relative weight, i.e. they are all equally critical, while flow 3 has zero weight since it only influences strategic decisions. Furthermore, for flow 4 only latency incurred by activity planning, and sequence generation and validation is considered. Other sources of latency such as unscheduled DSN time or "keep out" windows due to RF interference are assumed to be anecdotal[2] from the perspective of expected system performance and are therefore neglected.

Step 5-2.3. Characterization of Data Utility

Figure 1-9 that notionally describes the operations of MSL. The rover is operated using two downlinks and one uplink per Martian sol. At the beginning of each day, the rover receives commands that describe the tasks to perform that day. During the next 6 hours, while in daylight, it conducts the necessary roving and science activities, and then returns all critical data at through a relay pass scheduled at the end of the sol. This data is analyzed by mission planners at JPL, and the activities for the next sol are planned, specific commands are generated and finally validated. Additionally, non-critical data not necessary for planning the next day of operations is returned, if needed, through a night pass with the MRN. From this description, it is apparent that MSL returns two types of data: On the one hand, latency-sensitive information is returned in the afternoon pass so that it can be used to plan the next day of operations. It is categorized as latency-sensitive since its delivery must be performed with enough time to allow JPL planners to successfully analyze it and deliver a good activity plan for the next day. On the other hand, there is latency-insensitive information that is returned over the night pass and does not necessary influence the next day of operations, but is rather integrated in the long-term strategic planning process.

Although humans are significantly more capable and autonomous than current Mars rovers, conducting science activities without feedback from scientists on Earth is highly unlikely [209]. For instance, during the Apollo era communication infrastructure was put in place

[2] Note that this does not mean unimportant. Indeed, successfully managing these contingency events is essential to the correct operation of all assets around Mars.

to allow scientists on Earth to be involved in the activities conducted by astronauts on the surface of the Moon. In particular, "the science backroom role was to understand and support traverse activities in real time, using suit-mounted and rover-based video streams and data" [209]. Importantly, note that this was possible in the case of Apollo missions due to short light-time delays between Earth and the Moon. On the other hand, NASA analog missions such as DRATS and NEEMO [210] have also explored the ability to conduct science activities using the MER and MSL model. In this case, scientists on Earth are divided into the *tactical team* and the *strategic team.* The former is intended to support activities in near real-time throughout the day, while the latter analyzes all the obtained data at the end of daily operations and provides recommendations on how to improve the tactical process and, if needed, re-plan the next day science [209].

Ultimately, the choice between Apollo's or MER/MSL's concept of operations as studied in NASA's analog missions is based on science output. In that sense, it was found that the "MER model had significant advantages over the Apollo model in that it decreased individual workloads and increased the scientists' ability to evaluate specific scientific hypotheses through individual analysis and interaction among science team members" [209]. Yet, some drawbacks also emerged: The strategic backroom had to process large amounts of information to function smoothly, all of which happened overnight and incurred increased fatigue for the team members [209]. This issue, also reported for current Mars rovers, has been addressed by allowing the science team to work according to Earth time rather than Mars time, and allowing for infrequent "restricted sols" to occur in which no downlink data is available for planning the next day [70]. Finally, experience with the Mars rovers demonstrates that having a backroom engaged and ready to receive and process data from Mars all the time is challenging from the perspective of personnel management on Earth. In fact, it is well documented that an important part of the learning process of operating rovers at Mars was related to having teams not work overtime (or overnight) and yet ensure that its science instruments are optimally utilized [211], [70].

To mimic the distinction between Apollo's or MER/MSL's concept of operations, I created four canonical time lines that define the cadence with which data would move between Earth and Mars. The primary variable modified in each of them is the number of feedback loops allowed between on-site astronauts and Earth-based science teams. Next, I describe their primary characteristics:

- Near real-time (NRT): At all times there is an active link between the astronauts and the surface of Mars. Latency introduced by the network that supports the Mars ecosystem is minimal (not taking into consideration the light time delay), thus allowing the operations to be conducted analogously to the Apollo program.

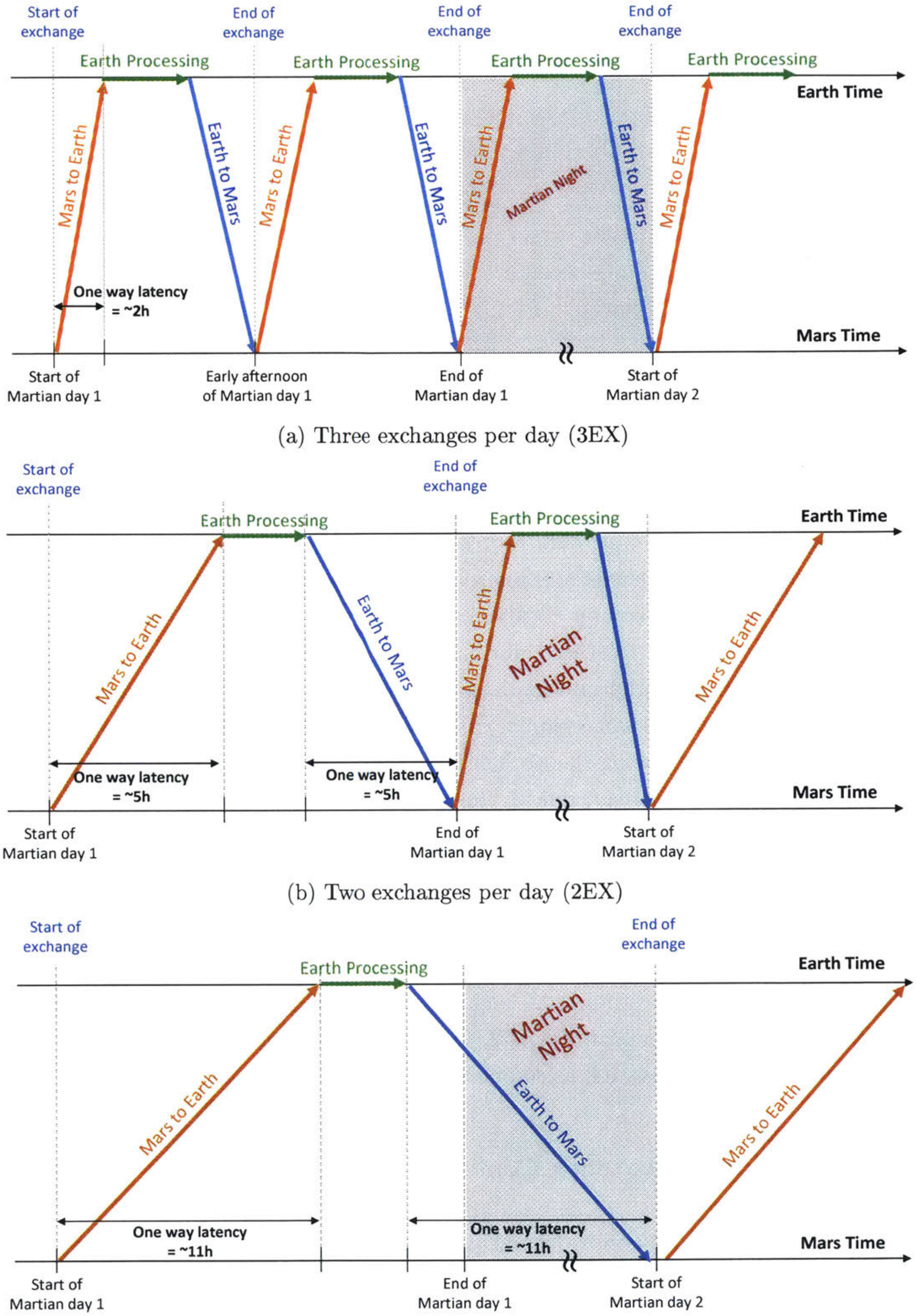
- Three exchanges per day (3EX): Every day the astronauts are granted three exchanges per day with enough bandwidth to return all scientific data. They include a morning briefing contact, a mid-day tag-up and an evening debriefing pass (see Figure 5-5a). This is equivalent to an enhanced MER/MSL concept of operations.

- Two exchanges per day (2EX): Similar to the 3EX concept of operations, but without the mid-day exchange. This is analogous to current MER/MSL concept of operations (see Figure 5-5b).
- One exchanges per day (1EX): In this degraded mode of operations, only one exchange per day is provided for science purposes at different times depending on the network availability (see Figure 5-5c).

Observe that, as depicted in Figure 5-5, each concept of science operations results in a different one-way maximum latency requirement for the system. The exact values utilized for this case study are as follows:

- NRT operations: 20 minutes of one way latency allowance, including 20 minutes for worst case one-way light time delay. Nominally, the turn around time of data sent to Earth is limited by how fast the backroom science on Earth can provide meaningful feedback. In some instances, 1 hour delays in parts of the data might be observed due to retransmissions of the CFDP protocol.
- 3EX operations: 1.7 hours of one way latency allowance, including 20 minutes for worst case one-way light time delay. It mimics a 8am, 2pm, and 8pm pass, Mars time. The nominal processing time at the Earth backroom is assumed to be 2 hours, but can be reduced to 1 hour if retransmission of the CFDP protocol is required.
- 2EX operations: 4.7 hours of one way latency allowance, including 20 minutes for worst case one-way light time delay. It mimics a 8am and 8pm pass, Mars time. The nominal processing time at Earth is assumed to be 2 hours, but can be reduced to 1 hour if retransmission of the CFDP protocol is required.
- 1EX operations: 10.7 hours of one way latency allowance, including 20 minutes for worst case one-way light time delay. The nominal processing time at Earth is baselined at 2 hours per day, but can be reduced to 1 hour if retransmission of the CFDP protocol is required.

These four modes of science operations were presented to 5 experts that have participated in past NASA analog missions. They were then asked to provide a qualitative score from 1 to 5 for each of them. Using an approach similar to Reference [189], I normalized these qualitative scores into the utility function presented in Figure 5-6. Interestingly, observe that the marginal benefit of delivering continuous communications for science and outreach purposes is significantly lower than between two and three exchanges per day. As one interviewee said, it is possible to design the science operations activities so that astronauts take advantage of the non-continuous communication structure [I19]: Use the morning pass



(a) Three exchanges per day (3EX)

(b) Two exchanges per day (2EX)

(c) One exchange per day (1EX)

Figure 5-5: Science Operational Modes

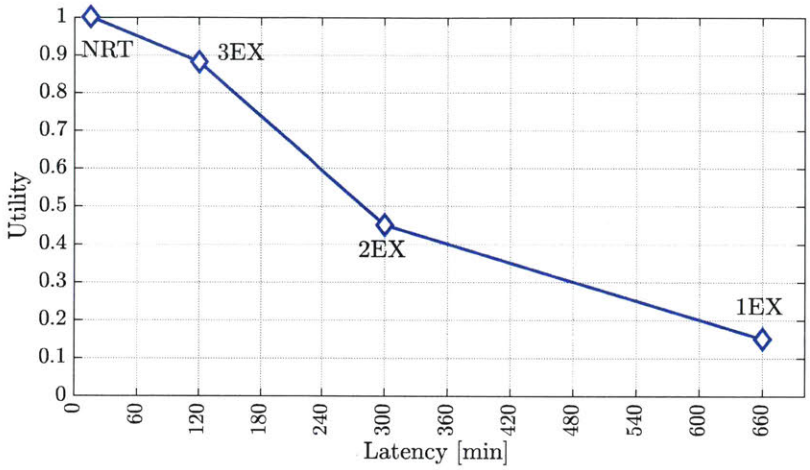


Figure 5-6: Normalized Utility Function from Expert Interviews

to brief astronauts on the activities to perform; conduct a morning EVA where they primarily map and tag an area of interest; data returned during this step is simultaneously analyzed on Earth by the tactical science room team and feedback on morning activities is delivered by mid-day; conduct an afternoon EVA to collect the previously identified samples; curate the samples and return all relevant information for the strategic science backroom team to process [I17]. Without the mid-day tag-up, the need for a tactical backroom science disappears[3] and part of the strategic backroom work can be offloaded to regular day time hours on Earth, thus facilitating the work of Earth scientist during long duration campaigns [I16]. Yet, the inability to react quickly to new information can significantly impact the efficiency of all science activities even if astronauts have significant training [I16].

Step 5-2.4. Definition of a normalization scheme

Similar to the previous case studies, all rankings and centrality measure outputs will utilize sum normalization as defined in Step 2-2. This facilitates understanding the extent of a latency contributor in the system as compared to all the other contributors.

Step 5-3. Ranking of Latency Contributors

Given the set of canonical nodes defined in Table 5.1 for the Mars-Earth system, Table 5.2 defines the DSM that is used as adjacency matrix to feed the centrality measure. The orange 1s are representative of the uplink flow of information, the while the blue and green 1s

[3] Part of the functionalities of the tactical science backroom could be provided by astronauts in the DSH. These would always be present independently of the science operational profile as I assume that there is always direct communication links from the surface of the planet to the DSH.

Table 5.2: Mars-Earth System DSM

		A	B	C	D	E	F	G	H	I
Rovers	A	▨	1							
PRX link	B		▨		1					
DTE link	C	1		▨			1			
Mars Relays	D		1	1	▨	1				
DSN schedule	E				1	▨	1			
DSN sites	F			1			▨	1		
WAN	G						1	▨		1
Ops. Plan	H				1	1		1	▨	
Scientist	I								1	▨

Table 5.3: Latency Contributor Quantification

Node	Latency [min]	Rationale
PRX link	10	Reliable CFDP with 1.24Mbps link (bandwidth limited)
DTE link	60	Reliable CFDP with 6Mbps link (propagation limited)
MRN	120	Revisit time of MRO and ODY, without scheduling constraints
DSN schedule	0	DSN assets always available unless contingency or critical event
DSN sites	<1	Low level digital processing, no store&forward performed
WAN	<1	Ground network performance with 45Mbps lines
Ops. Plan	360	Rover sequence generation and validation
Scientist	120	Science tactical decision time

represent the tactical downlink and service scheduling flows respectively. For the latter, the scientist is assumed to be the source of information. It then influences the science operations plan and command generation, which in turn affects the DSN and MRN schedules.

Table 5.3 lists the average latency introduced by each element in the system, as well as the primary rationale for its inclusion. Observe that the DSN schedule has a no expected latency contribution since it only impedes the return of data in a timely manner in the event of critical or contingency events. This is important from a systems perspective since the DSN influences other elements such as the rover operations teams. However, since they can count on practically unlimited access to the network, it in fact imposes no restrictions on how to design and implement the rover operations.

The three data flows defined in Step 5-2 and Table 5.2, as well as the utility function elicited in Step 5-2 and the expected latency contributions from Table 5.3, can now be used as inputs to the centrality measure defined in Equation (2.9). This results in the ranking of latency contributors from Figure 5-7. Observe that over 40% of utility loss is attributable to the time it takes for the science operations team to generate and validate sequences. However, this is performed independently from the network that supports rover operations and therefore does not have a direct infrastructure cost associated. Furthermore, this lengthy process is only required in the case of robotic rovers due to their limited autonomy. If humans are stationed on the surface of Mars, it is expected that all science activities are conducted by

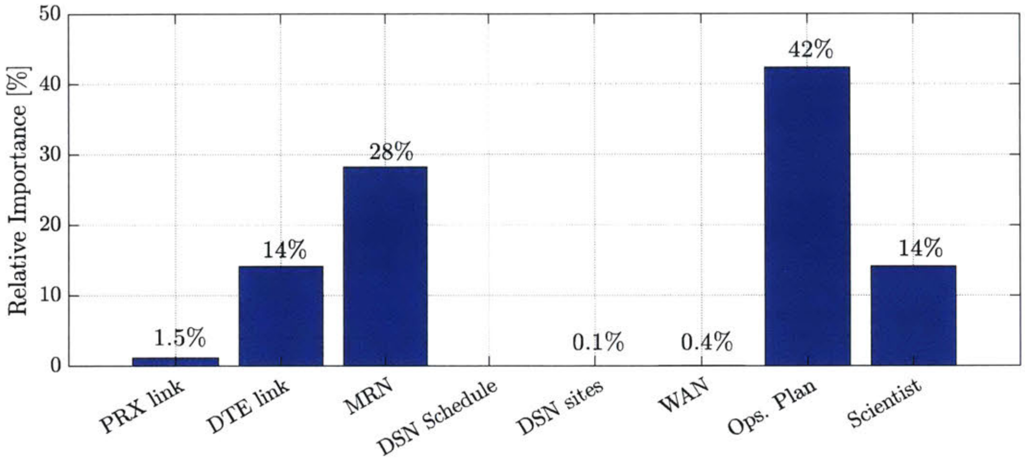


Figure 5-7: Ranking of Latency Contributors

them and therefore the need to generate commanding sequences to supporting equipment is largely reduced.

The second latency contributor to be considered is the science backroom, specifically the time it takes to generate a valid operations plan given data from the previous day of rover operations. In that sense, and from a NASA enterprise perspective, the preferred solution would be to spend resources so that efficient science activities at the backroom minimize the amount of time and effort spent providing feedback to astronauts. This would be implemented, for instance, by having scientists work overtime or overnight and paying extra hours, the cost of which would be compensated by lowering the communication requirements on both the Mars relay network and the DSN. That being said, experience with NASA's analog missions demonstrates that this approach is suboptimal from a science perspective, specially for long duration stays in the Red Planet. Indeed, scientists that are forced to work overtime for extended periods of time experience fatigue, specially if overwhelmed with large amounts of near real-time data from cameras that provide context information from the astronaut's current location [209], [212]. Similarly, it has been demonstrated that, when faced with time constraints, scientists tend to prioritize tactical decisions and, consequently, they are "unable to develop mature hypotheses and science rationales that potentially inform operational decisions for subsequent days" [209]. Therefore, in this case study I will assume that the time alloted to scientists to generate feedback is constant, and instead focus on other sources of performance loss within the Earth-Mars production system.

The third latency contributor to be considered is the MRN, specifically the fact that it is now optimized for science activities rather than communication coverage. The decision between a fully dedicated relay satellite system or a hybrid science-communication alternative from the perspective of constellation coverage and cost has already been studied in the literature. In particular, the Space Communication Architecture Working Group compared, in Reference [213], how to support astronauts on the surface of Mars using MRO-class orbiters, a cruise stage converted into a relay satellite, or a dedicated spacecraft placed at either an areostationary orbit, a critically inclined orbit or a circular 1000km orbit. They demonstrate that the areostationary alternative not only provides continuous coverage over the ±70 deg latitude range, but it result in better operability since no conflicting stakeholders are involved in the spacecraft operations. Similarly, it also increases system flexibility by providing low delta-V opportunities to re-adjust the coverage area after Mars insertion [213]. Therefore, they recommend that during the human exploration era dedicated relay satellites should be deployed, a finding that concurs with current analyses within JPL's Communications Architectures and Research Section and NASA's DRA5.0 study. Therefore, in the rest of this chapter I will assume that no latency is induced by lack of coverage from the MRN to its users.

Next, both the DTE link and science decision cycles are flagged as primary latency contributors by the centrality measure. For the latter, a baseline window of two hours has been assumed. It defines the total time allowance that scientists have from the instant in time when they first receive information from the rover to the time they have to deliver a valid tactical science and operations plan for the next day (without including command generation and validation). In that sense, the two hours have been selected as an expected value given current lessons learned for both MER and MSL, as well as the analog missions described in Step 5-2. That being said, observe that once again this latency contributor is not directly related to the cost of the communication network. Indeed, in this case the limiting factors are related to science assimilation time, which will be increasingly hard to diminish as new rovers and eventually humans generate and return more data to review [209]. Finally, I observe the DTE link is the first latency contributor in the ranking that can be effectively utilized to save cost in the overall communication infrastructure. Therefore, its architecture will be considered in the subsequent sections of this chapter.

Step 5-4. Problem Formulation

Step 5-4.1. Definition of Case Study Assumptions and Goals

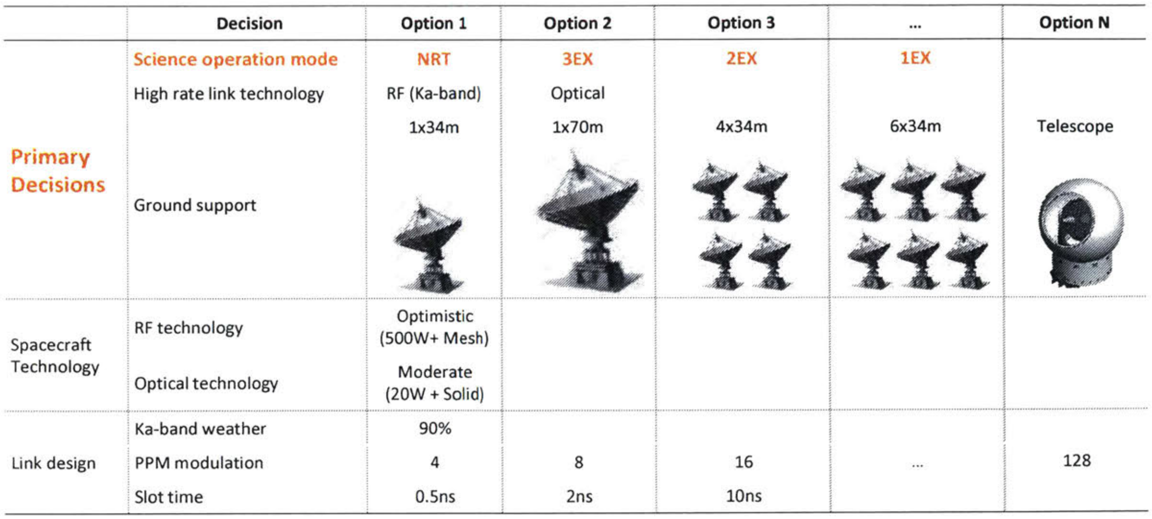
Given that the DTE has been identified as the primary latency contributor in the system that has a direct impact on NASA's infrastructure cost, the goal of this system architecting exercise is to quantify the mass savings of the DTE/DFE payload that provides the Earth-Mars link during human science exploration activities at the surface of Mars as a function of the science operational profile and DSN ground infrastructure.

The primary set of assumptions and constraints utilized in this part of the case study are as follows:

- Data is returned to Earth using two areostationary relay satellites, as well as the DSH. One of the relay satellite is always positioned in direct line of sight from the astronaut landing site.
- The exact astronaut landing site is unknown, but bounded by two constraints: First, it must be within ±60 deg of latitude; second, it must be at less 2 km of latitude [214].
- Orbits of all non-surface assets are not known except for their general configuration: The DSH is stationed at a high elliptical orbit; Phobos orbiters follow its trajectory; and all other assets are assumed to be at low Mars orbit.
- Frequency band allocations are based on the current SCaN services, as well as recommendations available from the SFCG.
- Modulation and coding for all links are based on recommendations from the SFCG and the CCSDS. In particular, I assume that GSMK modulation is used with a spectral efficiency of 1.29Hz/bps, and LDPC coding at 2/3, 3/4 and 4/5 is available.
- Technology available in 2040 for implementing the DTE link includes 500W TWTAs, mesh antennas with densities of 1-2kg/m^2, as well as lasers that deliver up to 20W of optical power.
- Non latency-sensitive data is always returned with high priority and without any latency allowance.
- Critical data is returned over a reliable X-band link with limited capacity. Data criticality is assessed based on application type and allowable latency as defined in Step 5-2. Furthermore, two copies are provided for redundancy mimicking the current MRN "custodian" mechanism.
- Non-critical data is returned over a trunk line implemented using either Ka-band or optical communications. One copy is returned and another one is stored on-board the relay satellites and DSH as part of the "custodian" mechanism.

Step 5-4.2. Definition of Architectural Space

Figure 5-8 provides a visual representation of the morphological matrix that defines the options available to design the DTE link between Mars and Earth. Three primary decisions are available: First and foremost, selecting the concept of operations for science activities. Recall that this decision will determine the latency requirement imposed to latency-sensitive applications and therefore will drive the bandwidth requirements of the system. Second,



	Decision	Option 1	Option 2	Option 3	...	Option N
Primary Decisions	Science operation mode	NRT	3EX	2EX	1EX	
	High rate link technology	RF (Ka-band)	Optical			
	Ground support	1x34m	1x70m	4x34m	6x34m	Telescope
Spacecraft Technology	RF technology	Optimistic (500W+ Mesh)				
	Optical technology	Moderate (20W + Solid)				
Link design	Ka-band weather	90%				
	PPM modulation	4	8	16	...	128
	Slot time	0.5ns	2ns	10ns		

Figure 5-8: Morphological Matrix for the DTE link

selection of the frequency band to implement the high rate link between Mars and Earth. This decision is directly linked to the third one, i.e. ground support, in which the DSN provides service through five distinct options: A traditional 34 meter or 70 meter antenna, four 34 meter antennas arrayed, six 34 meter antennas arrayed or an optical telescope.

Additionally, a set of secondary decisions are available to this case study. First, technology available to implement the high rate DTE link includes the maximum transmit power, or the type of antenna/telescope used to achieve high gains. To limit the scope of the results, only one option per frequency band is assumed based on private conversations with members of JPL's Communication Architectures and Research Section, as well as predicted values from the literature (see [215] and [216] for Ka-band technology, and [217] and [218] for optical technology). Finally, a minor set of decisions is required to properly design the links in the system. For RF links, atmosphere effects are evaluated at 90% according to the values provided by the DSN link design handbook [219], [216]. Similarly, the optical link is implemented using M-PPM modulation [218], with both the modulation order and slot time variable and adjusted for optimal performance.

Step 5-4.3. Model Development and Validation

Several interdependent models are required to answer the research questions of this case study. Figure 5-9 provides the high level structure of the process followed. First, and given the current MRN, I ranked latency contributors (Step 5-3) and identified which parts of the system are critical to deliver information in a timely manner. Once this step is performed, the pool of users at the vicinity of Mars and science operation profile are used to estimate the data rate required to support them at any point in time. This information is used to set

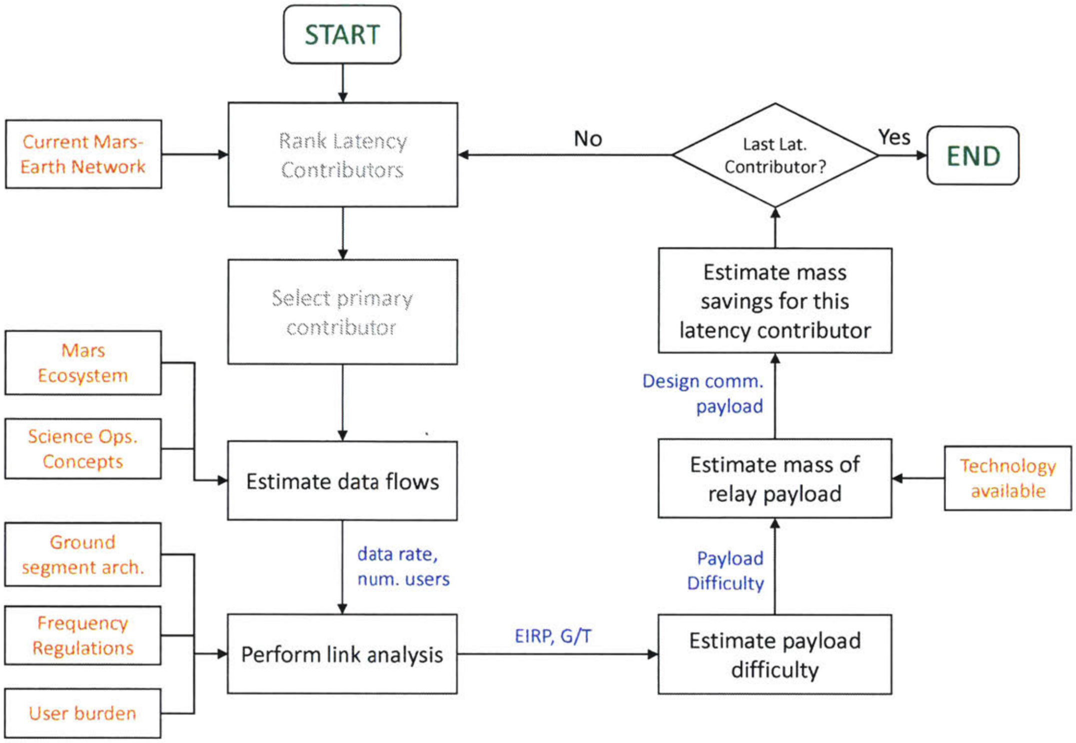


Figure 5-9: Case Study Structure

a data rate requirement for the DTE and DFE links. Next, using link analysis tools, I obtain the EIRP and G/T requirements to support the downlink and uplink respectively. Finally, I use the concept of payload difficulty[4] to characterize the mass of the payload that supports this link, and estimate the mass savings obtained when latency-sensitive information is not returned in real-time.

Astronaut Activity Modeling

Three primary scenarios have been considered for identifying the data flows required at the Earth-Mars network: Astronauts arriving at Mars, astronauts at the surface conducting science operations, and astronauts departing the Red Planet. They were generated by JPL's Communication Architectures and Research Section and treated as inputs for this case study. Of them, the scenario with astronauts at the surface of Mars will be used for sizing the required relay network as it requires larger data rates and is, consequently, the more stringent case.

Each scenario is characterized by four elements:

- Days: Astronauts and other robotic entities generate and receive data during a period

[4] See Appendix B for a detailed description of payload difficulty is a useful construct to size space communication payloads.

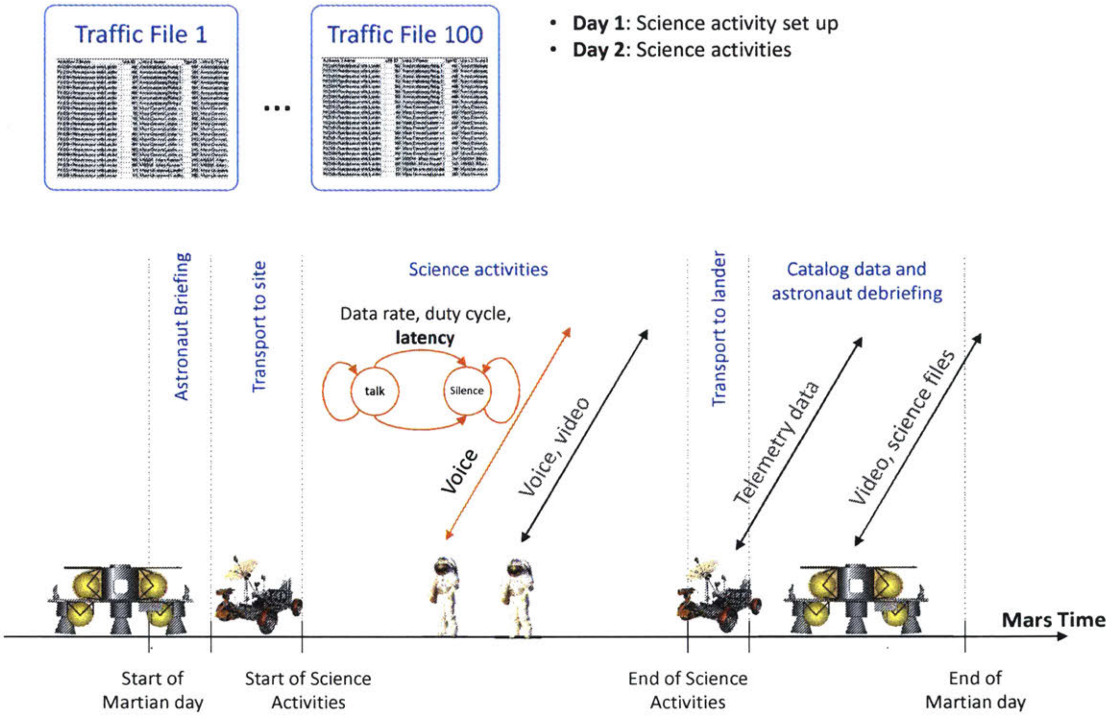


Figure 5-10: Mars Scenario with Activities and Data Flows

of one or two days.

- Activities: Set of high level tasks that define the operations of astronauts at Mars within a day. Examples of activities include *Descent to Mars surface*, *surface drilling operations*, or *checkout and ascent to Mars orbit* among others.

- Data flows: Each activity contains a collection of data flows from different users, both human and robotic (including orbiters). A data flow is characterized by five primary properties: The application type (e.g. telemetry, video, voice, science data), the application data rate, the average on-time period, the duty cycle and the allowable latency.

- ON/OFF periods: The average on-time period and duty cycle are utilized to model the data flow as a two state Markov process, where the ON state represents the transmission of data at the specified data rate.

Note that this model has been found to be suitable for space communication applications [220]. Note also that, given the randomness introduced by the Markov model, multiple observation per scenario must be analyzed in order to obtain statistically significant measurements of required bandwidth and storage for the different elements in the system. Therefore, for the purposes of this case study, results reported were obtained using a pool of 100 observations per scenario (see Figure 5-10).

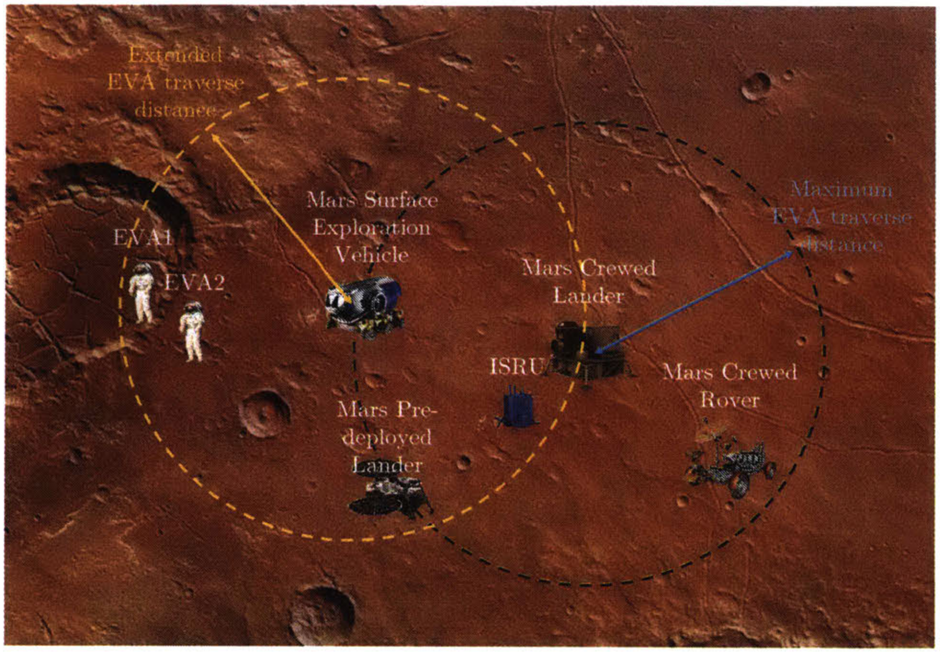

Extended EVA traverse distance
EVA1
EVA2
Mars Surface Exploration Vehicle
Mars Crewed Lander
Maximum EVA traverse distance
ISRU
Mars Pre-deployed Lander
Mars Crewed Rover

Figure 5-11: Assumed Mars Surface Topology (Adapted from Reference [12])

The typical structure of science operations at the surface of Mars include a combination of field camp setup and EVA science operations. This minimizes the risk of burning out astronauts, which increases safety risks and decreases their productivity [I16]. Consequently, a representative set of activities for a two day period would include:

- Day 1: Field camp setup
  - PAO activity: 30 minute visit to a pre-deployed lander and return of HD imagery.
  - OPS relocation: 1 hour transport of the ISRU oxygen production system to a new optimal site.
  - 30 minute transport of the Mars Surface Exploration Vehicle [221] to drilling site. The Mars Surface Exploration Vehicle performs functions similar to a Crew Mobility Chassis. It extends the range of EVA traverses by providing increased life support system capabilities (see Figure 5-11).
  - 30 minute return to Mars Crewed Lander.
  - 20.5 hours of preparation of science activity day including, food storage, off-duty and recreation activities, exercise, general housekeeping, and sleep.
- Day 2: Science data collection and analysis
  - Initial briefing for science operations.
  - 30 minute transport to Mars Surface Exploration Vehicle at drilling site.
  - 6 hours of field work and other science activities during EVAs (two astronauts at a time). Other astronauts are supervising the EVAs from the Mars Crewed

Table 5.4: Mars Application Types

Application	Data Rate	One Way Latency	Application Type
Voice	20 kbps	1 sec	Real-time
Biomedical	4 kbps	1 sec	Real-time
Caution and Warning	2 kbps	1 sec	Real-time
Command and Teleoperations	200 kbps	1 sec - 1 min	Near real-time
Navigation	2 kbps	1 sec - 5 min	Near real-time
SD video	1-3 Mbps	5 min - 12 hours	Latency-sensitive
Science data	2.4 Mbps	5 min - 12 hours	Latency-sensitive
HD video	10-20 Mbps	5 min - 12 hours	Latency-sensitive
PAO video	10-20 Mbps	5 min - 12 hours	Latency-sensitive
Files	24Mbps	5 min - 12 hours	Latency-sensitive

Lander and the Deep Space Habitat.

- 30 minute return to Mars Crewed Lander after EVAs have concluded.
- 17 hours for sample curation and analysis, as well as de-briefing to Earth science back-room. Other PAO activities, exercise, food preparation, housekeeping tasks and sleep are also included.

As previously mentioned, within each activity multiple data flows from all surface and orbital assets in the Mars vicinity would be established. Table 5.4 summarizes the set of application types supported by these data flows, as well as their expected data rate and latency requirements. Interestingly, note that the three application types described in Section 1.3.1 are clearly applicable in the context of the MRN and support of astronauts. For latency-sensitive applications, a wide range for the latency requirement is provided for now since the insights from Step 5-2 are required to narrow it down. Finally, observe that significant data rate requirements on the order to 10 to 20 Mbps will be required to successfully support these latency-sensitive applications if no latency allowance is accepted. They are one order of magnitude larger than the current MRN bandwidth capability and therefore vindicate the need for this case study (see Figure 5-12 for current DTE link rates).

ArchNet

ArchNet is a cross-platform high-level network simulator intended for architectural studies in the context of space communication networks [140]. It was originally developed to assess the bandwidth requirements for the international lines that connect the different DSN sites with JPL [58]. Furthermore, it implements a two-state Markov leveling scheme that can be used to obtain coarse bandwidth requirements for space communication networks as a function of instantaneous data rate, pass duration and allowable latency [46].

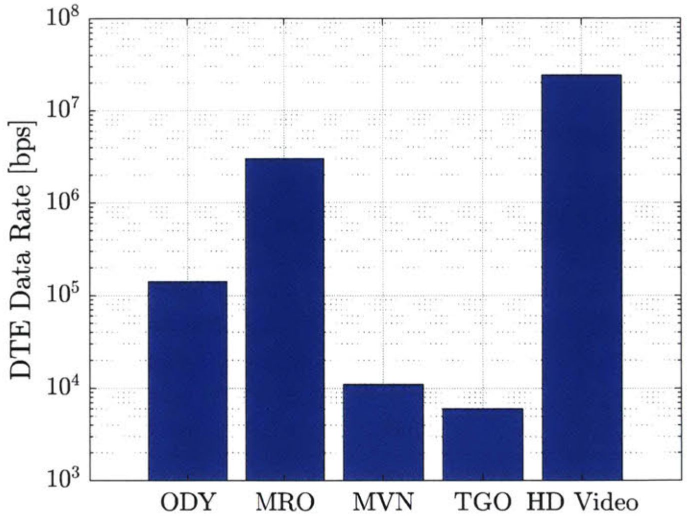


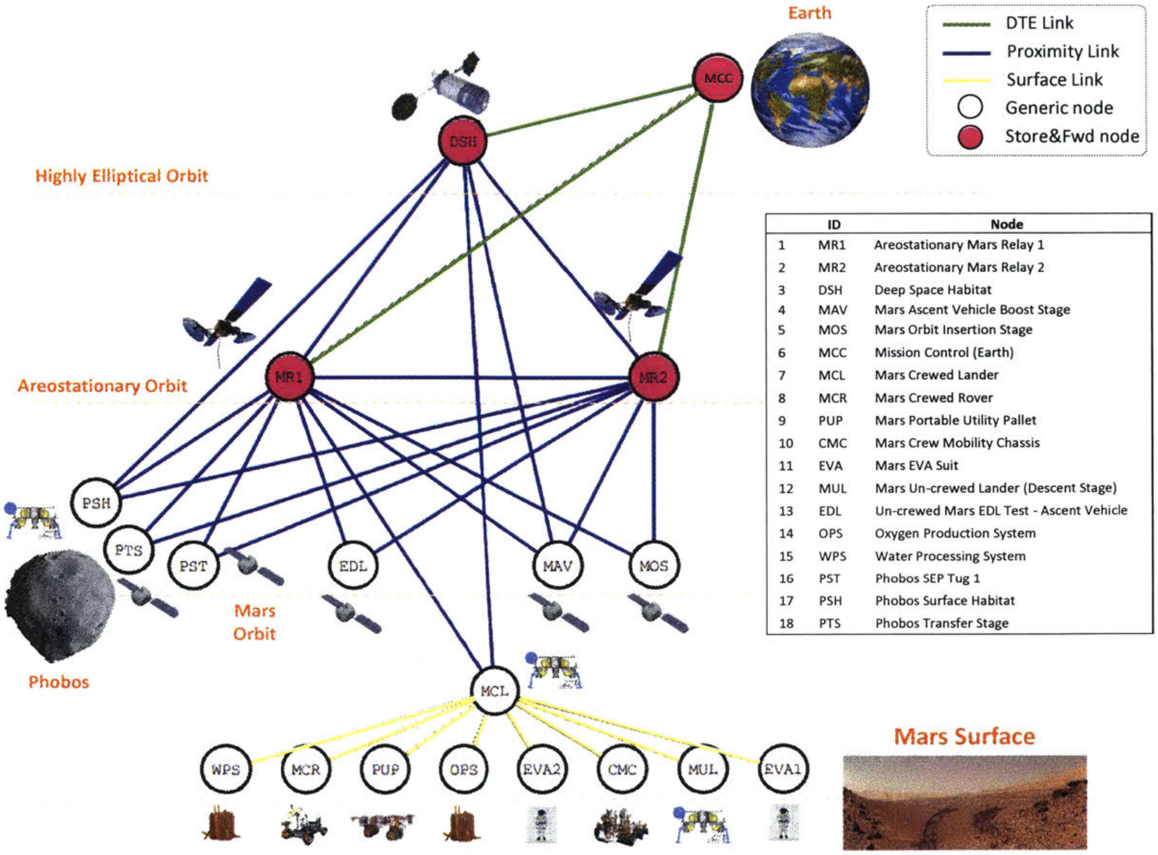
Figure 5-12: DTE Link Rates from MRN (Data from Reference [13])

Several improvements were required to obtain data rate requirements for the Mars-Earth network[5]. The most important include:

1. Processing of data flows with random ON/OFF periods.
2. Processing and data extraction multiple scenarios to obtain statistically significant bandwidth measurements.
3. Modeling of multi-band links and the mechanism that assigns a given data type to a given frequency band.
4. Modeling of proximity links to be supported with multiple access payloads, including statistics on the number of users active as a function of time.
5. Modeling of rules that govern the routing of data from origin to destination. This includes differentiation between the primary and secondary paths as part of the custodian mechanism.

Figure 5-13 provides a visual representation of the network topology as modeled in ArchNet. As previously indicated, a wide pool of users are modeled, both human and robotic, including support elements such as ISRU oxygen and water systems, as well as rover and landers. Four ArchNet evaluations runs are performed, one per science operations mode. In turn, each evaluation simulates 100 scenarios that describe the random traffic flows to be supported during the different activities that the astronauts are conducting on the Mars surface. After each simulation, a time line that describes the capacity required in a given link as a function

[5] All ArchNet simulations utilize data rates net of coding and other low level communication artifacts.



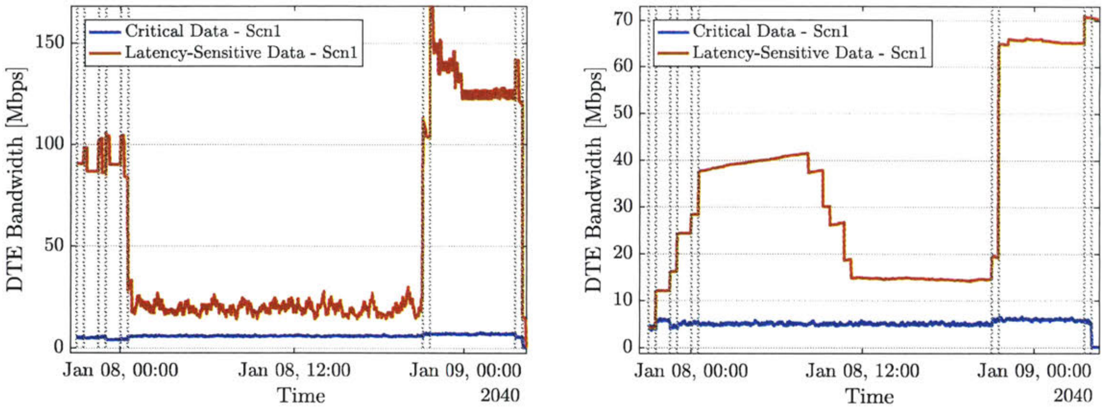
	ID	Node
1	MR1	Areostationary Mars Relay 1
2	MR2	Areostationary Mars Relay 2
3	DSH	Deep Space Habitat
4	MAV	Mars Ascent Vehicle Boost Stage
5	MOS	Mars Orbit Insertion Stage
6	MCC	Mission Control (Earth)
7	MCL	Mars Crewed Lander
8	MCR	Mars Crewed Rover
9	PUP	Mars Portable Utility Pallet
10	CMC	Mars Crew Mobility Chassis
11	EVA	Mars EVA Suit
12	MUL	Mars Un-crewed Lander (Descent Stage)
13	EDL	Un-crewed Mars EDL Test - Ascent Vehicle
14	OPS	Oxygen Production System
15	WPS	Water Processing System
16	PST	Phobos SEP Tug 1
17	PSH	Phobos Surface Habitat
18	PTS	Phobos Transfer Stage

Figure 5-13: Mars Network Topology Modeled in ArchNet

of time is available, as well as other related metrics such as data storage for the custodian mechanism.

Figure 5-14 provides an example of ArchNet's output. The left plot depicts the required capacity in the DTE link between the MRN areostationary relay satellite and a DSN site assuming that science operations are performed in near real-time (NRT). Vertical dotted bars are used to indicate the start and end of each simulated activity. In turn, Figure 5-14b provides the bandwidth estimates for the same link, but assuming that science operations are performed in 1EX mode. Observe the significant differences in the total bandwidth requirements, as well as variations in the instant of time at which data reaches Earth. In particular, in the NRT case data is delivered to the final user almost immediately after finalizing each activity. Alternatively, in the 1EX mode of operations data from the first set of activities is progressively trickled back to Earth and arrives at JPL 8 to 10 hours later.

Observe also that all capacity estimates from Figure 5-14 specifically differentiate between critical and latency-sensitive data. Indeed, while ArchNet simulates the DTE link it keeps track of which data is critical and must be returned without any latency allowance and which data is latency-sensitive. Consequently, the required bandwidth for the former type



(a) NRT Science Operations

(b) 1EX Science Operations

Figure 5-14: Estimated DTE Link Capacity Time Line

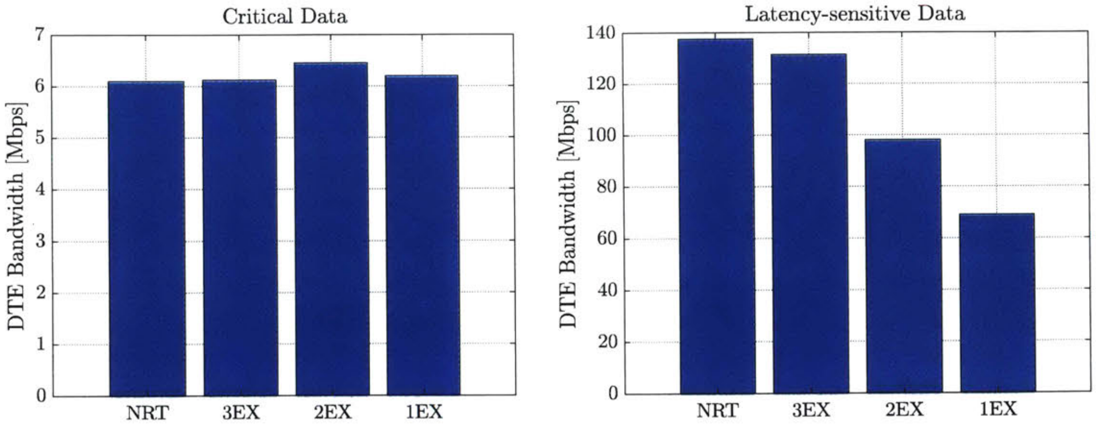
of data is insensitive to the choice of science operations.

Finally, recall that data flows in the system are random due to the Markov process that characterizes them (see Step 5-2). Therefore, 100 time lines such as those represented in Figure 5-14 are available after each simulation. To condense this information into a single data rate requirement, I proceed as follows: First compute the 95% percentile bandwidth required in the time domain for each scenario. Then, treat them as an independent observation of an IID process, and compute the definitive requirement as the median value across all available observations. The result of this process is exemplified in Figure 5-15, where the estimated bandwidth for the DTE link is provided as a function of the science operations mode. Note that all provided values have been estimated using the 95% percentile over the time lines rather than the maximum value. Indeed, this approach minimizes the effect of spurious capacity estimates generated during the discrete time simulation. Similarly, observe that the capacity requirement for critical data is not exactly equal but exhibits minor differences. These differences of less than 1Mbps are once again attributed to negligible artifacts of the simulation process and are therefore not considered during the design of the DTE/DFE payload.

Link Analysis

Once the capacity required between Mars and the Earth has been properly modeled, I now turn my attention to modeling the deep space link between the relay satellite and the DSN ground segment. Two primary configurations are possible given the Morphological matrix from Figure 5-8:

- RF system: Critical data is returned to Earth using an X-band link while latency-



(a) Critical Data

(b) Latency-sensitive Data

Figure 5-15: DTE Link Capacity Requirement

sensitive data is transmitted at Ka-band. A single DTE payload supports both links at the same time similar to how TDRSS operates (see Figure 5-16a).

- Optical system: Critical data is returned to Earth using an X-band link while latency-sensitive data is transmitted using an optical link. Two communication payloads must be carried by the Mars relay satellite (see Figure 5-16b).

Consequently, a total of 3 types of links have to be modeled: X-band downlink, Ka-band downlink and optical downlink. Observe that uplinks are not modeled as they will not drive the mass and power requirements of the relay satellite payload. Indeed, not only do they require one order of magnitude less data rate, but the DSN is being equipped with 80kW high power amplifiers that would provide a very large EIRP when combined with the 34m or 70m apertures [42]. If that is not enough, uplink arraying is also being considered as a possibility to further increase the system performance [222].

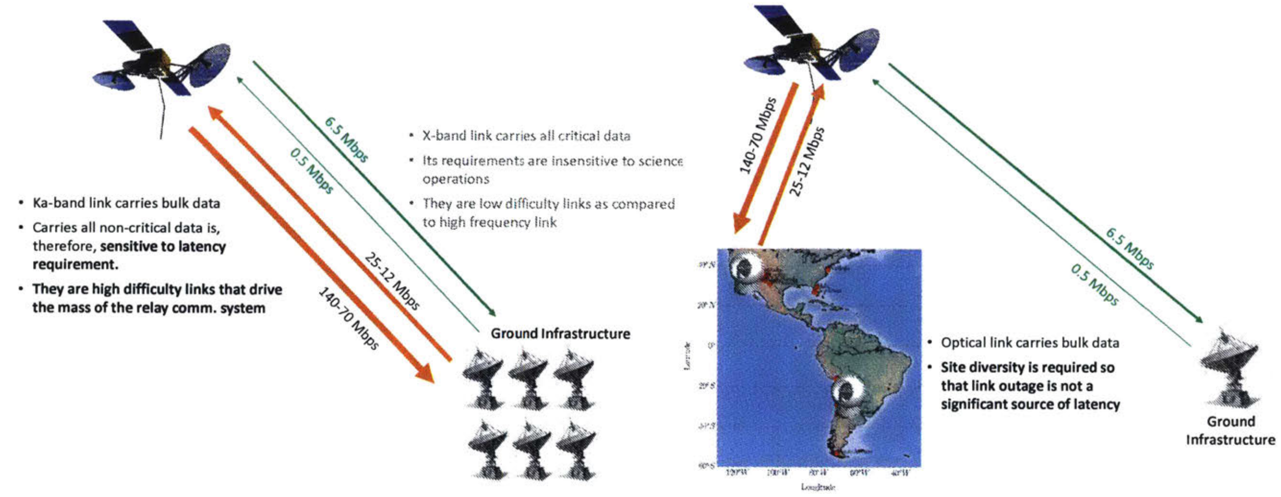
The RF downlink between Mars and the DSN assets have been computed using data from the DSN link design handbook [219]. Next, we summarize the primary set of assumptions utilized:

- The link budget is computed at maximum Earth-Mars distance (i.e. conjunction), with an Sun-Earth-Probe angle of 3 deg.

- The weather effect is estimated at a 90% confidence level.

- The link minimum elevation angle is 10 deg and 20 deg at X and Ka-band respectively, based on current DSN practices.

- Implementation, pointing and polarization losses incur in 2.07dB of losses.


6.5 Mbps
0.5 Mbps
X-band link carries all critical data
Its requirements are insensitive to science operations
They are low difficulty links as compared to high frequency link
Ka-band link carries bulk data
Carries all non-critical data is, therefore, sensitive to latency requirement.
They are high difficulty links that drive the mass of the relay comm. system
25-12 Mbps
140-70 Mbps
Ground Infrastructure
140-70 Mbps
25-12 Mbps
6.5 Mbps
0.5 Mbps
Optical link carries bulk data
Site diversity is required so that link outage is not a significant source of latency
Ground Infrastructure
Longitude

(a) RF System

(b) Optical System

Figure 5-16: DTE Link Configurations

- The modulation index is set optimistically at 80 deg. This is representative to the data transmit phase of the downlink rather than signal acquisition phases.

To validate the obtained link model, comparison against link budgets computed by the DSN program office is performed. For instance, Table 5.5 compares the link budget between Juno and a 34 meter antenna at X-band (inputs are highlighted in yellow). Truth values are obtained from Reference [223], and results indicate a very moderate discrepancy of 1.45dB. A similar exercise was conducted with a downlink from Odyssey during the signal acquisition phase. In this case, the true margin was provided to the model and the comparison was performed over the required EIRP. Discrepancy in this case was limited to 0.12dB, once again validating the link budget calculator.

On the other hand, the optical link budget was performed assuming an M-PPM modulation with variable number of modulation levels and time slot, both of which were optimized to minimize the required EIRP to close the link. References [224] and [225] were used as primary sources of information to perform all link budget calculations, as well as assessing the performance of the M-PPM modulation and coding formats. Similarly, estimation of background noise photons was based on values provided in the Reference [226].

Given the insights from these references, the baseline set of assumptions utilized during the computation of the optical link budget include:

- The link budget is computed at maximum Earth-Mars distance, with an Sun-Earth-Probe angle of 3 deg.

- The ground station has a diameter of 12 meters.

Table 5.5: Juno X-band Downlink Design Control Table

	Truth	Model	Units	Id	Legend
LINK CONSTANT					
Antenna Type	34m	34m	-	-	-
Frequency band	X	X	-	-	-
Carrier frequency	8.43e9	8.43e9	Hz	-	-
Carrier wavelength	0.036	0.036	m	-	-
Link distance	4.75e11	4.75e11	m	-	-
User data rate	19991.66	19991.66	bps	-	-
Tracking planet	None	None	-	-	-
TRANSMITTER PARAMETERS					
EIRP	59.2	59.1	dBW	1	-
Tx circuit loss	0.9	1.0	dB	2	-
Tx pointing loss	0.93	1.0	dB	3	-
Radiated power	57.37	57.1	dBW	4	(1-2-3)
PATH PARAMETERS					
Space loss	284.6	284.48	dB	5	-
Atmospheric losses	0.06	0.15	dB	6	-
RECEIVER PARAMETERS					
Rx antenna gain	68.26	68.3	dB	7	-
Rx pointing loss	0.1	0.08	dB	8	-
Polarization loss	0.05	0.07	dB	9	-
TOTAL POWER SUMMARY					
Total rx power (C)	-	-159.38	dBW	10	(4-5-6+7-8-9)
Noise due to antenna MW HW	16.33	15	K	11	-
Noise due to atmosphere	3.68	4.26	K	12	-
Noise due to Cosmic Background	2.69	2.68	K	13	-
Noise due to Sun	0	0	K	14	-
Noise due to Planet	0	0	K	15	-
Rx noise temperature	22.7	21.94	K	16	(11+12+13+14+15)
Noise spectral density (No)	-215.04	-215.19	dBW/Hz	17	-
Noise bandwidth	389045	309471	Hz	18	-
Received noise (N)	-	-163.29	dBW	19	(17+18)
Received C/N	-	3.19	dB	20	(10-19)
Received C/No	55.9	55.81	dB-Hz	21	(10-17)
CHANNEL PERFORMANCE					
Command and Ranging data suppression	0.74	0.03	dB	22	-
Received Cd/No	55.15	55.78	dB-Hz	23	(21-22)
Link data rate	-	119949.93	bps	24	-
Received Eb/No	4.36	4.99	dB	25	(23-24)
Radio loss	0.48	0.8	dB	26	-
Output Eb/No	4.04	4.91	dB	27	(25-26)
Required Eb/No	-0.1	-0.1	dB	28	-
Eb/No margin	3.56	5.01	dB	29	(27-28)

- The modulation level can vary between 4 and 128 levels due to laser technology limitations.

- The minimum slot time is 0.5ns.

- The atmospheric losses have a worst, nominal and best values of 2.2, 0.6, 0.3dB.

- The pointing losses have a worst, nominal and best values of 2.0, 1.61, 1.25dB.

- The space telescope has a worst, nominal and best optical efficiency of 58.3%, 65.2% and 71.9%.

- The ground telescope has a worst, nominal and best optical efficiency of 27.7%, 31.3% and 34.8%.

- The link uses the standardized wavelength of 1064nm.

- 3dB margin is always assumed.

- 0.75dB of coding gap is always required.

- The synchronization and quantization losses are estimated to be 1dB.

- The optical filter at the ground station has a bandwidth of 1Å. It includes a polarization filter that eliminates 3dB of noise background photons.

- The receiver quantum efficiency is constant at 40%.

Figure 5-17 provides an example of the optical link budget model output in the form of a payload difficulty[6] vs. data rate plot. It has been validated against internal data provided by JPL's Communications Architectures and Research Section. Several singularities are worth noting: First, vertical discontinuities are visible. They occur when the modulation order changes to avoid channel capacity limitations of an M-PPM system. Second the plot exhibits the typical logarithmic behavior that is expected given that the y-axis is measured in decibels while the x-axis is not. Finally, observe that the difference between the best, nominal and worst cases is significant, and consequently mass estimates for the DTE payload might be subject to a large degree of uncertainty.

Site Diversity

Transmission of optical links through the Earth atmosphere is particularly challenging. In the absence of clouds, their performance is typically limited by turbulence, background noise incident on the ground telescope, atmospheric seeing and angle of arrival fluctuations

[6] See appendix B.

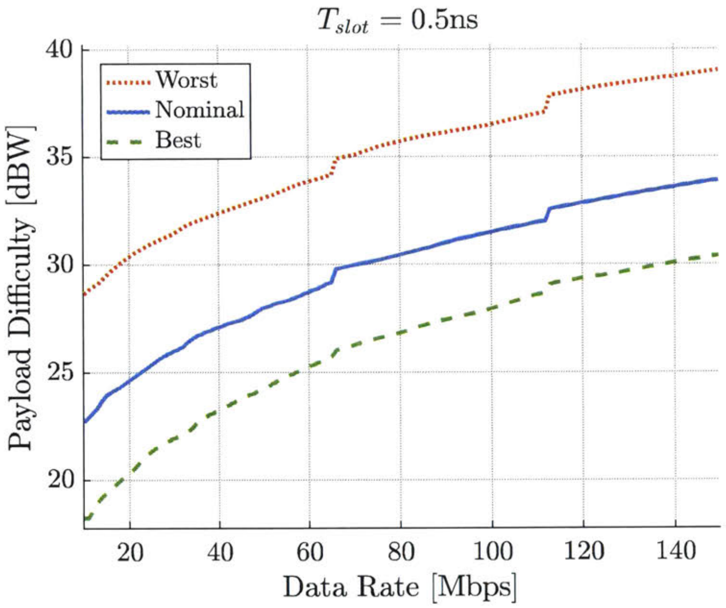


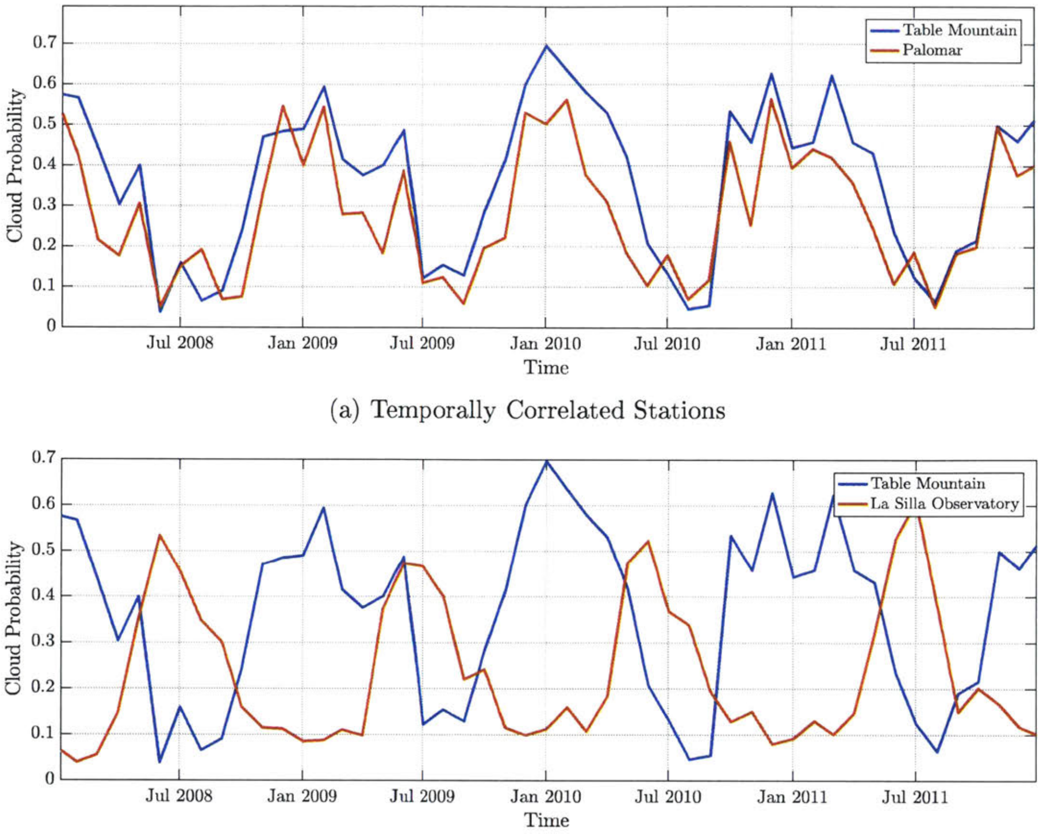
Figure 5-17: Deep Space Optical Link Performance

[227]. These can be mitigated, at the physical layer level using aperture averaging techniques, adaptive optics, background noise rejection filters or hybrid RF/Optical systems [227]. Notwithstanding the importance of these link impairments[7] and their respective mitigation techniques, their effect is secondary when compared to cloud coverage. Indeed, if a cloud impairs a space-to-ground optical link, then the attenuation experienced is so large that no transmission can occur.

To solve this problem, site diversity has been typically proposed as a possible technique for mitigating the deleterious effects of clouds. In a spatially diverse network, multiple ground telescopes are built to provide service to a single satellite. When a specific site is clouded, the space optical payload points the laser towards another station that is unclouded, and transmission continues. This greatly increases the availability of the system and ensures that operation of the optical space-to-ground link can occur with high probability [228], [229], [230], [231].

Understanding the impact of site diversity in the context of this thesis is necessary for two reasons. First, all Ka-band links are sized using a 90% weather confidence level. Therefore, fair comparison with the optical system requires the a space-to-ground network with a similar level of availability. Second, if the probability of having the optical link disrupted due to cloud coverage, then this might become a significant latency contributor that was not properly accounted for in ArchNet.

[7] They have in fact been taken into account when obtaining the worst, nominal and best case performance of the optical link



(a) Temporally Correlated Stations

(b) Temporally Uncorrelated Stations

Figure 5-18: Cloud Probability

To quantify the number of ground stations required to provide 90% availability in a space-to-ground optical system, I developed a simple ON/OFF model identified using the cloud fraction data set[8]. Effects such as temporal and spatial correlation are both well captured. For instance, Figure 5-18 presents the cloud probability between January 2008 and January 2012 for two sets of ground stations. Observe that Table Mountain and Palomar, both located in California, USA, are subjected to the same seasonal variability. In contrast, Table Mountain and La Silla Observatory in Chile are temporally anti-correlated and therefore we expect large network availability gains by utilizing these two sites.

Given that stations lying in separate hemispheres are far apart from each other and can therefore be considered spatially uncorrelated, the probability of having two sites clouded at the same time can be simply computed as the product of the two corresponding cloud fractions over time. In that sense, Figure 5-19 provides the network availability for a system that incorporates a receiver at both Palomar and La Silla Observatories over a period of eight years. The mean and 5% confidence value are also provided as horizontal lines. Observe that

[8] The implementation details can be found in Appendix C.

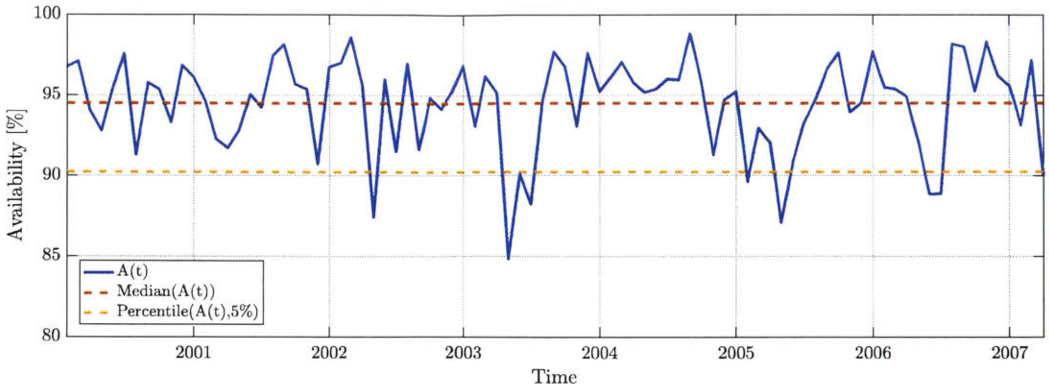


Figure 5-19: Two Station Availability

the expected availability of such a system is well above 90% and, in fact, 90% availability is provided with an 95% probability. Consequently this configuration would be equivalent or better to a Ka-band system designed with 90% weather effects confidence interval, and therefore I will henceforth require two optical sites to achieve the same level of performance as a Ka-band system.

Payload Difficulty and Payload Mass

As indicated in Figure 5-9, a critical step of this case study is to be able to estimate the mass of a communication payload given traditional link budget requirements such as EIRP or G/T and available RF and optical technology. To facilitate this step, I developed the notion of payload difficulty, a frequency normalized quantity that can be used to obtain characteristic function for payload mass given a very limited set of technology parameters[9].

For a DTE payload with a parabolic antenna or optical telescope, the mass of the system can be estimated using three characteristic technology parameters:

- Antenna density, expressed in kg/m^2. Current solid antennas have antenna densities of $\geq 6kg/m^2$. However, future deployable and mesh antennas will deliver densities on the order of 1 to 2 kg/m^2.

- HPA power generation capacity, measured in W/kg. The function that relates power to mass for communication payloads can be assumed linear in the worst case, specially if power combining techniques are utilized [216].

- Baseband electronics, frequency sources, harnesses, diplexers, waveguides, etc. They are assumed to have constant mass.

[9] See Appendix B for a detailed explanation of payload difficulty for SA and MA payloads.

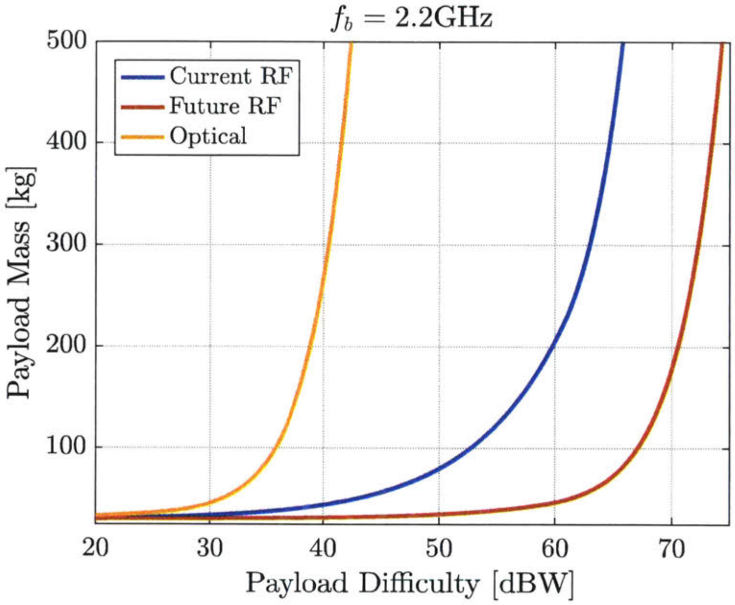


Figure 5-20: SA Payload Characteristic Curves

Observe that the provided approach is equally valid for an RF SA payload or an optical telescope. In fact, previous system level studies for optical communications have used a similar approach [217]. Figure 5-20 provides the assumed characteristic curves assuming the technology parameters specified in the Morphological Matrix from 5-8 using a normalized frequency of 2.2GHz (S-band allocation). Note that the mass of a SA is highly non-linear with payload difficulty. In fact, each characteristic curve has three distinct regions:

- Electronics-constrained payload: If the payload difficulty is too low, then its mass will be primarily determined by the mass of the electronics, harnesses, waveguides, among others. Since these have been considered constant, they set a lower bound on the payload mass.

- Power-constrained payload: The payload difficulty is too high, then its mass will be driven by the mass of the antenna used to provide the required gain. In this region, the mass of the payload increases exponentially with constant slope depending on the type of antenna considered.

- Power-optimized payload: The payload difficulty is such that the available technology can be used to optimize the total system mass, i.e. the required EIRP is provided by an optimal combination of transmit power and antenna size. This is visible in the curve knees of Figure 5-20.

In practice, the shape of the obtained payload difficulty characteristic curves imposes practical limitations on the performance and feasibility of space communication systems. In particular, if the link that should be supported by a given payload is beyond the threshold value at which you enter the power-constrained region, then it becomes exponentially difficult to implement it. Therefore, even one dB of payload difficulty is penalized with an enormous increase in total system mass.

On the other hand, the effect of technology advances in RF communication systems is clearly understandable through the concept of payload difficulty and the resulting characteristic curves. Indeed, with current RF amplifier technology that provides a maximum output power of 200W and solid dishes, providing more than 60dBW of payload difficulty becomes significantly taxing from a systems perspective. Alternatively, up to 10dBW of extra payload difficulty are achievable if future technology can provide HPAs of 500W and antenna densities of 1 to 2 kg/m^2.

Finally, payload difficulty can also be used to directly compare RF and optical technology. In Figure 5-17 I quantify the payload difficulty required to close a 20-100Mbps optical link from Mars at maximum range. Observe that only 20 to 40dBW of payload difficulty are required. Therefore, using the yellow line from Figure 5-20 we conclude that the SA payload will be at most in the power-optimized region, and consequently it can be implemented with reasonable payload mass. As the next section will demonstrate, a similar link at Ka-band will require on the order of 70dBW and consequently it will only be implementable if the assumed 2040's technological is available.

Ground Segment Cost

The cost of DSN ground segment was estimated based on the insights of References [175] and [177] and includes some upgrades with respect to the model presented for Chapter 4. The most salient include:

- The cost of an array includes the penalties indicated in Reference [175], most notably the correlation equipment use to generate the optimally arrayed signal.

- A learning factor of 95% is utilized for the construction cost of antennas within the same array [39].

- The total number of antennas to deliver an arraying factor N is computed as $N + K$, where K is the number of spare antennas required to provide the same operational reliability as a monolithic system [173]. In that sense, maintenance failures across antennas in an array have been assumed independent, and the overall reliability target has been set to that of a current 34 BWG meter antenna.

- The cost of a ground telescope for optical communications is obtained assuming that all costs not related to telescope assembly are equal to those of the DSN (without the arraying equipment), while the telescope assembly estimate is obtained from Reference [232] assuming a monolithic mirror design and no experience factor.

All antenna models have been calibrated using a similar approach to Chapter 4. 34 meter antennas are assumed to have a construction cost of FY2016 $\$25M$, while a 12m optical telescope is assumed to cost FY2016 $\$120M$. Prices of the antenna are scaled based on the diameter using $\gamma = 2.4$. For instance, the construction cost of a 70 meter antenna is estimated at FY2016 $170M approximately.

Step 5-5. Analysis of Results

Step 5-5.1. Network Bandwidth Requirements

Table 5.6 summarizes the bandwidth requirements elicited with ArchNet as a function of the network customers, activities, data flows, and science operational profile. As previously mentioned, the estimates provided are the median value for the 95% capacity over 100 stochastic network simulation, rounded to a 500kbps resolution. It includes both the DTE/DFE links, as well as two types of proximity links, the space-to-ground link[10] (SGL) and the MA links. For the DTE/DFE links, two estimates are provided depending on the data type, near real-time or latency-sensitive. Note that the near real-time data, due to its criticallity, is returned always without any latency margin. Consequently, the resulting bandwidth requirement is constant across science operational profiles. On the other hand, the Ka-band portion of the DTE/DFE links carries all the non-critical data and, therefore, has a total capacity requirement that depends on the type of science operations. In that sense, observe that the bandwidth required in 1EX operations is half what is expected in NRT operations, from 140Mbps to 70Mbps.

Significant bandwidth savings are also observed in the Mars proximity links. For the SGL, only 75Mbps are required if 1EX operations, while 160Mbps must be provided in the case of a NRT system. Finally, the MA links are also impacted, especially for the forward services. In that sense, latency has dual effect: In the NRT case, high rate links between the relay satellite and the user are demanded, but the number of users to support simultaneously is limited to two. If 3EX, 2EX or 1EX science operations are conducted, one order of magnitude reduction in the required data rate is observed, but now links are active for longer periods of time and consequently the system must be designed to support up to 5 users at the same

[10] The Space-to-Ground link provides high data rate communications to the Mars surface through a SA payload. The link is established between the relay satellite and the Mars lander, and multiplexes all data flows arriving and departing the martian surface.

[11] Critical data, returned over the X-band link in near real-time.

[12] Latency-sensitive data, returned over the Ka-band or optical link.

[13] Aggregate of critical and latency-sensitive data, all multiplexed in the same link.

Table 5.6: Bandwidth Requirement Elicitation for MRN

Link Type		**Data Type**	**NRT**		**3EX**		**2EX**		**1EX**	
			Mbps	**Users**	**Mbps**	**Users**	**Mbps**	**Users**	**Mbps**	**Users**
DTE:										
	-	NRT[11]	6.5	1	6.5	1	6.5	1	6.5	1
	-	LS[12]	140	1	130	1	100	1	70	1
DFE:										
	-	NRT	0.5	1	0.5	1	0.5	1	0.5	1
	-	LS	25	1	17	1	15	1	12	1
SGL:										
	FWD	NRT&LS[13]	7	1	3.5	1	2.5	1	2.5	1
	RTN	NRT&LS	160	1	130	1	115	1	75	1
MA:										
	FWD	NRT&LS	12	2	2	4	1.2	5	1.2	5
	RTN	NRT&LS	0.5	4	0.5	4	0.5	4	0.5	4
TOTAL	-	-	805	-	737.5	-	687.5	-	614.5	-
Δ[%]	-	-	-	-	9.20%	-	17.10%	-	31.00%	-

time. This is detrimental due to the self-interference problem inherent to CDMA systems, an issue that will be addressed in Step 5-6.

All in all, this section highlights the impact in total Mars relay satellite capacity as a function of the science operation profile selected. Indeed, almost 10% reduction is observed if we switch from the NRT to 3EX operational profile. Similarly, approximately 20% and 30% savings are obtained if only 2 and 1 exchange per day are budgeted.

Step 5-5.2. Link Analysis

Using the data rate requirements from Table 5.6, as well as the link models developed in Step 5-4, I can now estimate the performance of Mars-Earth link. Table 5.7 summarizes the required EIRP and antenna gain to be provided by the DTE/DFE payload on-board each Mars relay satellite. Several important observations should be emphasized: First, the DTE service requires 50dB more of EIRP as compared with the DFE service's gain. Given that a maximum RF power of 500W (27dBW) has been assumed as the technological limit of TWTAs in 2040, the difference of 23dBs must be provided through larger apertures. Therefore, the DTE service will always be the limiting factor in this system. Second, the DTE EIRP and DFE gain for the critical data is constant across all science operation modes. This matches our expectations since no latency is allowed when returning this type of data. Lastly, the difference between continuous and non-continuous science operations is confined in the 0 to 3dB range for RF systems, but can be as high as 4dB in the case of optical

Table 5.7: Required EIRP and Gain for the DTE Payload

Link	Units		Critical				Latency-Sensitive			
			NRT	3EX	2EX	1EX	NRT	3EX	2EX	1EX
RF Only System:										
DFE	dBi	Max	27.7	27.7	27.7	27.7	51.7	50.1	49.5	48.6
DFE	dBi	Min	20.3	20.3	20.3	20.3	44.4	42.7	42.1	41.2
RF/Optical Only System:										
DFE	dBi	Max	27.7	27.7	27.7	27.7	-	-	-	-
DFE	dBi	Min	20.3	20.3	20.3	20.3	-	-	-	-
RF Only System:										
DTE	dBW	Max	77.0	77.0	77.0	77.0	98.7	98.4	97.2	95.7
DTE	dBW	Min	69.6	69.6	69.6	69.6	91.3	91.0	89.9	88.3
RF/Optical Only System:										
DTE	dBW	Max	77.0	77.0	77.0	77.0	140.8	140.6	138.8	137.4
DTE	dBW	Min	69.6	69.6	69.6	69.6	132.3	131.9	130.4	128.3

systems. For the former, the maximum difference of 3dB should also be expected since the data rate in the 1EX operational mode is half what is needed for continuous operations (140 vs. 70Mbps). In contrast, in optical communications an extra dB of penalty is incurred due to the non-linear channel capacity of an M-PPM link with Poisson distributed noise photons.

As indicated in Appendix B, both the EIRP and gain for a multi-band satellite system are not directly comparable. Therefore, Table 5.8 provides the payload difficulty assuming a baseline frequency of 2.2GHz. In this case, we can clearly observe that the DTE link to carry latency-sensitive data is significantly more challenging to implement than the X-band link and the DFE links, and consequently will be the driving data type that will constrain the mass of the DTE payload. Note, however, that direct comparison between the Ka-band and optical DTE link (i.e. 75-72dBW vs. 38-35dBW approximately in the worst case) is not possible due to the fact that they would be implemented different technology. To address this issue, Figure 5-21 provides a visual representation of the payload difficulty for the different DTE services. In that sense, both the X-band link and the optical link will be in the power-optimized region of the payload difficulty characteristic curve and will require a mass of less than 100kg approximately. Alternatively, the Ka-band link will be in the power-constrained region of the characteristic curve, and consequently will (1) require a large mass (>100kg) and (2) its estimate will be very sensitive to the outcome of the link budget analysis.

Table 5.8: Payload Difficulty for the DTE Payload

Link	Units		Critical NRT	Critical 3EX	Critical 2EX	Critical 1EX	Latency-Sensitive NRT	Latency-Sensitive 3EX	Latency-Sensitive 2EX	Latency-Sensitive 1EX
RF Only System:										
DFE	dBi	Max	17.4	17.4	17.4	17.4	27.8	26.2	25.6	24.7
DFE	dBi	Min	10.0	10.0	10.0	10.0	20.5	18.8	18.2	17.3
RF/Optical Only System:										
DFE	dBi	Max	17.4	17.4	17.4	17.4	-	-	-	-
DFE	dBi	Min	10.0	10.0	10.0	10.0	-	-	-	-
RF Only System:										
DTE	dBW	Max	65.3	65.3	65.3	65.3	75.4	75.1	74.0	72.4
DTE	dBW	Min	57.9	57.9	57.9	57.9	68.1	67.8	66.6	65.1
RF/Optical Only System:										
DTE	dBW	Max	65.3	65.3	65.3	65.3	38.7	38.5	36.6	35.3
DTE	dBW	Min	57.9	57.9	57.9	57.9	30.2	29.7	28.3	26.1

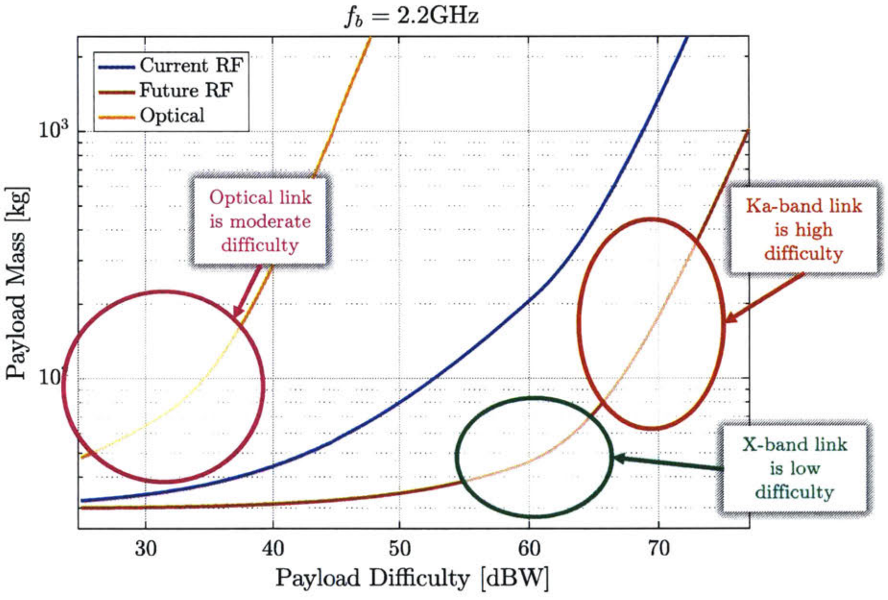


Figure 5-21: DTE Mass vs. Payload Difficulty

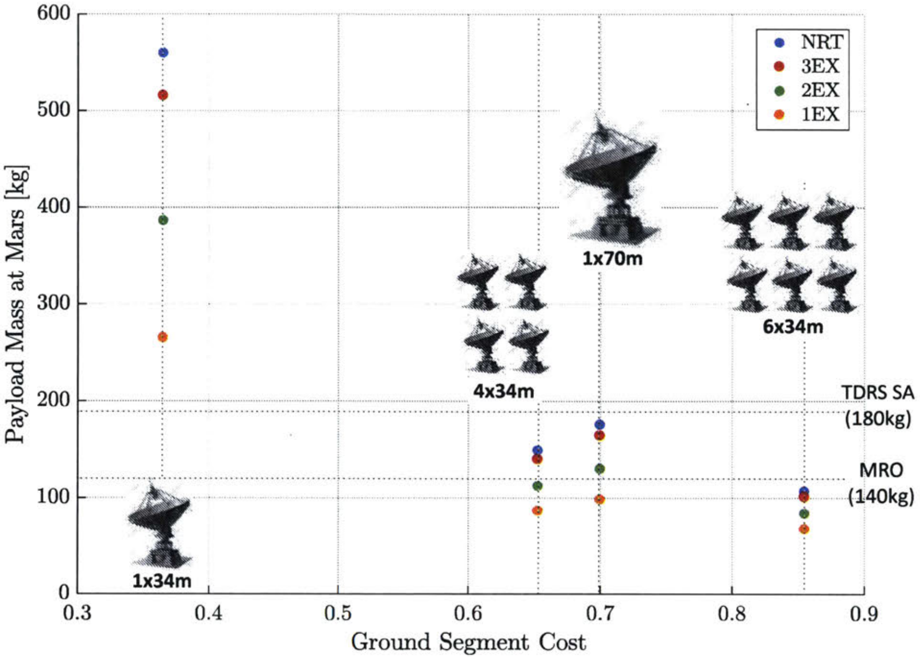


Figure 5-22: DTE Payload Mass vs. Ground Segment for RF Only Architectures

Step 5-5.3. DTE Payload Mass and Mass Savings

Given the results of the link analysis and payload difficulty characteristic curves, it is now possible to estimate the effect of performing science operations in non-real time mode from the perspective of the DTE payload. For instance, 5-22 presents the tradespace of DTE payload mass vs. ground segment cost if the MRN implements the DTE high rate link using Ka-band technology. For comparison purposes, the mass of the total payload carried by the MRO spacecraft is also depicted, along with the type of ground support provided by the DSN. Costs are normalized against the cost of one DSN site, i.e. it is implicitly assumed that the current configuration with three regions of coverage is maintained. Also, for each type of ground support, only the best link configuration is plotted (e.g. LDPC code with rate 2/3 usually obtains optimal performance).

The tradespace general structure has the expected shape. Support with a single 34 meter antenna is impractical due to the lack of gain in the Earth receiver, which results in antenna diameters of 12 to 16 meters on the relay side. Even with deployable technology, this results in DTE payload masses of 300 to 650 kg which would be difficult to implement (the TDRSS SA payloads weight on less than 200kg for instance). On the other hand, significant savings are obtained when arraying four 34 meter antennas. Not only is the required DTE payload mass lower as compared with one 70 meter antenna (mostly due to increased pointing losses in the larger dish), but it is also less costly, albeit not significantly. In short, results are consistent with the well known result that four 34 meter antennas are essentially equivalent to a 70 meter antenna. More interesting is the fact that supporting the DTE link is now feasible with a parabolic antenna of 5.5 to 8 meters depending on the science operational profile. This, combined with the forecasted technology improvements, yield a payload mass of 85 to 150 kg depending on the science operational profile. Finally, support from six 34 meter antennas results in yet another system improvement. Now the DTE link can be supported with a payload that weighs on the order of 100kg, but the cost of the ground segment infrastructure for the MRN network is almost equal to all the cost of a current DSN site.

To properly quantify the savings obtained when utilizing 1EX, 2EX or 3EX instead of NRT science operation profiles, I utilize the well-known concept of *main effects*. In particular, Figure 5-23 plots the average mass reduction on the DTE payload mass when latency-sensitive data is returned in non-real time mode. Observe that marginal benefits are obtained when operations are switched from the NRT to the 3EX mode, with an expected mass reduction of 6.5% in the DTE payload. Alternatively, if 2EX or 1EX operations are accepted, then the mass savings are 25% and 45% approximately, thus resulting in a significant impact with respect to the design of the MRN. On the other hand, Figure 5-23 also plots the interaction between science operations and ground network support. In this case, we observe that the expected mass savings from utilizing a given number of science exchanges per day is reduced are the ground system on Earth is increasingly capable. Essentially, the more gain provided by the DSN the lower the payload difficulty of the DTE link, and consequently its design is in a flatter region of the payload difficulty characteristic curve. In the limit, if you could array an infinite number of ground antennas, the DTE payload at Mars could potentially close the link with a non-directional antenna with constant mass, and changes in data rate could be accommodated by simply arraying more antennas on the ground, so no mass benefits would be observed from trickling latency-sensitive data back to Earth.

Finally, the effect of including optical communications in the system is presented in Figure 5-24. Each ground system configuration is depicted asa combination of two 12 meter optical telescopes to provide site diversity, combined with an RF system to serve the X-band link that carries critical data. Following the approach from Reference [224], optical links are evaluated assuming best, nominal and worst conditions depending on the state of the atmosphere and the optical equipment efficiencies. Observe that under best conditions, the DTE link can provide the required data rate for human Mars exploration activities with current technology assuming that a 20cm terminal is deployed at Mars. Nevertheless, if the link must work under the worst possible conditions (e.g. low optical efficiencies, low elevation angles), then telescopes of 50 to 60cm are required to close the link. This, in turn, results in a 200 to 300% increase in DTE payload mass, which yields it impractical for the MRN and uncompetitive with respect to the Ka-band alternative.

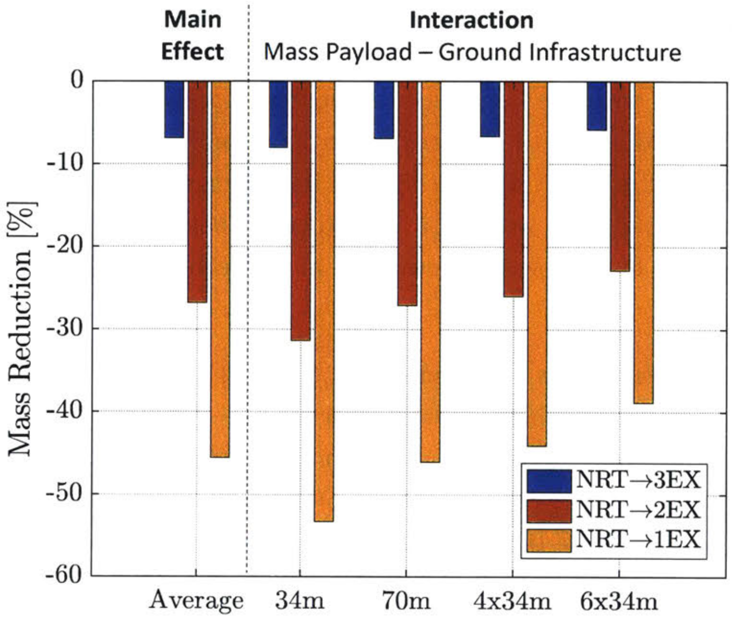


Figure 5-23: Main Effects and Interaction on DTE Payload Mass for RF Architectures

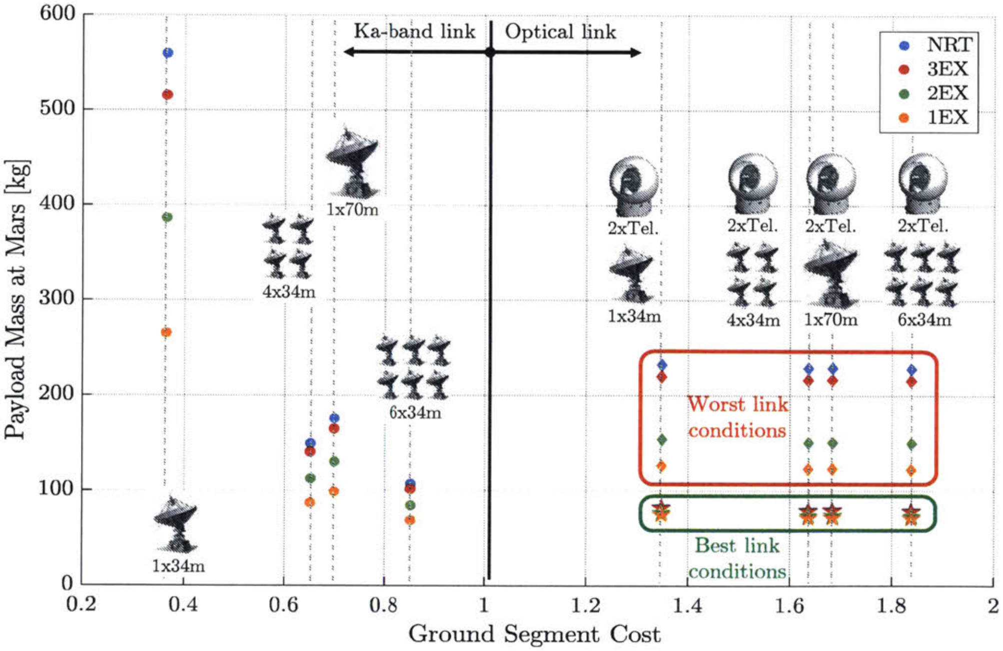


Figure 5-24: DTE Payload Mass vs. Ground Segment for RF and Optical Architectures

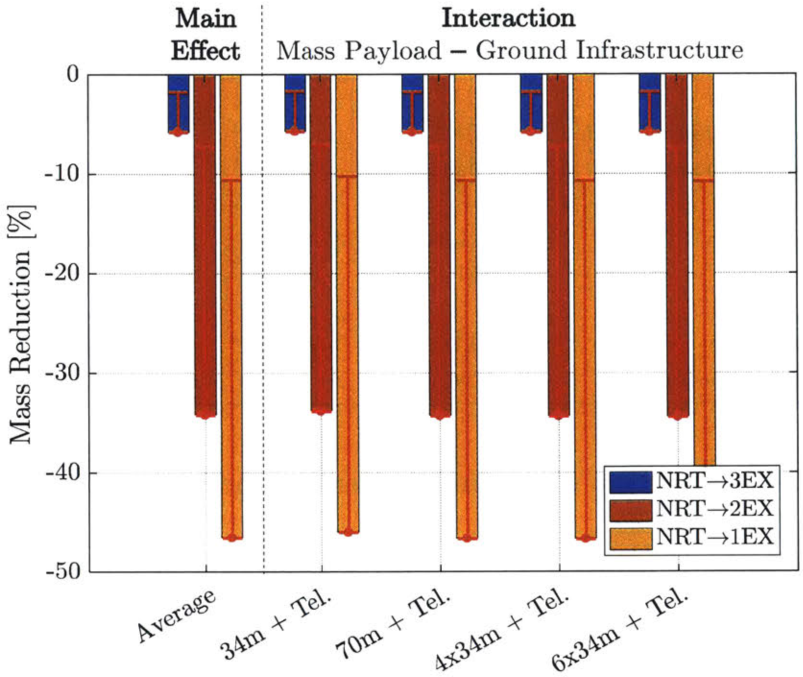


Figure 5-25: Main Effects and Interaction on DTE Payload Mass for Optical Architectures

Finally, Figure 5-25 presents the main effect and interaction analysis analogous to Figure 5-23 for architectures where the high-rate DTE link is provided through an optical link. The red error bars provide the band of uncertainty as estimated with the best and worst cases for the optical link. Observe that the mass savings in the DTE payload are difficult to predict in this case and can vary between 45% and less than 10% depending on the science operational profile and the performance of the optical system. In that sense, if nominal optical link conditions are assumed, the data rates to be returned from Mars are not high enough to result in significant mass savings for the system, which indicates that NRT operations should probably be recommended from the network communication perspective. Note that this fact does not imply that NRT operations must be conducted. Indeed, as indicated by all interviewees in this case study, the selection of a given science operational profile depends on many factors that are not necessarily related to the communication infrastructure. Therefore, the primary conclusion of this case study is that optical communications are a promising technology that can facilitate Mars exploration by granting full flexibility to scientists and ensuring that astronaut operations can be optimized based uniquely on science considerations. This contrasts with the current Mars rover operations, which are highly constrained by bandwidth limitations between Earth and Mars and can therefore only return a fraction of all the images they obtain.

Table 5.9: Morphological Matrix for the PRX link

Decisions	Option 1	Option 2	Option 3	Option 4
Primary Decisions:				
Science operations mode	NRT	3EX	2EX	1EX
Frequency band	S-band	X-band		
Customer Service:				
User burden	Omni (180deg)	LGA (90 deg)	MGA (20 deg)	
Coverage	LMO	Phobos		
Link Design:				
Coding rate	4/5 LDPC	5/6 LDPC	7/8 LDPC	
MA implementation	CDMA			

Step 5-6. Identification of Second-Order Latency Contributors

Based on the insights from Step 5-2 and Step 5-3, the second latency contributor to be considered is the proximity link between the areostationary relay orbiters and users in the Mars vicinity. Once again, the primary intent of this analysis is to quantify the mass savings obtained in the proximity link part of the network when science activities are supported using the four science operational profiles defined in Step 5-2. To limit the extent of the analysis, I assume three primary architecting decision: Science operations mode, frequency band selection, and customer service (see Table 5.9). Note that customer service is specified as a combination of coverage and user burden, where the latter quantifies what type of communication payload would be imposed on the customer to close the link with the MRN.

Step 5-6.1. Multiple Access Payload Technology

Several technological considerations must be taken into account when estimating the performance of a MA payload in the Mars vicinity. First, the choice of MA scheme is set to CDMA in accordance with directives from JPL's Communications Systems and Research Section. As a result, all links established with this part of the communication system will be affected by deleterious effects of self-interference. To estimate their extent, I first quantify the proximity link budget as if the user was isolated, and then I estimate the $\frac{E_b}{N_o}$ degradation that must be taken into account to ensure that the desired bit error rate is achieved. This degradation factor can be easily derived by comparing the system SNR and SINR [233], and can be expressed as

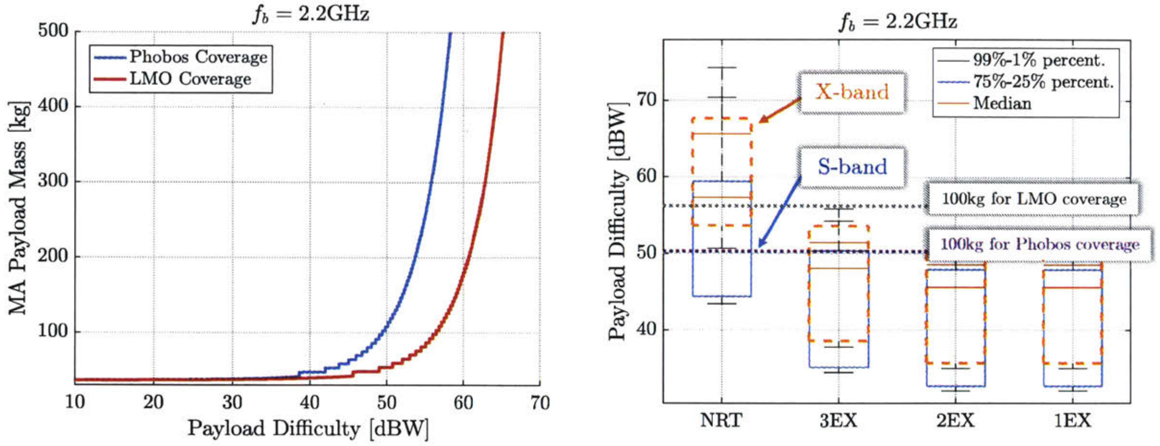
$$D = \frac{\left(\frac{E_s}{N_o}\right)_M}{\left(\frac{E_s}{N_o}\right)_1} = \frac{1}{1 - (N-1)\frac{R_s}{R_c}\left(\frac{E_s}{N_o}\right)_1}, \tag{5.1}$$

Table 5.10: CDMA Self-interference

Inputs	**S-band**				**X-band**			
	NRT	**3CD**	**2CD**	**1CD**	**NRT**	**3CD**	**2CD**	**1CD**
Forward Service:								
R_b [Mbps]	12	2.0	1.2	1.2	12	2.0	1.2	1.2
N [Users]	2	4	5	5	2	4	5	5
R_c [Mcps]	70	70	70	70	35	35	35	35
Return Service:								
R_b [Mbps]	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
N [Users]	4	4	4	4	4	4	4	4
R_c [Mcps]	78	78	78	78	39	39	39	39
FWD Service Degradation Factor [dB]:								
4/5 LDPC	2.42	1.04	0.81	0.81	8.39	2.42	1.82	1.82
5/6 LDPC	2.52	1.08	0.84	0.84	9.19	2.52	1.88	1.88
7/8 LDPC	2.67	1.13	0.88	0.88	10.89	2.67	1.99	1.99
RTN Service Degradation Factor [dB]:								
4/5 LDPC	0.21	0.21	0.21	0.21	0.44	0.44	0.44	0.44
5/6 LDPC	0.22	0.22	0.22	0.22	0.45	0.45	0.45	0.45
7/8 LDPC	0.23	0.23	0.23	0.23	0.47	0.47	0.47	0.47

where N is the number of users to be supported simultaneously, R_c is the chip rate, R_s is the user symbol rate, and $\left(\frac{E_s}{N_o}\right)_1$ is the SNR required to close the link for one user. In that sense, Table 5.10 estimates the CDMA self-interference degradation factor assuming that the MA system utilizes the entire spectrum at S and X-band, and the maximum chip rate is obtained from a 1.29Hz/bit spectral efficiency. Note that the provided values are categorized depending on the science operation profile, since it affects the data rate per user and number of users to support simultaneously (see Step 5-5). Observe that self-interference effects are mostly present in the FWD service as it supports larger data rates, and is particularly intense if X-band is used and NRT science operations are conducted. In this case, between 8 and 10dB of link degradation can occur, mostly due to the limited bandwidth allocated at X-band for space-to-space links.

On the other hand, we also need to estimate the the mass of a MA communication payload as a function of EIRP and gain. To that end, I use the concept of payload difficulty once again, albeit the transformation to use in the case of MA payloads is slightly different from that used with SA payloads (see Appendix B). Furthermore, I assume that payload mass can be estimated as the sum of three components: phased-array antenna, high-power amplifiers and electronics. For the first element, I assume that an S-band antenna weighs 0.5kg and requires 5kg of electronics [234], while the power amplifiers can deliver 500W of total RF power. Similarly, the gain of the phased array elements is determined based on coverage requirements, which results in the two characteristic curves provided in Figure 5-26a. Observe that they differ depending on whether the system is optimized for Phobos



(a) MA Payload Characteristic Curve

(b) Payload Difficulty for MRN MA System

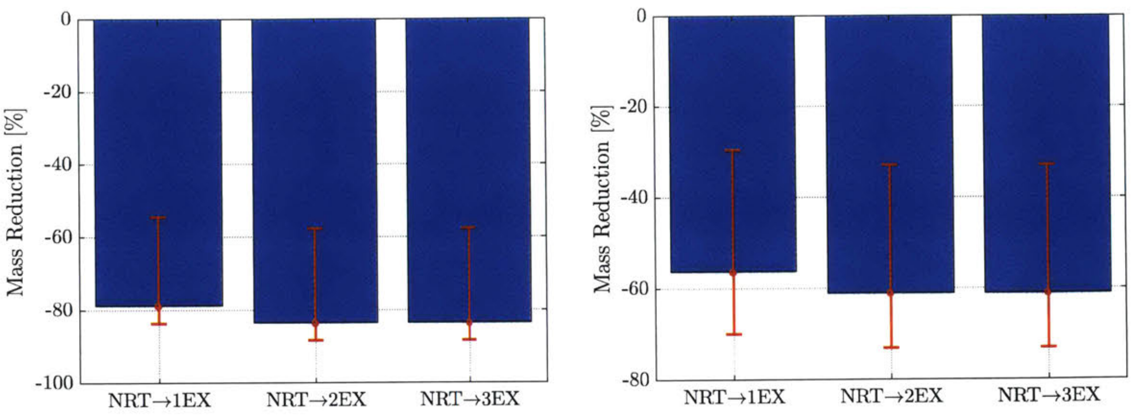
Figure 5-26: MRN MA System

or low Mars orbit (LMO) coverage. In the former case, each element of the phased array has almost 30 deg of field of view to maximize the percentage of time that Phobos lies within the main lobe of the phased array antenna. Alternatively, in the LMO case all antennas have a field of view of 12.4 deg, and consequently the same number of elements can deliver higher directivity.

Step 5-6.2. Multiple Access Payload Difficulty

Using the set of link budget tools developed in Step 5-5, I estimate the EIRP and payload difficulty required to ensure that the MA system on-board the MRN successfully supports the data rates and number of users for each science operational profile and the user burden. Several interesting remarks are possible:

- If the MA system is implemented at X-band, then the required payload difficulty is larger and consequently more mass to implement this link will be required. There are two primary rationales for this increase: First, the user and MA antenna have gains determined by user burden and coverage considerations respectively. Therefore, increasing the carrier frequency does not result in higher gain at either end of the communication link, it just results in resizing the antenna to satisfy the field of view requirement. Consequently, the link budget is penalized due to the increased losses at X-band. On the other hand, we also observe that the differential between X and S-band is particularly large if NRT operations are conducted. As demonstrated in Table 5.10, up to 10dB of self-interference need to be accounted for in this case due to bandwidth limitations in Mars orbit-to-orbit links at X-band. Therefore, the self-interference further increases the payload difficulty.



(a) Phobos Coverage

(b) Low Mars Orbit Coverage

Figure 5-27: Mass Savings in the MRN MA Payload. Red Bars Indicate Variability across MA Configurations

- Assuming that the MA system is implemented at S-band and operations are conducted with a fixed set of exchanges, a MA payload can be implemented with less than 35 antenna elements and consequently the mass is contained to 100kg or less. In that sense, if only LMO coverage is required and the best system configuration is assumed, then the MA payload is in the electronics-constrained region of the characteristic curve. Alternatively, if operations are conducted using the NRT approach, then payload is in the power-optimized region of the characteristic curve.

Step 5-6.3. Multiple Access Payload Mass Savings

Finally, the mass savings of utilizing the a non-continuous science operations mode for the MRN proximity links can be estimated directly from the characteristic curve in 5-26a. Figure 5-27 provides the obtained results, with band of uncertainty as computed from all possible configurations from the original morphological matrix. Observe that mass savings are highly significant in both cases, but particularly if Phobos coverage is assumed. Observe also that there is no significant difference between 3EX, 2EX and 1EX operational profiles, but rather it is a binary scenario. If all communications are provided in near real-time, then the high data rate to be provided vastly impacts the design of the MA system on-board the MRN satellites. In contrast, if a limited amount of latency is allowed, then the required data rates decrease by one order of magnitude and render the system implementation significantly easier.

Step 5-7. Development of Recommendations

Given the lessons learned from the latency-centric approach for architecting the Mars Relay Network, the following set of recommendations summarize the primary findings:

Question 1: Is is possible to conduct science activities on the Mars surface during the human exploration era without providing continuous communications to the astronauts?

Performing effective science on the surface of Mars is possible without near real-time communications between both planets. Several analog missions show that performing science using operations analogous to current Mars rovers is advantageous because (1) well-trained astronauts can perform effective science without the need for constant supervision, (2) sufficient time is allowed for the back-room science team to make informed recommendations on how to proceed, and (3) the large amount of data to be processed on Earth becomes rapidly unmanageable from a human factors perspective (e.g. fatigue) if real-time operations are required.

Question 2: How many exchanges per day should be utilized in the Mars human exploration era?

According to test subjects from past analog missions, 3 exchanges per day are almost as effective in conducting science activities as continuous communications in the presence of large propagation delays. This operations mode is largely preferred over 1 and 2 exchanges per day, as it allows scientists on Earth to provide feedback within the same Martian day, which is especially effective if sample tagging and collecting are performed during a morning and afternoon EVA.

Question 3: What is/are the most influential latency contributors in the current Mars-Earth network?

The current Earth-Mars data production system is largely limited by ground operations, most notably sequence planning and validation. Other significant sources of latency include lack of rover coverage by the MRN, as well as time allocated for data activity planning and total DTE transmission and propagation time.

Question 4: What is/are the most influential latency contributors in the future Mars-Earth network?

During the human exploration era, dedicated areostationary relay satellites will be deployed around Mars to provide continuous coverage to astronaut landing sites. Similarly, command generation and validation will not be needed. Therefore, the primary latency contributors will be data processing time on Earth by the science backroom, as well as DTE and proximity links used to return data from the Red Planet.

Question 5: Which data rates should be supported in the network if science operations are performed in near real-time, or with 1, 2, and 3 exchanges per day?

For the DTE link, between 140 and 70Mbps are required to return all latency-sensitive data over the trunk line that connects Mars and Earth. A similar effect can be observed in the DFE link, albeit data rates are contained in the 25 to 12Mbps range. For critical data returned over an X-band link, 6.5Mbps and 0.5Mbps should suffice. Finally, the proximity link data rates in the MA part of the MRN are specially sensitive to continuous vs. non-continuous operations. In that sense, up to 12Mbps might be needed in the forward service if NRT operations are required, while 2 to 1Mbps is expected depending on the number of exchanges per day.

Question 6: What are the savings in the DTE links that connect Mars and Earth if science operations are not conducted in real-time?

If the high rate DTE link is implemented at Ka-band, mass savings in the payload required at the MRN range from 7% to 25% to 55% approximately, depending on the number of exchanges supported per day (3, 2 and 1). The savings obtained in this case are inversely correlated with the cost of the DSN ground infrastructure, i.e. more capable antennas on Earth result in smaller savings on the MRN when science operations are optimized. On the other hand, if optical communications are utilized to implement the high rate link from Mars, then mass savings from non-continuous latency-sensitive services are highly dependent on the optical link performance, but expected values are significantly lower than for RF only networks.

Question 5: What are the secondary and tertiary latency contributors for data relayed from the surface of Mars back to Earth?

Proximity links and repatriation lines on Earth should also be considered as part of the end-to-end system that delivers latency-sensitive data. For the former, mass savings from different science operation modes will depend primarily on desired coverage in the Mars vicinity. If Phobos orbiters must be supported by the MRN, up to 80% mass savings can be obtained by deploying a payload that is only able to deliver 2Mbps and therefore supports 3 exchanges per day. Alternatively, if LMO coverage is required, then mass savings are moderate and range from 30% to 60% approximately.

5.3 Summary

In this case study, I have studied the effect of provisioning latency-sensitive services in the context of planned science operations on the surface of Mars during the human exploration era. The first half of the case study has been devoted to understanding if science operations can successfully be performed without continuous communications to and from the Red Planet, i.e. there is tactical back-room science on Earth that analyzes data immediately after it is produced, sent and received on Earth. Results from both previous analog missions and expert interviews with test subjects indicate that a science operational scheme where data is exchanged between both planets at a predefined cadence (similar to current Mars rover operations) is preferred in the presence of large light-time delays. In that sense, alternating sample tagging and sample collection, together with a mid-day feedback from the Earth's science back-room has been identified as valid mode of operations that is expected to deliver almost the same scientific value as having near real-time communications. Interestingly, if the mid-day feedback is not provided and operations are conducted analogously to current Mars rovers, then the expected scientists' satisfaction drops to almost 50%.

The second part of this case study has been devoted to understanding the latency contributors in the current and future Mars-Earth network, as well as the impact of not providing real-time communications for astronauts exploring the Red Planet's surface. To that end, I initially considered the current implementation of the MRN to understand with factors contribute to latency, both technical and non-technical. Using the centrality measure presented in Chapter 2, which in the context of this case study simply produces a ranking based on the relative amount of latency introduced by each part of the system, I identified both the DTE and proximity links as the primary sources of latency. The DTE was prioritized due to its high difficulty, specially at Mars superior conjunction. Results indicate that supporting science in non real-time mode has significant savings for the portion of the network, albeit improved support from the DSN progressively reduces them. If optical communications are used to provide the high data rate between Earth and Mars, then the savings from utilizing a limited number of exchanges per day is reduced, albeit uncertainty in link calculations given available information about deep space optical systems hinder our ability to provide accurate estimates on the mass savings. Finally, a similar study for the MA payload to be carried by the future MRN was also considered, which proved significant savings in this part of the network if non real-time operations are assumed and coverage for Phobos is required.

THIS PAGE INTENTIONALLY LEFT BLANK

6 Conclusions

6.1 Thesis Summary

Latency has long been a secondary requirement for space communication networks. Indeed, with the exception of some notable programs such as NPOESS, most studies related to space networks treat latency *in passing*, without understanding its implications in the infrastructure's ability to satisfy its customer needs. This has led, over the last decade, to two realizations: First, data from space satellites is sometimes delivered too late to the end user, who could have performed better science had it been delivered before. Second, levying blind end-to-end latency requirements in space communications networks results in significant cost and complexity increases, which then need to be effectively managed to satisfy tight budgetary and avoid undesired program cancellations.

To address the aforementioned issues, the primary goal of this thesis to provide an efficient approach to manage end-to-end latency requirements from a systems perspective and architect space communication networks around them. To do so, I decomposed the problem into four main tasks and structured the thesis accordingly: First, analyze current space networks to understand what part of the infrastructure we have in place induce latency. Second, based on current and future missions, survey which applications are constrained by latency, and categorize them depending on the type of requirement imposed. Third, develop a systems level approach to manage end-to-end latency based on the concept of network centrality. And fourth, apply the proposed approach to three case studies to understand its validity and demonstrate its usefulness when architecting future space communication networks.

The first two parts of the thesis are contained within Chapter 1. Initially, I studied the architecture of three networks used by NASA from a system perspective, understanding the different types of functions they provide and how they result in latency contributors that should be considered when quantifying the performance of the end-to-end system. In that sense, I argued that both service execution and service management functions are responsible for inducing latency, both over a wide range of values and due to a wide range of root causes. This resulted in the first important take-away of this thesis: End-to-end latency cannot be studied from a traditional communication or networking point of view with single conceptual model for all latency contributors, but is rather better analyzed from a systems perspective where end-to-end latency is decomposed in its constituents and then analyzed separately.

Next, I surveyed the literature for past, present and future missions concepts for which latency has been flagged as an important requirement. This exercise resulted in a categorization of space-related applications as a function of their latency requirements. On the one hand, certain missions return data that is not limited by latency in any capacity. Examples of these latency-unconstrained applications include climate data products build on daily, weekly, monthly or even yearly averages, as well as most data products from deep space probes. On the other hand, latency-constrained applications are characterized by generating data with a given expiration date. In particular, depending on the level of data timeliness and the type of requirement (hard vs. soft), they are further sub-divided into real-time, near real-time or latency-sensitive applications. Of those, I concentrate on the latter as they typically require returning large amounts of data over a pre-defined time interval that can be effectively traded against infrastructure cost.

In Chapter 2, I described a new latency-centric approach to architecting space communication networks that must provide support to latency-sensitive applications. The approach is rooted on three core concepts: System architecture analysis, namely functional and formal decomposition; utility theory; and centrality measures. System architecture analysis is used to decompose the system into a set of nodes (or elements) that abstract one or multiple latency contributors. On the other hand, utility theory is combined with the concept of betweenness centrality to obtain a metric that quantifies the amount of utility lost in a given node of the system, and consequently in a given latency contributor. This centrality measure is then used to rank contributors and focus the system architecting exercise towards parts of the network that are particularly deleterious with respect to overall data timeliness. Consequently, the proposed centrality measure performs the same role as a *heuristic function* in an heuristic optimization scheme: It estimates in which direction the system architecting approach must proceed for further detailed analysis.

The proposed latency-centric approach to architecting space communication networks is first benchmarked against a high fidelity simulation of an IP-like network in Chapter 3. Initially, an arbitrary network topology is assumed, with two primary latency contributors, packet transmission time (for connections) and packet processing time (for routers). Using an approach analogous to social network analysis, the ability to identify the critical latency contributor in the system is tested under a baseline nominal scenario and four stress cases that vary a wide range of operating conditions: Utility function concavity, routing strategy, data importance, and cost-heterogeneity. The results of this exercise demonstrate that the proposed centrality measure can successfully identify the ranking of latency contributors that most hinder data timeliness in the system, provided that the movement of data across the system is properly represented. It also provides a first batch of empirical evidence to support the usefulness of the proposed centrality measure.

Chapter 4 applies the latency-centric approach to architecting space communication networks to the design of networks that return satellite products used for weather prediction purposes. This case study has a dual purpose in the context of this thesis. First, it exemplifies how to use the proposed approach for one of the previously identified latency-sensitive space exploration applications. Second, it also serves as a retrospective case study that compares the rankings obtained using the proposed centrality measure against the design process of two real systems, namely NPOESS' SafetyNet and JPSS' CGI. Results indicate that the ranking produced by the centrality measure can successfully explain the core upgrades implemented by both programs, thus providing a second level of validation evidence against a realistic systems. Furthermore, additional results of the case study include optimal placement of ground stations for weather forecasting at mesoscale and global resolutions as a function of cost and programmatic risk, as well as direct comparison between as ground and space-based infrastructures.

Finally, Chapter 5 utilizes the latency approach to architecting space communication networks in a forward-looking case study, namely the design of the Mars relay network that provides services to astronauts on the Mars surface conducting science exploration activities. The first part of this case study was devoted to understanding how science can be conducted without continuous communications. Through expert interviews, I postulate that a network capable of delivering three exchanges per day, one at the beginning of the Martian day, one early in the afternoon, and one late at night can provide almost the same satisfaction to scientists that a network designed to deliver all non-critical data without any latency allowance. Given these findings, and the set of latency contributors present in the future MRN, two detailed system architecting studies are performed, one for the DTE link between Mars and Earth, and one for the MA part of the proximity links. For the former, significant data rate and mass savings are possible if 3 or 2 exchanges per day are provided, specially if deep space optical communications are not used to supported data from Mars. For the latter, implementing the required links for real-time communications is found to be significantly more difficult than with any other science operational profile due to the large data rates required despite the need to provide a lower number of simultaneous links.

6.2 Thesis Contributions

The primary set of contributions mirror the high-level structure of this thesis. The can be summarized into five primary categories:

1. **Identify** the latency contributors that should be considered when architecting communication networks that deliver services to space exploration applications.

2. **Characterize and classify** current and future space exploration activities, both human and robotic, as a function of the latency requirement they impose on the networks that support them.

3. **Propose** a latency-centric approach to architecting space communication networks based on the concept of network centrality for ranking latency contributors as function

of the utility loss they introduce.

4. **Quantify** the impact of trading latency vs. infrastructure cost for two latency-sensitive applications, namely weather satellite data and human science exploration on the Mars surface.

5. **Develop** models and tools to support the aforementioned system architecting studies. These include an extensible network simulation tool, detailed deep space link analysis models, satellite coverage and latency analysis tools, and a new approach for estimating the availability of space-to-ground optical network.

The first two contributions of this thesis are domain specific. In a sense, they can be viewed as the initial grind that must be done before getting to the interesting part of the problem, i.e. proposing and comparing new system implementations. Nevertheless, its insights are invaluable since they stress the importance of considering latency when defining user requirements for space networks. Note that my analysis focused on space exploration applications. Yet, there is nothing unique about them. A similar analysis could be conducted for traditional fixed or mobile satellite services, especially now that providing high data rate global connectivity through space and airbone-based platforms is highly popular.

The third contribution is methodological and is built upon concepts that have been traditionally used in the domain of systems engineering: Design structure matrices (or equivalently graph adjacency matrix for networks), utility theory and network analysis. The original motivation for this approach comes from the heterogoenity of latency contributors elicited in Chapter 1. Indeed, creating a unified model to capture all of them appropriately is typically challenging, if feasible at all, a fact that vindicates tackling the problem from a systems perspective instead. In that sense, the proposed latency-centric approach to architecting space communication networks utilizes the concept of network centrality to first understand and rank different latency contributors in the network, and then guide the system architecting process towards analyzing areas that will have significant impact in reducing end-to-end latency at the expense of increased infrastructure cost.

The fourth contribution is, once again, domain specific, and quantifies the impact of trading latency vs. infrastructure cost for two latency-sensitive applications. The first application considered is return of weather satellite data. This case study was originally motivated by the NPOESS program, its failure, and its evolution towards the a less capable yet more affordable CGI. By ranking the latency contributors and performing detailed tradespace exploration of the space-to-ground infrastructure that supports weather satellites, I was able to demonstrate that the proposed approach can be used to identify the primary latency contributors and gain insight in which parts of the system need to be improved to deliver the desired level of end-to-end performance. On the other hand, the second case study was forward-looking and demonstrated how the proposed approach is applied in the domain of deep space communications, namely the Mars Relay Network and its support of human exploration activities circa 2040.

Finally, the fifth contribution is intricately linked to the fourth one, and can be categorized in the domain of specific models and tools. They were developed to quantify performance and cost drivers for current and future space communication networks. For instance, a new end-to-end network simulation tool for interplanetary space networks was developed. It allows network architects to quantify the effect of setting latency requirements in the system, both from the perspective of required link capacity and relay storage. It can also be easily extended to model multi-band satellite systems, as well as gather statistics on the number of satellites being supported at once by the different elements of the network. On the other hand, detailed link analysis tools in the context of deep space networks, both in RF and optical communication systems were developed. They were coupled with technology performance forecasting to obtain first order estimates of mass savings in the Mars relay network if larger latency requirements are allowed. Finally, a novel approach for estimating the availability of space-to-ground optical ground networks was developed, both from a theoretical and statistical standpoint. Utilizing the widely available cloud fraction data set, I demonstrated that numerical approximation methods can be used to efficiently estimated the probability of having a certain number of space-to-ground links clouded.

6.3 Future Work

Several areas of future work were identified during the completion of this dissertation. First, applying the proposed framework to other space networks developed and under development would be beneficial to better understand its limitations. Traditionally, system studies have used the Iridium and Globalstar systems as examples of complex communication networks that would have been better managed with the principles of systems engineering and systems architecture. Luckily, new space communication networks are under development while this thesis is being completed. Not only is Iridium rolling out their next generation of satellites, but other companies such as OneWeb, Google, or Facebook, are competing to develop new satellite and sub-orbital networks that provide Internet to rural and remote areas of the world. It would therefore be interesting to understand whether latency-sensitive services would be required in these new upcoming networks, whether they are the limiting factor to consider when architecting them, and which latency contributors should be prioritized.

Another potential area of future work is related to the centrality measure definition. In that sense, section 2.5 provided an overview of centrality measures for system architecture in a generic context, and defined the different building blocks required for specifying their functional form as a function of the type of knowledge required, from application agnostic to application specific. A possible area of future work could build upon the proposed categorization and, for a diverse set of systems and applications, define centrality measures with different levels of domain-specific knowledge, and quantify how well they identify bottlenecks or elements of interest by comparison with their real counter parts. This could include, for instance, definitions build upon the concept of return on investment, and could yield optimal evolution paths for system from a cost-benefit perspective. Similarly, adaptations for systems with with other desired "ilities" could be envisioned.

Future work in the context of the two developed case studies has also been identified. For the Mars relay network, significant enhancements in capturing astronauts activities while conducting science activities should be incorporated in the analysis. This entails two primary tasks. One the one hand, direct interaction with members of NASA analog mission should be fostered in order to better understand what they expect from the network that will support their operations. In that sense, it is paramount that traffic models for interplanetary networks are adapted depending on the science operational profile rather than assuming a generic science day structure. On the other hand, additional interesting system considerations for the Mars-Earth network should be considered. For instance, I have implicitly assumed that the high rate DTE link must be provided by either a Ka-band or an optical link. In fact, a network using a tri-band trunk line between both planets is also conceivable and would result in smaller communication payloads that can be used to return data with different levels of quality of service.

Finally, this body of research and the results herein contained would also benefit from increased model fidelity, both in the set of effects they capture and the input values that they assume. As an example, optical communications impaired by cloud coverage should be better characterized as a latency contributor. The models presented in this thesis argue that two optical telescopes per site would suffice to ensure cloud free line of sight with high probability. Cloud models that also capture statistics on cloud duration would be invaluable in assessing the distribution of latency they introduce, and thus allow a more informed analysis on what level of site redundancy should be provided to deliver the expected quality of service.

A NASA Space Communication Networks

This appendix summarizes the descriptive decomposition exercise conducted in order to identify common functional and form elements for current space communication networks. In particular, the architecture and services provided by four NASA networks, namely the DSN, the NEN, the SN and the CSO is first presented. Then, the mapping between elements of form and the functionality they perform is presented through a modified N^2 diagram [235].

A.1 The Deep Space Network

The DSN is a space communication network that provides communication, navigation and scientific services for missions and celestial bodies across the solar system. It is specifically architected to satisfy the needs of customers that operate at deep space distances and therefore experience high difficulty links.

The network is composed by four main locations: Three deep space communication complexes (DSCC), one in Goldstone, CA (GDSCC), one in Madrid, Spain (MDSCC), and the last one in Canberra, Australia (CDSCC); and a network control center (NCC) at the Jet Propulsion Laboratory (JPL) in Pasadena, CA. The three DSCCs are equipped with multiple 34m and one 70m antennas that are used to send and receive signals from remote spacecraft, track them, and investigate celestial bodies of scientific interest [236].

The exact set of services provided to DSN customers can be found in reference [237] and includes:

- Command Services
- Telemetry Services
- Tracking Services (including Delta-DOR)
- Calibration and Modeling Services
- Radio Science Services
- Radio Astronomy/VLBI Services
- Radar Science Services

Mission that want to utilize these services have to interface with the DSN both during the planning and the operations phase. For the planning phase, the DSN Commitments Office [47] serves as the primary point of contact between the network and the customer, assisting in the development of the mission communication capabilities and ensuring that they remain compatible with the DSN. In contrast, during the operations phase mission planners interface directly with the DSN assets in order to schedule communication passes, provide trajectory information, send and receive information from the remote spacecraft, and receive performance measurements for the services they request.

A.2 The Near-Earth Network

The NEN is a space communication network composed of 14 remote ground stations that support near-Earth orbiting spacecraft. Half of these ground stations are owned and maintained by NASA while support from the other ones is achieved through commercial contracts with ground station providers such as the Universal Space Network (USN) or Kongsberg Satellite Services (KSAT) [176].

Reference [42] summarizes the set of services offered by the NEN in four categories. Note that, compared to the DSN, the NEN does not offer science services.

- Forward Data Delivery
- Return Data Delivery
- Radiometric Services (excluding Delta-DOR)
- Trajectory Services

Missions that require support from the NEN interface initially with Network Integration Management Office (NIMO) located at the Goddard Spaceflight Center (GSFC). Its responsibilities include performing network loading analyses, RF link margin and coverage analyses, compatibility testing and orbital analyses. Once the mission is in its operations phase, it interfaces with the Flight Dynamics Facility (FDF) for trajectory determination and the Wallops Orbital Tracking Information System (WOTIS) for service provision. The latter is composed of three main elements: a message handling system, a scheduling engine and a database. The scheduling engine allows mission operators to define generic scheduling requirements in the form of rules. They are then processed by the WOTIS in order to generate conflict-free schedules. Alternatively, missions may also request specific antennas and pass times or schedule NEN support manually by directly interfacing with the NEN Scheduling Office (NENSO) [176].

A.3 The Space Network

The SN is a space communication network established in the early 1980s that is capable of providing continuous tracking and communication services for spacecraft operating at LEO. In contrast to the DSN and the NEN, the SN is composed of both a space and a ground segment. The Tracking and Data Relay Satellite System (TDRSS) is a constellation of GEO satellites located approximately 120deg apart in the equatorial plane. Each TDRS carries two 5 meter SA antennas that allow high data rate communications through S, Ku and Ka-band frequency channels, as well as an MA phase arrayed antenna that simultaneously supports up to 5 customers through a low data rates S-band CDMA-based system [238]. The TDRSS constellation is supported by three ground terminals, two of them located at NASA's White Sands Complex (WSC) in NM, and the other one located in Guam. Information sent to a TDRS satellite is downlinked to one of these ground stations and then forwarded to the appropriate MOC.

Reference [238] provides a thorough description of the services offered by the SN as well as the performance parameters that characterize them. They are divided in the following categories:

- MA telecommunication services
- SSA telecommunication services
- KuSA telecommunication services
- KaSA telecommunication services
- Tracking and clock calibration services

Note that the service categorization is in this case significantly different from that of the DSN and the NEN. Services are primarily characterized from a *form* perspective (e.g. the frequency band) rather than by the functionality they accomplish. In that sense, the MA, SSA, KuSA and KaSA all services provide both forward and return data delivery services. On the other hand, the tracking and clock calibration services are functionally equivalent to the radiometric services from the NEN. Finally, SN customers obtain trajectory services from the FDF which processes the radiometric products provided by TDRSS.

On the other hand, missions requiring support from the SN interface with the NIMO for planning, testing and compatibility purposes. The set of provided functionality is analogous to that of the NEN. In contrast, mission operations are conducted through the SN Web Services Interface (SWSI). They provide the necessary functionality to request SN contacts and monitor their execution. Data from the SN ground stations is sent first to the Network Control Center Data System (NCCDS) in WSC and then forwarded to the customer MOC through the NASA Integrated Services Network (CSO).

A.4 The Communications Service Office

The CSO is a WAN that provides terrestrial voice, video and data services across all NASA facilities. Its services are not exclusive to support of remote spacecraft, but also include corporate network services, as well as dedicated center and facility services for local area network deployment and maintenance. Its implementation is part of the IT Infrastructure Integration Program (I3P) contract currently awarded to the Science Applications International Corporation (SAIC) [239] [240].

The resulting IP-based infrastructure is known as the Internet Protocol Operational Network (IONet). It is structured in four tiers (closed, restricted, open and external) that provide enhanced levels of security to the information that flows through them. For instance, the closed IONet is used to communicate tracking stations and network control centers and it has no connectivity to the external internet. In contrast, the open IONet is used by missions to route incoming scientific data from the spacecraft to the MOCs [176].

The CSO offers three types of services for mission support purposes: Mission routed data, dedicated mission data and dedicated mission voice. Missions utilizing the routed data service obtain communication services through a carrier managed[1] backbone IP-network. They interface with it at Service Demarcation Points with a local area network (LAN) interface. Common commercial standards such as 10 Base T, 100 Base TX, 100 Base FX or Gigabit Ethernet can be used, as well as several legacy analogical interfaces [239].

In contrast, missions that use the dedicated mission data and/or voice services are supported through the CSO Mission Network. For data purposes, the network provides them with data rates between 9.6kbps and 1.5Mbps, while voice is distributed at 8 to 64kbps. The CSO Mission Small Conversion Devices (SCD) can be used to support legacy spacecraft, as well as CCSDS/SLE formatted data streams. They are capable of converting them into commonly supported data formats such as UDP/IP or TCP/IP [239].

Within each communication service, the CSO offers four service categories depending on the level of performance required. As shown in table A.1, the main difference between each category lies on the percentage of availability and the restoral time and is mainly a function of the data criticality. For instance, in order to meet the stringent requirements from Real-Time Critical services, the CSO guarantees full redundancy on all routes between origin and destination. In contrast, Mission Critical Level A services provide redundancy at the hardware level, but no physically disjoint paths between origin and destination are guaranteed.

[1] The infrastructure belongs to an external telecommunications service provider

Table A.1: CSO Service Categories

	Service Category			
	Real-Time	Mission Critical Level A	Mission Critical Level B	Mission Critical Level C
Mission Routed Data:				
Availability	99.98%	99.95%	99.90%	99.50%
Restoral Time	< 1min	2hr	< 4hr	< 4hr
Coverage Period	24 × 7	24 × 7	24 × 7	24 × 7
Acceptable Packet Loss	0.001	0.001	0.001	0.001
RTT	< 120ms	< 120ms	< 120ms	< 120ms
Dedicated Mission Data:				
Restoral Time	99.98%	99.95%	99.90%	99.50%
Coverage Period	< 1min	2hr	< 4hr	< 4hr
Dedicated Mission Voice:				
Availability	99.98%	99.95%	99.90%	99.50%
Restoral Time	< 5min	2hr	< 4hr	< 4hr
Coverage Period	24 × 7	24 × 7	24 × 7	24 × 7
RTT	< 500ms	< 500ms	< 500ms	< 500ms

A.5 Function to Form Mapping for Space Communication Networks

Table A.2 presents the architecture of the DSN, NEN, SN and CSO using a modified N^2 diagram with elements of form represented by rows and functionality represented by columns. Note that the latter has been grouped according to the categorization presented in section 1.5.1.

Table A.2: Function to Form Allocation for NASA's DSN, NEN, NISN and SN

			Service Execution Functions							**Service Mgmt. Functions**			**Network Mgmt. Functions**		
Element	**Location**	**Status**	**Beamforming**	**Sampling**	**Modulating**	**Framing**	**Coding**	**Routing**	**Store&Forwarding**	**Interfacing**	**Scheduling**	**Navigating**	**Controlling**	**Testing**	**Insuring**
SN															
TDRSS															
TDRS 3	AOR	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 5	POR	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 6	AOR	Stored	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 7	IOR	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 8 (H)	IOR	Operational	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 9 (I)	AOR	Operational	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 10 (J)	POR	Operational	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 11 (K)	POR	Stored	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
TDRS 12 (L)	AOR	Stored	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
Ground Segment															
WSGT															
SGLT-4	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	
SGLT-5	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	
WSGT-DIS	WSC	Deprecated	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	
GDIS	WSC	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗	
WDISC	WSC	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗	
WART	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	

Continued on next page

Table A.2 – continued from previous page

Element	Location	Status	Beamforming	Sampling	Modulating	Framing	Coding	Routing	Store& Forwarding	Interfacing	Scheduling	Navigating	Controlling	Testing Insuring
STGT														
SGLT-1	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
SGLT-2	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
SGLT-3	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
STGT-DIS	WSC	Deprecated	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
WDISC	WSC	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗
GRGT														
SGLT-6	Guam	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
GDIS	Guam	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗
ETGT	WSC	Operational	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
OTHER														
MILA	Merrit Island	Operational	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
MTRS	Antarctica	Operational	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗
SPTR	Antarctica	Operational	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗
ATF	Dongara, Aus	Operational	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
BRTS-1	WSC	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓
BRTS-2	Australia	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
BRTS-3	Ascension Island	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
BRTS-4	American Samoa	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
Control Centers														
NCCDS	WSC	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✓	✗	✓	✗
NIMO	GSFC	Operational	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
FDF	GSFC	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
NISN														

Continued on next page

Table A.2 – continued from previous page

Element	Location	Status	Beamforming	Sampling	Modulating	Framing	Coding	Routing	Store& Forwarding	Interfacing	Scheduling	Navigating	Controlling	Testing Insuring
DFRC	-	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
OAFS	-	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
PFLT	-	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
JSC	JSC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
KSC	KSC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
NYC	NYC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
ARC1 router	Ames	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
ARC2 router	Ames	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
MSFC-CHI router	GRC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
GSFC1 router	GSFC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
GSFC2 router	GFSC	Operational	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
CSO	NASA	Operational	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗
NEN														
Ground Segment														
SGS	Norway	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
WGS	Virginia	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
MGS	Antarctica	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
ASF	Alaska	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
WSC	New Mexico	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
MILA	Florida	Operational	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
KSAT	Norway	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
USN Alaska	Alaska	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
USN North Pole	Alaska	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
USN Hawaii	Hawaii	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗

Continued on next page

Table A.2 – continued from previous page

Element	Location	Status	Beamforming	Sampling	Modulating	Framing	Coding	Routing	Store& Forwarding	Interfacing	Scheduling	Navigating	Controlling	Testing Insuring
USN Australia	Australia	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
SSS	Chile	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
WAL	Germany	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
KIR	Sweden	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
HBK	South Africa	Comercial	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
Control Centers														
WOTIS	WSC	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
WFF	Wallops	Back-up	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
NENSO	Wallops	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
NENPAO	Wallops	Operational	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗
NIMO	GSFC	Operational	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
FDF	GSFC	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
DSN														
Ground Segment														
GDSCC														
DSS-14	Goldstone, CA	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-15	Goldstone, CA	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-23	Goldstone, CA	Planned (2024)	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-24	Goldstone, CA	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-25	Goldstone, CA	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-26	Goldstone, CA	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
SPC	Goldstone, CA	Operational	✗	✗	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
MDSCC														
DSS-53	Madrid, Spain	Planned (2020)	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Continued on next page

Table A.2 – continued from previous page

Element	Location	Status	Beamforming	Sampling	Modulating	Framing	Coding	Routing	Store& Forwarding	Interfacing	Scheduling	Navigating	Controlling	Testing Insuring
DSS-54	Madrid, Spain	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-55	Madrid, Spain	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-56	Madrid, Spain	Planned (2019)	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-63	Madrid, Spain	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-65	Madrid, Spain	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
SPC	Madrid, Spain	Operational	✗	✗	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
CDSCC														
DSS-33	Canberra, Aus	Planned (2022)	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-34	Canberra, Aus	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-35	Canberra, Aus	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-36	Canberra, Aus	Planned (2026)	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-43	Canberra, Aus	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
DSS-45	Canberra, Aus	Operational	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
SPC	Canberra, Aus	Operational	✗	✗	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗
Control Centers														
DSNPSO	JPL, CA	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
DSNCO	JPL, CA	Operational	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
ROC	Monrovia, CA	Back-up	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✓
ECC	Goldstone, CA	Back-up	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✓
CTT-22	Movable	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
MIL-71	Florida	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
DFT-21	JPL, CA	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
JPL-Central	JPL, CA	Operational	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✓	✗
MGSS	JPL, CA	Operational	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗

B Payload Difficulty

Communication payloads on-board relay satellite systems oftentimes provide the capability to transmit and receive at multiple frequency bands and data rates. For instance, all TDRS satellites carry two SA antennas that support three different frequency bands, S, Ku and Ka, and provide service at data rates more than two orders of magnitude different from each other. Therefore, when designing relay satellite systems that potentially operate at multiple frequency bands, it is important to understand which of the offered services drives the relay spacecraft mass and power.

In this appendix, the concept of *payload difficulty* is introduced as an intermediate construct to help design space communication payloads. To that end, I first provide a succinct review of the link budget equation and how it can be utilized to compare links at different frequency bands. Then, I describe the process of transforming traditional communication requirements (e.g. data rates) into system-level figures of merit (e.g. payload mass and power). Finally, the framework is applied to the case of a SA parabolic antenna, an optical telescope and a phased array MA payload.

B.1 The Link Budget Equation

The primary function addressed by a space communication system such as TDRSS is the transmission and reception of information reliably through a set of wireless links across different entities: Users and relay satellites, relay satellites and ground stations or multiple relay satellites. The fundamental tool used by communication engineers to design such links is a *link budget*, i.e. a power balance equation that ensures that enough power is received at end of the communication link to successfully decode the data is embedded in the electromagnetic signal that propagates through space. Numerous authors have explained the basics of link budget equations in the context of space systems. For instance, Wertz [39] provides an excellent treatment of its different constituents for system-level studies. In turn, Jo and Maral provide in References [241] and [40] a more formal and in-depth treatment they specifically target communication engineers. Consequently, the reader is referred to these references for an exhaustive introduction on the link budget equation (and key concepts such as dB), which will be used frequently throughout this appendix.

The performance of a communication link from the perspective of the receiver is a function of two elements: The signal power received C, and the level of noise measured at the receiver entrance N. The ration between them, known as signal-to-noise ration SNR is the fundamental FOM for any RF link, as it is directly related to the probability of making an error when reconstructing the data bits being transmitted. The signal power received is typically computed as

$$C = P_t + G_t + G_r - L_{fs} - L \tag{B.1}$$

where

- P_t is the transmitter power.
- G_t and G_r are the transmitter and receiver gain respectively.
- L_{fs} are the free space losses.
- L are other miscellaneous losses incurred in the end-to-end signal transmission path. Examples of losses included in L are implementation losses at receiver and transmitter, polarization losses, atmospheric losses or synchronization and quantization losses among others.

Frequently, the sum of transmitter power and gain is referred to as Equivalent Isotropically Radiated Power (EIRP). On the other hand, noise is typically assumed to be additive, white and Gaussian for RF communication systems (AWGN), and is assumed to be proportional to the receiver's noise temperature T_{rx}, expressed in Kelvin:

$$N = kT_{rx}B_n \tag{B.2}$$

where

- k is the Boltzmann constant.
- T_{rx} is the receiver noise temperature.
- B_n is the receiver noise bandwidth.

A related and more useful expression for the link budget equation in digital communication systems is based on the energy per bit rather than signal power:

$$\frac{E_b}{N_o} = SNR + B_n - R_b \tag{B.3}$$

where R_b is the link data rate in bits per second. If coding is applied to protect the link against errors, then R_b is the coded data rate also measured in bits per second.

Characterizing all parameters of the link budget equation to a certain degree of accurateness is usually a lengthy process that depends on many factors: Link frequency band, atmosphere and planet conditions, antenna pointing profile, etc. Rather than focusing on these details, in this appendix I will focus on two primary system level elements of the link budget equation:

Frequency band and data rate. For instance, it is apparent from equation B.3 that two links with equal SNR and different data rate will result in different $\frac{E_b}{N_o}$. It can be proven that the link bit error probability can be expressed as a function of the $\frac{E_b}{N_o}$ [43], and consequently a link that operates at higher data rates will yield, on average, more erroneous bits. Note that, intuitively, this is a sensible conclusion. If the power transmitted remains constant but the data rate increases, each bit of information is encoded with less energy. Therefore, given a constant level of noise energy, it is more likely that this noise will "trick" the receiver into thinking that a '1' is a '0' and vice versa.

On the other hand, three main parameters are affected by the choice of frequency band: Transmitter gain, receiver gain and free space losses. Additionally, other factors such as atmospheric losses are also dependent on frequency, especially for $f \geq 3\text{GHz}$[1]. Disregarding the dependence of noise temperature and atmospheric losses with frequency, the link budget equation can be expressed as:

$$\frac{E_b}{N_o} = P_t + h_t(f_c^2) + h_r(f_c^2) - l_{fs}\left(f_c^2\right) - L + B_n - R_b \tag{B.4}$$

where

- f_c is the link central frequency, dependent on the band at which the link operates.
- $h_t(\cdot)$ is a function that estimates the gain of the transmitting antenna. For instance, for a parabolic antenna of diameter D,

$$h_t(D, f_c) = \eta \left(\frac{\pi D}{\lambda}\right)^2, \tag{B.5}$$

  with $\lambda = \frac{c}{f_c}$.
- $h_t(\cdot)$ is a function that estimates the receiving antenna gain.
- $l_{fs}\left(\cdot\right)$ is the function that estimates the free space losses: $l_{fs}\left(d, f_c\right) = \left(\frac{4\pi d}{\lambda}\right)^2$, $\lambda = \frac{c}{f_c}$.

Note that equation B.4 is significantly more informative than equations B.1 and B.3 to a systems engineer. For instance, it clarifies why a link between two directional antennas is easier to close at higher frequency bands. Indeed, the transmitter gain will increase as f_c^2, and so will the receiver gain and free space losses. Since the latter has a negative sign in front, the increase in free space losses will cancel out the increase in receiver gain. Yet, the extra gain at the transmitter will be left as a "net" gain from increasing the link's frequency.

Another interpretation of this argument, even more amenable to those not familiar with communication concepts such as gain, can be provided through the data rate. Consider an RF SA payload and antenna that is used to implement an inter-satellite link that offers two

[1] At $f \leq 3\text{GHz}$, the Earth atmosphere can be considered "transparent".

services: S-band link at 1Mbps and Ka-band service at 100Mbps. The goal of a systems engineers is to understand which of the two links will be driving the payload mass and power requirements. An initial assessment of the problem could lead to the following conclusion: Since 100Mbps is two orders of magnitude larger than 1Mbps, the Ka-band service is the limiting link in this system. Alternatively, we can approach the problem using Equation B.4 and rolling the effect of the frequency band change into the link data rate. In particular, assuming that both services are implemented using a single communication payload with constant RF power P_t and parabolic antenna of diameter D, the difference in data rate between the X-band and Ka-band links, in logarithmic scale, can be estimated as

$$R_b^{(Ka)} - R_b^{(S)} = 10\log_{10}\left(\frac{f_c^{(Ka)}}{f_c^{(S)}}\right)^2. \tag{B.6}$$

Using NASA's band allocations [42], we estimate $R_b^{(Ka)} - R_b^{(S)} \approx 21dB$ and therefore $R_b^{(Ka)} \approx R_b^{(S)} + 21dB \approx$ 125Mbps. In other words, it is equally difficult, from the perspective of a communication payload, to provide a link at S-band and 1Mbps, or a link at Ka-band and 125Mbps. Consequently, since our Ka-band service only requires 100Mbps, the S-band payload will drive the mass and power requirements of our relay satellite system.

B.2 Payload Difficulty

Given the realization that frequency band is a paramount factor when sizing space communication payloads, we define the concept of ***payload difficulty*** as the equivalent EIRP that a payload must deliver to provide a forward service[2] at fixed data rate normalized to a baseline frequency. If the return services[3] are the constraining factor, then ***payload difficulty*** is equally defined but using the G/T figure of merit instead. Note that the choice of baseline frequency is arbitrary, i.e. it can be set to any value as long as it is consistent throughout all calculations.

Care must be taken when defining the normalization factor for payload difficulty. Most communication engineers are familiar with the ability to transform link budget quantities such as EIRP, gain or data rate from one frequency band to another using Equation B.6. However, since payload difficulty is concept fundamentally related to mass, the normalization expression must be derived based on the equation that characterizes this quantity. To that end, in Section B.3 I derive the normalization factor for an SA payload with parabolic antenna, while Section B.4 provides the corresponding equations for an MA phased array system. Note that, as a first approximation, the derivations from Section B.3 are also valid for optical telescopes [217].

[2] Forward services represent the transmission of data from the relay to the user.

[3] Return services represent the transmission of data from the customer to the relay.

B.3 Payload Difficulty for Single Access Payloads

It is well documented [216] [217] that the mass of a parabolic antenna is a function of the diameter and can be expressed as

$$mass_{HGA} = k_1 D^{\beta}, \tag{B.7}$$

where k_1 is a constant that depends on the antenna technology and $2 < \beta < 2.7$ typically. Observe that, intuitively, the exponent β must be > 2 so that the mass of the parabolic dish increases with its surface area. In practice, $\beta \approx 2.4 - 2.7$ are observed, especially if the mass of the arm/gimbal necessary to point the dish is taken into account.

Consider an SA payload that must deliver $EIRP_{ka}$ dBW to close a link at Ka-band. Then, using Equation B.5 it is immediate to prove that the antenna mass can be easily estimated as

$$m_{HGA}^{ka} = k_1 D^{\beta} = k_1 \left(\frac{\lambda_{ka}}{\pi}\right)^{\beta} \left(\frac{G_{ka}}{P_t \eta}\right)^{\beta/2}, \tag{B.8}$$

where λ_{ka} is the central wavelength at Ka-band, G_{ka} is the gain, in linear scale, to be provided by the antenna, and P_t is the transmission power in Watts from the HPAs. Define now the following transformation:

$$PD_b = EIRP_{ka} + 10 \log_{10} \left(\frac{f_b}{f_{ka}}\right)^2, \tag{B.9}$$

where PD_b is the payload difficulty at an arbitrary baseline band B, and f_b and f_{ka} are the central frequency of B and Ka-band respectively. Then, we can compute the mass of this SA payload at band B as follows:

$$\begin{aligned} m_{HGA}^{b} =& k_1 D^{\beta} = k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_b}{P_t \eta}\right)^{\beta/2} = \\ =& k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{G_{ka}}{P_t \eta}\right)^{\beta/2} \left(\frac{f_b}{f_{ka}}\right)^{\beta} = \\ =& k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{G_{ka}}{P_t \eta}\right)^{\beta/2} \left(\frac{\lambda_{ka}}{\lambda_b}\right)^{\beta} = \\ =& k_1 \left(\frac{\lambda_{ka}}{\pi}\right)^{\beta} \left(\frac{G_{ka}}{P_t \eta}\right)^{\beta/2}. \end{aligned} \tag{B.10}$$

Comparing the right hand side of Equations B.8 and B.10 demonstrates that m_{HGA}^{b} computed using PD_b is in fact equal to m_{HGA}^{ka} computed using the original requirement $EIRP_{ka}$. Therefore, by virtue of transformation B.9, we can now compute the mass of the SA payload at B-band even if the service requirements (e.g. data rate) are specified in another band.

On the other hand, other significant drivers for the mass of an SA payload include the HPAs, electronics, frequency sources or mounting case. Assume that, out of these, only the mass required to provide RF power depends on P_t, while the electronics and other factors can be considered constant [216]. Then, the total mass of the payload can be estimated as

$$\begin{aligned} mass =& mass_{HGA} + mass_{HPA} + mass_{other} = \\ =& k_1 D^{\beta} + f\left(P_t\right) + k_2, \end{aligned} \tag{B.11}$$

where $f\left(\cdot\right)$ is a function that translates Watts of RF power into kilograms of payload mass. This function can be typically assumed linear or quasi-linear [216] as more power is either provided by using larger power amplifiers or by combining power from multiple smaller amplifiers. In either case, Equations B.10 and B.11 can be combined to obtain the final expression for the mass of a communication payload normalized to a baseline frequency band B:

$$mass_{SA} = k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_b}{P_t \cdot \eta}\right)^{\beta/2} + f\left(P_t\right) + k_2. \tag{B.12}$$

Observe that Equation B.12 is particularly useful for systems engineering purposes. Indeed, parameters k_1, β and η define the technology available to implement the parabolic dish. For instance, a solid antenna will have $k_1 \approx [6-9] kg/m^{\beta}$ [216] and an efficiency of $\eta \approx 0.55$. In contrast, a large deployable dish will have $k_1 \approx< 1 kg/m^{\beta}$ [216] and an efficiency of $\eta \leq 0.55$. Similarly, $f\left(P_t\right)$ quantifies how easy it is to procure RF power in space communication payloads, while k_2 is representative of constant elements such as baseband electronics, frequency sources and mounting cases.

B.3.1 Optimal Mass for Single Access Payloads

A typical problem to solve during the design of an SA payload is to find its optimal mass based on the communication requirements of its forward service. To that end, we proceed as follows:

1. Compute the EIRP required to close the link budget equation for all services to be provided (typically at different frequency bands).

2. Select a baseline frequency band (for instance, the lowest of all bands from the payload services).

3. Transform the EIRP requirements at different frequency bands to *payload difficulty* at the baseline band using transformation B.9 and select the worst case. Note that *payload difficulty* is expressed in comparable dBW as they are all measured at the same frequency band.

4. Utilize Equation B.12 to estimate the payload mass assuming a sensible range of transmit power (e.g. 50-200W for TWTAs), and select the best case.

Alternatively, and assuming that the antenna is parabolic, we can perform the last step analytically by deriving the optimal transmit power to use in the payload assuming that $f(P_t)$ is a continuous function:

$$\begin{aligned}\frac{\partial mass}{\partial P_t} =& k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_t}{\eta}\right)^{\beta/2} \frac{\partial}{\partial P_t}\left(\frac{1}{P_t}\right)^{\beta/2} + \frac{\partial}{\partial P_t} f(P_t) = \\ =& -k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_t}{\eta}\right)^{\beta/2} \left(\frac{1}{P_t}\right)^{\beta/2-1} \frac{1}{P_t^2} + f'(P_t) = \\ =& -k_1 \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_t}{\eta}\right)^{\beta/2} \frac{1}{P_t^{\beta/2+1}} + f'(P_t)\end{aligned} \qquad \text{(B.13)}$$

Equating B.13 to zero, we obtain

$$P_t^* = \left[\frac{k_1}{f'(P_t)} \left(\frac{\lambda_b}{\pi}\right)^{\beta} \left(\frac{PD_t}{\eta}\right)^{\beta/2}\right]^{\frac{1}{\beta/2+1}}. \qquad \text{(B.14)}$$

Once P_t^* has been computed, direct substitution into Equation B.12 will yield the optimal mass, a step that is not analytically performed since it is not particularly informative. On the other hand, Equation B.14 is certainly interesting. Note that $f'(P_t)$ is the marginal cost, in units of mass per unit of RF/optical power (i.e. it is expressed in units of kg/W), of procuring transmission power in a space-based system. For instance, if $f'(P_t) = 0$, then transmitting more power incurs in no penalty and therefore the optimal solution is to provide enough power so that the link closes and no antenna is needed. On the other hand, if $f'(P_t)$ is high enough then the amount of power utilized will be limited in favor of a larger aperture.

Utilizing the concept of payload difficulty we can now compare the performance of RF and optical communications for high rate transmitting. As an example, assume that a high rate (~ 100Mbps) link between Mars and Earth requires 65-75dBW of payload difficulty if transmitted at Ka-band using a baseline frequency of 2.2GHz, while the same link at optical frequency only requires between 26-38dBW of payload difficulty. Similarly, assume that future deployable antennas will have densities of 0.6kg/m^2 approximately, while TWTAs at Ka-band will be able to provide RF power at rates of 0.015kg/W (e.g. two redundant 3kg TWTA delivering 500W [216]). Finally, assume that optical telescopes will require 250kg/m^2 [217] and 1.15kg/W respectively [I21].

Figure B-1 plots the mass vs. payload difficulty curves for the RF and optical payload assuming the technology values previously specified and the optimal transmit power from Equation B.14. Observe that, in the Mars context, the payload mass required to provide a 100Mbps class link between Earth and the Red Planet is lower if implemented at Ka-band

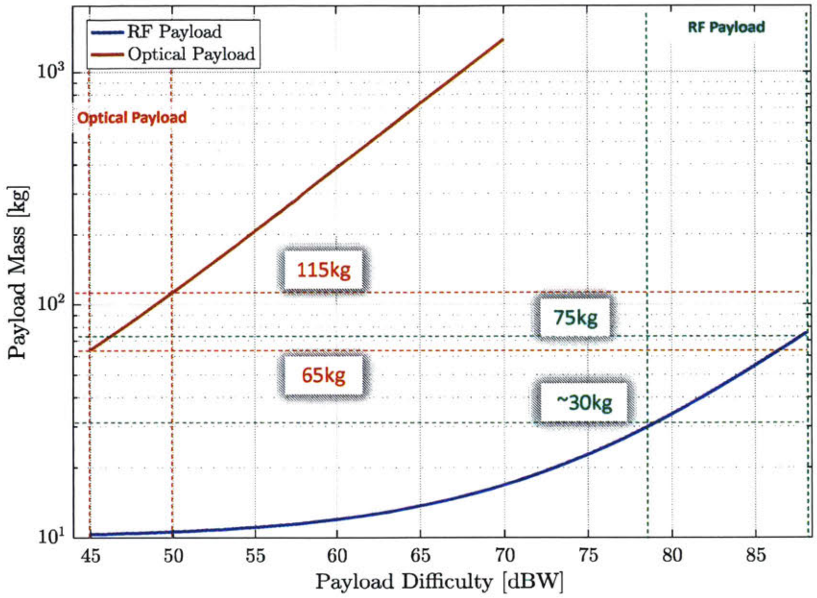


Figure B-1: Mass vs. Payload Difficulty

instead of optical (30-75kg vs. 65-115kg), even though the EIRP required in the latter case is on the order of 40dBW lower. This, of course, assumes that no progress towards reduction of mass and power requirements for space telescopes occurs and therefore cannot be considered a final result (refer to Chapter 5 for the in-depth analysis). However, it illustrates how *payload difficulty* is a useful construct that helps systems engineers translate traditional communication requirements such as EIRP and G/T into payload mass and power given a reduced set of technological parameters.

B.4 Payload Difficulty for Multiple Access Payloads

The derivation of *payload difficulty* for a MA payload is performed analogously to that of a SA payload. Yet, the transformation between EIRP and PD is necessarily different since the equation that relates EIRP and payload mass has a different functional form. To initiate the discussion for an MA payload, let us consider the mass of a phased array antenna. Based on reference [234], it can be approximated as

$$m_{PHA} = N_{elem} \left(k_1 + k_2\right) \lambda^{\gamma} \tag{B.15}$$

where

- N_{elem} is the number of elements in the phased array and can be estimated as $\frac{EIRP}{G_{elem} P_t}$ with all variables in linear scale and G_{elem} the gain of one antenna element.
- k_1 is the normalized mass of one antenna element. For instance, if an S-band antenna

element weighs 500g at S-band, then $k_1 = \frac{500}{\lambda_S^\gamma}$.

- k_2 is the normalized mass for the electronics of one antenna element, including phase shifters and low noise amplifiers.

- γ is a factor that indicates how the mass of an antenna element and its electronics varies as a function of the payload operating wavelength.

Consider now the following transformation

$$PD_b = EIRP_{b'} + 10\log_{10}\left(\frac{f_b}{f_{b'}}\right)^\gamma. \tag{B.16}$$

Then, the mass of a phased array operating at band b' and a requirement of $EIRP_{b'}$ dBW is

$$m_{PHA}^{b'} = \frac{EIRP_{b'}}{G_{elem}P_t}(k_1 + k_2)\lambda_{b'}^\gamma. \tag{B.17}$$

Similarly, the mass of a phased array operating at band b and with a *payload difficulty* requirement PD_b is

$$\begin{aligned} m_{PHA}^{b} =& \frac{PD_b}{G_{elem}P_t}(k_1 + k_2)\lambda_b^\gamma = \\ =& \frac{EIRP_{b'}}{G_{elem}P_t}\left(\frac{f_b}{f_{b'}}\right)^\gamma (k_1 + k_2)\lambda_b^\gamma = \\ =& \frac{EIRP_{b'}}{G_{elem}P_t}\left(\frac{\lambda_{b'}}{\lambda_b}\right)^\gamma (k_1 + k_2)\lambda_b^\gamma = \\ =& \frac{EIRP_{b'}}{G_{elem}P_t}(k_1 + k_2)\lambda_{b'}^\gamma. \end{aligned} \tag{B.18}$$

Once again, comparison of the right hand side of Equations B.15 and B.18 yields the desired result: The phased array antenna mass will be the same if estimated with an EIRP requirement at band b' or a payload difficulty requirement at band b. Therefore, transformation B.16 translates an EIRP at band b' to payload difficulty at band b' and vice versa. Importantly, observe that the transformations for the SA and MA payloads are not equal (B.9 vs. B.16) unless $\gamma = 2$. That being said, the interpretation of payload difficulty and its usefulness from the perspective of a systems engineer remains the same.

Let us now consider the total mass of a MA payload. Once again, we assume that it can be computed as the sum of three components: Antenna, high power amplifiers and baseband electronics. Using equation B.18, we obtain

$$\begin{aligned} mass_{MA} =& mass_{PHA} + mass_{HPA} + mass_{other} = \\ =& \frac{PD_b}{G_{elem}P_t}(k_1 + k_2)\lambda_b^\gamma + f(P_t) + k_3, \end{aligned} \tag{B.19}$$

where

- $mass_{PHA}$ is the total phased array mass.
- λ_b is the wavelength of the selected baseline frequency.
- k_1 and k_2 are defined as in Equation B.15.
- $f(\cdot)$ is a function, typically linear or quasilinear [216], that transforms RF power into payload mass.
- k_3 is a constant that captures the mass of other baseband electronics and is assumed to be independent for the EIRP requirement.

B.4.1 Optimal Mass for Multiple Access Payloads

Given a payload difficulty PD_b, then the optimal mass of the multiple access payload can be computed through simple derivation:

$$\frac{\partial mass}{\partial P_t} = -\frac{PD_b}{G_{elem} P_t^2} (k_1 + k_2) \lambda_b^{\gamma} + f'(P_t) = 0, \tag{B.20}$$

and consequently

$$P_t^* = \sqrt{\frac{PD_b}{G_{elem} f'(P_t)} (k_1 + k_2) \lambda_b^{\gamma}}, \tag{B.21}$$

with G_{elem} equal to the gain of one array element in linear scale. Note that both equations have been derived assuming perfect arraying, i.e. $G_{array} = N \cdot G_{elem}$.

B.5 Single Access vs. Multiple Access

In some instances it might be desirable to compare the performance of SA payloads to MA payloads. For instance, what is the extra mass penalty if you deliver a certain EIRP at band B using a parabolic dish that is highly directional vs. a phased array with multiple non-directional antennas? Payload difficulty can, once again, be a useful construct for this purpose, even if the SA and MA payloads operate at different frequency bands. However, since transformations B.9 and B.16 are not exactly the same, payload difficulty from a SA and MA payloads cannot be directly compared.

Fortunately, deriving a new transformation across payload difficulties is straightforward. In particular,

$$\begin{aligned} PD_b^{SA} =& EIRP_{b'}\left(\frac{f_b}{f_{b'}}\right)^2 \\ PD_b^{MA} =& EIRP_{b'}\left(\frac{f_b}{f_{b'}}\right)^\gamma , \end{aligned} \tag{B.22}$$

and consequently

$$\frac{PD_b^{SA}}{PD_b^{MA}} = \left(\frac{f_b}{f_{b'}}\right)^{2-\gamma} . \tag{B.23}$$

Note that now payload difficulty depends on the link frequency band b', as well as the baseline frequency band b. However, since both parameters are known a priori Equation B.23 can be successfully used to translate requirements between these two different payload types.

THIS PAGE INTENTIONALLY LEFT BLANK

C Approximation Methods for Estimating the Availability of Space-to-Ground Networks

In this appendix I present the details of the new method utilized to estimate the availability of space-to-ground optical networks. The explanation herein presented, as well as the results used to back it up, are extracted from Reference [242] by the same author.

C.1 Introduction

Optical communications are widely accepted as the dominant technology to build terrestrial high-speed networks. In the space domain, it is also widely accepted that optical communications can drastically increase the achievable data rates when transmitting information to and from remote scientific probes that orbit the Earth or other planetary bodies of the solar system [243].

Nevertheless, several challenges have delayed the deployment of optical space communication networks during the last decades. Among them is the sensitivity of these networks to atmospheric conditions. Indeed, cloud coverage at the network ground sites can easily disrupt all communications between the remote spacecraft and its mission operations center. To address this issue, site diversity has been proposed as a possible mitigation strategy: At any point in time, the spacecraft is in visibility with multiple ground stations and can therefore choose which one to communicate to. This redundancy progressively decreases the probability that all ground stations are clouded at the same time, thus improving the space-to-ground network availability.

C.1.1 Past Approaches to Estimating Optical Network Availability

Multiple references have addressed the issue of optical ground network availability, or the probability that at least one link between the spacecraft and a ground station will be available at any point in time. Estimation of this metric, also referred to as cloud-free line of sight (CLOS) or the complement of the link outage probability (LOP), has been typically achieved using two complimentary approaches, experimental and analytic.

The experimental approach provides tools that predict the network availability by simulating the network performance over a given time period for which historical cloud observations are available. For instance, References [244] and [245] use the Lasercom Network Optimization Tool (LNOT) to determine optimal optical ground network architectures to support a deep space probe. To that end, LNOT uses raw visible and infrared radiance atmosphere measurements as inputs to estimate the cloud probability. The same tool is utilized in Reference [246] to obtain the data return probability distribution for spacecraft operating at low Earth orbit, geosynchronous orbit and at moon distances. Finally, other references propose simulation tools similar to LNOT and apply them to their specific problems (e.g. [228], [229], [230], [231]).

On the other hand, the analytic approach studies the optical network availability by modeling the space-to-ground link as a random variable with a given probability distribution. References [247] and [231] propose a similar generic formulation and conceptually demonstrate its applicability in a wide variety of hypothetical scenarios: Space-to-ground links from a geosynchronous satellite with and without correlated ground stations, link availability to and from high altitude platforms, and space-to-ground data throughput based on a continuous atmospheric attenuation model. Similarly, communication outage in free-space optical communications due to atmospheric effects (not only clouds but also turbulence, fog or rain) has also been studied in the context of serial and parallel relay systems assuming typical statistical atmospheric models [248], [249], [250], [251]. In most cases, these references exploit the channel statistical formulation to simplify the link outage probability expressions, albeit they are typically only applicable to a limited set of optical technologies and network configurations.

Next, we provide a summary of the limitations identified in both the experimental and analytic approaches for computing an optical ground network availability. For the former, the proposed tools typically utilize minute-by-minute simulations that are computationally expensive and therefore have limited performance when conducting broad architectural optimization studies. Furthermore, they typically rely on low level data products from Earth observation missions (e.g. GOES satellites [252] or the EUMETSAT system [253]), which require both a deep understanding of cloud modeling and processing vasts amounts of information. Finally, tools such as LNOT have been developed by commercial entities and are therefore not readily available for the academic and research community.

In contrast, analytic approaches provide mathematical formulations that allow the user to directly obtain the estimates for the network availability without the need for expensive simulations. This is clearly desirable for the purposes of broad architectural and optimization studies. Nevertheless, they do not provide any recommendations on how to obtain the parameters that define the probability distributions that are proposed, nor do they validate them against other literature references. Finally, they typically rely on unrealistic simplifications (e.g. equal cloud probability across all ground stations) that render them inadequate for real architecting studies.

C.2 Network Modeling

C.2.1 Network Availability for Optical Ground Networks

Let us consider a space-to-ground optical network composed of N ground stations. At any point in time, a spacecraft locks its downlink laser to one of them and starts transmitting data stored in its on-board memory system. If the link is interrupted due to cloud coverage, then the spacecraft automatically and instantaneously locks onto another ground station and continues the download process. Following the notation from Reference [247], we define the downlink outage probability (LOP) as the probability of not having any link available at any point in time. In turn, we define the optical ground network availability (ONA) as the probability of having at least one space-to-ground link available.

Once again following Reference [247], we simplify the problem by assuming an ON/OFF channel in which p_i denotes the probability of having a cloud between the spacecraft and the ground telescope. This channel is mathematically modeled as a random variable $\mathcal{X}_i$ distributed according to a Bernoulli distribution

$$\mathcal{X}_i \sim B\left(p_i\right) = \begin{cases} 1 & \text{w.p. } p_i \\ 0 & \text{w.p. } 1 - p_i \end{cases} \quad \forall i \in [1, N] \tag{C.1}$$

Furthermore, let us also define $\mathcal{X}$ as the sum of all $\mathcal{X}_i$ in the system:

$$\mathcal{X} = \sum_{i=1}^{N} \mathcal{X}_i \tag{C.2}$$

Then, the state of the network at any point in time can be characterized by estimating $f_{\mathcal{X}}\left(X\right)$, that is, the probability of having a given number of space-to-ground links fail due to cloud coverage. Note that, with this formulation, the link outage probability and network availability can be simply estimated as

$$\text{LOP} = f_{\mathcal{X}}\left(X = N\right) = \mathcal{P}\left(\mathcal{X} = N\right) \tag{C.3}$$

$$\text{ONA} = 1 - \text{LOP} \tag{C.4}$$

As Section C.1.1 indicated, estimating $f_{\mathcal{X}}\left(X\right)$ analytically has traditionally been accomplished by imposing a set of restrictive and unrealistic constraints to the problem. A paradigmatic example would be independent ground stations with equal cloud probabilities: $p_i = p \; \forall i \in [1, N]$. In this case, $\mathcal{X}$ becomes a binomially distributed random variable, were the probability of k successes (or clouded links) over N experiments (ground stations) is

$$\mathcal{P}(\mathcal{X} = k) = \binom{N}{k} p^k (1-p)^{N-k} \tag{C.5}$$

Unfortunately, a similar equation when $p_i \neq p_j$, $i \neq j$ cannot be in general expressed in a short convenient form, especially in the case where ground stations are correlated to one another. In order to circumvent this limitation, we propose three alternative approximation methods to characterize $f_{\mathcal{X}}(X)$: Monte Carlo Sampling (MCS); the Lyapunov Central Limit Theorem (CLT) [254]; and the Chernoff Bound [255]. Next, we provide a succinct introduction to the three proposed alternatives with emphasis on the intuition behind these mathematical constructs and their practical implementation rather than providing a fully formal description.

Monte Carlo Sampling

Monte Carlo methods have been used in the past for a wide variety of purposes (sampling, estimation, optimization) and applications (e.g. operations research, finance and economics, computation statistics) [256]. They are currently widely popular due to their flexibility and efficiency when modeling complex random-driven scenarios where analytic results cannot be expressed in convenient closed form solutions.

The MCS strategy constructs $f_{\mathcal{X}}(X)$ through a two step process. First, all $\mathcal{X}_i$ are sampled directly from their respective Bernoulli distributions $B(p_i)$ using available functions from typical scientific software (e.g. Matlab [257] or Python's Numpy [258]). Then, samples for $\mathcal{X}$ are estimated using Equation C.2 and the probability mass function is computed as the relative frequency with which $X = k$, $k \in [1, N]$ occurs. Furthermore, confidence intervals (CI) for $f_{\mathcal{X}}(X)$ can be also easily constructed through either the bootstrapping method or by repeating the MCS experiment multiple times. For instance, consider drawing samples for $\mathcal{X}_i$ repeatedly S times, calculate $X^{(s)}$ using the relative frequency method, and then estimate $f_{\mathcal{X}}\left(X^{(s)}\right)$. Finally, using the S available samples for $f_{\mathcal{X}}\left(X^{(s)}\right)$, compute the standard deviation and CIs for the probability density function.

The Lyapunov Central Limit Theorem

The Lyapunov CLT is an extension to the classical CLT in which the random variables $\mathcal{X}_i$ are assumed to be independent but not identically distributed. In our case, $\mathcal{X}_i \sim B(p_i)$ with $\mu_i = \mathrm{E}[\mathcal{X}_i] = p_i$ and $\sigma_i^2 = \mathrm{Var}[\mathcal{X}_i] = p_i(1-p_i)$, where p_i is the probability of cloud coverage at the i-th ground station. Under certain conditions, the Lyapunov CLT states that

$$\frac{1}{s}\sum_{i=1}^{N}\left(\mathcal{X}_i-\mu_i\right)\xrightarrow{d}\mathcal{N}\left(0,1\right) \tag{C.6}$$

or equivalently $\mathcal{X}\xrightarrow{d}\mathcal{N}\left(\mu,s^2\right)$ with $\mu=\sum_{i=1}^{N}\mu_i=\sum_{i=1}^{N}p_i$ and $s^2=\sum_{i=1}^{N}\sigma_i^2=\sum_{i=1}^{N}p_i\left(1-p_i\right)$. In other words, $\mathcal{X}$ converges in distribution to a normal random variable with parameters μ, s and we can therefore estimate the network's characteristic probability distribution $f_{\mathcal{X}}\left(X\right)$ using the normal cumulative distribution function $\Phi\left(x\right)$ and the De Moivre-Laplace approximation [259]:

$$\begin{aligned}\mathcal{P}\left(\mathcal{X}=x\right)\approx&\mathcal{P}\left(x-0.5<\mathcal{X}<x+0.5\right)\\ \approx&\Phi\left(x+0.5\right)-\Phi\left(x-0.5\right)\end{aligned} \tag{C.7}$$

Note that the Lyapunov CLT holds as long as the Lyapunov condition is satisfied, which, as noted in Reference [260], intuitively states that all $\mathcal{X}_i$ should have a similar variance. In that sense, our ON/OFF model assumes that all $\mathcal{X}_i$ follow a Bernoulli distribution with $\sigma_i^2=p_i\left(1-p_i\right), p_i\in\left[0,1\right]$. Therefore, the maximum variance difference between $\mathcal{X}_i$ and $\mathcal{X}_j$, $i\neq j$ is 0.25 and occurs for $p_i=0.5$ and $p_j=0$ respectively. Therefore, the Lyapunov condition is always satisfied.

The Chernoff Bound

The Chernoff Bound can be used to obtain an upper bound for the tail distribution of a set of independent non-identically distributed random variables. Two versions of the Chernoff Bound exist, the additive and the multiplicative form [261], albeit only the the latter will be considered for this paper. The Multiplicative Chernoff Bound (MCB) states that given N random variables $\mathcal{X}_i$ taking values $\{0,1\}$ and a real value $\delta>1$, then

$$\mathcal{P}\left(\mathcal{X}\geq\delta\mu\right)<\left(\frac{e^{\delta-1}}{\delta^{\delta}}\right)^{\mu} \tag{C.8}$$

Note that in Equation C.8, the parameter μ is an input that can be directly computed once the cloud coverage statistics for each ground station have been characterized. Furthermore, combining Equations C.8 and C.3 results in $\delta=\frac{x}{\mu}$ and we can therefore approximate $f_{\mathcal{X}}\left(X\right)$ as

$$\mathcal{P}\left(\mathcal{X}=x\right)\approx\mathcal{P}\left(\mathcal{X}\geq\delta_-\mu\right)-\mathcal{P}\left(\mathcal{X}\geq\delta_+\mu\right) \tag{C.9}$$

with $\delta_-=\frac{x-0.5}{\mu}$ and $\delta_+=\frac{x+0.5}{\mu}$. This allows us to obtain an upper bound on the link outage probability and consequently a lower bound on the optical network availability.

Finally, we consider the effect of constraints in δ and their interpretation. Specifically, for the MCB to hold it is assumed that $\delta > 1$ which, in turn, indicates that $\mu < x$. Since $\mu < N$ is always true unless all ground station are clouded 100% of the time, it is clear that the Chernoff bound can be used to estimate the LOP. Further, we know that this bound is only valid for the tail end of $f_{\mathcal{X}}(X)$. Inequality $\mu < x$ sets the extend of this tail so that networks with low average cloud probability can apply the bound for a wider range of x values.

C.2.2 Spatially Correlated Ground Stations

Up until this point, we have assumed that all ground stations are independent. This has allowed us to formulate the problem using independent but not identically distributed Bernoulli random variables and then estimate the mass probability function of their sum using three alternative methods. However, in a realistic problem ground stations can be situated at arbitrary locations, thus violating the independence assumption.

Unfortunately, if that is the case no analytic formulation can be used and, consequently, the only alternative is to use MCS approximations to obtain $f_{\mathcal{X}}(X)$. Sampling for a correlated binomial distribution has already been studied in the literature and efficient methods have been proposed. For this paper, we follow the procedure described in Reference [262]. In particular, assume we want to obtain samples from N binomial random variables with mean p_i, $i \in [1, N]$ and covariance Σ_{ij}, $i, j \in [1, N]$. Then, we proceed as follows:

1. Draw samples $y^{(s)}$, $s \in [1, S]$ from a multivariate normal distribution $\mathcal{N}(\gamma, \Lambda)$, also known as latent distribution.

2. Generate the binomial distribution by truncating the normal samples:

$$x^{(s)} = \begin{cases} 1 & \text{iff } y^{(s)} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{C.10}$$

The moments of the desired binomial distribution p, Σ and the moments of latent distribution γ, Λ are related as follows:

$$p_i = \Phi(\gamma_i) \tag{C.11}$$

$$\Sigma_{ij} = \Phi_2(\gamma_i, \gamma_j; \Lambda_{ij}) - \Phi(\gamma_i)\Phi(\gamma_j) \tag{C.12}$$

$$\Lambda_{ii} = 1, \forall i \tag{C.13}$$

where $\Phi_2(x, y; \lambda)$ denotes to the cumulative probability distribution of a bivariate normal with correlation λ evaluated at (x, y). Three important points must be clarified when considering this procedure: First, the variance of the latent distribution is fixed for all ground stations (see Equation C.13). This is due to the fact that binomial distributions have mean and variance specified by a single parameter p, while normal distributions are specified through two parameters. By letting $\Lambda_{ii} = 1 \; \forall i$, we eliminate the extra degree of freedom. Second, the correlation between two binomial random variables is necessarily constrained in order to avoid violating the axioms of probability theory [262]. These constrains include

$$\Sigma_{ij} \leq \min \{p_i (1 - p_j), p_j (1 - p_i)\} \tag{C.14}$$

$$\Sigma_{ij} \geq - p_i p_j \tag{C.15}$$

$$\Sigma_{ij} \geq - (1 - p_i)(1 - p_j) \tag{C.16}$$

which, if violated, result in Λ not being semi-definite positive, a necessary condition for it to be a valid covariance matrix for the latent distribution. Finally, we note that the computational performance of the proposed methodology scales poorly for large values of N. Indeed, since Equation C.12 does not have a closed form solution, all Λ_{ij} have to be computed numerically and consequently the number of solver calls increases as $\mathcal{O}(N^2)$.

In order to mitigate this problem, we propose to approximate the bivariate normal distribution $\Phi_2(x, y; \lambda)$ present in Equation C.12. Following the notation from Reference [263], we define

$$L(x, y; \lambda) = \mathcal{P}(\mathcal{X} > x, \mathcal{Y} > y) \tag{C.17}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are two arbitrary random variables jointly distributed following a bivariate normal distribution of zero mean, unitary variance and correlation λ. Without loss of generality, we assume $x \geq y$ and realize that, for the bivariate normal,

$$\Phi_2(x, y; \lambda) = \Phi(x) + \Phi(y) + L(x, y; \lambda) - 1 \tag{C.18}$$

Therefore, we can efficiently compute $\Phi_2(x, y; \lambda)$ as long as we can approximate $L(x, y; \lambda)$. Again following Reference [263], we note that

$$L(x, y; \lambda) \approx \Phi(-x) \Phi\left(\frac{\lambda \mu(x) - y}{\sqrt{1 - \lambda^2}}\right) \tag{C.19}$$

which allows us to obtain a closed form solution for Equation C.12 that obtains the latent distribution covariance matrix as a function of the original binomial mean and correlation:

$$\Lambda_{ij} = \frac{c_1 c_2 \pm \sqrt{c_1^2 - c_2^2 + 1}}{1 + c_1^2}, \forall i, j, i \neq j \tag{C.20}$$

$$c_1 = \frac{\phi(\gamma_i)}{\xi \Phi(-\gamma_i)} \tag{C.21}$$

$$c_2 = \frac{\gamma_j}{\xi} \tag{C.22}$$

$$\xi = \Phi^{-1}\left(\frac{1 - \Sigma_{ij} + p_i p_j - p_i - p_j}{1 - p_i}\right) \tag{C.23}$$

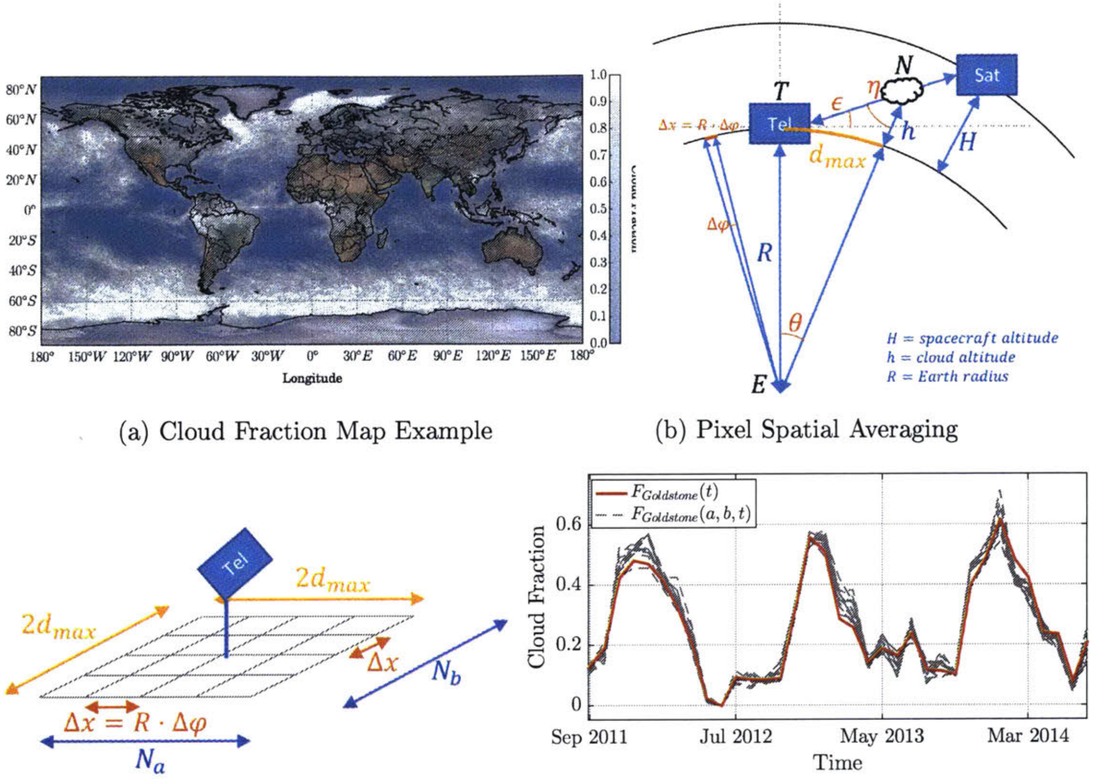
In Equation C.21, $\phi(x)$ denotes the standard normal probability density function. Note that the sign ambiguity in Equation C.20 is resolved by ensuring that the correlation coefficient is appropriately bounded $0 \leq \Lambda_{ij} \leq 1$. Note also that better approximations for the bivariate normal distribution have been proposed in the literature. For instance, Reference [264] states that the approximation used in Equation C.19 results in large errors when $\lambda \geq 70\%$ and proposes an alternative solution. However, its formulation is too complex to yield an efficient closed form solution for Λ_{ij}, thus indicating that a numerical solver over Equation C.12 would be required in that case. Evidently, this defeats the purpose of using an approximation in the first place.

C.3 Cloud Modeling

Up until this point, a framework for approximating the availability of an optical ground network has been introduced. The proposed approach is based on an ON/OFF channel that is characterized by the probability p_i of having a cloud disrupting the space-to-ground link.

In order to estimate p_i for any ground station across the world, we take advantage of the preprocessed Cloud Fraction data product from NASA's Terra and Aqua satellites [265]. Similar data products are available also from other providers such as the GOES satellites, the EUMETSAT system, the Himawari and Fengyun satellites, the Calipso Satellite and the International Satellite Cloud Climatology Project. As explained in References [266] and [267], the cloud fraction is a level 2 data set that indicates the probability of having a cloud at a given latitude and longitude (see Figure C-1a). It is available since the year 2000 on a daily, weekly and monthly basis, and is provided as a set of timestamped and geolocated images, each one with $N_x \times N_y$ pixels. In that sense, each pixel defines a region of Earth with constant cloud probability p_i, defined with a single numerical value in the $[0, 1]$ range. The extent of this region is directly related to the dataset angular resolution $\Delta\varphi$, which we select to be 0.1 deg for the rest of the paper. This results in a pixel size equal to $\Delta x \approx \Delta\varphi \cdot R = 11.12$km (see Figure C-1b).

Let $\mathcal{F}_i\left(a, b, t\right), i \in [1, N], a \in [1, N_x], b \in [1, N_y]$ denote the random variable that models



(a) Cloud Fraction Map Example

(b) Pixel Spatial Averaging

(c) Pixel Spatial Averaging

(d) Monthly Cloud Fraction Spatial Averaging

Figure C-1: Cloud Fraction Figures

the cloud fraction value for a pixel located at ground coordinates (a, b) at time t. Then, we define the steady state cloud probability for the i-th ground station as

$$p_i = \mathrm{E}_{a,b}^{\Delta}\left\{\mathcal{F}_i\left(a, b, t\right)\right\} \tag{C.24}$$

where the expectation is taken over the spatial dimensions and we implicitly assume that $\mathcal{F}_i\left(a, b, t\right)$ is stationary during Δ units of time. When computing this expectation, four main factors have to be taken into account:

- The telescope-spacecraft pointing profiles that define which clouds can impair the space-to-ground link as the latter is tracked by the former.

- The possible parallax error in the cloud fraction data set caused by the swath with which Earth observation instruments take measurements of the atmosphere.

- The seasonality of the cloud fraction time series.

- The spatial and temporal correlations of the cloud fraction time series between two or more ground stations.

Next, we present a succinct explanation of how these four factors have been taken into account when obtaining estimates for the probability of cloud coverage p_i.

Telescope pointing

Assume that a telescope is located at latitude λ_i and longitude ϕ_i. Assume also that its minimum elevation angle is $\epsilon_{i,min}$. Then, as the telescope tracks a spacecraft the optical beam will traverse not only the clouds located exactly at coordinates (λ_i, ϕ_i), but also in the surrounding vicinity. To estimate the extent of this vicinity, we define the telescope central angle as the angle between the telescope and a cloud, measured from the center of Earth, when it is pointing at an elevation angle ϵ (see Figure C-1b). Using triangle $\widehat{ETN}$, we first compute the central angle as

$$\theta = \frac{\pi}{2} - \epsilon_{min} - \arcsin\left(\frac{R}{R+h}\cos\epsilon_{min}\right) \tag{C.25}$$

with R being the mean Earth radius (6371km), h being the maximum cloud altitude (10-12km approximately) and ϵ_{min} being the telescope's minimum elevation angle (10-20deg typically). Next, we use θ to calculate the maximum distance, on the ground, where a cloud impeding the space-to-ground link would be located assuming no parallax error: $d_{max} = \theta \cdot R$. Therefore, all pixels with center at a distance less or equal than d_{max} should be considered when constructing p_i for a given ground station. Now, let N_a and N_b denote the number of pixels in the vicinity of the site that satisfy the distance condition $d \leq \theta \cdot R$ (see Figure C-1c). Then, in general we define the expected cloud probability at time t as

$$\mathcal{F}_i\left(t\right) = \mathrm{E}_{a,b}\left\{\mathcal{F}_i\left(a,b,t\right)\right\} \approx \sum_{a=1}^{N_a}\sum_{b=1}^{N_b}\mathcal{F}_i\left(a,b,t\right)\mathcal{P}\left(a,b\right) \tag{C.26}$$

where $\mathcal{P}\left(a,b\right)$ represents the probability of the antenna pointing to a certain direction in space and depends on the type of spacecraft under consideration. Since these profiles are unlikely to be available during the first stages of the optical network design process, for the rest of the paper we assume equal probabilities over all $N_a \times N_b$ pixels. For instance, Figure C-1d presents the results of the described spatial averaging process for a geostationary satellite supported from Goldstone with a minimum elevation angle of 20 deg. Since the exact position of the satellite is not known, we estimate that 13 pixels need to be averaged. They are represented as dotted lines in Figure C-1d along with a solid line that represents their average.

Parallax Error

The parallax error is incurred when space-based instruments take measurements in a direction different from the spacecraft's nadir. To assess the magnitude of the parallax error, we take advantage of the model proposed by Wang [268] for the MODIS instrument, which incidentally is the same instrument used to construct the the Cloud Fraction dataset. In particular, using Equation C.27, Terra's altitude (H = 710km [269]), MODIS' swath (SW = 2330km [269]), and maximum cloud height (h = 12km), we estimate a maximum parallax error of 7.4km.

$$r = \frac{hH\tan\varphi}{H-h} = \frac{hH^2}{SW\,(H-h)} \tag{C.27}$$

Importantly, we observe that this parallax error results in a location error smaller than the pixel size of 11km approximately. This indicates that it will be a second order factor that is already included in the averaging process described in Section C.3.

Cloud Fraction Seasonality

Once $\mathcal{F}_i\,(t)$ has been obtained for all ground stations under consideration, the next step is to compute the cloud probability as

$$\hat{p}_i\,(t, t+\Delta) = \mathrm{E}^{\Delta}\left\{\mathcal{F}_i\,(t)\right\} \approx \frac{1}{\Delta}\sum_{t}^{t+\Delta} F_i\,(t) \tag{C.28}$$

where we have assumed that the sample mean is used as estimator for the temporal expectation and Δ denotes its bandwidth. Since $\mathcal{F}_i\,(t)$ is, in general, non-stationary over long time horizons, we need to ensure that Δ is selected so that it has optimal performance.

To illustrate the problem, assume that the average link failure probability at any ground station is non-stationary. Its seasonality can be modeled as

$$p_t = \begin{cases} c_1 & 0 \leq t \leq t_1 \\ \dots & \\ c_K & t_{K-1} < t \leq t_K \end{cases} \tag{C.29}$$

where c_k $\forall k \in [1, K]$ are constant levels uniformly distributed between 0 and 1. Assume also that the duration of a fictional season is constant, denoted by $T_s = t_k - t_{k-1}$, and measured in arbitrary units of time (see Figure C-2). If one fictional cloud measurement per unit of time is available, then at most T_s consecutive measurements should be used to estimate the values of c_k. In other words, the estimator bandwidth is bounded by $\Delta \leq T_s$.

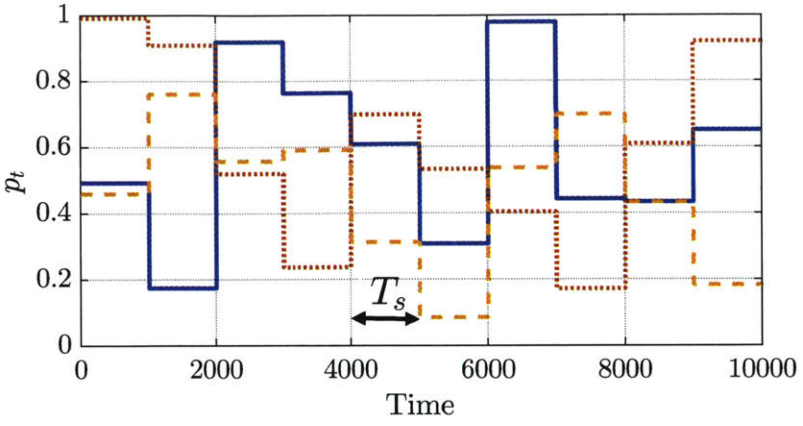


Figure C-2: Three Realizations for p_t

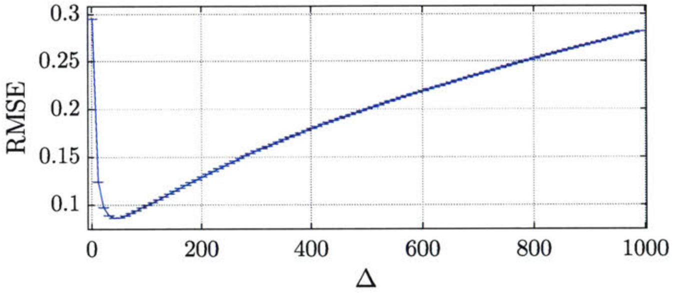


Figure C-3: Effect of Δ and T_s mismatch in p_t Estimation Error

We can now utilize this illustrative model to visualize the effect of a mismatch between T_s and Δ. To that end, we select $K = 10$ seasons and $T_s = 10^3$ units of time, and we generate 100 fictional sample paths for the average link failure probability p_t (Figure C-2 shows three of them). Then, we estimate the values of c_k for each path using Equation C.28, i.e. a sample average over a rolling window of size Δ, with Δ varying from 1 to T_s. Finally, we collect statistics on the c_k's estimator root mean square error (RMSE) and plot them in Figure C-3. We observe that, approximately, a 45 sample bandwidth is optimal. This is due to the fact that the sample estimator is applied over a rolling window rather than over pre-fixed time intervals synchronized with the season changes (since those would not be known for cloud fraction time series). In other words, if $\Delta = T_s$ is selected, then there is a high likelihood that the rolling window uses samples from season c_{k+1} to estimate the value of season c_k.

From a practical standpoint, these findings indicate choosing Δ is non-trivial. Since we do not know the true stationarity structure of the cloud time series, we propose to define Δ during the benchmarking phase (see Section C.4.2). In other words, we first compute the network availability by averaging the cloud fraction time series at different time intervals (e.g. daily, weekly, monthly, yearly), and then we compare the obtained results to those mentioned in the literature. Then, we select Δ such that the validation error is minimized. Note that, as mentioned in Section C.1, the proposed benchmarks for validation purposes are based on hourly or even minute-based simulations that are therefore not subject to errors caused by mismatches in Δ.

Finally, given that $\hat{p}_i\left(t, t+\Delta\right)$ is only valid for a limited span of Δ units of time, it is clear that the optical network availability will now be time-dependent. Therefore, we define $\mathcal{X}_t$ as the random variable that models the state of the network between time t and $t+\Delta$. Similarly, we redefine the overall optical network availability as the probability of having at least one link available with a 95% certainty over all possible states of $\mathcal{X}_t$.

Ground Station Spatial and Temporal Correlation

Lastly, we consider the spatial and temporal correlation between the cloud fraction data set at different ground stations. The former captures the notion that sites located close enough should be subject to the same cloud conditions at any given instant of time. In contrast, the latter captures the effect of yearly seasonality in the cloud fraction which, for instance, causes networks with ground sites in the Northern and Southern Hemisphere to have better availability than those with only sites in continental US [245].

Assessing the magnitude of the temporal correlation can be easily achieved with the cloud fraction data set and is inherently taken into account when estimating the cloud fraction probability $\hat{p}$. For instance, consider two negatively correlated ground stations such as Goldstone, California and La Silla, Chile (see Figure C-4). Then, at $t = 01/01/2013$ we will assess the state of the network $\mathcal{X}_t$ by first estimating $\hat{p}_{Goldstone} \approx 0.15$ and $\hat{p}_{LaSilla} \approx 0.60$ and then using them as inputs for the proposed approximation methods.

In contrast, assessing the spatial correlation cannot be done directly through the cloud fraction data set as it does not indicate what is the probability of having a site clouded provided that another one is too. To circumvent this limitation and yet maintain a simplified approach, we follow the results from Reference [270] and model the cloud spatial correlation as

$$\begin{aligned} \lambda_{ij} &= \exp\frac{-d_{ij}}{d_0},\ i,j \in \left[1, N\right],\ i \neq j \\ d_0 &\in [200, 400] \approx 300km. \end{aligned} \tag{C.30}$$

Therefore, we assume that the spatial correlation is only dependent on the distance between two ground stations and a normalizing factor d_0. Then, we use the results from Section C.2.2 to obtain Bernoulli distributed samples with the appropriate correlation structure.

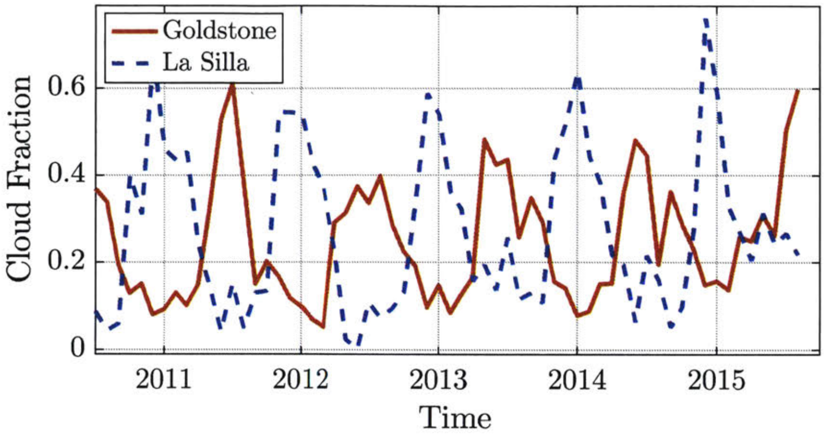


Figure C-4: Negatively Time Correlated Ground Stations

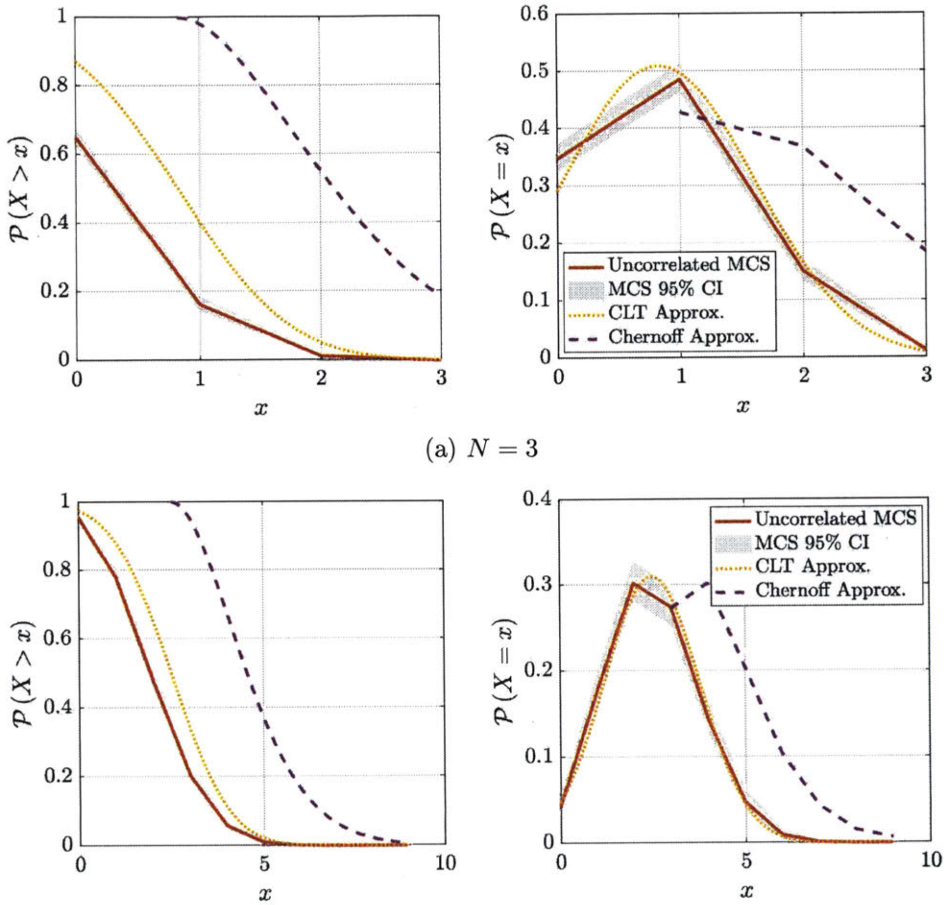
C.4 Empirical Tests and Benchmarks

C.4.1 Numerical Benchmarks

The goal of this section is to explore the performance of the proposed approximation methods for assessing $f_{\mathcal{X}}(X)$ using numerical simulations in both uncorrelated and correlated settings. Importantly, the results of this section are only provided as clarification examples and do not assert the validity of the approximation methods for computing the optical ground network availability. For this discussion, please refer to Sections C.4.2 and C.4.3.

Figure C-5 presents the obtained results for the proposed approximations when considering 3 and 9 independent Bernoulli variables. In the 3 variable case, their means are equal to 0.20, 0.50, 0.13 respectively. For the 9 variable case, the same values are used repeating them three times and maintaining their order. On the other hand, each $f_{\mathcal{X}}(X)$ is obtained through three approaches: MCS with 1000 samples and repetition to obtain the CI, the Lyapunov CLT and the Chernoff bound. It can be observed that, assuming no correlation, the Lyapunov CLT provides a good approximation even for $N = 3$ while the Chernoff bound consistently overestimates the tail probabilities. Therefore, we suspect that the usefulness of the Chernoff bound as a method for estimating the link outage probability is limited even for large networks $(N > 9)$.

Next, we consider the effect of correlated Bernoulli random variables. To that end, we assume three highly correlated random variables $\mathcal{X}_1$, $\mathcal{X}_2$ and $\mathcal{X}_3$ with the following characteristic



(a) $N = 3$

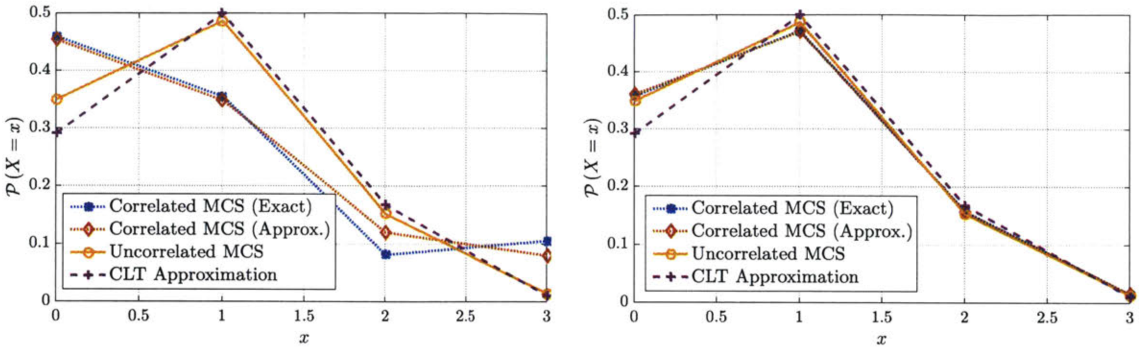
(b) $N = 9$

Figure C-5: CLT and Chernoff Approximations

moments:

$$p = \begin{bmatrix} 0.20 \\ 0.50 \\ 0.13 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & 0.30 & 0.65 \\ 0.30 & 1 & 0.40 \\ 0.65 & 0.40 & 1 \end{bmatrix} \tag{C.31}$$

Figure C-6a plots the obtained $f_{\mathcal{X}}(X)$ using four methods: Uncorrelated MCS (see Section C.2.1), exact and approximated correlated MCS (see Section C.2.2), and the Lyapunov CLT approximation (see Section C.2.1). Note the significant difference between the obtained correlated and uncorrelated distributions, especially in the tail portion of the distribution. While the uncorrelated model estimates $\mathcal{P}(X = 3) = 0.0106$, the correlated one is in fact bimodal with $\mathcal{P}(X = 3) \approx 0.10$. In other words, using the uncorrelated model would result in overestimating the network availability by 9% approximately. Additionally, Figure C-6b plots the same results when $\lambda_{ij} = 0.05, \forall i, j,\ i \neq j$, that is, the correlation effect should be almost unnoticeable. It is clear that in this case the correlated and uncorrelated sam-



(a) Sum of Largely Correlated Bernoulli Variables

(b) Sum of Uncorrelated Bernoulli Variables

Figure C-6: Correlation Effect on Bernoulli Random Variables

pling methods yield similar results, thus indicating that the described correlated Bernoulli sampling procedure can be used under all circumstances.

Finally, we study the computational performance of the five available methods to generate $f_{\mathcal{X}}(X)$, namely uncorrelated MCS, exact correlated MCS, approximated correlated MCS, CLT approximation and Chernoff bound. Figure C-7 summarizes the obtained results when repeating each estimation procedure 1000 times and letting N vary between 1 and 25. It can be observed that, even for $N = 3$, the exact correlated Bernoulli sampling procedure is an order of magnitude more computationally intensive than the approximated correlated MCS procedure and all the non-correlated methods. More importantly, this difference vastly increases as more correlated ground stations are added, with a two order of magnitude difference observed at around 10 sites and almost a three order magnitude difference at 25 sites. In summary, these results confirm our initial intuition: The exact correlated MCS method has limited scalability for large N due to $\mathcal{O}(N^2)$ solver calls. Finally, numerical approximation methods such as the CLT or the Chernoff bound always outperform their MCS counterparts from a computational perspective and, therefore, their use should be prioritized whenever the network architecture allows it.

C.4.2 Literature Benchmarks

Another approach to validating the proposed method is by comparing its results with previously reported optical availability estimates in the literature. To restrict the extent of this study, we select two previously studied architectures, one with correlated ground stations (Architecture 1: Goldstone, California; Kitt Peak, Arizona; the McDonald Observatory, Texas; and Mauna Kea, Hawaii) and another one with uncorrelated ground stations (Architecture 2: Kitt Peak, Arizona; Arequipa, Peru; and La Silla, Chile), and use the proposed approximation methods using different stationarity bandwidths. In that sense, References [244] and [245] report that the ONA for both architectures is 90% and 96 – 98% approxi-

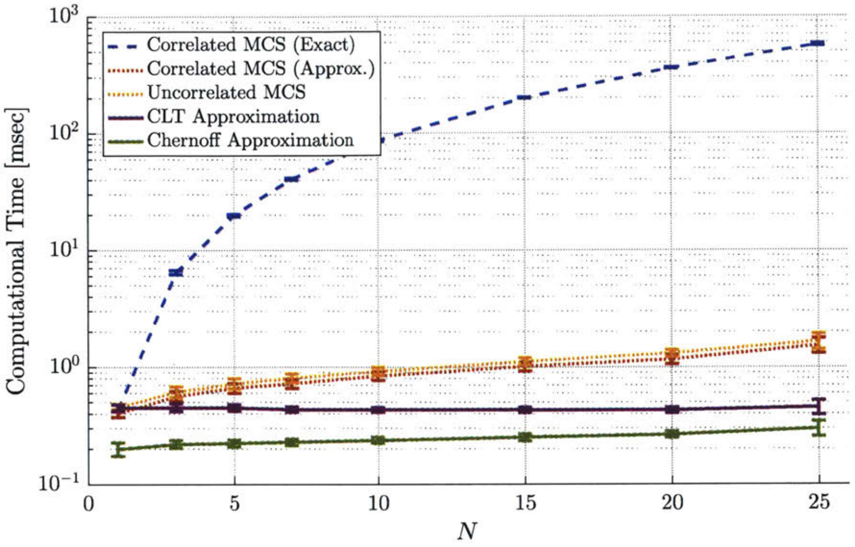


Figure C-7: Computational Burden for ONA Estimation Methods

mately, so these numbers will be considered *truth*.

In order to benchmark these estimates, we collect cloud fraction data for the aforementioned ground sites and their vicinity. The spatial averaging process (see Equation C.26) is then performed assuming a uniform distribution across all pixels visible with a telescope minimum elevation angle of 10 deg and a maximum cloud altitude of 12km. On the other hand, the temporal averaging (see Equation C.28) is performed assuming full (historical), yearly, monthly, weekly and daily stationarity. Finally, the spatial correlation between sites is estimated based on the simplified exponential model presented in Section C.3 and results in significant correlation for the first architecture (12.8% for Goldstone and Kitt Peak, 8.6% for Kitt Peak and McDonald) and negligible correlation for the second one (less than 1% for Arequipa and La Silla).

Table C.1 provides the obtained benchmark results, both in terms of the methods used to obtain $f_{\mathcal{X}}(X)$ and the stationarity of the cloud fraction time series. We note that the uncorrelated MCS method overestimates the availability of the first architecture by almost 2% in the best case, thus reinforcing the importance of not assuming independence when ground sites are closely located. Furthermore, we note that the CLT approximation provides good estimates for the optical network availability for uncorrelated or slightly correlated networks (less than $5-10\%$) of three or more ground sites assuming weekly or longer stationarity. If that is the case, the error incurred by the approximation method as compared to simulation is less than 1% (even 0.5% for monthly stationarity). Finally, the Chernoff bound underestimates the optical network availability by $20-25\%$ consistently and therefore is not recommended under any scenarios.

Table C.1: Optical Ground Network Availability for Benchmark Architectures

ONA[%]	Uncorrelated MCS		Correlated MCS (Exact)		Correlated MCS (Approx.)		CLT Approximation		Chernoff Bound	
	Arch 1	Arch 2	Arch 1	Arch 2	Arch 1	Arch 2	Arch 1	Arch 2	Arch 1	Arch 2
Historical	98.7	98.8	98.2	98.7	98.2	98.7	98.9	99.3	78.9	80.5
Yearly	98.1	97.8	97.3	97.8	97.3	97.7	98.2	98.3	75.6	75.1
Monthly	92.4	97.4	91.0	97.4	91.0	97.4	92.5	97.0	68.0	67.8
Weekly	89.8	96.0	88.1	96.0	88.1	96.0	90.2	95.3	67.8	64.8
Daily	85.3	94.2	85.7	94.7	85.7	94.7	85.6	92.6	67.7	62.7

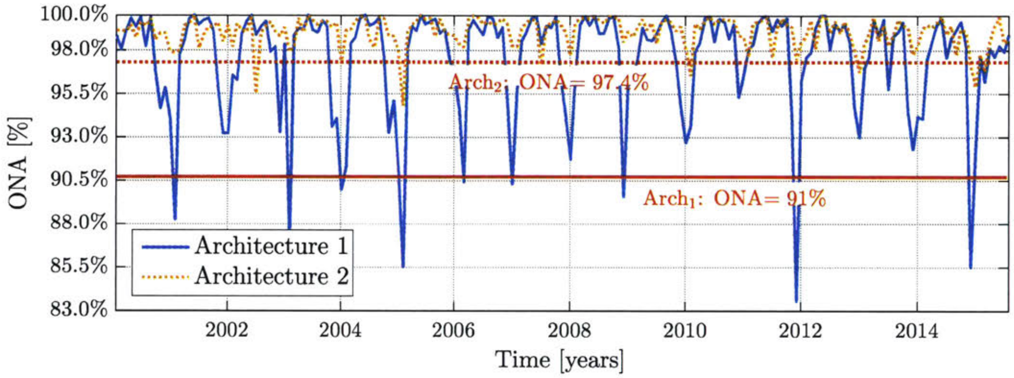


Figure C-8: Time Series ONA for Benchmark Architectures

On the other hand, the issue of choosing the appropriate level of stationarity for the cloud fraction time series is clearly captured by the obtained results. Daily and weekly averages tend to largely underestimate the network availability, thus indicating that not enough samples have been utilized in order to obtain a stable estimate for p_i. In contrast, historical and yearly averages tend to overestimate the network availability because changes in the cloud fraction distribution due to seasonality effects are smoothed out.

Finally, monthly averages result in the best estimates for the network availability, with values very close to those reported by the benchmark references. In particular, using the correlated simulation method with monthly averaged cloud fractions results in ONA $\approx$ 91% for the first architecture, a 1% error with respect to the value reported by previous studies. Similarly, the second architecture obtains ONA $\approx$ 97.4%, also consistent with the 96 to 98% availability from the literature (see Figure C-8).

C.4.3 Simulation Benchmarks

Having identified monthly stationarity as "optimal" for using the proposed approximation methods, we now compare the results obtained against discrete-event simulation. In this case, the Cloud Fraction data set comes from the EUMETSAT system and contains cloud imagery every two hours during four years for Europe, Africa and the Middle East. For comparison purposes, we compute the optical network availability for 2000 different architectures, half of which are non-correlated and the other half are highly correlated. These architectures are generated by choosing randomly from the list of ground stations in Reference [231]. Furthermore, in order to force high correlation, we selecting only ground stations within Germany if necessary.

Figure C-9 presents eight comparative plots for the ONA time series obtained through hourly simulation, MCS approximation and CLT approximation. Each point in the time series indicates, for a given month t, the probability of not having all ground stations in the network clouded at the same time. We can observe that the simulation, MCS approximation and CLT method results are consistent in all uncorrelated architectures (see Figure C-9a), while significant errors are observed in the correlated case for the CLT approximation (see Figure C-9b). Note, however, that the approximated MCS method remains a good approximation in all cases, thus confirming that it can be used in combination with the proposed distance-based correlation model to successfully model cloud cover in a correlated optical ground network.

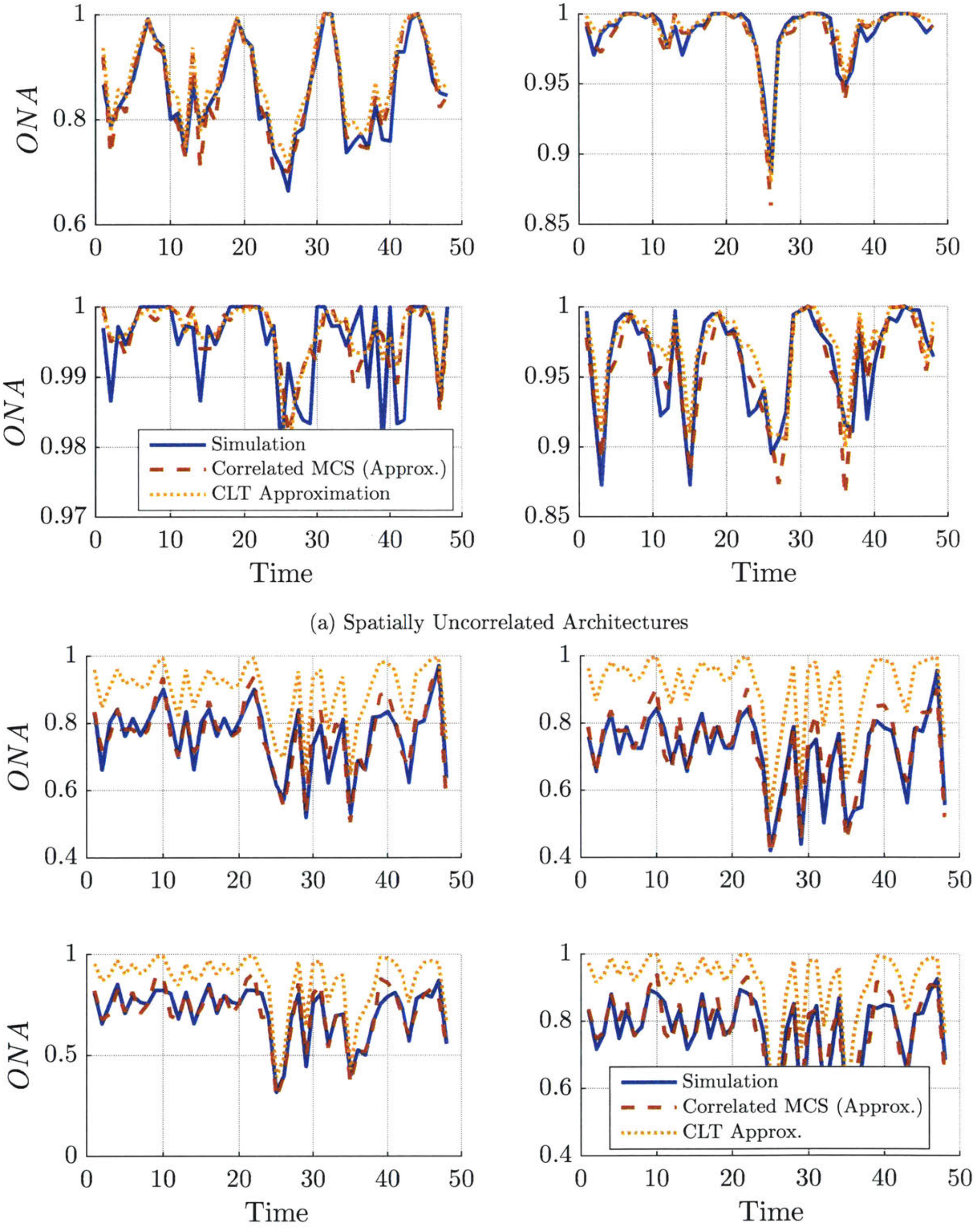
To further emphasize this point, figures C-10a and C-10b plot a histogram of the RMSE between the simulation obtained ONA and the two approximation methods across the 1000 uncorrelated and correlated random architectures respectively. In this case, we observe that the RMSE is confined to the $\approx 2\%$ range for the uncorrelated case for both approximation methods. In other words, the error between the ONA value from the simulation and its approximation is confined to the $\pm 2\%$ range for almost 70% of cases. In contrast, when the same histogram is plotted for the correlated architectures we notice a significant increase of the RMSE for the CLT approximation ($15-20\%$), while the correlated MCS RMSE is limited to 5% on average. Therefore, we conclude that the proposed approximation methods are suitable for high level architectural studies and provide first order estimates of the network availability at significantly lower computational cost than baseline the simulation method used in past literature studies.

C.5 Final Recommendations

This section summarizes the findings from the conducted empirical tests and benchmark exercises, and provides concise recommendations on how to efficiently compute $f_{\mathcal{X}}(X)$ and the optical network availability. In particular:

- To estimate the cloud probability p_i, use the monthly averaged Cloud Fraction data set at the ground station location, as well as its vicinity.

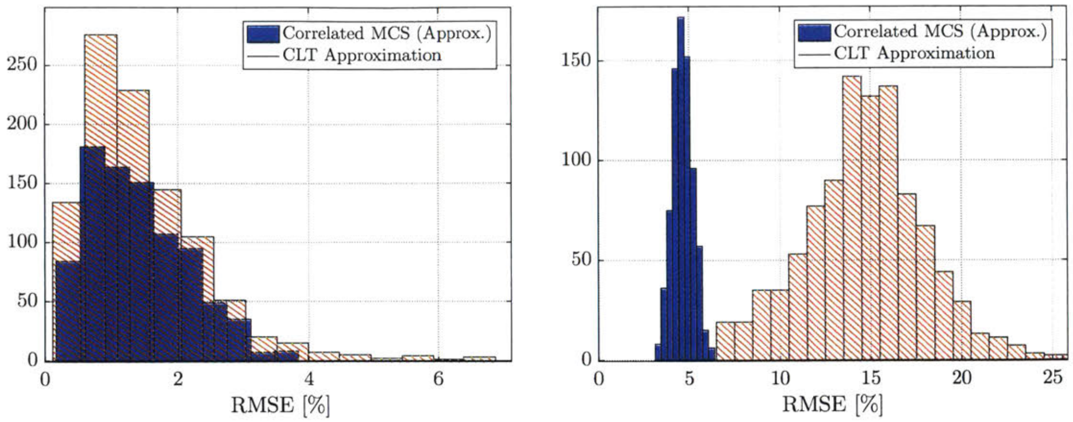
- If the correlation $\lambda_{ij} < 5 - 10\%$ for all $i, j \in N$, $i \neq j$ and $N < 3 - 4$, then obtain $f_{\mathcal{X}}(X)$ using the uncorrelated Monte Carlo sampling method.



(a) Spatially Uncorrelated Architectures

(b) Spatially Correlated Architectures

Figure C-9: Simulated vs. Approximated Cloud Fraction Time Series



(a) Spatially Uncorrelated architectures

(b) Spatially Correlated architectures

Figure C-10: Network Availability RMSE

- If the correlation $\lambda_{ij} < 5 - 10\%$ for all $i, j \in N$, $i \neq j$ and $N \geq 3 - 4$, then obtain $f_{\mathcal{X}}(X)$ using the CLT approximation.
- If $5 - 10\% \leq \lambda_{ij} < 100\%$ for any $i, j \in N$, $i \neq j$, then obtain $f_{\mathcal{X}}(X)$ through the approximated correlated MCS.

For all MCS methods, results reported in Section C.4 were generated using 1000 samples per ONA estimate, which was found to be a good trade-off between sampling error and computational performance.

Finally, throughout this paper no reference to a specific space mission was assumed. That being said, some considerations that should be taken into account when applying the proposed methods: First, for low Earth orbit spacecraft and deep space probes, line-of-sight between the spacecraft and the N ground stations available has to be pre-computed. In other words, at any point in time, the methods have to be applied using only the subset of sites in direct visibility with the spacecraft. This is obviously not necessary for geostationary satellite. On the other hand, for airborne platforms, other approaches such as the ones described in Reference [247] are more adequate and should, therefore, be favored. Lastly, the proposed approximation methods are well suited for high-level architectural studies when a large number of candidate sites are under consideration (e.g. 20-30). Ideally, they can be used to reduce the space of candidates sites to $5 - 7$, after which full simulation should be applied to avoid the 2 to 5% error induced by the approximation methods. Note, however, that the reduction in computational burden is significant. Given, for instance, 30 initial candidate sites, more than 1 trillion network architectures are possible. Yet, after the down-selection process with the approximation methods only 128 options remain for detailed simulation and assessment.

THIS PAGE INTENTIONALLY LEFT BLANK

D Interview List

Table D.1: Interviews for Chapter 1

Mission	Affiliation	Role	Spacecraft Type	Reference
CAS	JPL	Lead Scientist	Touring	[I1]
CAS	JPL	Operations Manager	Touring	[I2]
MRN	JPL	Chief Architect	Mars orbiter/rover	[I3]
Spitzer	JPL	Program Manager	Earth Trailing	[I4]
Voyager	JPL	Program Manager	Flyby	[I5]
MMS	GSFC	Mission Director	Earth Trailing	[I6]
NIMO	GSFC	Office Chief	Ground System	[I7]

Table D.2: Interviews for Chapter 4

Affiliation	Role	Interview Focus	Reference
NCEP	Assimilation System Specialist	Utility function elicitation	[I8]
Meteo-France	Ingenieur de Recherche	Utility function elicitation	[I9]
MIT	Research Scientist	Utility function elicitation	[I10]
ECMWF	Research Scientist	Utility function elicitation	[I11]
SN	DSP expert	Latency characterization	[I12]
SN	MA and beamforming expert	Latency characterization	[I13]
NEN	CFDP expert	Latency characterization	[I14]
CAS	Mission operator	Latency characterization	[I15]
MMS	Mission operator	Latency characterization	[I16]

Table D.3: Interviews for Chapter 5

Affiliation	Role	Interview Focus	Reference
JSC	Researcher in geological sciences	Utility function elicitation	[I17]
JSC	Planetary scientist	Utility function elicitation	[I18]
JSC	Planetary scientist	Utility function elicitation	[I19]
GSFC	Researcher in geological sciences	Utility function elicitation	[I20]
DSN	Senior Engineer	Utility function elicitation	[I21]
MRN	Chief Architect	Latency characterization	[I22]

THIS PAGE INTENTIONALLY LEFT BLANK

BIBLIOGRAPHY

[1] Molly E. Brown, Mark L. Carroll, and Vanessa M. Escobar. Study on data latency needs and requirements. Technical report, NASA, November 2013.

[2] Deep space communications and navigation center of excellence. `https://descanso.jpl.nasa.gov/performmetrics/stairstep.pdf`. Accessed: 04/28/2017.

[3] SWIFT Science Center. The SWIFT technical handbook version 12.0. Technical report, NASA, 2015.

[4] Ilott P. Makovsky A. and Taylor J. Mars science laboratory telecommunications system design. Technical report, NASA-JPL, November 2009.

[5] Europa Study Team. Europa study 2012 report - europa multiple flyby mission. Technical report, NASA-JPL, May 2012.

[6] Ravi Kumar and Sergei Vassilvitskii. Generalized distances between rankings. In *Proceedings of the 19th international conference on World wide web*, pages 571–580. ACM, 2010.

[7] Carla Cardinali and Sean Healy. Impact of gps radio occultation measurements in the ecmwf system using adjoint-based diagnostics. *Quarterly Journal of the Royal Meteorological Society*, 140(684):2315–2320, 2014.

[8] European organisation for the exploitation of meteorological satellite. https://www.nesdis.noaa.gov/content/our-satellites#flyout. Accessed: 03/20/2017.

[9] J. Schmetz and K. Holmlund. EUMETSAT's current and future satellite programmes. Online Presentation, November 2015.

[10] Tamotsu Igarashi. Jaxa's earth observation: Gcom, gpm, earthcare, gosat. Online Presentation, March 2010.

[11] Mars exploration. `http://mars.nasa.gov/programmissions/overview/`. Accessed: 10/24/2016.

[12] Stephen J Hoffman. The mars surface reference mission: a description of human and robotic surface activities. Technical report, NASA, Johnson Space Center, 2001.

[13] Dsn schedule explorer. `http://www.mit.edu/~portillo/dsn/index.html`. Accessed: 10/25/2016.

[14] Dan Lester and Harley Thronson. Human space exploration and human spaceflight: Latency and the cognitive scale of the universe. *Space Policy*, 27(2):89–93, 2011.

[15] NASA. `https://swift.gsfc.nasa.gov/about_swift/mission_flow/dataprod_timeline.html`. Online. Accessed: 03/20/2017.

[16] Office of the Inspector General. Review of NASA's tracking and data relay satellite system. Technical report, NASA, 2011.

[17] Space Communications and Navigation Program. Space communications and navigation program (SCaN) architecture definition document (ADD). volume 1: Executive summary. Technical Report 4, National Aeronautics and Astronautics Administration, NASA Headquarters. Washington, D.C., April 2014.

[18] Morgan Dwyer. *The Cost of Jointness: Insights from Environmental Monitoring Systems in Low Earth Orbit.* PhD thesis, Massachusetts Institute of Technology, September 2014.

[19] Dimitri P Bertsekas, Robert G Gallager, and Pierre Humblet. *Data networks*, volume 2. Prentice-Hall International New Jersey, 1992.

[20] Molly E. et al Brown. NASA earth science division study on data latency needs and requirements. Technical report, NASA, 2013.

[21] Mark W Maier. *The art of systems architecting.* CRC press, 2009.

[22] I Standard. Systems and software engineering–system life cycle processes. *ISO Standard*, 15288:2008, 2008.

[23] Edward F. Crawley, Bruce G. Cameron, and Daniel Selva, editors. *System Architecture.* Pearson, New York, first edition, 2015.

[24] Esd34. system architecture. `http://ocw.mit.edu/courses/engineering-systems-division/esd-34-system-architecture-january-iap-2007/`. Accessed: 04/14/2015.

[25] ISO/IEC/IEEE. Iso/iec/ieee 42010:2011 systems and software engineering – architecture description. Technical Report 4, ISO/IEC/IEEE, NASA Headquarters. Washington, D.C., April 2014.

[26] Daniel Selva Valero. *Rule-Based System Architecting of Earth Observation Satellite Systems.* PhD dissertation, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, June 2012.

[27] Willard L. Simmons. *A Framework for Decision Support in Systems Architecting.* PhD thesis, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, February 2008.

[28] Daniel Selva. *Rule-based System Architecting of Earth Observation Satellite Systems.* PhD thesis, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, June 2012.

[29] Applied Sciences Program NASA Earth Science. Measuring socioeconomic impacts of earth observations. Technical report, NASA, April 2012.

[30] Olivier L De Weck, Daniel Roos, and Christopher L Magee. *Engineering systems: Meeting human needs in a complex technological world.* Mit Press, 2011.

[31] Hugh McManus, Mathew Richards, Adam M Ross, and Daniel E Hastings. A framework for incorporating ilities in tradespace studies. In *AIAA Space*, volume 1, pages 941–954, 2007.

[32] Klaus Krippendorff. Combinatorial explosion. *Web Dictionary of Cybernetics and Systems. Princia Cybernetica Web*, 2010.

[33] Cihan H Dagli, Atmika Singh, Jason p Dauby, and Renzhong Wang. Smart systems architecting: computational intelligence applied to trade space exploration and system design. In *Systems Research Forum*, volume 3, pages 101–119. World Scientific, 2009.

[34] Alexander Rudat. Hexane: Architecting manned space exploration missions beyond low-earth orbit. Master's thesis, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, May 2013.

[35] Sysml open source specification project. `http://sysml.org/`. Accessed: 04/15/2015.

[36] Marc Sanchez, Daniel Selva, Antonios Seas, Bernard Seery, Bruce G. Cameron, and Edward F. Crawley. Exploring the architectural trade space of nasas space communication and navigation program. In *2013 IEEE Aerospace Conference*, Big Sky, Montana, March 3-10 2013. Institute of Electrical and Electronics Engineers.

[37] Object Management Group. Omg systems modeling language. Technical report, Object Management Group, 2012.

[38] Agi system toolkit. `http://www.agi.com/products/stk/`. Accessed: 04/15/2015.

[39] Wiley J Larson and James Richard Wertz. Space mission analysis and design. Technical report, Microcosm, Inc., Torrance, CA (US), 1992.

[40] Gerard Maral and Michel Bousquet. *Satellite Communications Systems: Systems, Techniques and Technology.* Wiley, 4 edition, 5 2002.

[41] James L Duda, Joseph Mulligan, James Valenti, and Michael Wenkel. Capabilities of SafetyNet ground systems architecture for the national polar-orbiting operational environmental satellite system (NPOESS). In *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, volume 2, pages 1045–1048. IEEE, 2004.

[42] Space Communications and Navigation Program. Space communications and navigation program (SCaN) service catalog. Technical report, National Aeronautics and Astronautics Administration, NASA Headquarters. Washington, D.C., September 2011.

[43] James Richard Wertz, David F Everett, and Jeffery John Puschell. *Space mission engineering: the new SMAD.* Microcosm Press, 2011.

[44] GOES-R Program/Code 410. Goes-r series level 1 requirements (lird). Technical report, DOC, NOAA, NESDIS, NASA, October 2013.

[45] NOAA. New satellite capabilities. News report, 2015.

[46] Kar-Ming Cheung and Esther Jennings. Coarse-grain bandwidth estimation techniques for large-scale network. In *Aerospace Conference, 2013 IEEE*, pages 1–11. IEEE, 2013.

[47] Dsn commitments office. http://deepspace.jpl.nasa.gov/advmiss/. Accessed: 10/08/2015.

[48] James L Duda, Joseph Mulligan, James Valenti, and Michael Wenkel. Latency features of safetynet ground systems architecture for the national polar-orbiting operational environmental satellite system(npoess). In *Proc. SPIE*, volume 5659, pages 233–241, 2004.

[49] Martin Agnew, L Renouard, and A Hegyi. Edrs–spacedatahighway: Near-real-time data relay services for leo satellites and haps. In *30th AIAA International Communications Satellite System Conference, ICSCC*, 2012.

[50] Alan M Goldberg. Environmental data production and delivery for npoess. In *International Geoscience and Remote Sensing Symposium*, pages 999–1001, 2002.

[51] Ensight performance measurements: Active network measurements. `https://ensight.eos.nasa.gov/`. Accessed: 08/15/2016.

[52] Leonard Kleinrock. *Communication nets: Stochastic message flow and delay.* Courier Corporation, 2007.

[53] Eytan Modiano. Burke's theorem and networks of queues. Class notes, 2015.

[54] Chuck Fraleigh, Sue Moon, Bryan Lyles, Chase Cotton, Mujahid Khan, Deb Moll, Rob Rockell, Ted Seely, and S Christophe Diot. Packet-level traffic measurements from the sprint ip backbone. *IEEE network*, 17(6):6–16, 2003.

[55] Joel Sommers, Paul Barford, Nick Duffield, and Amos Ron. Improving accuracy in end-to-end packet loss measurement. In *ACM SIGCOMM Computer Communication Review*, volume 35, pages 157–168. ACM, 2005.

[56] Kar-Ming Cheung and Douglas S Abraham. End-to-end traffic flow modeling of the integrated scan network. *the Interplanetary Network Progress Report*, 42:189, 2012.

[57] Marc Sanchez, Inigo del Portillo, Bruce G. Cameron, and Edward F. Crawley. Assessing the impact of real-time communication services on the space network ground segment. In *2016 IEEE Aerospace Conference*, 2016.

[58] Kar-Ming Cheung, Douglas Abraham, Marc Sanchez Net, Kristy Tran, and Carlyn-Ann Lee. Traffic modeling for deep space network in the human mars exploration era1. In *14th International Conference on Space Operations*, 2016.

[59] IU Sung and Jay L Gao. *CFDP performance over weather-dependent Ka-band channel.* Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2006.

[60] F Flentge. Study on cfdp and dtn architectures for esa space missions. In *The Third International Conference on Advances in Satellite and Space Communications. Houston, Texas: Space Commerce Conference and Exposition*, 2011.

[61] Tomaso De Cola, Harald Ernst, and Mario Marchese. Performance analysis of ccsds file delivery protocol and erasure coding techniques in deep space environments. *Computer Networks*, 51(14):4032–4049, 2007.

[62] General mission analysis tool. `https://gmat.gsfc.nasa.gov/`. Accessed: 04/18/2017.

[63] Gregory W Donohoe and Pen-Shu Yeh. Sensor data processing on a reconfigurable processor. In *Proc. NASA Earth Sciences Technology Conference*, pages 24–26. Citeseer, 2003.

[64] GOES-R/Code 416. Goes-r series - level i requirements (lird). Technical report, NOAA, 2013.

[65] Bradley J Clement and Mark D Johnston. The deep space network scheduling problem. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1514. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[66] Mark D Johnston. Multi-objective scheduling for NASA's future deep space network array. In *Proceedings of the 5th International Workshop on Planning and Scheduling for Space (IWPSS 2006)*, pages 27–35, 2006.

[67] Mark D Johnston and Daniel Tran. Automated scheduling for nasaâĂŹs deep space network. Technical report, Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2011., 2011.

[68] Kar-Ming Cheung, Charles H Lee, William B Gearhart, Tai Vo, and Suzanne Sindi. Link-capability driven network planning and operation. In *Aerospace Conference Proceedings, 2002. IEEE*, volume 7, pages 7–3281. IEEE, 2002.

[69] DN Baker, L Riesberg, CK Pankratz, RS Panneton, BL Giles, FD Wilder, and RE Ergun. Magnetospheric multiscale instrument suite operations and data system. *Space Science Reviews*, 199(1-4):545–575, 2016.

[70] Deborah S Bass, Roxana C Wales, and Valerie L Shalin. Choosing mars time: analysis of the mars exploration rover experience. In *Aerospace Conference, 2005 IEEE*, pages 4174–4185. IEEE, 2005.

[71] John L Bresina, Ari K Jónsson, Paul H Morris, and Kanna Rajan. Activity planning for the mars exploration rovers. In *ICAPS*, pages 40–49, 2005.

[72] Brook R. Sullivan. *Technical and economical feasibility of telerobotic on-orbit satellite servicing.* PhD thesis, University of Maryland, 2005.

[73] Charles R Doarn, Arnauld E Nicogossian, and Ronald C Merrell. Applications of telemedicine in the united states space program. *Telemedicine Journal*, 4(1):19–30, 1998.

[74] Levente Kovács, Tamás Haidegger, and Imre Rudas. Surgery from a distance. application of intelligent control for telemedicine. In *Applied Machine Intelligence and Informatics (SAMI), 2013 IEEE 11th International Symposium on*, pages 125–129. IEEE, 2013.

[75] The Consultative Committee for Space Data Systems. Voice communications. Technical report, The Consultative Committee for Space Data Systems, September 2010.

[76] Dapeng Wu, Yiwei Thomas Hou, Wenwu Zhu, Ya-Qin Zhang, and Jon M Peha. Streaming video over the internet: approaches and directions. *IEEE Transactions on circuits and systems for video technology*, 11(3):282–300, 2001.

[77] Ashwin Rao, Arnaud Legout, Yeon-sup Lim, Don Towsley, Chadi Barakat, and Walid Dabbous. Network characteristics of video streaming traffic. In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, page 25. ACM, 2011.

[78] ESA. Soho science operations plan. Technical report, ESA, 1995.

[79] The Consultative Committee for Space Data Systems. Delta-DOR - technical characteristics and performance. Technical report, The Consultative Committee for Space Data Systems, May 2013.

[80] M. Goldberg. NOAA direct broadcast data initiative to meet NWP latency requirements. Technical report, Coordination Group for Meteorological Satellites, July 2013.

[81] P. G. Edwards and D. Pawlak. Metop: The space segment for Eumetsat's polar system. Technical report, ESA, May 2000.

[82] Joint Polar Satellite System. Level 1 requirements document - final. Technical report, Joint Polar Satellite System, June 2015.

[83] JL Burch, TE Moore, RB Torbert, and BL Giles. Magnetospheric multiscale overview and science objectives. *Space Science Reviews*, 199(1-4):5–21, 2016.

[84] Wenyu Jiang and Henning Schulzrinne. Modeling of packet loss and delay and their effect on real-time multimedia service quality. In *Proceedings of Nossdav'2000*. Citeseer, 2000.

[85] I Cisco. Cisco visual networking index: Forecast and methodology, 2011–2016. *CISCO White paper*, pages 2011–2016, 2012.

[86] Sebastian Egger, Peter Reichl, Tobias Hoßfeld, and Raimund Schatz. Time is bandwidth? narrowing the gap between subjective time perception and quality of experience. In *2012 IEEE International Conference on Communications (ICC)*, pages 1325–1330. IEEE, 2012.

[87] Tobias Hoßfeld, Sebastian Egger, Raimund Schatz, Markus Fiedler, Kathrin Masuch, and Charlott Lorentzen. Initial delay vs. interruptions: between the devil and the deep blue sea. In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 1–6. IEEE, 2012.

[88] Sunny Tsiao. Read you loud and clear. *The Story of NASA's Spaceflight Tracking and Data Network*, 2007.

[89] Peter C Fishburn. Utility theory for decision making. Technical report, DTIC Document, 1970.

[90] Theo J Stewart. A critical survey on the status of multiple criteria decision making theory and practice. *Omega*, 20(5):569–586, 1992.

[91] Ales Cerný. *Mathematical techniques in finance: tools for incomplete markets*. Princeton University Press, 2009.

[92] Marc Sanchez Net. Architecting space communications networks. Master's thesis, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, June 2014.

[93] Alessandro Aliakbargolkar. *A framework for space systems architecting under stakeholder objectives ambiguity.* PhD thesis, Massachusetts Institute of Technology, 2012.

[94] Mark Davis. *Option pricing in incomplete markets*, volume 1 of *Mathematics of Derivative Securities*, pages 216–226. Cambridge University Press, New York, 2009.

[95] Lenos Trigeorgis. *Real options: Managerial flexibility and strategy in resource allocation.* MIT press, 1996.

[96] Joshua T Knight. *A Prospect Theory-Based Real Option Analogy for Evaluating Flexible Systems and Architectures in Naval Ship Design.* PhD thesis, University of Michigan, 2014.

[97] Julien Hugonnier, Dmitry Kramkov, and Walter Schachermayer. On utility-based pricing of contingent claims in incomplete markets. *Mathematical Finance*, 15(2):203–212, 2005.

[98] Gergana Assenova Bounova. *Topological evolution of networks: Case studies in the US airlines and language wikipedias.* PhD thesis, Massachusetts Institute of Technology, 2009.

[99] Stephen P Borgatti and Martin G Everett. A graph-theoretic perspective on centrality. *Social networks*, 28(4):466–484, 2006.

[100] Stephen P Borgatti. Centrality and network flow. *Social networks*, 27(1):55–71, 2005.

[101] Phillip Bonacich and Paulette Lloyd. Eigenvector-like measures of centrality for asymmetric relations. *Social networks*, 23(3):191–201, 2001.

[102] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[103] Linton C Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.

[104] Phillip Bonacich. Some unique properties of eigenvector centrality. *Social networks*, 29(4):555–564, 2007.

[105] Mark EJ Newman. A measure of betweenness centrality based on random walks. *Social networks*, 27(1):39–54, 2005.

[106] Jae Dong Noh and Heiko Rieger. Random walks on complex networks. *Physical review letters*, 92(11):118701, 2004.

[107] Yannick Rochat. Closeness centrality extended to unconnected graphs: The harmonic centrality index. In *ASNA*, 2009.

[108] V Latora and M Marchiori. A measure of centrality based on the network efficiency. arxiv: con-math 0402050. *The open-access journal of physics*, 2004.

[109] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[110] Tharaka Alahakoon, Rahul Tripathi, Nicolas Kourtellis, Ramanuja Simha, and Adriana Iamnitchi. K-path centrality: A new centrality measure in social networks. In *Proceedings of the 4th Workshop on Social Network Systems*, page 1. ACM, 2011.

[111] Xingqin Qi, Eddie Fuller, Qin Wu, Yezhou Wu, and Cun-Quan Zhang. Laplacian centrality: A new centrality measure for weighted networks. *Information Sciences*, 194:240–253, 2012.

[112] RHA Lindelauf, HJM Hamers, and BGM Husslage. Cooperative game theoretic centrality analysis of terrorist networks: The cases of jemaah islamiyah and al qaeda. *European Journal of Operational Research*, 229(1):230–238, 2013.

[113] Jialun Qin, Jennifer J Xu, Daning Hu, Marc Sageman, and Hsinchun Chen. Analyzing terrorist networks: A case study of the global salafi jihad network. In *International Conference on Intelligence and Security Informatics*, pages 287–304. Springer, 2005.

[114] Kevin Lewis, Jason Kaufman, Marco Gonzalez, Andreas Wimmer, and Nicholas Christakis. Tastes, ties, and time: A new social network dataset using facebook. com. *Social networks*, 30(4):330–342, 2008.

[115] Zhong Deng and Phillip Bonacich. Some effects of urbanism on black social networks. *Social Networks*, 13(1):35–50, 1991.

[116] Raja Kali and Javier Reyes. The architecture of globalization: a network approach to international economic integration. *Journal of International Business Studies*, 38(4):595–620, 2007.

[117] Stefania Vitali, James B Glattfelder, and Stefano Battiston. The network of global corporate control. *PloS one*, 6(10):e25995, 2011.

[118] Paul Hines and Seth Blumsack. A centrality measure for electrical networks. In *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pages 185–185. IEEE, 2008.

[119] GA Kohring. Complex dependencies in large software systems. *Advances in Complex Systems*, 12(06):565–581, 2009.

[120] Manuel E Sosa, Steven D Eppinger, and Craig M Rowles. A network approach to define modularity of components in complex products. *Journal of mechanical design*, 129(11):1118–1129, 2007.

[121] Rusnak John Baldwin Carliss, MacCormack Alan. Hidden structure: Using network methods to map system architecture. Technical report, Harvard Business School, 2014.

[122] Kaushik Sinha et al. *Structural complexity and its implications for design of cyber-physical systems.* PhD thesis, Massachusetts Institute of Technology, 2014.

[123] Farah Alibay. *Evaluation of multi-vehicle architectures for the exploration of planetary bodies in the Solar System.* PhD thesis, Massachusetts Institute of Technology, 2014.

[124] Howard Seltman. Approximations for mean and variance of a ratio. `http://www.stat.cmu.edu/~hseltman/files/ratio.pdf`. Accessed: 08/16/2016.

[125] Gene Lee, OL de Weck, N Armar, E Jordan, R Shishko, Afreen Siddiqi, and J Whiting. Spacenet: Modeling and simulating space logistics. In *AIAA Space 2009 Conference and Exposition*, 2008.

[126] Ronnie Emile Thebeau. *Knowledge management of system interfaces and interactions from product development processes.* PhD thesis, Massachusetts Institute of Technology, 2001.

[127] João L Sobrinho. Algebra and algorithms for qos path computation and hop-by-hop routing in the internet. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 727–735. IEEE, 2001.

[128] Sergio D Servetto and Guillermo Barrenechea. Constrained random walks on random graphs: routing algorithms for large scale wireless sensor networks. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 12–21. ACM, 2002.

[129] Jon Postel. User datagram protocol. Technical report, Internet Engineering Task Force, 1980.

[130] Alessandro Golkar and Edward F Crawley. A framework for space systems architecture under stakeholder objectives ambiguity. *Systems Engineering*, 17(4):479–502, 2014.

[131] Marc Sanchez Net, Iñigo del Portillo, Bruce Cameron, Edward F Crawley, and Daniel Selva. Integrated tradespace analysis of space network architectures. *Journal of Aerospace Information Systems*, 12(8):564–578, 2015.

[132] Richard Bellman. *Dynamic Programming.* Princeton University Press, Princeton, New Jersey, 1957.

[133] Koki Ho, Olivier L de Weck, Jeffrey A Hoffman, and Robert Shishko. Dynamic modeling and optimization for space logistics using time-expanded networks. *Acta Astronautica*, 105(2):428–443, 2014.

[134] Sydney Do. *Towards earth independence - tradespace exploration of long-duration crewed mars syrface system architectures.* PhD thesis, Massachusetts Institute of Technology Department of Aeronautics and Astronautics, Cambridge, Massachusetts, June 2015.

[135] Rania Hassan, Babak Cohanim, Olivier De Weck, and Gerhard Venter. A comparison of particle swarm optimization and the genetic algorithm. In *Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference*, pages 18–21, 2005.

[136] Olivier de Weck and Karen Willcox. Multidisciplinary system design optimization (MSDO). Class Notes, April 2014.

[137] Tim Weilkiens. *Systems engineering with SysML/UML: modeling, analysis, design.* Morgan Kaufmann, 2011.

[138] Cyrus D Jilla. *A multiobjective, multidisciplinary design optimization methodology for the conceptual design of distributed satellite systems.* PhD dissertation, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2002.

[139] R Hevner Von Alan, Salvatore T March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS quarterly*, 28(1):75–105, 2004.

[140] Archnet. `http://www.mit.edu/~msnet/ArchNet/index.html`. Accessed: 08/05/2015.

[141] Marco Farina and Paolo Amato. Fuzzy optimality and evolutionary multiobjective optimization. In *EMO*, volume 3, pages 58–72. Springer, 2003.

[142] Average height. `http://www.averageheight.co/average-male-height-by-country`. Accessed: 09/01/2016.

[143] Charles Spearman. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101, 1904.

[144] Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[145] Leo A Goodman and William H Kruskal. Measures of association for cross classifications. In *Measures of association for cross classifications*, pages 2–34. Springer, 1979.

[146] Robert H Somers. A new asymmetric measure of association for ordinal variables. *American sociological review*, pages 799–811, 1962.

[147] Ben Carterette. On rank correlation and the distance between rankings. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 436–443. ACM, 2009.

[148] Persi Diaconis. Group representations in probability and statistics. *Lecture Notes-Monograph Series*, 11:i–192, 1988.

[149] SA Teukolsky, WT Vetterling, and BP Flannery. *Numerical recipes in C: the art of Scientific Computing.* Cambridge University Press, Cambridge, 1986.

[150] Nikolai Matni, Ao Tang, and John C Doyle. A case study in network architecture tradeoffs. In *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*, page 18. ACM, 2015.

[151] Hyojun Lim and Chongkwon Kim. Multicast tree construction and flooding in wireless ad hoc networks. In *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pages 61–68. ACM, 2000.

[152] Jennifer E Manuse. *The strategic evolution of systems: Principles and framework with applications to space communication networks.* PhD thesis, Massachuetts Institute of Technology, 2009.

[153] Climate and Meteorology Section. The socioeconomic value of climate and weather forecasting: A review. Technical report, Midwestern Climate Center, 1989.

[154] US Senate. Weather research and forecasting innovation act of 2016, h.r. 1561. US Senate Bill, November 2016.

[155] Deep space communications and navigation center of excellence. `https://descanso.jpl.nasa.gov/performmetrics/stairstep.pdf`. Accessed: 04/28/2017.

[156] G Kelly and JN Thèpaut. The relative contributions of the various space observing systems to the ecmwf forecast system. In *Proceedings of the 15th International TOVS Study Conference, Maratea*, 2006.

[157] Carla Cardinali, Sergio Pezzulli, and Erik Andersson. Influence-matrix diagnostic of a data assimilation system. *Quarterly Journal of the Royal Meteorological Society*, 130(603):2767–2786, 2004.

[158] Shawn W Miller, Michael Jamilkowski, and Kerry D Grant. Joint polar satellite system (JPSS) common ground system (CGS) overview and evolution. In *AIAA SPACE 2013 Conference and Exposition*, page 5534, 2013.

[159] Michael L Jamilkowski, Shawn Miller, and Kerry D Grant. Joint polar satellite system (JPSS) common ground system (CGS) multimission support. In *AIAA SPACE 2013 Conference and Exposition*, page 5499, 2013.

[160] Joint Polar Satellite System (JPSS) Code 470. Joint polar satellite system (JPSS) ground system requirements document (GSRD). Technical report, NASA, 2013.

[161] Mitchell D Goldberg, Heather Kilcoyne, Harry Cikanek, and Ajay Mehta. Joint polar satellite system: The united states next generation civilian polar-orbiting environmental satellite system. *Journal of Geophysical Research: Atmospheres*, 118(24), 2013.

[162] D Klaes and Kenneth Holmlund. The EPS/Metop system: Overview and first results. In *Proc. Joint EUMETSAT Meteorol. Satell. Conf., 15th Satell. Meteorol. Oceanogr. Conf. Amer. Meteorol. Soc.* Citeseer, 2007.

[163] T Andreassen, T Beck, J Bolle, A Haaland, and A Jensen. Polar bears and spacecraft tracking. *ESA bulletin*, pages 118–121, 2002.

[164] ENSIGHT network performance tests. http://ensight.eos.nasa.gov. Accessed: 02/25/2017.

[165] GOES-R/Code 416. GOES-R series - ground segment (GS) project functional and performance specification. Technical report, NOAA, 2015.

[166] GOES-R/Code 416. GOES rebroadcast (GRB) downlink specifications for users. Technical report, NOAA, 2012.

[167] Peter Michael Inness and Steve Dorling. *Operational weather forecasting*. John Wiley & Sons, 2012.

[168] Observing systems capability analysis and review tool. https://www.wmo-sat.info/oscar/. Accessed: 02/25/2017.

[169] The Coordination Group for Meteorological Satellites. NOAA direct broadcast data initiative to meet nwp latency. Technical report, The Coordination Group for Meteorological Satellites, 2013.

[170] T Tsuyuki and T Fujita. Outline of the operational numerical weather prediction at the japanese meteorological agency. *JMA Report, Tokyo*, 2002.

[171] Matplotlib basemap toolkit. `http://matplotlib.org/basemap/`. Accessed: 02/27/2017.

[172] Bruce E MacNeal and WJ Hurd. Parametric cost analysis of NASA's DSN array. In *Space OPS 2004 Conference*, page 433, 2004.

[173] Vahraz Jamnejad, Tom Cwik, and George Resch. Cost and reliability study for a large array of small reflector antennas for JPL/NASA deep space network (DSN). In *Aerospace Applications Conference, 1993. Digest., 1993 IEEE*, pages 121–132. IEEE, 1993.

[174] Sander Weinreb and L D'Addario. Cost equation for the ska. *SKA Memorandum*, 1, 2001.

[175] JI Statman, DS Bagri, CS Yung, S Weinreb, and BE MacNeal. Optimizing the antenna size for the deep space network array. *IPN Progress Report*, pages 1–8, 2004.

[176] Goddard Space Flight Center. Near earth network (NEN) user's guide. Technical report, National Aeronautics and Space Administration, January 2010.

[177] Department of Defense. Dod facilities pricing guide. Technical report, Department of Defense, 2015.

[178] Baard Eilertsen. Ground station networks vs. geo relay satellite systems for polar orbiting satellites. In *SpaceOps 2012*, page 1290826. American Institute of Aeronautics and Astronautics, 2012.

[179] Too big to fail? the green bank telescopeâĂŹs uncertain future. `http://www.scientificamerican.com/article/too-big-to-fail-the-green-bank-telescopes-uncertain-future/`. Accessed: 02/27/2017.

[180] Sunny Tsiao. Historical evolution and legacy of nasa's near-earth space communications networks. In *AIAA Space 2009 Conference & Exposition*, page 6409, 2009.

[181] Llewellyn D Howell. International country risk guide methodology. *East Syracuse, NY: PRS Group*, 2011.

[182] Claude B Erb, Campbell R Harvey, and Tadas E Viskanta. Political risk, economic risk and financial risk. *Financial Analysts Journal*, 1996.

[183] Stephen J Kapurch. *NASA Systems Engineering Handbook*. Diane Publishing, 2010.

[184] Robert C Merton. An intertemporal capital asset pricing model. *Econometrica: Journal of the Econometric Society*, pages 867–887, 1973.

[185] Campbell R Harvey. Predictable risk and returns in emerging markets. *Review of Financial studies*, 8(3):773–816, 1995.

[186] C Herb, CR Harvey, and T Viskanta. Country credit risk and global portfolio selection,". *Journal of Portfolio Management*, 1995.

[187] Mohamed A Ramady et al. *Political, economic and financial country risk.* Springer, 2013.

[188] Claude B Erb, Campbell R Harvey, and Tadas E Viskanta. Expected returns and volatility in 135 countries. *The Journal of Portfolio Management*, 22(3):46–58, 1996.

[189] Theodore K. Seher. *Campaign-level Science Traceability for Earth Observation System Architecting.* PhD thesis, Massachusetts Institute of Technology Engineering Systems Division, June 2009.

[190] Brandon H Suarez. *Integrating spacecraft and aircraft in Earth Observation System architectures.* PhD thesis, Massachusetts Institute of Technology, 2011.

[191] Justin M Colson. *System architecting of a campaign of earth observing satellites.* PhD thesis, Massachusetts Institute of Technology, 2008.

[192] Harry M Markowitz. *Portfolio selection: efficient diversification of investments*, volume 16. Yale university press, 1968.

[193] Joint Polar Satellite System (JPSS) Code 470. Joint polar satellite system. (june 2014). level 1 requirements document - final. Technical report, NASA, 2014.

[194] Office of Inspector General. Space communications and navigation: NASA's management of the space network. Technical report, NASA, 2014.

[195] Office of Inspector General. Audit of the joint polar satellite system: Continuing progress in establishing capabilities, schedules, and costs is needed to mitigate data gaps. Technical report, U.S. Department of Commerce, 2012.

[196] Office of Inspector General. Cost estimates, long-term savings, milestones, and enterprise architecture policy are needed for common satellite ground system program. Technical report, U.S. Department of Commerce, 2015.

[197] Charles D Edwards, Bradford W Arnold, David J Bell, Kristoffer N Bruvold, Roy E Gladden, Peter A Ilott, and Charles H Lee. Relay support for the mars science laboratory and the coming decade of mars relay network evolution. In *Aerospace Conference, 2012 IEEE*, pages 1–11. IEEE, 2012.

[198] Robert J Cesarone, Douglas S Abraham, and Leslie J Deutsch. Prospects for a next-generation deep-space network. *Proceedings of the IEEE*, 95(10):1902–1915, 2007.

[199] The Consultative Committee for Space Data Systems. Proximity-1 space link protocolâĂŤrationale, architecture, and scenarios. Technical report, The Consultative Committee for Space Data Systems, December 2013.

[200] GJ Kazz and E Greenberg. Mars relay operations: Application of the CCSDS proximity-1 space data link protocol. *JPL, California Institute of Technology*, 2002.

[201] E Glenn Lightsey, Thomas Campbell, Andreas Mogensen, P Daniel Burkhart, Todd Ely, and Courtney Duncan. Expected performance of the electra transceiver for mars missions. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2005.

[202] Daniel J McCleese et al. Robotic mars exploration strategy 2007–2016. *Jet Propulsion Laboratory, Pasadena, CA*, 2006.

[203] Mars Architecture Steering Group. Human exploration of mars design reference architecture 5.0. Technical report, NASA, 2009.

[204] Mars Architecture Steering Group. Human exploration of mars design reference architecture 5.0 - addendum. Technical report, NASA, 2009.

[205] Andre Makovsky, Peter Ilott, and Jim Taylor. Mars science laboratory telecommunications system design. Technical report, Jet Propulsion Laboratory, 2009.

[206] Neal R Kuo. Mars network operations concept. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 2, pages 209–216. IEEE, 2000.

[207] John P Grotzinger, Joy Crisp, Ashwin R Vasavada, Robert C Anderson, Charles J Baker, Robert Barry, David F Blake, Pamela Conrad, Kenneth S Edgett, Bobak Ferdowski, et al. Mars science laboratory mission and science investigation. *Space science reviews*, 170(1-4):5–56, 2012.

[208] Jim Taylor, Dennis K Lee, and Shervin Shambayati. Mars reconnaissance orbiter telecommunications. *DESCANSO Design and Performance Summary Series*, 12, 2006.

[209] RA Yingst, BA Cohen, DW Ming, and DB Eppler. Comparing apollo and mars exploration rover (mer)/phoenix operations paradigms for human exploration during nasa desert-rats science operations. *Acta Astronautica*, 90(2):311–317, 2013.

[210] Marcum Reagan, Barbara Janoiko, James Johnson, Steven Chappell, Ph. D, and Andrew Abercromby. NASA's analog missions: Driving exploration through innovative testing. In *AIAA SPACE 2012 Conference & Exposition*, page 5238, 2012.

[211] Jim Taylor, Andre Makovsky, Andrea Barbieri, Ramona Tung, Polly Estabrook, and A Gail Thomas. Mars exploration rover telecommunications. *Deep Space Communications and Navigation Systems. Jet Propulsion Laboratory*, 2005.

[212] Dean Eppler, Byron Adams, Doug Archer, Greg Baiden, Adrian Brown, William Carey, Barbara Cohen, Chris Condit, Cindy Evans, Corey Fortezzo, et al. Desert research and technology studies (drats) 2010 science operations: Operational approaches and lessons learned for managing science during human planetary surface missions. *Acta Astronautica*, 90(2):224–241, 2013.

[213] Space Communication Architecture Working Group (SCAWG). NASA space communications and navigation architecture recommendation for 2005-2030. Technical report, NASA, 2006.

[214] Mars Exploration Program Analysis Group (MEPAG) Presentation. Selecting a landing site for humans on mars. Technical report, NASA, 2016.

[215] Leri et al Datashvili. Advanced architectures of large space deployable mesh reflectors: From medium to very large sizes. In *Advanced Lightweight Structures and Reflector Antennas*, pages 7–20, 2014.

[216] W Dan Williams, Michael Collins, Richard Hodges, Richard S Orr, O Scott Sands, Leonard Schuchman, and Hemali Vyas. High-capacity communications from martian distances. Technical report, NASA, 2007.

[217] Abhijit Biswas, Hamid Hemmati, Sabino Piazzolla, Bruce Moision, Kevin Birnbaum, and Kevin Quirk. Deep-space optical terminals (dot) systems engineering. *IPN Progress Report*, 42:183, 2010.

[218] H Hemmati, K Wilson, MK Sue, LJ Harcke, M Wilhelm, C-C Chen, J Lesh, Y Feria, D Rascoe, and F Lansing. Comparative study of optical and radio-frequency communication systems for a deep-space mission. *TDA Progress Report*, February 1997.

[219] Deep Space Network. Dsn telecommunications link design handbook. Technical report, Jet Propulsion Laboratory, 2000.

[220] Tudor Stoenescu and Loren Clare. Traffic modeling for NASA's space communications and navigation (SCaN) network. In *Aerospace Conference, 2008 IEEE*, pages 1–14. IEEE, 2008.

[221] Stephen J Hoffman. Evolvable mars campaign development. Technical report, NASA Johnson Space Center, 2016.

[222] James A. Nessel. Performance analysis of a nasa integrated network array. Technical report, NASA, 2012.

[223] R Mukai, D Hansen, A Mittskus, J Taylor, and M Danos. Juno telecommunications. *NASA DESCANSO Design and Performance Summary Series*, 2012.

[224] Abhijit Biswas and Sabino Piazzolla. Deep-space optical communications downlink budget from mars: System parameters. *IPN Progress Report*, 42(154):0–1, 2003.

[225] Bruce Moision and Jon Hamkins. Deep-space optical communications downlink budget: modulation and coding. *IPN Progress Report*, 42(154):1–28, 2003.

[226] Shinhak Lee, Keith E Wilson, and Mitchell Troy. Background noise mitigation in deep space optical communications using adaptive optics. *The Interplanetary Network Progress Report, IPN PR*, pages 42–161, 2005.

[227] Hemani Kaushal and Georges Kaddoum. Optical communication in space: Challenges and mitigation techniques. *IEEE Communications Surveys & Tutorials*, 2016.

[228] Radha A Venkat and David W Young. Cloud-free line-of-sight estimation for free space optical communications. In *SPIE Defense, Security, and Sensing*, pages 873205–873205. International Society for Optics and Photonics, 2013.

[229] Sabino Piazzolla and Stephen Slobin. Statistics of link blockage due to cloud cover for free-space optical communications using NCDC surface weather observation data. In *High-Power Lasers and Applications*, pages 138–149. International Society for Optics and Photonics, 2002.

[230] Sylvain Poulenard, Michael Crosnier, and Angélique Rissons. Ground segment design for broadband geostationary satellite with optical feeder link. *Journal of Optical Communications and Networking*, 7(4):325–336, 2015.

[231] Christian Fuchs and Florian Moll. Ground station network optimization for space-to-ground optical communication links. *Journal of Optical Communications and Networking*, 7(12):1148–1159, 2015.

[232] H Philip Stahl. Survey of cost models for space telescopes. *Optical Engineering*, 49(5):053005–053005, 2010.

[233] Alessandra Babuscia, Dariush Divsalar, and Kar-Ming Cheung. CDMA communications systems with constant envelope modulation for cubesats. In *Aerospace Conference, 2015 IEEE*, pages 1–8. IEEE, 2015.

[234] GSFC's Exploration and Space Communications Projects Division. Earth regimes network evolution study. Technical report, NASA, May 2015.

[235] Robert Shishko and Robert Aster. NASA systems engineering handbook. *NASA Special Publication*, 6105, 1995.

[236] Jet Propulsion Laboratory. Dsn telecommunications link design handbook. Technical report, California Institute of Technology, March 2012.

[237] Jet Propulsion Laboratory. Deep space network services catalog. Technical report, California Institute of Technology, February 2015.

[238] Goddard Space Flight Center. Space network (SN) user's guide. Technical report, National Aeronautics and Space Administration, August 2007.

[239] Communications service office. `https://cso.nasa.gov/`. Accessed: 04/16/2015.

[240] NASA integrated communications services. `http://itcd.hq.nasa.gov/NICS.html`. Accessed: 04/16/2015.

[241] Kenneth Y. Jo. *Satellite Communications Network Design and Analysis.* Artech House Publishers, 10 2011.

[242] Marc Sanchez Net, Inigo Del Portillo, Edward Crawley, and Bruce Cameron. Approximation methods for estimating the availability of optical ground networks. *Journal of Optical Communications and Networking*, 8(10):800–812, 2016.

[243] Hamid Hemmati. *Near-Earth Laser Communications.* CRC Press, Taylor & Francis Group, LLC, 6000 Broken Sound Parkway NW, Suite 300, Boca Raton, FL 33487-2742, 2009.

[244] Robert Link, Mary Ellen Craddock, and Randall J Alliss. Mitigating the impact of clouds on optical communications. In *Aerospace Conference, 2005 IEEE*, pages 1258–1265. IEEE, 2005.

[245] Gary S. Wojcik, Heather L. Szymczak, Randall J. Alliss, Robert P. Link, Mary Ellen Craddock, and Michael L. Mason. Deep-space to ground laser communications in a cloudy world. *Proc. SPIE*, 5892:589203–589203–11, 2005.

[246] Optical Link Study Group. Optical link study group final report. Technical report, Interagency Operations Advisory Group, June 2012.

[247] Nicolas Perlot and Josep Perdigues-Armengol. Model-oriented availability analysis of optical GEO-ground links. In *SPIE LASE*, pages 82460P–82460P. International Society for Optics and Photonics, 2012.

[248] Mohammadreza A Kashani, Majid Safari, and Murat Uysal. Optimal relay placement and diversity analysis of relay-assisted free-space optical communication systems. *Journal of Optical Communications and Networking*, 5(1):37–47, 2013.

[249] Bernhard Epple. Simplified channel model for simulation of free-space optical communications. *Journal of Optical Communications and Networking*, 2(5):293–304, 2010.

[250] Alexander Vavoulas, Harilaos G Sandalidis, and Dimitris Varoutas. Weather effects on FSO network connectivity. *Journal of Optical Communications and Networking*, 4(10):734–740, 2012.

[251] Harilaos G Sandalidis. Performance analysis of a laser ground-station-to-satellite link with modulated gamma-distributed irradiance fluctuations. *Journal of Optical Communications and Networking*, 2(11):938–943, 2010.

[252] Steven J Goodman, James Gurka, Mark DeMaria, Timothy J Schmit, Anthony Mostek, Gary Jedlovec, Chris Siewert, Wayne Feltz, Jordan Gerth, and Renate Brummer. The GOES-R proving ground: accelerating user readiness for the next-generation geostationary environmental satellite system. *Bulletin of the American Meteorological Society*, 93(7):1029–1040, 2012.

[253] K Dieter Klaes, Marc Cohen, Yves Buhler, Peter Schlüssel, Rosemary Munro, Axelvon Engeln, EoinÓ Clérigh, Hans Bonekamp, Jörg Ackermann, and Johannes Schmetz. An introduction to the EUMETSAT polar system. *Bulletin of the American Meteorological Society*, 88(7):1085–1096, 2007.

[254] Hans Fischer. *A history of the central limit theorem: From classical to modern probability theory.* Springer Science & Business Media, 2010.

[255] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *The Annals of Mathematical Statistics*, pages 493–507, 1952.

[256] Dirk P Kroese, Tim Brereton, Thomas Taimre, and Zdravko I Botev. Why the monte carlo method is so important today. *Wiley Interdisciplinary Reviews: Computational Statistics*, 6(6):386–392, 2014.

[257] The Mathworks Inc. MATLAB. `http://www.mathworks.com/products/matlab/`. 2015-09-29.

[258] Numpy.

[259] Dimitri P. Bertsekas and John N. Tsitsiklis. *Introduction to Probability.* Athena Scientific, 2002.

[260] Kar-Ming Cheung. Statistical link analysis. a risk analysis perspective. *Interplanetary Network Directorate Progress Report*, pages 42–183, 2010.

[261] Torben Hagerup and Christine Rüb. A guided tour of Chernoff bounds. *Information processing letters*, 33(6):305–308, 1990.

[262] Jakob H Macke, Philipp Berens, Alexander S Ecker, Andreas S Tolias, and Matthias Bethge. Generating spike trains with specified correlation coefficients. *Neural Computation*, 21(2):397–423, 2009.

[263] David R Cox and Nanny Wermuth. A simple approximation for bivariate and trivariate normal integrals. *International Statistical Review/Revue Internationale de Statistique*, pages 263–269, 1991.

[264] Willem Albers and Wilbert CM Kallenberg. A simple approximation to the bivariate normal distribution with large correlation coefficient. *Journal of multivariate analysis*, 49(1):87–96, 1994.

[265] EOS Project Science Office. NASA earth observations - cloud fraction.

[266] Claire L. Parkinson and Raynold Greenstone. EOS data products handbook, volume 2. Technical report, National Aeronautics and Space Adminsitration, October 2000.

[267] MODIS Cloud Mask Team. Discriminating clear-sky from cloud with MODIS. Technical report, Cooperative Institute for Meteorological Satellite Studies, October 2010.

[268] Chunpeng Wang, Zhengzhao Johnny Luo, and Xianglei Huang. Parallax correction in collocating cloudsat and moderate resolution imaging spectroradiometer (modis) observations: Method and application to convection study. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 116(D17):80–88, 2011.

[269] X Xiong, K Chiang, J Sun, WL Barnes, B Guenther, and VV Salomonson. NASA EOS Terra and Aqua MODIS on-orbit performance. *Advances in Space Research*, 43(3):413–422, 2009.

[270] Pedro Garcia, Ana Benarroch, and Jose Manuel Riera. Spatial distribution of cloud cover. *International Journal of Satellite Communications and Networking*, 26(2):141–155, 2008.

GLOSSARY

ACE	Advanced Composition Explorer	38
Apollo	Project Apollo	38, 46
AQUA	AQUA Spacecraft (EOS PM-1)	38
ArchNet	Architecture Network Simulator	201–203, 209
ARQ	Automatic Repeat Request	185
AWGN	Additive White Gaussian Noise	248
BWG	DSN 34m Beam Wave Guide	146
CAS	Cassini Huygens	38, 281
CCSDS	The Consultative Committee for Space Data Systems	180, 181, 186, 196, 240
CDMA	Code Division Multiple Access	18, 222, 223, 239
CDSCC	Canberra Deep Space Communication Complex	237
CER	Cost Estimation Relationship	70
CFDP	CCSDS File Delivery Protocol	35, 185, 190, 281
CGI	Common Ground Infrastructure	10, 17, 52, 122–124, 126, 127, 141, 159, 165, 170, 171, 175–177, 233, 234
CLASS	Comprehensive Large Array-Data Stewardship System	126
CMA	China Meteorological Administration	121
CONUS	Continental US	118, 124, 142, 149, 162, 171
CPU	Central Processing Unit	69
CSO	Communications Service Office	18, 31, 237, 239–241
dB	Decibel	247
Delta-DOR	Delta-Differential One-Way Ranging	237, 238
DFE	Direct from Earth	181, 186, 195, 198, 204, 214–217, 227
DMSP	Defense Meteorological Satellite Program	121, 123, 127, 128, 135, 136, 139, 176
DoD	US Department of Defense	21, 121, 148, 175
DRA5.0	NASA Mars Design Reference Architecture 5.0	38, 180, 195
DRATS	Desert Research and Technology Studies	189
DS0	Digital Signal 0 Line	95
DS1	Digital Signal 1 Line	95
DSCC	Deep Space Communication Complex	237

DSH	Deep Space Habitat	181, 183, 192, 196
DSM	Design Structured Matrix	15, 18, 50, 55, 65, 75–77, 83, 138, 183, 184, 192, 193
DSN	Deep Space Network	13, 15, 17, 21, 22, 31–34, 36, 40, 43, 48, 62, 87, 146, 148, 170, 179–181, 183–188, 193–195, 197, 201, 203–206, 213, 218, 219, 228, 229, 237–239, 241, 245, 281
DSP	Digital Signal Processing	186, 281
DTE	Direct to Earth	15, 18, 181, 185, 193, 195–198, 201–206, 208, 211, 214–221, 227–229, 233, 236
ECMWF	European Center for MediumâĂŞRange Weather Forecasts	14, 17, 118–120, 132, 138, 281
EDL	Entry Descent and Landing	183
EDR	Environmental Data Record	175
EIRP	Equivalent Isotropically Radiated Power	18, 198, 205, 206, 211, 212, 215, 216, 223, 224, 248, 254, 255
EOS	Earth Observing System	14, 38, 117, 123–125, 136, 139, 168
ESA	European Space Agency	42, 121, 123, 124
ESPC	Environmental Satellite Processing Center	126
EVA	Extra Vehicular Activity	183, 192, 200, 201, 226
FDF	Flight Dynamics Facility	238, 239
FEC	Forward Error Correcting	34, 186
FERMI	Fermi Gamma-ray Space Telescope	38
FNMOC	Fleet Numerical Meteorology and Oceanography Center	127
FOM	Figure of Merit	143, 167, 169, 247
FTP	File Transfer Protocol	34

GCOM	Global Change Observation Mission	14, 125, 126
GDSCC	Goldstone Deep Space Communication Complex	237
GEO	Geosynchronous Orbit	238
GFS	Global Forecasting System	138
GNSS	Global Navigation Satellite System	134, 136
GOES	Geostationary Operational Environmental Satellite	14, 36, 38, 121, 123–126, 128, 136, 139, 141, 159
GPS	Global Positioning System	119
GPSRO	GPS Radio Occultation	119, 133
GRB	Gamma Ray Burst	39, 40
GSFC	Goddard Space Flight Center	124, 149, 281
GSFC	Goddard Spaceflight Center	238
GSMK	Gaussian Minimum Shift Keying	196
HD	High Definition	184, 200, 201
HPA	High Power Amplifier	147, 211, 213, 251, 252
HVAC	Heating, ventilation and Air Conditioning	149
IID	Independent and Identically Distributed	204
IJPS	Initial Joint Polar System	123
IONet	Internet Protocol Operational Network	240
IP	Internet Protocol	34, 81, 94, 95, 232, 240
IR	Infra-red	133
ISRU	In-situ Resource Utilization	179, 183, 200, 202
ISS	International Space Station	37–39, 71
JAXA	Japan Aerospace Exploration Agency	121, 125, 126
JPL	Jet Propulsion Laboratory	43, 181, 185, 188, 195, 197, 198, 201, 203, 208, 222, 237, 281
JPSS	Joint Polar Satellite System	10, 14, 17, 21, 38, 42, 52, 117, 121–123, 126, 127, 133, 136, 138, 139, 141–144, 146, 156, 159, 163, 165, 168, 170, 171, 173–178, 233

JSC	Johnson Space Center	281
KaSA	Ka-band Single Access	239
KSAT	Kongsberg Satellite Services	238
KuSA	Ku-band Single Access	239
LAN	Local Area Network	240
LANCE	Land, Atmosphere Near real-time Capability for EOS	31, 42
LDPC	Low-Density Parity-Check Code	196, 222, 223
LEO	Low Earth Orbit	121, 127, 174, 238
LGA	Low Gain Antenna	222
LMO	Low Mars Orbit	222
LOS	Line of Sight	14, 15, 35, 127, 142, 172, 184, 185
MA	Multiple Access	15, 124, 211, 214, 215, 222–225, 227, 229, 233, 239, 247, 250, 254–256, 281
MA	Multiple Access	239
MDSCC	Madrid Deep Space Communication Complex	237
MER	Mars Exploration Rover A/B	38, 43, 187–190, 195
Meteo-France	Center National de Recherches Meteorologiques	281
MetOp	EUMETSAT Polar Orbiting satellites	9, 14, 42, 118, 121, 123, 124, 132, 133, 136, 139, 171
MGA	Medium Gain Antenna	222
MIT	Massachusetts Institute of Technology	281
MMS	Magnetospheric Multiscale Mission	36, 38, 43, 281
MOC	Mission Operations Center	32, 43, 180, 186, 239, 240
MOE	Mean Opinion Score	44, 101
MRN	Mars Relay Network	15, 18, 43, 179–181, 183–185, 187, 188, 193–197, 201–203, 215, 218, 219, 222, 224, 225, 227–229, 233, 281

MRO	Mars Reconnaissance Orbiter	185, 187, 188, 195, 218
MSL	Mars Science Laboratory	15, 38, 42, 184, 187–190, 195
MW	Micro-wave	133
NAM	North American Model	162
NASA	National Aeronautics and Space Administration	9, 19, 20, 22, 31, 35, 42, 46, 121–124, 136, 146, 149, 151, 153, 155, 179, 180, 189, 190, 194, 195, 231, 236–239
NCC	Network Control Center	237
NCCDS	Network Control Center Data System	239
NCEP	National Centers for Environmental Prediction	118, 281
NEEMO	NASA Extreme Environment Mission Operations	189
NEN	Near Earth Network	17, 22, 31, 170, 237–239, 241, 244, 281
NENSO	NEN Scheduling Office	238
NESDIS	National Environmental Satellite, Data, and Information Service	176
NIMO	Network Integration Management Office	238, 239, 281
NOAA	National Oceanic and Atmospheric Administration	42, 121–126, 138, 176
NPHC	New Horizons	38
NPOESS	National Polar-orbiting Operational Environmental Satellite System	10, 13, 17, 21, 36, 122, 138, 141, 142, 165, 175, 176, 178, 231, 233, 234
NSF	National Science Foundation	177
NuSTAR	Nuclear Spectroscopic Telescope Array	38
NWP	Numerical Weather Prediction	9, 14, 17, 117–119, 121, 122, 124, 127–139, 141, 143–145, 168, 170, 172, 174, 177
ODY	2001 Mars Odyssey	187
OPN	Object-Process Network	29, 30
OPS	Oxigen Production System	200
OSCAR	Observing Systems Capability Analysis and Review Tool	130–133

OSE	Observing System Experiment	119
PAO	Public Affairs Office	200, 201
POES	Polar-orbiting Operational Environmental Satellite	14, 42, 117, 122, 124, 176
PoP	Point of Presence	126
PRX	Proximity Link	18, 185, 222
RF	Radio Frequency	15, 145, 187, 188, 205, 211–213, 215, 219, 220, 223, 235, 247, 250, 252, 253, 256
SA	Single Access	15, 211–214, 218, 223, 239, 247, 249–252, 254–256
SAIC	Science Applications International Corporation	240
SAP	System Architecture Problem	30, 69
SAS	System Architecting Synthesis	27, 28
SAT	System Architecture Task	28
SCaN	Space Communications and Navigation Program	13, 19–21, 23, 151, 196
SCD	CSO Mission Small Conversion Device	240
SD	Standard Definition	201
SFCG	Space Frequency Coordination Group	184, 196
SGL	Space-to-Ground Link	214, 215
SINR	Signal-to-Interference-plus-Noise Ratio	222
SLE	Space Link Extension	34, 126, 127, 186, 240
SN	Space Network	17, 22, 31, 47, 48, 87, 124, 170, 176, 237–239, 241, 242, 281
SNR	Signal-to-Noise Ratio	222, 223
SOC	Science Operations Center	43
SOHO	Solar and Heliospheric Observatory	38, 40
SSA	S-band Single Access	239
SSO	Sun-Synchronous Orbit	123, 127, 144, 166
STEREO	Solar Terrestrial Relations Observatory A/B	38
STK	Systems ToolKit	30, 145
SWaP	Size, Weight and Power	184
SWIFT	Swift Gamma-Ray Burst Mission	38, 40

SWSI	SN Web Services	239
SysML	System Modeling Language	30
TCM	Trajectory Correction Maneuvers	43
TCP	Transmission Control Protocol	34, 64, 65, 69, 96, 240
TDRS	Tracking and Data Relay Satellite	19, 124, 177, 238, 239, 242, 247
TDRSS	Tracking and Data Relay Satellite System	14, 15, 22, 47, 125, 143, 168–170, 174, 205, 218, 238, 239, 242, 247
TERRA	TERRA Spacecraft (EOS AM-1)	38
TWTA	Traveling Wave Tube Amplifiers	196, 215
UDP	User Datagram Protocol	69, 240
UHF	Ultrahigh Frequency	184
US	United States of America	52, 117, 118, 121–124, 127, 136, 141, 150, 152, 155, 156, 158, 166, 175, 176
USN	Universal Space Network	238
VLBI	Very Long Baseline Interferometry	237
WAN	Wide Area Network	13, 21, 52, 64, 65, 69, 81, 94–96, 101, 114, 127, 128, 149, 169, 171, 175, 177, 184, 185, 193, 239
WBS	Work Breakdown Structure	145, 146, 149, 151
WOTIS	Wallops Orbital Tracking Information System	238
WSC	White Sands Complex	239