Results of the MIT Space Communication and Navigation Architecture Study

Marc Sanchez, Daniel Selva, Bruce Cameron, Edward Crawley
Massachusetts Institute of Technology
77 Massachusetts Ave 33-409
Cambridge, MA 02139
617-682-6521
{msnet,dselva,bcameron,crawley}@mit.edu

Antonios Seas, Bernie Seery
NASA Goddard Space Flight Center
8800 Greenbelt Road
Greenbelt, MD 20771
301-286-7979
{Antonios.A.Seas, Bernard.D.Seery}@nasa.gov

Abstract—**NASA is currently conducting an architecture study for the next-generation Space Communication and Navigation system. This is an extremely complex problem with a variety of options in terms of band selection (RF, from S-band to Ka-band and beyond, or optical), network type (bent-pipe, circuit-switched, or packet-switched), fractionation strategies (monolithic, mother-daughters, homogeneous fractionation), orbit and constellation design (GEO/MEO/LEO, number of planes, number of satellites per plane), and so forth. When all the combinations are considered, the size of the tradespace grows to several millions of architectures. The ability of these architectures to meet the requirements from different user communities and other stakeholders (e.g., regulators, international partners) needs to be assessed. In this context, a computational tool was developed to enable the exploration of such large space of architectures in terms of both performance and cost. A preliminary version of this tool was presented in a paper last year. This paper describes an updated version of the tool featuring a higher-fidelity, rule-based scheduling algorithm, as well as several modifications in the architecture enumeration and cost models. It also discusses the validation results for the tool using real TDRSS data, as well as the results and sensitivity analyses for several forward-looking scenarios. Particular emphasis is put on families of architectures that are of interest to NASA, namely TDRSS-like architectures, architectures based on hosted payloads, and highly distributed architectures.**

TABLE OF CONTENTS

1. INTRODUCTION

Background

The Space Communication and Navigation Program (SCaN) is currently in charge of maintaining and upgrading the different networks that provide communication services between space missions and mission control centers, namely the Near-Earth Network (NEN), the Space Network (SN), the Deep Space Network (DSN) and the NASA Integrated Services Network (NISN). The main driving requirements of the program are to develop a unified space network that can support human and robotic mission needs across the Solar System and specifically in the Earth, Moon and Mars domain. This network must evolve to provide the best feasible data rates with the current technologies and protocols, and must foster interoperability with similar systems from other US agencies and NASA partners [1].

As part of the SCaN program, the first concept studies for the SN evolution are being undertaken at NASA Goddard Space Flight Center (GSFC) under the Space Based Relay Study (SBRS). Given that the SN is currently being upgraded through the launch of TDRS K, L and M (3rd generation), the goal of SBRS is to look forward (beyond 2020) and investigate new architectures for the TDRS system that have the potential of reducing costs and boosting the network capabilities [2].

Reference [3] presented the first version of the computational tool developed at MIT as part of the SBRS. This tool intends to quickly evaluate large spaces of future candidate SN architectures both in terms of performance and cost in order to (1) identify optimal subsets of architectures in the benefit-cost space and (2) conduct fast what-if analyses to complement the SBRS effort. While the focus of [3] was on modeling space communication networks, this paper extends the work by validating the tool and presenting results obtained when exploring multiple architectural decisions.

Research Goals

The first research goal addressed by this paper is related to the development of a rule-based fast scheduling algorithm that is suitable to evaluate thousands of network architectures. In particular, this paper introduces the assumptions and limitations of the current problem formulation and provides an overview of the data structures used to model the system. It also describes the heuristics used to guide the scheduling

978-1-4799-1622-1/14/$31.00 ©2014 IEEE.

[1] IEEEAC Paper #2327, Version 3, Updated 01/12/2014.

process and how they are captured in the forms of rules.

The second area of focus of this work is the validation of the tool's performance and cost models. In order to do that, four main actions need to be performed: first, check the communication systems design module that sizes the antennas and payloads for the relay satellites. Second, compare the performance of the scheduling algorithm with TDRSS operational data so as to ensure that its inherent heuristics are representative of how the system operates. Third, validate the spacecraft design algorithm that designs the subsystems of the relay satellite based on the communication payloads it carries; and fourth, assess the accuracy of the cost estimation module that predicts the space and ground segment cost. Although all these actions have been or are currently being performed, this paper will focus exclusively on the validation of the scheduling algorithm.

Finally, the third objective of this paper is to demonstrate the tool usefulness by presenting its outputs given a predefined scenario for the 2020-2030 time period. This scenario is based on the stakeholder analysis conducted as part of this effort (also introduced in [3]) but not explicitly addressed in this paper. The results are presented in the form of two-dimensional tradespaces that use system performance and cost as the two main metrics. Additionally, the tradespaces are categorized based on the options for the considered architectural decisions. This allows identification of generic trends and shared characteristics for non-dominated architectures that can be insightful to decision makers.

Paper Structure

The reminder of this paper is organized as follows: Section 2 provides an overview of the tool along with a short description of the main components of the performance and cost models. Section 3 introduces the newly developed scheduling algorithm, its assumptions, data structures and inherent heuristics. In turn, section 4 describes the tool validation results, with especial emphasis on the benchmarking of the scheduling algorithm with real TDRSS operational data. Finally, section 5 presents the results of exercising the model and looking at different cuts of the architectural tradespace.

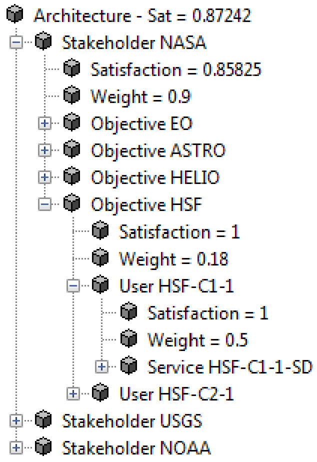
2. Model Overview

The goal of this section is to provide an overview of the architectural tool developed to analyze the performance and cost of space communication network that provide support to near-Earth satellites. It first synthesizes the set of decisions used to generate the space of candidate network architectures along with their allowed values. It then describes the inputs to the tool with respect to the demand forecast for the system, that is, how to specify what missions will be using it and what will their communication needs be. Finally, it introduces the different modules of the tool, focusing on those directly related to the computing the performance and the cost of the system.

Architectural Decisions

Table 1 lists the set of decisions used to generate different system architectures. Note that this list is an extension of what was presented in [3] in order to include architectures that use inter-satellite links, vary the number of ground stations and use fractionated spacecraft [4]. Note that disaggregation refers to distribution of relay-to-user payloads whereas fractionation refers to distribution of the space-to-ground payload.

Figure 1: Stakeholder decomposition



- Architecture - Sat = 0.87242
  - Stakeholder NASA
    - Satisfaction = 0.85825
    - Weight = 0.9
    - Objective EO
    - Objective ASTRO
    - Objective HELIO
    - Objective HSF
      - Satisfaction = 1
      - Weight = 0.18
      - User HSF-C1-1
        - Satisfaction = 1
        - Weight = 0.5
        - Service HSF-C1-1-SD
      - User HSF-C2-1
  - Stakeholder USGS
  - Stakeholder NOAA

Scenario definition

In order to evaluate the performance of a space communication network, it is necessary to understand the set of potential users of the system and their requirements. In this case, these users are near-Earth missions that connect to the relay satellites in order to return their data to Earth and receive commands from their respective mission control centers. Since the goal of this work is to identify good architectures for the 2020-2030 time frame, specifying the set of potential customers requires forecasting the missions that will be flying in the next decade (along with their communication subsystem needs) and creating a scenario that is representative of a typical day of operations for the system.

Two main inputs are required to fully define a scenario. First, a stakeholder decomposition that lists all the potential stakeholders for the network and assigns a weight to measure their relative importance. Each stakeholder has, in turn, different objectives which contain sets of missions that require the network support. Additionally, each mission has a set of demanded services (telemetry, command, science return, etc.) that capture its communication needs.

As an example, figure 1 presents a stakeholder decomposition for a NASA centric space communication network. For stakeholder NASA, four objectives are identified, each one corresponding to a major scientific community in the agency (Astrophysics, Heliophysics, Earth Observation and Human Space Flight). Additionally, the U.S. Geological Survey (USGS) and the National Oceanic and Atmospheric Administration (NOAA) are also known to use the current TDRSS through the Landsat [5] and the Polar Operational Environmental Satellites. Therefore, they are also included as potential stakeholders to the system.

At the very end of the stakeholder decomposition lay the different communication services requested by the network customers. For each service, five parameters have to be specified: number of desired contacts, duration per contact, data rate, maximum acceptable latency and minimum time between contacts. These parameters are used by the scheduling algorithm in order to compute a plausible schedule for the whole system and assess how well the network can meet a mission communication requirements.

Table 1: Architectural decisions

Decision	Range of values
Antenna selection	Any parabolic antenna supporting S-, X-, Ku-, Ka-band. Optical telescopes (1550nm)
Payload type selection	bent-pipe vs circuit-switched vs packet-switched
Payload-to-spacecraft allocation (disaggregation)	All the possible partitions of N_{instr} instruments into $1 \leq N_{sat} \leq N_{instr}$ satellites
Inter-Satellite Link payload allocation	Yes or no for each constellation of satellites
Contract modality	100% procurement, hosted payloads, or 100% commercial
Orbit selection	GEO, MEO or LEO
Constellation design	# planes: N_p # sats per plane: N_{sp}
Fractionation strategy	all-mothers vs. mother-daugther
Ground station	Subset of White Sands, Guam and a new site

Tool structure

The general structure of the tool is composed by four main parts: An STK [6] model that computes the physical information of the system for all possible configurations on the network; an architecture generator that creates all possible architectures based on the set of architectural decisions and their allowed values; an architecture *evaluator* that, given an input architecture, computes its cost and benefit; and a resource manager that parallelizes the evaluation process by allocating architectures among different *evaluators* as they become available.

The STK model performs the orbital computations for the tool. In particular, STK is used to simulate a typical day of operations of the network by propagating the movement of both the customer missions, the relay satellites and the ground stations. Additionally, it computes the contact opportunities between the different nodes of the network, be it through a direct link or through a multi-hop path, and obtains an estimate of the distances of the inter-satellite and space-to-ground links.

The architecture generator is composed by a set of rules that enumerates all possible combinations for the specified architectural decisions and outputs an array of architectures that are ready for evaluation. The formulation of these rules is generic, i.e. it uses architectural patterns that formulate the architecture generation process as a combinatorial problem [7]. As an example, the *payload-to-spacecraft allocation* decision can be viewed as an instance of a partitioning problem in which, given a predefined set of communication payloads, one is interested in finding the best grouping into satellites (all in a unique spacecraft, all distributed in different spacecrafts flying as a cluster, or combination of these options). Note also that in some cases the decisions are not independent but rather tightly coupled. For instance, the *payload-to-spacecraft allocation* decision can be preceded by a *payload band selection* problem in which a subset of communication payloads first are being selected and then allocated into the corresponding satellites.

The architecture *evaluator* takes as input one architecture and obtains an estimate of its benefit and cost. In order to do that, the following high level process is followed: First, the architecture is decomposed into its constituents (constellations, clusters, satellites, antennas, payloads) and their properties (e.g. antenna diameter, payload frequency band, etc.) are imported from a database. Next, the information of the system decomposition is used along with the outputs of the STK model in order to compute a possible schedule for the network and design the different types of relay satellites. Once this has been done, the last step is to compute the benefit and cost of the overall system. For the former, the satisfaction of a mission is assessed by comparing its desired communication requirements with what the system has granted him through the schedule. This satisfaction is then aggregated to a single metric through the stakeholder decomposition previously presented (figure 1). On the other hand, the cost of the network is obtained by computing an estimate of the space and ground segment cost, based on the design of the different relay satellites and the building and operations cost for the network ground stations.

Finally, the last part of the tool is a resource manager that enables parallel computing. Given an array of architectures produced by the architecture generator, the resource manager creates an independent thread for each and queues them for evaluation. It then creates a pool of architecture *evaluators* and progressively wakes the threads as *evaluators* in the pool become free.

The architecture evaluator

*Overview of performance model—*The performance of the satellite communication network is assessed based on the outputs of the scheduling algorithm (a detailed description is presented in section 3). For each service demanded by a mission three metrics are computed: number of granted contacts, maximum latency and user burden.

The number of granted contacts indicates how many times the customer mission connects with a relay satellite to download and/or upload information. Since all contacts for a particular service and mission have the same nominal data rate and duration, computing the number of granted contacts is effectively equivalent to computing the fraction of the data volume returned to Earth (scaled by a factor).

$$\%\text{ returned DV} = \frac{DV_{ret}}{DV_{des}} = \frac{R_b \cdot T_c \cdot \#GC}{R_b \cdot T_c \cdot \#DC} = \frac{\#GC}{\#DC} \quad (1)$$

where DV_{ret} is the returned data volume, DV_{tot} is the total data volume to return, R_b is the contact nominal data rate, T_c is the contact duration, $\#GC$ is the number of granted contacts and $\#DC$ is the number of desired contacts.

On the other hand, the maximum latency computes the maximum amount of time that a mission will have to hold its collected data in the on-board memory before being able to download it. This metric is important in some applications where the scientific information from the satellites is used to obtain periodic data products (e.g., images from weather satellites might be needed every three hours to update weather predictions and issue alerts if necessary). Similarly, this metric can be important for mission controllers that do not want satellite operating without supervision for long periods of time. As a worst case estimate, this maximum latency is computed as the maximum time between consecutive contacts.

Finally, the user burden is used as a proxy for how difficult it is for a mission to connect to the network. This burden is measured in terms of the *Equivalent Isotropically Radiated Power* (EIRP) that the communication subsystem of customer mission has to provide in order to close the link with the relay satellite. It is assumed that this link will always operate at nominal data rate so that if all contacts get scheduled, then the returned data volume will equal the desired daily data volume for the mission.

Once these three metrics have been computed, the satisfaction of a particular service can be directly assessed by comparing their values to a set of step functions that capture each missions preferences. For instance, a human space flight mission might heavily value not dropping contacts because continuous communication with the astronauts is essential to the mission safety. Therefore, if a contact is lost then its satisfaction might be 20% (in a normalized 0 to 100% scale) or even almost null. Alternatively, a small astrophysics mission might not consider critical to have all its desired contacts scheduled since the scientific data it gathers is not time sensitive. Thus, if a contact is lost then only a 5% of the satisfaction is lost.

Overview of cost model—The cost model provides a rough estimate of the lifecycle cost of the architecture. The model distinguishes between recurring and non-recurring cost, and includes payload cost, bus cost, launch cost, operations cost, ground segment cost, and service fees. The obtained estimate depends on the contract modality of the architecture. For 100% procurement architectures, payload, bus, launch, operations and ground segment cost, both non-recurring and recurring, are paid by NASA. No service fees are incurred.

For 100% hosted payloads, the tool uses two different pricing models taken from [8], namely "pay per mass" and "resource pricing." The pay per mass pricing model includes systems engineering, integration and testing cost which is a fraction of the total bus and payload cost, and then insurance cost, operator cost, and a certain profit are added as constant percentages that get compounded to yield the final estimate. The resource pricing model takes into account everything in the pay per mass mode, plus a marginal bus and launch cost. These delta costs are based on the fraction of resources (mass, power) of the host spacecraft that are used by the hosted payload. Lastly, 100% commercial services costs are computed at a fixed price per MB of data.

Payload cost estimates include the cost of the antennas and the cost of all electronics. They are based on Cost Estimating Relationships (CER) provided in [9]. Bus cost is also estimated using CER from [9]. These CER provide estimates of recurring and non-recurring cost at the subsystem level, and therefore require subsystem masses as an input. Subsystem masses are computed by an iterative spacecraft design algorithm that sizes the bus subsystems based on payload requirements and orbital considerations.

On the other hand, launch cost is computed as the total required number of launches times the price of one launch for a set of launch vehicles, and the less costly option is selected. The number of launches required is calculated by taking into account the mass and dimensions of the spacecraft, as well as the orbit, and the performance envelope of each launch vehicle. Operations cost represents the cost of operating the relay satellites, and it is simply computed as a fixed number times the number of relays in the constellation. Finally, ground segment cost includes the cost of constructing a new ground station when necessary, plus the recurring maintenance and operations costs.

3. Scheduling Algorithm

The goal of this section is to explain the scope, assumptions and heuristics inherent to the rule-based scheduling algorithm developed for rapid evaluation of space communication networks.

Assumptions

The first and most important assumption for the proposed rule-based scheduling algorithm is that of non-optimality. This assumption is grounded in two facts: First, the scheduling process is done following a greedy approach, i.e. contacts are scheduled whenever it is possible regardless of future states of the network. Second, when more than one network resource can serve a contact, deciding which one to chose is done based on a set of heuristic rules.

On the other hand, it is also important to note that the proposed scheduling algorithm only addresses the problem of the access network, that is, the allocation of resources between network customers and network nodes. This simplification is done in order to facilitate the scheduling process and decouple it from the intricacies of the routing algorithm implemented in the backbone network.

Finally, the third assumption for the scheduling algorithm is related to its inputs. It is assumed that all the required information will be available at the start of the process and no numeric computations (e.g. propagating satellites) will have to be done inline with the scheduling process. This fact is further explored in the following subsection.

High-level Structure of the Scheduling Algorithm

The inputs of the scheduling algorithm come from (1) the set of users that want to connect with the network and their relative priorities, (2) the contact opportunities between them and the relay satellites, and (3) the contact opportunities between these satellites and the system ground stations. While (1) is an input to the tool based on demand forecasts, (2) and (3) are dependent on the structure of the network (i.e. where relay satellites are placed and what communication payloads they carry) and involve computing the motion of the satellites over a time period (typically one day).

Two strategies for computing the contact opportunities are possible: Performing the orbital computations inline with the evaluation of the network architectures, or pre-computing the contact opportunities and then tailoring this information to the specific implementation of the network architecture. In the first strategy, the orbital and *line of sight* computations are performed each time that an architecture has to be evaluated. This is easiest approach since the network is fully defined and therefore there is no ambiguity with respect to the satellites and payloads that are placed into orbit. Nevertheless, it also limits the computational performance of the tradespace exploration tool because it increases the time required to evaluate one architecture.

On the other hand, the second strategy decouples the problem of computing contact opportunities from the actual physical implementation of the network. In this approach, contact opportunities are only calculated between orbital positions. These orbital positions can contain either one satellite or a set of satellites with multiple communication payloads each. The main advantage of this apporach is that the orbital positions for the network nodes are a fixed input and do not depend on the architectural decisions. Being that the case, one can first calculate contact opportunities between all possible orbital positions and then replicate or eliminate them as needed, depending on the details of the architecture under evaluation.

Once the contact opportunities between the network customers and the relay satellites have been computed, the next step is to understand which of them are valid given the system characteristics. For instance, a contact from a bent-pipe architecture will only be valid if the relay satellite is at the same time connected to a ground station. Alternatively, a packet-switched architecture with on-board storage can use a store and forward mode of operation where it first communicates with a customer, then stores temporarily the information and finally dumps it to a ground station.

Taking into account these considerations is done by trimming the customer-to-relay contact opportunities (from now on *visibility windows*) based on the relay-to-ground-station contact opportunities (from now on *paths*, since they might require multi-hop communication through inter-satellite links) and the network type under consideration. Note that this process cannot be pre-computed since it requires information on the architecture being evaluated, but it can be optimized by also pre-computing the paths with and without inter-satellite links and then creating rules that trim the visibility windows' starting and ending times accordingly.

Once this process is complete, the scheduling algorithm is ready to start allocating network resources to customers. It now has a complete description of the access network through the visibility windows, and it has also captured the first order effects of the type of backbone network. Next, the scheduling algorithm has to decide the order with which contacts will be served. This is done by assigning a priority to each contact (from now on *required contact*). The details of the heuristics used for contact prioritization are explained later in this section.

Finally, the last step of the scheduling algorithm is to compare the *visibility windows* with the *required contacts* until there are no more contacts to schedule or there are no more free resources on the network. This comparison has to take into account the following factors:

1. Frequency band compatibility: A visibility window is suitable for scheduling a contact if it uses the same frequency band (or optical wavelength) than the customer mission.
2. Data rate: A visibility window is suitable for scheduling a contact if the nominal data rate it can support is higher than the requested data rate by the customer mission.
3. Duration: A visibility window is suitable for scheduling a contact if its duration is longer than that of the contact, or if it can be concatenated with other visibility windows to achieve an overall duration longer than that of the contact.
4. Network status: A visibility window is suitable for scheduling a contact only if its state (starting and ending times) has been updated based on the past scheduled contacts.

Note that (1) and (2) are not tied to the physical geometry of the system but rather to the communication payloads that are carried by the customers and relay satellites. Alternatively, (3) depends primarily on the satellites' motion and captures the idea that the system performs handovers as the customer loses support from a relay and then connects to another one. The specific heuristics used for selecting the best visibility window and performing handovers is explained later in this section. Lastly, (4) takes into account the evolution of the network resources as contacts are progressively scheduled. In particular, once a contact is allocated it is mandatory to update the status of the visibility windows so that they are accordingly trimmed or eliminated.

Data Structures

Based on the description from the previous subsection, four main data structures are required in order to capture the state of the network and perform the scheduling process: *visibility-window*, *path*, *required-contact* and *customer*. In a rule-based expert system these data structures come in the form of *facts*, i.e. a container of any sort of information. As an example, a fact ”car“ can contain multiple properties (slots) such as color, mass, power or make.

Table 2 lists the four main facts of the scheduling algorithm along with their slots. It can be seen that the *visibility-windows* and *paths* have a similar structure: an origin node (a customer for the first and a satellite for the second), a destination node (a satellite vs. a ground-station), a starting time and an ending time. Note also that they have the slot ”orbital-position“ that is used to pre-compute these facts. The *visibility-windows* also include information regarding the band and nominal-data-rate so that it is easily accessible when determining their suitability to service a particular contact. Similarly, a *path* has a slot ”ISL“ that is used as proxy for multi-hop or single-hop contact between a relay and a ground station. As previously stated, if the architecture under evaluation does not support ISLs, then the contact opportunities will be trimmed according to paths computed without ISLs.

On the other hand, the facts *required-contact* contain two different types of data: information regarding the characterization of the contact such as the user requesting it, the service, expected duration, priority, frequency band and nominal data rates. On the other hand, it also contains information on how well the contact has been served by the network.

Finally, the *customers* facts keep track of the temporal evolution of the customer satellites during the scheduling process. All satellites start with their slot ”time“ at 0 and progressively increase its value as contacts get scheduled. If at a particular instant in time the network cannot provide access to the customer, then the ”time“ slot is increased $n \cdot \Delta t, n > 0$ until the access can be granted or the scheduling algorithm reaches its simulation end. Note also that this process is repeated for each customer and service, i.e. a satellite will be able to independently schedule contacts used for telemetry and science data return.

Table 2: Scheduling Algorithm Facts

Fact Name	Fact Slots
Visibility-Window	id, customer, satellite, orbital-position, tStart, tEnd, band, data-rate
Path	id, satellite, orbital-position, ground-station, ISL, tStart, tEnd
Required-Contact	id, user, service, priority, duration, band, nominal-data-rate, satisfied, tStart, tEnd, data-volume
Customer	id, service, time

Heuristics

This subsection presents a description of the different heuristics inherent to the scheduling algorithm. They have been categorized into two main areas: heuristics for customer prioritization and heuristics for visibility window selection and concatenation.

Customer prioritization— As previously stated, in order to build a schedule that maximizes the system benefit it is important to prioritize the customers and the services they require. In this case, this is done through the following heuristic:

$$Priority = M \cdot \textit{mission class} + \textit{service priority} \quad (2)$$

The *mission class* can take values $1, 2, 3, 4, 5$ (5 being maximum priority) is used to capture the idea that some missions will always have higher priority due to specific circumstances. For instance, a human space flight mission will always have more priority than any robotic mission since astronaut lives are at stake. Alternatively, *service priority* captures the importance of the the mission and service in the stakeholder-objective-mission-service decomposition used to compute the benefit of the system. This metric is included so that the computed scheduled is maximizing the benefit score of the network. Finally, M is a control parameter that captures the relative importance between *mission class* and *service priority*. It is typically set to high value (e.g., 10000) to ensure that *mission class* is always the most important factor.

Visibility window selection and concatenation—Let's assume that a *required contact* has been selected as the next one to be served based on the prioritization. When tackling the problem of scheduling it based on the available *visibility windows*, three questions arise:

1. Are there any restrictions that limit the time at which the contact should be scheduled?
2. If there is more than one visibility window that can completely overlap a contact, which one has to be chosen?
3. If there is more than one visibility window that can partially overlap a contact and none that can fully serve it, which one has to be chosen for concatenation?

The answer to the first question is, in most cases, yes. Missions tend to prefer scheduling contacts when they have significant amounts of information to dump, or when their batteries are almost fully charged. Even operational constraints might arise, with contacts being scheduled preferably during the normal working hours of the staff at the mission control center. As a result, most schedulers allow customers to provide their desired contact times along with a flexibility window over which the event should be preferably scheduled [10].

Despite the importance of these factors in real-life operations of satellite communication networks, they are currently disregarded in the current scheduling algorithm. The rationale for this is as twofold: On the one hand, the goal of the scheduler is to assess the overall capacity of network and capture the effect having extra satellites and communication payloads in orbit. The authors realize that having extra scheduling constraints reduces the flexibility of allocating the network resources, thus potentially reducing the overall data flows through the network. However, they also assume this fact to be a second order effect compared to extra amount of resources that become available if more relay satellites are put into orbit. On the other hand, the scheduling process for the tool is based on a demand forecast beyond the 2020 time frame. This implies high levels of uncertainty both on the number and types of missions that will be flying, as well as on the operational aspects of the services they will demand. Therefore, taking into account scheduling constraints will be of limited usefulness since they will typically be unknown and difficult to predict from existing missions.

Alternatively, answering questions (2) and (3) is a complex problem that requires taking into account the duration of the *visibility windows* and how they can be concatenated in order to provide long periods of mission support. In particular, the following distinction is made: A *visibility window* is said to *completely overlap* with a contact if its duration duration d_{wnd} satisfies equation 3, where d_{con} is the contact duration. If this inequality does not hold, then the visibility window is said to *partially overlap* with the contact and requires concatenating multiple visibility windows until complete overlap is achieved. This happens when equation 4 becomes true, where d'_{wnd_i} represents the duration of the *i*-th concatenated *visibility window*, trimmed accordingly with of the previous selected windows $d'_{wnd_{j-1}}$.

$$d_{wnd} >= d_{con} \quad (3)$$

$$\sum_{i=1}^{n} d'_{wnd_i} >= d_{con} \quad (4)$$

Distinguishing between *partial* and *complete overlap* is important because it provides the condition to identify when a *required contact* can be immediately fully satisfied served versus the problem of using handovers to provide continuous support. As it will now be explained, these two situations entail different heuristics.

Partial vs. complete overlap heuristics—Having identified the difference between *partial* and *complete overlap*, the next step is to identify the temporal conditions that characterize both situations.

Let t_0 be the time at which a customer wants to schedule a contact, d the duration of the contact, and t_{s_i} and t_{e_i} the starting and ending times of the *i*-th visibility window. Then, a *partial overlap* exists if equation 5 holds, while *complete overlap* requires equations 5 and 6 to be true at the same time.

$$t_{s_i} \leq t_0 < t_{e_i} \tag{5}$$

$$d \leq t_{e_i} - t_0 \tag{6}$$

Note also that equation 5 is a necessary but not sufficient condition for *partial overlap*. That being the case, two strategies can be used to identify a *partial overlap*: Look for all the visibility windows that satisfy condition 5 and dissatisfy condition 6; or eliminate all visibility windows that satisfy conditions 5 and 6 at the same time.

Once the conditions for *complete* and *partial overlap* are understood, the final and most important step is to define the heuristics that will chose the best visibility window among the set that satisfy equations 5 and 6 (dissatisfy in the latter case). For *complete* overlap, the following heuristic will be used: *The best visibility window is the one that minimizes the non-overlapping time with the contact*. This is mathematically expressed in equation 7, where $(t_0 - t_{s_i})$ expresses the non-overlapping time between the start of the visibility window and the start of the contact, and $(t_{e_i} - (t_0 + d))$ accounts for the non-overlapping time between the end of the visibility window and the end of the contact.

$$\begin{gathered}(t_0 - t_{s_i}) + (t_{e_i} - (t_0 + d)) < \\ \left(t_0 - t_{s_j}\right) + \left(t_{e_j} - (t_0 + d)\right)\end{gathered} \quad \forall j, i \neq j \tag{7}$$

Simplifying equation 7 is a straightforward process that leads to equation 8, which simply states that the best visibility window is the shortest one. Recall, however, that equation 6 must also be true for *complete overlap*, thus indicating that the selected window will be the one with duration closest to the contact duration.

$$t_{e_i} - t_{s_i} < t_{e_j} - t_{s_j} \forall j, i \neq j \tag{8}$$

The other heuristic that needs to be explored is the one used to select the visibility window to concatenate when the contact is under a *partial overlap* situation. In this case, the following heuristic will be used: *The best visibility window is the one with longer duration and shorter non-overlapping time between the start of the contact and the start of the visibility window*. This is mathematically expressed in equation 9, where $((t_0 + d) - t_{e_i})$ is the time between the ending of the contact and the ending of the visibility window, and $(t_0 - t_{s_i})$ is the non-overlapping time between the start of the contact and the start of the visibility window. As previously demonstrated with equations 7 and 8, equation 9 can be simplified to more compact and manageable form - equation 10.

$$\begin{gathered}((t_0 + d) - t_{e_i}) + (t_0 - t_{s_i}) < \\ \left((t_0 + d) - t_{e_j}\right) + \left(t_0 - t_{s_j}\right)\end{gathered} \quad \forall j, i \neq j \tag{9}$$

$$t_{s_j} - t_{s_i} < t_{e_i} - t_{e_j} \forall j, i \neq j \tag{10}$$

Table 3 summarizes all the heuristics presented in this section and formulates them in the form of *if-then* statements that can be directly input into the rule-based scheduling algorithm.

4. BENCHMARKING

The goal of this section is to describe the validation process for the scheduling algorithm by comparing its outputs with real TDRSS operational data.

Dataset Description

The Network Control Center Data System (NCCDS) is the network control facility for the TDRSS. Its scheduling activities include mission planning, event scheduling, scheduling conflicts and support scheduling among others [11]. Although it is located in White Sands NM, close to the White Sands Ground Terminal (WSGT) and Second TDRSS Ground Terminal (STGT), it controls the scheduling process for the entire network including the Guam Remote Ground Terminal (GRGT) and its supported satellites. The NCCDS interfaces with the mission operations centers (MOCs) and responds to their requests by either granting events or mediating in case of scheduling conflicts.

An extract of the NCCDS database used to schedule the TDRSS was obtained and analyzed. It contained on twelve days of real TDRSS operational data, containing information regarding the number and types of events scheduled each day, the satellites and antennas through which they were served, and the supported data rates for each of them. The lessons learned from the analysis of this dataset (total scheduled time, total data volume, antenna utilization, etc.) are not disclosed on this paper due to non-disclosure agreement (NDA) restrictions. For the same reason, the results for the validation case will only present percentages of error with respect to the real TDRSS performance.

Validation Methodology

The proposed validation strategy for the scheduling algorithm undertakes the following steps:

1. Define aggregate metrics that capture the performance of the overall TDRSS scheduling system.
2. Examine the validation data in order to characterize the TDRSS network and identify generic trends and typical days of operation.
3. Define representative mission scenarios that can be used as inputs to the tool.
4. Benchmark the results from the rule-based scheduler with the real TDRSS operational data based on the metrics identified in (1).

The goal of step (1) is to define a set of metrics that capture the load of the network as a whole, rather than understanding the amount of information that flows through the different satellites. The rationale for this approach follows from the initial intent of having a scheduling algorithm as part of this tool, i.e. to understand the global network capacity and the effect of putting new relay satellites in orbit. Two metrics were identified as useful: Total data volume DV_{total} and percentage utilization $\%Ut$.

The total data volume is used as a summary of how many contacts can be scheduled through the network. The available TDRSS dataset contains information on the nominal data rate at which the scheduled contacts run, the same information

Table 3: Scheduling Algorithm Heuristics

Heuristic	Conditions	Outputs
Prioritization	$\exists$ *req-con* $\mid$ priority $= null$	priority = M $\cdot$ *mission class* $+$ *service priority*
Complete overlap	$\exists$ *vis-wnd i* $\mid t_{s_j} \leq t_0 < t_{e_j}$ AND $d \leq t_{e_j} - t_0$ $\nexists$ *vis-wnd j* $\mid t_{s_i} \leq t_0 < t_{e_i}$ AND $d \leq t_{e_i} - t_0$ AND $t_{e_j} - t_{s_j} < t_{e_i} - t_{s_i}\ i \neq j$	Select *i*-th vis-wnd
Partial overlap	$\nexists$ *vis-wnd k* $\mid t_{s_k} \leq t_0 < t_{e_k}$ AND $d \leq t_{e_k} - t_0$ $\exists$ *vis-wnd i* $\mid t_{s_j} \leq t_0 < t_{e_j}$ $\nexists$ *vis-wnd j* $\mid t_{s_i} \leq t_0 < t_{e_i}$ AND $t_{s_i} - t_{s_j} < t_{e_j} - t_{e_i}\ i \neq j$	Select *i*-th vis-wnd

that can be used as an input for the model scenario. Therefore, the data volume can be simply be computed as

$$DV_{total} = \sum_{i=1}^{N} \sum_{j=1}^{M_i} DV_{ij},\ DV_{ij} = R_{b_{ij}} \cdot T_{c_{ij}} \qquad (11)$$

where N is the number customers for the network, M_i is the number of contacts the *i*-th customer wants to schedule, and $R_{b_{ij}}$ and $T_{c_{ij}}$ are the data rate and contact duration for the *j*-th contact of the *i*-th users. It is immediate to see the if the rule-based scheduler drops contacts due to lack of network resources, the total data volume DV_{total} metric will be penalized.

Alternatively, the percentage of utilization ($\%Ut$) measures how much time a payload, antenna or satellite has been active and serving a customer. The percentage of utilization is defined as follows:

$$\%Ut_{pay_i} = \frac{\cup_{p=1}^{P} C_{i_p}}{86400} \qquad (12)$$

$$\%Ut_{ant_j} = \frac{\cup a = 1^{A} C_{j_a}}{86400} \qquad (13)$$

$$\%Ut_{sat_k} = \frac{\cup s = 1^{S} C_{k_s}}{86400} \qquad (14)$$

where the operator $\cup$ indicates that all contacts served through that element are placed in a time axis and consolidated into single contacts when more than one overlap. Therefore, $\cup_{p=1}^{N} C_{i_p}$ indicates that all contacts C_{i_p} routed through the *i*-th payload are consolidated. Note that for validation purposes only $\%Ut_{ant}$ is really meaningful since the antennas are the effective network resource that the scheduler allocates. In other words, if an antenna is pointed to a customer it will not be able to serve another mission, therefore decreasing the pool of available resources. In contrast, if a contact is served through a particular satellite that does not imply that this same satellite cannot be serving another contact at the same time.

Once the two metrics for validation were identified, the next step is to define representative mission scenarios for the TDRSS operations. After analyzing the twelve available days, two options were selected: A typical day of operations, and a high load day. A day is considered to fall under the high load category if the TDRSS network is relaying a total data volume greater than one standard deviation. Finally, once these scenarios are selected, the rule-based scheduler can be used to generate an automatic distribution of contacts and compare both schedules.

Table 4: Benchmark results for a typical day of operations

Relative Error	$\%Ant_{ut}$	DV_{total}
Satellite	17.94%	5.96%
Antenna	7.19%	6.07%
S-band	55.25%	58.30%
Ku-band	6.08%	7.38%

Table 5: Benchmark results for the high load scenario

Relative Error	$\%Ant_{ut}$	DV_{total}
Satellite	20.44%	7.46%
Antenna	3.38%	6.07%
S-band	54.74%	43.26%
Ku-band	6.55%	4.70%

Validation Results and Discussion

Tables 4 and 5 present the relative error for $\%Ut$ and DV_{total} when comparing the two aggregate metrics for the real TDRSS schedule vs. the rule-based schedule. Results indicate that the rule-based scheduling algorithm is able to replicate the TDRSS schedule within less than a 10% error in all cases except for the satellite percentage of utilization, and S-band total data volume and percentage of antenna utilization. This is due to the following facts.

First, since the rule-based scheduler heuristics do not take into consideration whether antennas of the same satellite should or should not be activated simultaneously, there is no guarantee that its solution will match that of a real TDRSS schedule.

Second, the TDRSS scheduler allows customer missions to schedule multiple events at the same time as long as they operate through the same antenna and different frequency bands. In other words, missions typically schedule a Ku-band event overlapped with an S-band event so that science data, telemetry and commands are transmitted simultaneously. This feature is currently not implemented on the rule-based scheduler. As a result when modeling this type of contacts it is assumed that only the Ku-band link is active since it drives the amount of data being sent over the contact. This modeling inaccuracy results in big disparities for the S-band payload both in terms of $\%Ut$ and DV_{total}.

That said, it is important to note that the results in tables 4 and 5 do validate the scheduler. The two discrepancies previously identified are due to the how the scheduler models contacts. In contrast, it can be seen that both the total data volume for the network and the percentage of utilization of the antennas is accurate to less than 10%. This relative error is in fact accountable to exogenous sources of error during the validation process, i.e. missing information from the TDRSS dataset. As an example, the TDRSS schedule did not contain the names of the missions that the network is currently supporting, thus leaving the orbital parameters for the missions as an undesired degree of freedom.

5. Results

Architectural decisions

Table 6 lists the decisions included in the results herein presented with their set of allowed values. They create a tradespace of 4,450 architectures that was effectively reduced to 1,440 with two constraints: A maximum of 2 antennas can be placed on the same satellite; and a maximum of 9 satellites can be put in orbit. The first constraint is included in order to prevent complex satellite configurations with more than 2 high gain parabolic dishes. The second antenna is included to reduce the size of the tradespace to a manageable set and prevent highly disaggregated architectures (each communication payload flying separately) that should be studied together with fractionation options (not addressed in this paper).

Note the large span of architectures being explored. On the one hand, TDRSS-like architectures are reproduced as an instance of constellations of monolithic satellites procured and then operated by NASA. On the other hand, fully hosted payload architectures are considered, with one or two antennas being hosted. Finally, architectures that disaggregate TDRSS-like multi-band payloads into a low data rate payload and a high data rate payload are also analyzed (see figure 2). This becomes interesting due to the non-trivial couplings between the decisions. For instance, is it better to host a low data rate payload as opposed to tri-band TDRSS-like payload? Does having separate low and high data rate antenna provide greater network capacity due to greater scheduling flexibility?

Scenario description

As previously stated, the generation of a scenario for the 2020-2030 time frame implies forecasting the types and number of missions, and specifying their communication requirements. For this study, it is assumed that the number of missions stays approximately constant as compared to what the TDRSS is currently supporting through its *single access* services. It is also assumed that the missions use similar concept of operations as today and therefore the total requested scheduled time is on the same order of magnitude. However, in order to accommodate for the higher data volumes that missions are expected to require, the contact data rates are increased from the current hundreds of Mbps to more than one Gbps.

The resulting scenario consists of a set of 15 missions from NASA, NOAA and USGS. The NASA Earth observation community has eight missions using the network, 2 of them being major drivers (e.g. a DESDYNI class mission [12]), 2 more being medium sized missions and 4 more being smallsats. A mission becomes a driver for the network if its required data rate at least doubles the capabilities of the current TDRSS and the total desired data volume is greater than 10Tbit/day. On the other hand, the NASA astrophysics and heliophysics community are represented through 4 missions, one of which is considered to be a driver and the other three are medium sized missions. These medium sized missions typically return $\sim$1Tbit/day.

Similarly, the human spaceflight stakeholder is considered to have 2 missions in orbit, one similar to the International Space Station (ISS) and another one representative of a Multi-Purpose Crew Vehicle (MPCV) capsule. The ISS-like is a driver both in terms of required scheduled time (continuous contact throughout the day is required) and data volume. In turn, the MPCV-like missions only requires support for a long continuous periods of time ($\sim$5 hours). Finally, the NOAA and USGS mission are also included in the scenario, but are expected to be users of the system. In this case, it is assumed that they will use the network for contingency situations and therefore will schedule a maximum of two 5 to 10 minute contacts.

For each of the described mission, at least two types of service are included, a science return service and a telemetry tracking and command (TT&C) service. Additionally, the human spaceflight missions also have high definition (HD) video services required. As a result, the mission scenario has a total of 213 required contacts per day that account for 115 hours of total schedulable time and a total data volume of 130Tbit/day.

Payload band selection and allocation

The payload band selection and allocation problem looks at what are the best frequency bands to be included in the system and how to best allocate them in relay satellites. In order to provide a stable comparison, this study assumes that all satellites are procured and then operated by NASA. Figure 3 presents the obtained tradespace and flags the set of architectures that achieve the best performance for a given total life cycle cost (LCC). These architectures have the following characteristics:

1. Monolithic architecture with one single access antenna (SA) per satellite supporting only S-band
2. Monolithic architecture with one SA per satellite supporting S, Ku and Ka-bands simultaneously
3. Monolithic architecture with two SA per satellite, one supporting S-band and the other one supporting Ku and Ka-band.
4. Monolithic architecture with two SA per satellite, one supporting Ku and Ka-band and the other one supporting S, Ku and Ka-band simultaneously.
5. Disaggregated architecture with 3 satellites carrying two SA antennas, one supporting S-band and the other one supporting Ku and Ka-band, and 3 more satellites carrying two SA antennas, one supporting Ku and Ka-band and the other one supporting optical communications.

Based on the analysis of the overall tradespace, the following conclusions can be drawn. First, an all optical architecture achieves a very low score in benefit (0.1 approximately) but is potentially less expensive than most only RF and RF/optical architectures. This is because the mission scenario used for this simulation only included two missions with data rates high enough ($>$1Gbps) that they required optical communications. Second, the incremental value of having S-band for TT&C services is approximately 40% of the benefit. Similarly, having K-band for science data return services provides an extra 50 to 55% of the benefit. Third, the current TDRSS architecture is able to provide approximately 84% of the total benefit, although the same level of performance can be obtained with the less costly architecture 4 (of the previous list). Finally, architectures that obtain maximum benefit sup-

Figure 2: Tradespace definition

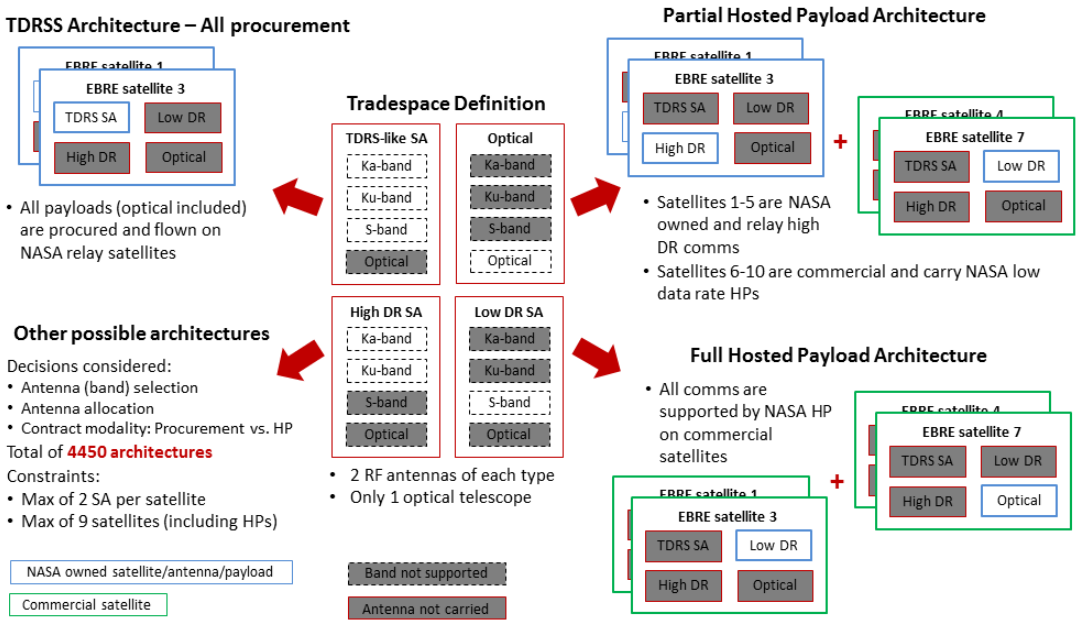

TDRSS Architecture – All procurement
EBRE satellite 1
EBRE satellite 3
TDRS SA
Low DR
High DR
Optical
• All payloads (optical included) are procured and flown on NASA relay satellites
Tradespace Definition
TDRS-like SA
Ka-band
Ku-band
S-band
Optical
Optical
Ka-band
Ku-band
S-band
Optical
High DR SA
Ka-band
Ku-band
S-band
Optical
Low DR SA
Ka-band
Ku-band
S-band
Optical
• 2 RF antennas of each type
• Only 1 optical telescope
Partial Hosted Payload Architecture
EBRE satellite 1
EBRE satellite 3
TDRS SA
Low DR
High DR
Optical
+
EBRE satellite 4
EBRE satellite 7
TDRS SA
Low DR
High DR
Optical
• Satellites 1-5 are NASA owned and relay high DR comms
• Satellites 6-10 are commercial and carry NASA low data rate HPs
Other possible architectures
• Decisions considered:
• Antenna (band) selection
• Antenna allocation
• Contract modality: Procurement vs. HP
• Total of 4450 architectures
• Constraints:
• Max of 2 SA per satellite
• Max of 9 satellites (including HPs)
Full Hosted Payload Architecture
• All comms are supported by NASA HP on commercial satellites
EBRE satellite 4
EBRE satellite 7
TDRS SA
Low DR
High DR
Optical
+
EBRE satellite 1
EBRE satellite 3
TDRS SA
Low DR
High DR
Optical
NASA owned satellite/antenna/payload
Commercial satellite
Band not supported
Antenna not carried

Table 6: Architectural decisions and values

Decision	Range of values
Antenna selection	TDRS-like SA (x2), High Data Rate SA (x2), Low Data Rate SA (x2), Optical
Payload type selection	bent-pipe
Payload-to-spacecraft allocation	All the possible partitions of N_{instr} instruments into $1 \leq N_{sat} \leq N_{instr}$ satellites
ISL payload allocation	No
Contract modality	100% procurement, hosted payloads
Orbit selection	GEO
Constellation design	# planes: $N_p = 1$ # sats per plane: $N_{sp} = 3$
Fractionation strategy	all-mothers (no fractionated spacecraft)
Ground station	White Sands, Guam

port all RF bands and also provide optical communications. Nevertheless, the addition of the optical payload in the system results in disaggregation of the monolithic satellites which, in turn, causes a significant increase in the overall system cost (25% approximately).

Figure 4 shows the same tradespace but color coded according the how communication payloads are allocated into relay satellites. Since the network constellation design is fixed (the geosynchronous plane, with three evenly spaced orbital positions), the number of relay satellites in the system is a multiple of three. If only three satellites are in orbit, then the architecture is monolithic and all communication payloads are put together in a single spacecraft. The results indicate that monolithic architectures are, in general, better from a cost perspective since the cost of launching more satellites outweighs the savings of reduced mass, power and volume for the satellites. Nevertheless, better performance can be achieved if the TDRSS-like antennas (supporting simultaneously S, Ku and Ka-band) are broken into separate low data rate (S-band) and high data rate antennas (Ku and Ka-band).

Figure 3: Band Selection

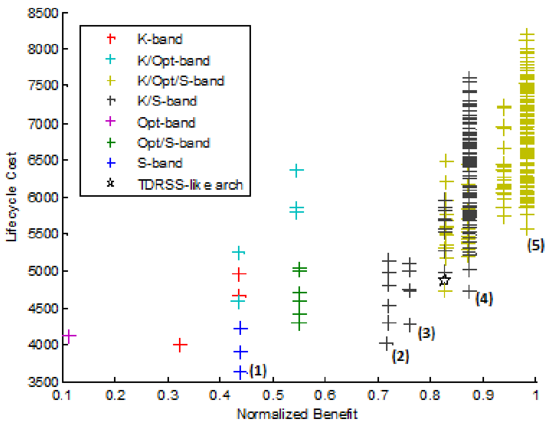


Figure 4: Payload Allocation

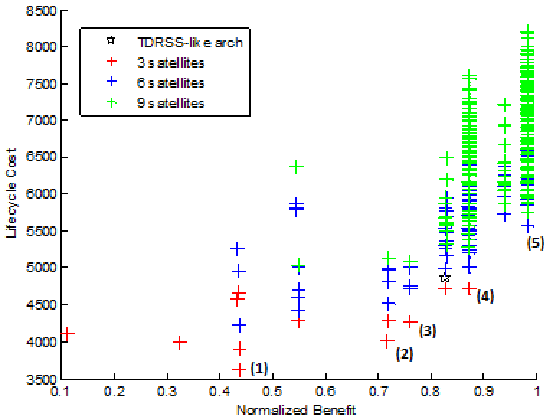
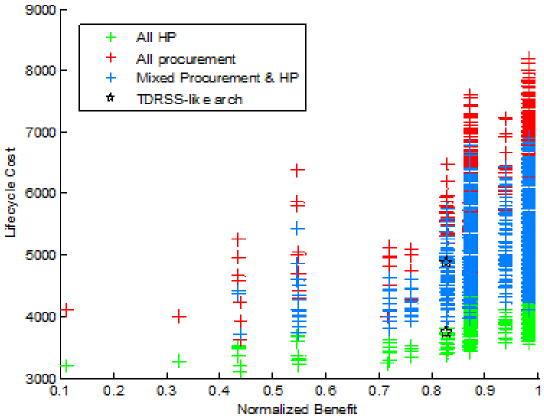


Figure 5: Procurement vs. Hosted Payloads



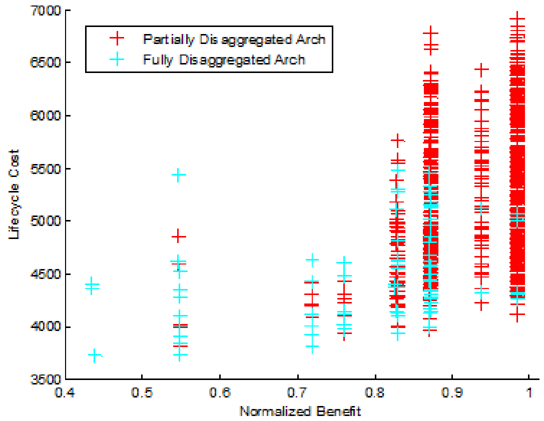
Hosted payloads vs 100% procurement

This case study focuses on the impact of including hosted payloads as part of the system. Figure 5 presents an extended representation of the tradespace where hosted payload architectures are included. The first immediate conclusion that can be drawn is that, based on the cost model from [8], hosted payloads offer the possibility of significant cost reductions (between 15% and 30%). As a result, all architectures on the Pareto front rely on hosted payload architectures.

On the other hand, concerns about using hosted payload assets for critical applications like astronaut support have been flagged as a major caveat. In this case, an optimal solution would entail using a portfolio of hosted payloads and privately owned satellites, and schedule the communications through ones or the others according to operational constraints. Therefore, building this portfolio can be done by analyzing what communication payloads are better suited for hosted payload configurations.

Figure 6 presents the tradespace of architectures that mix hosted payload and procured assets. Results indicate that including hosted payload assets in the system does not have a major impact on its performance but requires disaggregating the monolithic satellites identified as optimal in the previous case study. That being the case, the next questions arise: Which payload obtains better savings when being hosted? How many payloads do you want to fly in a hosted payload approach?

Figure 6: Portfolios of hosted payload and procured assets



The answer for the first question can be obtained by analyzing the non-dominated architectures from figure 6 and estimating the cost savings when hosting the different types of antennas that are considered in this analysis. Results indicate that hosting an optical payload can potentially save an 28% of the cost, while a low data rate antenna (S-band) only obtains a 16% cost reduction (note that these numbers are based on the system life-cycle cost and are heavily influenced by the hosted payload cost model). Therefore, it seems to be more advisable to put high data rate payloads like optical terminals in commercial satellites and retain control of S-band communications (which in turn ensures that contingency event will be addressed by privately owned assets not constrained by the operational limitations of host platforms).

Finally, figure 7 presents the tradespace of hosted payload architectures color coded by the number of antennas that are being hosted. By looking at the Pareto front, it can be seen that almost all architectures host one single antenna as opposed to two in order to minimize the burden on the host platform. This burden is also contingent on the mass and power of the hosted communication payload. This is the main reason why optical terminals become so attractive for hosted payload architectures as opposed to low frequency systems that require big parabolic antennas and power amplifiers to close the link budgets.

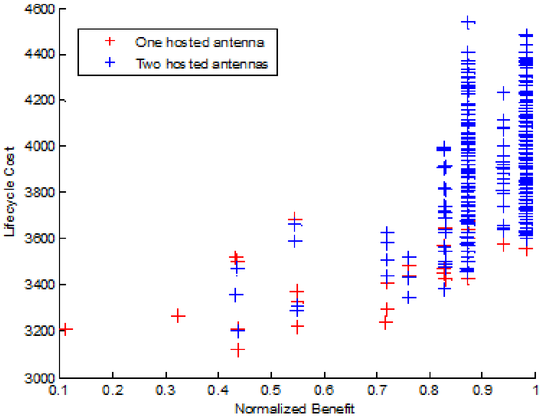
6. Conclusion

Summary

This paper has presented the first set of results obtained through the developed architectural tool to evaluate space communication networks. It has first provided a high level overview of the tool, with a description of four main constitutive elements: An STK model, an architecture generator, an architecture *evaluator* and a resource manager. Next, it has described the newly developed scheduling algorithm based on a rule-based expert system that efficiently allocates the network resources to satisfy near-Earth mission communication requirements.

Then, a discussion on the conducted validation exercise has

Figure 7: Portfolios of hosted payload and procured assets



been presented in order to gain some level of confidence on the results that the tool outputs. This discussion has focused on the validation of the scheduling algorithm by benchmarking it with operational schedules from the TDRSS. Finally, this paper has presented two case studies for future implementations of the TDRSS system. Based on a demand forecast for the network, a first case study has considered the problem of selecting the frequency band to be supported and how to allocate them into the relay satellites. It has been shown that maximum performance architectures require a mix of optical and RF payloads that support high throughput communications as well as reliable low data rate communications. If the traditional procurement strategy is assumed (NASA buys and operates the relay satellites), then monolithic architectures are preferable unless more than two high gain antennas render the resulting satellite configuration too complex.

In turn, the second case study has extended this analysis by introducing architectures with hosted payloads. It has been shown that, according to the current available pricing model, hosted payload architectures are clearly preferable than the traditional procurement strategy, with cost savings between 15% to 30% for the same level of system performance. It has then been discussed the advantages of having mixed procured and hosted payload architectures as a compromise to obtain networks with reduced lifecycle costs that can still address the requirements of highly sensitive and reliable applications. Results have demonstrated that high data rate payloads (specifically optical payloads) are the best candidates to be hosted (with savings up to 28%) thanks to their reduced mass and power requirement. On the other hand, low data rate communication payload should be allocated in privately owned satellites as they require bigger antennas and power amplifiers that increase the burden on the host platform.

Future Work

The main streams of future work are as twofold: On one hand, additional features should be added to the model in order to better capture the complexity of the network configurations (e.g. coupling between the costs of the communication payloads depending on the level of on-orbit processing the they perform). Additionally, the size of the tradespace is currently limited to less than two thousand architectures due to computational limitations, a stringent limitation given the possible combinations from the identified architectural decisions. The solution currently under development is to include a genetic algorithm that alleviates this problem by iteratively generates new populations of architectures by combining the best previously evaluated networks.

On the other hand, the other main stream of future work is related to exercising the tool in a variety of architectural decisions and mission scenarios. In the presented case studies, only geosynchronous constellations were considered although the tool allows comparing them with systems that place relays in MEO and LEO orbits. Additionally, the tool can also provide insight in valuing inter-satellite links and how the cost of putting in orbit their extra communication payloads can be leveraged by reducing the number of operating ground stations. Moreover, couplings between the payload allocation and fractionation strategy should also be further explored so as to understand if monolithic architectures still become preferable if relay satellites can be decomposed in clusters of independent antennas and payloads. Finally, supplementary what-if analysis can be conducted based on (1) the demand forecast for the 2020-2030 time frame, (2) granted spectrum allocations to NASA and (3) technology improvements that increase the spectral efficiency of the communication payloads.

APPENDIX

Table 7: **Acronyms**

Arch	Architecture
CER	Cost Estimating Relationship
DESDYNI	Deformation Ecosystem Structure and Dynamics of Ice
DSN	Deep Space Network
EIRP	Equivalent Isotropically Radiated Power
GEO	Geosynchronous Orbit
GRTG	Guam Remote Ground Terminal
GSFC	Goddard Space Flight Center
HD	High Definition
HP	Hosted Payloads
ISL	Intersatellite Link
ISS	International Space Station
LCC	Life Cycle Cost
LEO	Low Earth Orbit
MEO	Medium Earth Orbit
MIT	Massachusetts Institute of Technology
MOC	Mission Operating Center
MPCV	Multi-Purpose Crew Vehicle
NASA	National Aeronautics and Space Administration
NCCDS	Network Control Center Data System
NDA	Non-Disclosure Agreement
NEN	Near Earth Network
NISN	NASA Integrated Services Network
NOAA	National Oceanic and Atmospheric Administration
RF	Radio Frequency
SA	Single Access
SBRS	Space based Relay Study
SCaN	Space Communication and Navigation
SN	Space Network
STGT	Second TDRSS Ground Terminal
STK	Systems ToolKit
TDRSS	Tracking and Data Relay Satellite System
TT&C	Telemetry Tracking and Command
USGS	United States Geological Survey
WSGT	White Sands Grount Terminal

ACKNOWLEDGMENTS

This project is funded by NASA under grant #NNX11AR70G. Special thanks for Gregory Heckler, David Milliner, and Catherine Barclay at NASA GSFC, for their help getting the dataset and their feedback on our study.

REFERENCES

[1] National Aeronautics and Space Administration, "Space Communications and Navigation (SCaN) Network Architecture Definition Document (ADD) Volume 1 : Executive Summary," Tech. Rep., 2011.

[2] ——. [Online]. Available: http://www.nasa.gov/directorates/heo/scan

[3] e. a. Sanchez Net, Marc, "Exploring the architectural trade space of nasas space communication and navigation program," in *Aerospace Conference, 2013 IEEE*, 2013.

[4] P. Brown, O. & Eremenko, "Fractionated space architectures: a vision for responsive space," Tech. Rep.

[5] S. M. V. . D. C. E. Teles, J., "Overview of TDRSS," *Advances in Space Research*, vol. 16, pp. 67–76, 1995.

[6] Analytical Graphics, Inc. [Online]. Available: http://http://www.agi.com/

[7] D. Selva Valero, "Rule-Based System Architecting of Earth Observation Satellite Systems by," Ph.D. dissertation, Massachusetts Institute of Technology, 2012.

[8] K. D. W. . S. P. Davidson, A., "Pricing a hosted payload," in *Aerospace Conference, 2012 IEEE*, 2012.

[9] W. J. Larson and J. R. Wertz, *Space mission analysis and design*. Microcosm, Inc., 1992.

[10] M. Adinolfi and A. Cesta, "Contributed Paper Heuristic Scheduling of the DRS Communication System," vol. 8, 1995.

[11] National Aeronautics and Space Administration, *Space Network Users Guide (SNUG)*, 2007, no. August 2007.

[12] e. a. Tran, J. J., "Evaluating cloud computing in the nasa desdyni ground data system," in *Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing*, 2011.

BIOGRAPHY

***Marc Sanchez Net** is currently a second year M.S. student in the department of Aeronautics and Astronautics at MIT. His research interests include machine learning algorithms and rule-based expert systems, and their suitability to the fields of system engineering and space communication networks. Prior to his work at MIT, Marc interned at Sener Ingenieria y Sistemas as a part of the team that develops and maintains FORAN, a CAD/CAM/CAE commercial software for shipbuilding. Marc received his degrees in both Industrial engineering and Telecommunications engineering in 2012 from Universitat Politecnica de Catalunya, Barcelona.*

***Dr. Daniel Selva** received a PhD in Space Systems from MIT in 2012 and he is currently a post-doctoral associate in the department of Aeronautics and Astronautics at MIT, and an adjunct Assistant Professor in the Sibley School of Mechanical and Aerospace Engineering at Cornell University. His research interests focus on the application of multidisciplinary optimization and artificial intelligence techniques to space systems engineering and architecture. Prior to MIT, Daniel worked for four years in Kourou (French Guiana) as a member of the Ariane 5 Launch team. Daniel has a dual background in electrical engineering* *and aeronautical engineering, with degrees from Universitat Politecnica de Catalunya in Barcelona, Spain, and Supaero in Toulouse, France. He is a 2007 la Caixa fellow, and received the Nortel Networks prize for academic excellence in 2002.*

Dr. Bruce Cameron *is a Lecturer in Engineering Systems at MIT and a consultant on platform strategies. At MIT, Dr. Cameron ran the MIT Commonality study, a 16 firm investigation of platforming returns. Dr. Cameron's current clients include Fortune 500 firms in high tech, aerospace, transportation, and consumer goods. Prior to MIT, Bruce worked as an engagement manager at a management consultancy and as a system engineer at MDA Space Systems, and has built hardware currently in orbit. Dr. Cameron received his undergraduate degree from the University of Toronto, and graduate degrees from MIT.*

Dr. Edward F. Crawley *received an Sc.D. in Aerospace Structures from MIT in 1981. His early research interests centered on structural dynamics, aeroelasticity, and the development of actively controlled and intelligent structures. Recently, Dr. Crawleys research has focused on the domain of the architecture and design of complex systems. From 1996 to 2003 he served as the Department Head of Aeronautics and Astronautics at MIT, leading the strategic realignment of the department. Dr. Crawley is a Fellow of the AIAA and the Royal Aeronautical Society (UK), and is a member of three national academies of engineering. He is the author of numerous journal publications in the AIAA Journal, the ASME Journal, the Journal of Composite Materials, and Acta Astronautica. He received the NASA Public Service Medal. Recently, Prof Crawley was one of the ten members of the presidential committee led by Norman Augustine to study the future of human spaceflight in the US.*

Bernard D. Seery *is the Assistant Director for Advanced Concepts in the Office of the Director at NASA's Goddard Space Flight Center (GSFC). Responsibilities include assisting the Deputy Director for Science and Technology with development of new mission and measurement concepts, strategic analysis, strategy development and investment resources prioritization. Prior assignments at NASA Headquarters included Deputy for Advanced Planning and Director of the Advanced Planning and Integration Office (APIO), Division Director for Studies and Analysis in the Program Analysis and Evaluation (PA&E) office, and Deputy Associate Administrator (DAA) in NASA's Code U Office of Biological and Physical Research (OBPR). Previously, Bernie was the Deputy Director of the Sciences and Exploration Directorate, Code 600, at (GSFC). Bernie graduated from Fairfield University in Connecticut in 1975 with a bachelors of science in physics, with emphasis in nuclear physics. He then attended the University of Arizona's School of Optical Sciences, and obtained a masters degree in Optical Sciences, specializing in nonlinear optical approaches to automated alignment and wavefront control of a large, electrically-pumped CO2 laser fusion driver. He completed all the course work for a PhD in Optical Sciences in 1979, with emphasis in laser physics and spectroscopy. He has been a staff member in the Laser Fusion Division (L-1) at the Los Alamos National Laboratories (LANL), managed by the University of California for the Department of Energy, working on innovative infrared laser auto-alignment systems and infrared interferometry for target alignment for the HELIOS 10 kilojoule, eight-beam carbon dioxide laser fusion system. In 1979 he joined TRW's Space and Defense organization in Redondo Beach, CA and designed and developed several high-power space lasers and sophisticated spacecraft electro-optics payloads. He received the TRW Principal Investigators award for 8 consecutive years.*

Dr. Antonios A. Seas *is a Study Manager at the Advanced Concept and Formulation Office (ACFO) of the NASA's Goddard Space Flight Center. Prior to this assignment he was a member of the Lasers and Electro-Optics branch where he focused on optical communications and the development of laser systems for space applications. Prior to joining NASA in 2005 he spent several years in the telecommunication industry developing long haul submarine fiber optics systems, and as an Assistant Professor at the Bronx Community College. Antonios received his undergraduate and graduate degrees from the City College of New York, and his doctoral degree from the Graduate Center of the City University of New York. He is also a certified Project Management Professional.*