Methodologies for the Analysis of Value from Delay-Tolerant Inter-Satellite Networking

PhD Dissertation

Christopher John Lowe

Department of Mechanical and Aerospace Engineering

University of Strathclyde, Glasgow

May, 2017

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

Signed:

Date: May 20, 2017

Abstract

In a world that is becoming increasingly connected, both in the sense of people and devices, it is of no surprise that users of the data enabled by satellites are exploring the potential brought about from a more connected Earth orbit environment. Lower data latency, higher revisit rates and higher volumes of information are the order of the day, and inter-connectivity is one of the ways in which this could be achieved. Within this dissertation, three main topics are investigated and built upon. First, the process of routing data through intermittently connected delay-tolerant networks is examined and a new routing protocol introduced, called Spae. The consideration of downstream resource limitations forms the heart of this novel approach which is shown to provide improvements in data routing that closely match that of a theoretically optimal scheme. Next, the value of inter-satellite networking is derived in such a way that removes the difficult task of costing the enabling inter-satellite link technology. Instead, value is defined as the price one should be willing to pay for the technology while retaining a mission value greater than its non-networking counterpart. This is achieved through the use of multi-attribute utility theory, trade-space analysis and system modelling, and demonstrated in two case studies. Finally, the effects of uncertainty in the form of sub-system failure are considered. Inter-satellite networking is shown to increase a system's resilience to failure through introduction of additional, partially failed states, made possible by data relay. The lifetime value of a system is then captured using a semi-analytical approach exploiting Markov chains, validated with a numerical Monte Carlo simulation approach. It is evident that while inter-satellite networking may offer more value in general, it does not necessarily result in a decrease in the loss of utility over the lifetime.

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

Acr. / Abb.	Full definition
AEOLDOS	Aerodynamic end of life de-orbit system
BM	Buffer management
bps	Bits per second
BW	Bandwidth
CC	Congestion control
CER	Cost estimating relationship
CL-mode	Connectionless mode
CO-mode	Connection oriented mode
CTMC	Continuous time Markov chain
DTMC	Discrete time Markov chain
DTN	Delay- and disruption-tolerant network
EDRS	European data relay system
EG	Evolving graph
EO	Earth observation
FIFO	First in first out
FSDN	Fixed Schedule Dynamic Network
FSS	Federated satellite system
GEO	Geostationary orbit
GS	Ground station
GSN	Ground station network
IPN	Inter-planetary internet
ISL	Inter-satellite link
ISN	Inter-satellite networking
Lat	Latitude
LEO	Low Earth orbit
LIFO	Last in first out
Lon	Longitude
LP	Linear programming
LV	Launch vehicle
M2M	Machine to machine
MANET	Mobile ad-hoc network
MAU	Multi-attribute utility

Acr. / Abb.	Full definition
MAUT	Multi-attribute utility theory
MCS	Monte Carlo simulation
NP	Non-deterministic polynomial-time
NPV	Net present value
QoS	Quality of Service
RF	Radio frequency
SAU	Single-attribute utility
SSCM	Small satellite cost model
TCP	Transmission control protocol
TDRS	Tracking and data relay satellite
TTL	Time to live
UDP	User datagram protocol
UHF	Ultra high frequency
VANET	Vehicular ad-hoc network
VC	Virtual circuit
VCB	Virtual combined buffer
VHF	Very high frequency

Nomenclature

All units are SI units, unless otherwise stated.

Symbol	Definition	Section
A	Antenna (effective) area, and	5.5
A	Platform surface area	5.6
A_{SA}	Solar array area	All
C	Network connectivity	3
C	Cost (of a system)	4, 5, 6
C_0	Cost (of a system without inter-satellite networking)	All
C'	Cost (of a system with inter-satellite networking)	All
C_D	Drag coefficient	All
C_{ground}	Ground segment cost	All
C_{ISN}	Cost of inter-satellite networking system manufacture	All
C_L	Lifetime cost	All
C_{launch}	Launch segment cost	All
C_{maint}	Maintenance cost	All
C_{MP}	Maximum-price cost	All
C_{ops}	Operations cost	All
C_{pass}	Cost of a ground station pass	All
C_{space}	Space segment cost	All
c	Capacity of data transfer event	All
$\mathcal{C}$	Total journey cost	All
$\boldsymbol{D}$	Matrix of max. communication range between nodes ($\mathbb{R}^{n \times n}$)	All
D	Destination node	2
D	Data volume	5.1
D	Antenna diameter	5.5
D	Slant range to edge of imager footprint	5.6
DoD	Battery depth of discharge	All
d	Distance between nodes	2
d	Cell degradation factor (per year)	5
E	Set of edges in G	All
E_b	Energy per bit (link budget)	All
E_{bat}	Battery energy capacity	All

Symbol	Definition	Section
$E_{\mathcal{G}}$	Sub-set of edges in $\mathcal{G}$	All
F	Phase of satellite distribution in Walker constellation	All
f	Flow of data through network	All
G	Graph containing all nodes and edges	3, 5.1
G	Antenna gain	5.5, 5.6
G_T	Transmitter gain	All
G_R	Receiver gain	All
$\mathcal{G}$	Evolving graph	All
H	Number of hops in a journey	3
H	Orbit altitude	5
J	Set of feasible journeys	All
$\mathcal{J}$	Journey between nodes	All
$\mathcal{J}^*$	Highest value journey	All
K	Multi-attribute utility scaling factor	All
k	Single attribute weight parameter	4
k	Boltzman constant	5
L	Mission lifetime	All
L_A	Attenuation (atmospheric) losses (link budget)	All
L_F	Imager footprint length	All
L_L	Line losses between transmitter and antenna (link budget)	All
L_S	Free space losses (link budget)	All
M	Total number of edges in $\mathcal{G}$	3
M	System mass	5
M	Number of node states	6
M_{ant}	Antenna mass	All
M_{bat}	Battery mass	All
M_{dry}	System dry mass	All
M_{ISL}	Inter-satellite link sub-system mass	All
M_{power}	Power system mass	All
M_{SA}	Solar array mass	All
M_{stru}	Structure mass	All
M_{TWT}	Travelling wave tube amplifier mass	All
m	Number of edges in G	3
m	Number of data delivery events	5

Symbol	Definition	Section
m	Node state	6
$\mathcal{M}$	Set of node states	All
N	Total number of nodes	5, 6
N_0	Signal noise	All
n	Number of nodes in G	3
n	Number of satellites	5
n	Node	6
$\mathcal{N}$	Set of nodes	6
P	Set of packets	3
P	State transition probability matrix	4
P	Number of planes in Walker constellation	5
P	Probability of being in a particular state	6
P_{ave}	Orbit average power	All
P_{EOL}	End of life power	All
P_{ecl}	Power demand during eclipse portion of orbit	All
P_{ground}	Power required for communication to ground segment	All
P_{ISL}	Inter-satellite link sub-system power demand	All
Pr_{net}	Price one should be willing to pay for networking capability	All
P_{sun}	Power demand during sunlit portion of orbit	All
P_T	Radiated radio frequency (transmission) power	All
p	Number of data acquisition events	All
p_{ij}	Probability of transitioning between states i and j	All
$\mathcal{P}$	Path taken in a journey	All
Q	Set of nominal resources	All
Q'	Set of virtual resources	All
$\boldsymbol{R}$	Matrix of bandwidth (data-rate) between nodes ($\mathbb{R}^{n \times n}$)	All
R	Data rate	All
R_E	Earth radius	All
r	Pixel resolution	All
r_{ij}	Entry in $\boldsymbol{R}$ (i^{th} row, j^{th} column)	3
r_i	Orbit radius (initial)	5
r_f	Orbit radius (final)	5
S	Source node	2, 5.1
S	Solar flux (at 1 astronomical unit)	5.5, 5.6

Symbol	Definition	Section
X	Random variable	All
Y	Number of Monte Carlo simulations	All
α	Time of packet generation	All
β	Ground station pass rate (passes per day)	5
β	Weibull distribution shape parameter	6
$\boldsymbol{\Gamma}$	Node availability matrix ($\mathbb{R}^{n \times n}$)	All
γ	Orbit average fraction of solar array illumination	All
γ_{ij}	Entry in $\boldsymbol{\Gamma}$ (ith row, jth column)	All
δ_{ij}	Entry in $\boldsymbol{D}$ (ith row, jth column)	All
ε	Elevation above the horizon	All
ε_{bat}	Battery energy density (J/kg)	All
η	Angle from nadir to imager footprint-inner	All
θ	Instrument beam width	5
θ	Weibull distribution scale parameter	6
λ	Wavelength	All
λ_{F0}	Earth central angle to footprint outer edge	All
λ_{FI}	Earth central angle to footprint inner edge	All
η_{cell}	Solar cell power conversion efficiency	All
η_{ch}	Battery charge efficiency	All
η_{ecl}	Power transfer efficiency (to sub-system) during eclipse	All
n_{pack}	Solar cell packing efficiency	All
η_{sun}	Power transfer efficiency (to sub-system) during sunlit orbit	All
π	Probability of state residence	All
ρ	Volume of a packet	All
ρ_{SA}	Solar array density	All
σ	Times of edges in a journey	All
τ	Number of sub-graphs	All
τ_{ecl}	Duration of eclipse portion of orbit	All
τ_{sun}	Duration of sunlit portion of orbit	All
ν	True anomaly	All
φ	Time to live (TTL) of a packet	All
χ	Journey feasibility (0,1)	All

Acknowledgements

This is the part where I pay thanks to those who have made this whole process possible, and to whom I owe a great deal of gratitude. If you are reading this, then I ask you please to spare a minute of your time on this and the following page, in honour of the days, weeks, months and years that these people have given to help me.

First and foremost, to my wife Louise, I cannot thank you enough. You have not only shown patience beyond belief, but offered support without question, sympathy without judgement, and understanding without resentment. I owe this work in its entirety, and so much more, to you – thank you.

My supervisor and mentor, Dr Malcolm Macdonald, it has been a true pleasure to work alongside you for these years, I have learned a great deal. Your expert insight has allowed me to make this piece of work something that I am very proud of. Thank you also for your patience and support, it means a huge amount.

I would most certainly not be in a position as privileged as I am without the kind, supportive upbringing that I was so lucky to receive. To my Mum and Dad, Vanessa and Bruce, thank you for providing me a life that so many in this world are not fortunate enough to have. To my two brothers, Andrew & Simon, my Auntie and Uncle, Doreen and Chris and my parents-in-law John & Jaizie, thank you for always being there to support me.

No PhD student ever truly works alone, and I must thank those with whom I have had the pleasure to share an office. For helping me understand new concepts at times when all logic seemed to have left through our 8th floor window. In particular, Steve, Craig, Albert, Pam, Hina, Sarah, Giuliano, Dani, Andreas, Garrie, Daniele, Willem, Marcel, Thomas, Generoso, Ruaridh, Kristaps, Ciara, Emma, JJ, Ross, Philipp and Steve G from Strathclyde, and everyone from the LRT at TU Munich.

As important as time spent in the office is, the moments spent elsewhere have significance too. I owe so many thanks to friends that have helped me clear my head and recharge, be that in the mountains, on the bike, or in the pub. You have all played a crucial role in helping me achieve this lifelong goal.

To those from my life before I moved to beautiful Glasgow, who furnished me with the skills I needed and encouraged my ambition to embark on such a fulfilling endeavour. Academics, researchers, friends and class-mates from the University of Southampton, and all at Magna Parva Ltd, your impact has been more significant than you may know, so thank you.

"*Have fun, work hard and get smart, in that order*"

- Jeff Lowe, *Matenoia*

"Either this is madness or it is Hell." "It is neither," calmly replied the voice of the Sphere, "it is Knowledge; it is Three Dimensions: open your eye once again and try to look steadily."

– Edwin Abbott, Flatland

Chapter 1

Introduction

The birth of the internet sparked a modern age of global-scale networking, since which research into the concepts and realities of wireless sensor networks, cloud computing and the Internet of Things (IoT) has grown enormously. The world is undoubtedly becoming more connected, with an estimated 212 billion devices likely to be linked via the IoT by 2020 [1], and plans being made for broadband-like internet from satellites for all developing regions across the globe [2]. Connectivity has the ability to improve knowledge transfer, increase failure resilience and reduce cost over an equivalent, less connected system. The space industry is no exception to this phenomenon, and the ability and desire to share resources, such as communication bandwidth, data storage and energy, amongst spacecraft, is increasing.

For decades, satellites have been orbiting the Earth and other celestial bodies, collecting huge amounts data, but often oblivious to what is happening on other platforms elsewhere. The data is generally stored until the time comes to deliver to users on the ground for processing and exploitation, which could feasibly take hours. In certain situations, the ability to share information with other orbiting platforms, thus exploiting a greater set of resources, could be valuable. Long data-delivery latency can be overcome through deployment of large ground station networks (GSNs), such as for the ORBCOMM satellite constellation, however whether this offers a better solution than one using inter-satellite networking and few ground stations is often unclear. Other systems, such as the IRIDIUM constellation, do apply inter-satellite networking to great effect, but rely on a stable link to their nearest neighbours and a fully connected path between a data packet's source and destination, in order to function.

In space, and in particular in Low Earth Orbit (LEO), the number of satellites is growing, thanks in part to advances in micro-electronics enabling low-cost nano-satellites to be considered commercially useful, but also because of the growth in

satellite applications demanding increased data provision. This means that the potential, and indeed need, for wireless connections to be made between space-assets facilitating data transfer between nodes is increasing. Furthermore, developments in communication technology, for example widespread use of software defined radio (SDR), electronic beam-steering for phased array antennas and the development of high data-rate optical communication systems, means the prospect of inter-satellite communication is more attractive than ever before.

As identified by Shaw [3], almost every satellite system can be viewed as an information transfer network, with data being the product of interest. This data may be in the form of images, messages or other types of scientific measurements, but are typically reducible to a finite number of elements (bits) which are delivered to users/customers on the ground via a communications link. Furthermore, the utility of a mission is generally considered as some function of data provision, be it volume and latency as with most communication systems, or revisit rate relating to its acquisition as for some surveillance systems. It is the above characteristics related to data collection and provision that have directed efforts in this work, whereby mission topologies are modelled as generic dynamic networks and mission performance is measured as some function of data delivery metrics.

1.1   Motivation & Problem Statement

Consider the simple scenario of two satellites orbiting the Earth, one circling via the poles and the other peering continuously down onto the equator.

Each communicates with a unique ground station (GS) somewhere convenient along its ground-track, such that they will experience one opportunity per orbit during which data could be downloaded (Figure 1).
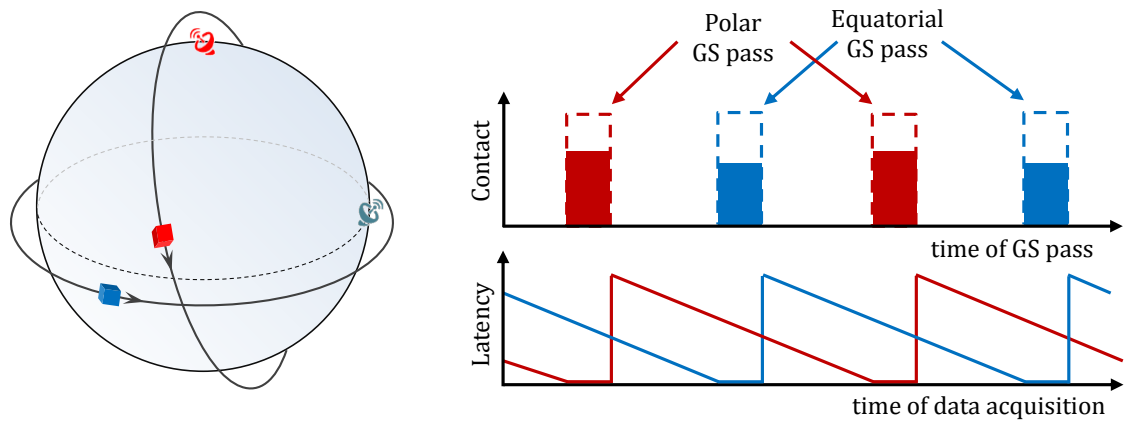
Figure 1 – Two-satellite scenario with no networking

In the top plot (Figure 1, right hand side), contact opportunities for each satellite are shown, with a dotted line indicating total available capacity, while in the bottom plot, delivery latency of data being continuously collected by each satellite is illustrated.

By including the potential to network and share data with each other during periods of close proximity, the download opportunity for both increase by some amount, offering either more data download prospects or a more even distribution of delivery opportunities. Here, benefits for both platforms are realised from data sharing each time they meet as the polar (red) satellite crosses the equator (Figure 2). As it descends it will transfer data to the equatorial satellite for delivery during its upcoming pass, and as it ascends it will receive data from the equatorial satellite for delivery at the pole.
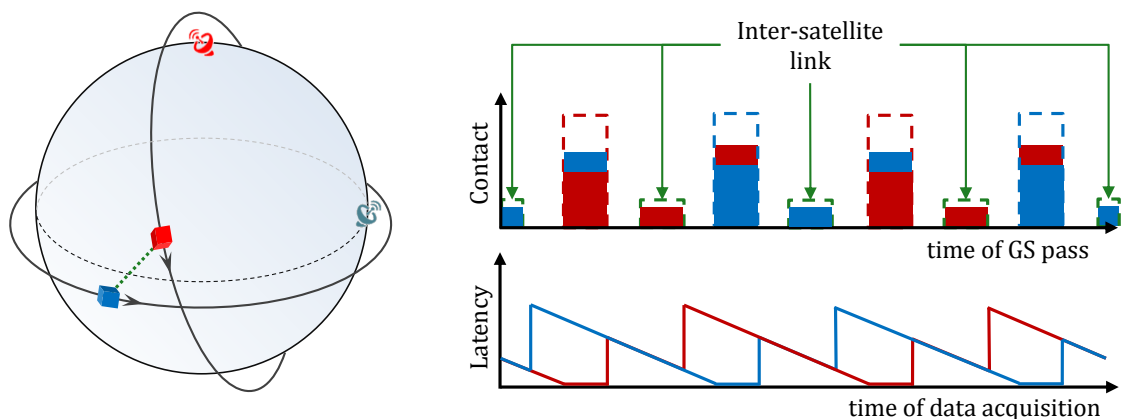
Figure 2 – Two-satellite scenario with networking

It is clear that there is some added value here, enabled by inter-satellite networking. In addition to this, failure of the download capability on one of the satellites would typically render that platform unusable, however the addition of networking enables a more graceful degradation, resulting in a mere reduction of the overall mission utility. Now expand the situation to include 10s, or even 100s of satellites, in different orbits, carrying out their own tasks, with access to various sub-sets of a large distributed ground station network and suddenly, quantifying the value added from incorporating a networking capability is far from trivial. It is however likely that the potential performance for each satellite increases with each addition to the network. The work presented in this dissertation, to the author's best knowledge, provides the first formal analysis aimed at answering the question:

Is the addition of an inter-satellite networking capability of value to a mission?

The answer to this question, given the current state of the art, requires a multi-disciplinary approach that considers implications at the operational-, system- and mission-levels, including assessment of both cost and performance in most cases. Indeed, interest could be attributed to a variety of stakeholder types, from platform developers wanting to define their on-board sub-systems, to national agency representatives wanting to drive forward collaborative efforts between organisations. As such, the methods herein are generalised wherever possible, with consideration of scalability at the forefront.

Regarding the above question, if the value function is formulated correctly and the answer to this question is "yes", then one can be certain that taking the relevant steps to include networking into the system is worthwhile. Based on this proposition, a number of gaps in the knowledge-base were identified:

i. an effective approach to routing data through a resource limited delay- & disruption-tolerant network (DTN) with deterministic mobility patterns,

ii. a definition of the requirements for a networked system to guarantee data delivery provisions,

iii. a formal approach to quantify the value added from an inter-satellite networking capability during nominal operations, and

iv. the effects of inter-satellite networking on the resilience to system failure and expected value over the mission lifetime.

Worth noting is that while inter-satellite networking has a positive influence on a lot of missions, it is not always worth the additional cost required to implement it. Nevertheless, the relationships between different performance parameters are often more complex than anticipated, and the methodologies introduced in this dissertation aim to enhance the mission design process.

1.2   Contributions

To facilitate efficient use of inter-satellite communication opportunities for the purposes of data-routing, a new routing scheme has been developed, called *Spae**. This strategy exploits the deterministic mobility patterns of spacecraft over ground stations in order to route data, while being considerate of the expected amount of available downstream resources (e.g. energy, buffer and bandwidth). Unlike other, similar routing strategies, *Spae* is topology-, time-, data- and protocol-independent, allowing convenient deployment into research and development applications as well as providing a robust platform on which a commercial application could be built. This addresses point i from the previous section.

Following establishment of a suitable routing strategy, mission value can be evaluated through comparison of performance (utility) and cost metrics. Unique to this investigation is the independence from inter-satellite networking technology development cost, which has typically been difficult to quantify and therefore, justify. This analysis instead identifies feasibility of a networked system in terms of data delivery provision and then calculates the added mission value provided from a networking capability. This quantifies the price a stakeholder should be willing to pay for the necessary technology developments and/or integration, to be carried out and is thus a useful contribution for mission designers, investors and the research community. This addresses points ii and iii from the previous section.

Finally, the value of inter-satellite networking in terms of resilience to off-nominal conditions caused by failure, either at the system or sub-system level, is analysed. This work offers a first look at tackling this concept through modelling state transitions of the system as a time-varying Markov-chain [4], to identify both expected conditions and the resulting reduction in mission capability. Through comparison with a nominal, non-networked system, the increased robustness provided as a function of mission

---

* Scottish word: to predict/foretell

value is investigated. Validation of this method is made using a numerical approach, exploiting Monte Carlo simulations [5]. This addresses point iv from the previous section.

The following articles have been published, by the author, in support of the work within this dissertation:

- Lowe, C., Macdonald, M., "*Resource considerate data routing through satellite networks*", AIAA Journal of Aerospace and Information Systems, July 2016, DOI: 10.2514/1.I010423

- Lowe, C., Macdonald, M., "*Rapid Model-based Inter-Disciplinary Design of a CubeSat Mission*", Acta Astronautica, Vol. 105, Iss. 1, Dec 2014

- Lowe, C., Macdonald, M., Greenland, S., "*Parametric CubeSat flight simulation architecture*", 64th International Astronautical Congress, Beijing, 2013

- Lowe, C., Macdonald, M., Greenland, S., "*Through-life modelling of nano-satellite power system dynamics*", 64th International Astronautical Congress, Beijing, 2013

1.3   Thesis Outline

In Chapters 2 & 3, the state of the art in data routing through delay-tolerant networks is discussed, and Spae routing is described, analysed and validated. Chapters 4, 5 and 6 provide an end-to-end approach to evaluating the value offered by inter-satellite networking under both nominal and off-nominal conditions. This begins with an assessment of current and applicable mission design and analysis methods, followed by an investigation into value quantification under the influence of a network-capable system and a look at how partial system failure impacts this value proposition. Finally, in Chapter 7, conclusions about the work and its findings are made, followed by recommendations for further work that could be carried out to extend the investigation.

Chapter 2

Delay- & Disruption-Tolerant Networks

A delay- and disruption-tolerant network (DTN) is defined as a set of nodes offering data networking capability in situations where contemporaneous end-to-end connections (between a source and destination) rarely exist. A DTN can be the result of various characteristics, such as short communication link ranges, physical node contact requirements, low node density, low levels of transmission energy or frequent attacks on network elements. In this chapter, an overview of DTN properties and a review of the DTN literature are provided, with a focus on the state of the art in data routing protocols.

2.1 Early Networking Developments

There exist two data transmission principals that dominate the field of computer networking; connection-oriented (CO-mode) and connectionless (CL-mode) communication. The former demands a two-way interaction between a message's source and destination prior to transmission of any useful data (referred to as packets), such that bandwidth can be reserved offering a guaranteed quality of service (QoS). Packets are sent over a specific path in the network and arrive in the order in which they were sent. The original public-switched telephone network was the first global-scale example of CO-mode communication. The latter, connectionless communication, offers a different approach, whereby data (referred to as datagrams) are sent from the source without need for prior arrangement with the destination, such that an end-to-end connection need not be established *a priori*. As such, each packet in a bit-stream may traverse a unique path, potentially resulting in arrival at the destination in an order different to that in which it was sent. It is the latter of these data transmission types that dominates the work presented in this dissertation.

Packet-switched networks, introduced by Baran in a series of 11 articles released by the RAND corporation in 1964 [6], generally operate over a CL-mode network, where

commonly used protocols include internet protocol (IP) and user datagram protocol (UDP). The alternative paradigm to a packet-switched network is circuit-switched network, which operates exclusively in a CO-mode owing to its need for a dedicated path through the network, which is traversed by all packets in the bit-stream. Packet-switched networks may exploit the benefits of circuit-switched networks through application of a virtual-circuit (VC), whereby all datagrams in a bit-stream are sent over the same path, offering increased QoS (Figure 3).
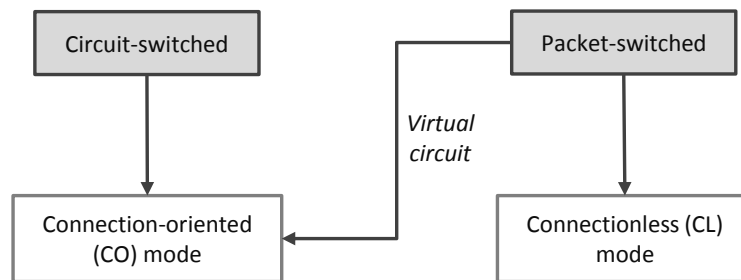
Figure 3 – Relationship between, and types of, circuit- & packet-switched networks

Research into networking methods, in which delay or disruption to data delivery are considered, dates back to 1961, when message switching data networks were introduced by Kleinrock [7]. This approach used store-and-forward routing of datagrams that were transferred along a journey toward their destination in a hop-by-hop fashion. Noteworthy is that for a journey of $n + 1$ hops, the address of hop $i + 1$ is derived at node i, $i \in [1, n]$, offering robustness to disruptions along the journey, a strategy still employed in modern DTNs.

The first mention of DTN-like networking in the field of space systems was in 1999 [8], where the need for intelligent networking, high storage volumes and high data rates was mentioned with respect to Earth science missions. Between 2001 and 2002, articles referring to the Interplanetary Internet (IPN) [9], [10], a communication system aimed at providing internet-like services in support of deep-space exploration, were published. The long propagation delays and frequent disruption of the links between rovers, orbiters and Earth terminals prompted research into this new field of communication. Shortly after, the term delay-tolerant network (DTN) was introduced by the same team[*] [10] in an article formally defining the architecture that encompasses intermittently connected networks, including the IPN. In this work, some fundamental DTN design principles are discussed, including the use of non-chatty

---

[*] With the exception of Travis and addition of Fall.

communication and store-and-forward data transfer techniques. The term delay-tolerant network was also used in 2003 by Fall [11], in which reference was made to challenged networks over which a DTN would reside and act as an overlay architecture – a characteristic also inferred in [10]. Challenged networks in this sense are defined as those that suffer regular disconnect between nodes or exhibit long queueing delays, thus presenting unsuitable environments for traditional network protocols to operate successfully. Since then, the term has been extended to include disruption-tolerant networks, and generally represents any network in which an end-to-end connection between a source and destination rarely exists.

Prior to [10] and [11], a number of articles by teams or individuals either exclusively in, or in collaboration with, the Institut National de Recherche en Informatique et en Automatique (INRIA) make reference to evolving graphs (EG) or dynamic networks [12]–[16]. These articles formally describe the time-varying topology of DTNs, but make no explicit reference to the term DTN. Also from a topology perspective, focussing primarily on node interactions, an excellent attempt to formalise the definition of a DTN was made in [17], with the introduction of time varying graphs (TVG).

2.2   DTN Properties

The scope of what is classed as a DTN is vast, but a particular network can be defined through a number of properties that describe the dynamic topology, contact schedule, data traffic, resources and knowledge characteristics.

2.2.1   Contact Schedule

Generally, a DTN, and more specifically its topology, can be defined by the set of nodes and the dynamic schedule of connections between node-pairs. Connections are typically brought about through some proximity-based measure resulting from node mobility, however this is not necessarily the case and the intermittent connectivity property may result simply from time-dependent availability of nodes for interaction. It is thus the network's contact schedule that is of most significance in describing the evolving topology, and not the node mobility per se.

A simple example of a time-varying topology is shown in Figure 4, where an end-to-end connection between source node S and destination node D is achieved via transfer

between intermediate nodes, *A* and *B*, before a direct link becomes present in the final time step.
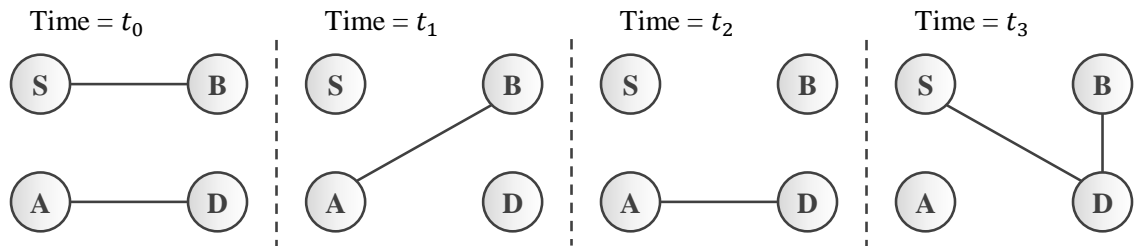
Figure 4 – Example of a non-contemporaneous end-to-end link from S to D

Perhaps the most important topology characteristic in the selection of an effective routing strategy is the predictability of the contact schedule. While some unpredictable social networks exhibit stochastic schedules, and reliable space-based networks present deterministic schedules, most DTNs exhibit something in between, where a level of periodicity and/or node unavailability exists.

2.2.2   Data Traffic

The generation of traffic, or data, in a network can be defined as the arrival of packets/messages/datagrams/bundles at nodes over time. The arrival of data may be stochastic or deterministic, and time or position dependent. A packet may have one or more destination nodes to which it should be delivered, intermediate nodes through which it must pass during its journey (perhaps for security reasons), priority with respect to the other traffic in a node's buffer and a time to live (TTL), which defines the time remaining until expiration. In addition, a packet will have a size/volume, which dictates the amount of resources required for its storage (buffer) and forwarding (bandwidth and energy).

2.2.3   Resources

In the majority of DTN literature, reference is typically made to three resource types;

i.   *Buffer* – the volume available on a node for storage of data
ii. *Energy* – the energy available for reception, processing and transmission of data
iii. *Bandwidth* – the rate at which data can traverse between two nodes

Resource limitations should be considered when routing packets through a network, since a deficiency could result in inefficient forwarding/copying and poor routing decisions. Recent developments in data storage could be argued to have removed many of the limitations in buffer capacity, however self-imposed restrictions may exist for the purposes of passive traffic control.

2.2.4   Knowledge

Knowledge about the network topology and traffic state is what enables one to make sensible routing and buffer management decisions. Consider a network in which nodes must deliver packets to their respective destination, but no knowledge of the network is available and there exists no way of obtaining this information. Nodes may wish to replicate the packets and provide a copy to every node with which it makes contact, in the hope that one will pass it to the destination, but this will of course have a detrimental effect on resource consumption. Now consider a network in which nodes have full knowledge of the contact schedule, resource availability on every node and traffic both in, and due to arrive to, the network. With this information, an optimal routing schedule could be derived according to the network objectives.

2.3   Routing Through DTNs

"Routing: the process of selecting paths from origins to destinations in a network" [18]

As with any system, the design of an effective routing strategy relies on definition of specific attributes that the network operator considers an important measure of performance.

Some of the most commonly applied attributes, and their associated objectives, are described in Table 1.

Attribute	Objective	Description/Example
Delay/latency	Minimise	Time from collection/generation of data at the source, to reception at the destination
Hop Count	Minimise	Number of nodes through which a packet has passed before reaching the destination. It is often used as an analogy for energy consumption
Delivery Ratio	Maximise	Ratio of collected/generated packets to delivered packets, i.e. a measure of the number of dropped packets
Quality of Service	Maximise	Fraction of data delivered from source to destination without error or loss of information. This is generally dependent on the signal to noise ratio of communication links between network nodes.

Table 1 – Typical network objectives

Of course, other application-specific objectives may exist, such as minimising the variance of delay over all delivered data, which should be considered on a case-by-case basis.

The following sections aim to provide an overview of the literature and developments surrounding routing in DTNs, however there exists a number of summary and review papers that are recommended to the reader. As one of the earliest publications on the subject, [11] offers a formal description of the underlying DTN architecture, which is reinforced by [19]. Articles [20] and [21] offer excellent overviews of routing strategies published prior to 2006 & 2007 respectively. Congestion control and buffer management methods are summarised in [22] and a survey of some of the transport-layer protocols used in DTN routing is presented in [23]. Also recommended is [24], which evaluates routing strategies from a performance perspective, highlighting the importance of the trade-off between high delivery ratio and low delay. Classification of different DTNs, with respect to their level of connectivity, is discussed in [25] and [26], in which routing protocols are categorised according to the message replication scheme and end-to-end path requirement. Indeed, the level of connectivity[*] is what fundamentally separates DTNs from connected networks, and has been shown to be highly influential in the routing efficiency [27].

---

[*] Connectivity – the fraction of total time that a node pair are in contact, averaged over all node pairs in the network

Research into network routing represents the vast majority of work in the field of DTNs, and encompasses a broad range of algorithms and protocols, as well as strategies to cope with resource limitations such as buffer congestion. Essentially, during a contact with another node in the network, the routing protocol aims to answer the fundamental question "*should this packet be copied/forwarded to the current neighbour?*" It is argued that if this question is answered correctly for every packet, during every contact, then one has an effective routing strategy.

By definition, DTNs are a type of connectionless (CL-mode) communication, which has motivated development of many new routing protocols. Connection-oriented routing methods designed for fully connected ad-hoc networks, including forms of Distance Vector Routing and Link State Routing, are not suitable in their nominal form due to their reliance on an end-to-end connection. However, some adaptations of these protocol families have emerged as solutions to the DTN routing problem and are discussed below.

For the purposes of this work, review of relevant work is separated into *i)* networks that exhibit mainly stochastic contact schedules (Section 2.3.1) and *ii)* those that exhibit mainly deterministic/predictable contact schedules (Section 2.3.2). Again, this phenomenon is not represented by a discrete divide, but more a sliding scale driven by the node behaviour, uncertainty, reliability and availability.

2.3.1   Stochastic Schedule

Much of the research in the field of DTN routing is focused on those networks in which the contact events between nodes are stochastic in nature, such that data is transferred on an opportunistic basis. These types of networks are generally referred to as mobile ad-hoc networks (MANETs) or in the vehicle-specific case, vehicular ad-hoc networks (VANETs). Note that MANETs and VANETs are traditionally thought of as fully connected networks with time varying graph topologies [28]–[30], for which routing is often achieved using CO-mode strategies. However their applicability to DTNs is justifiable, as a sub-set. Early efforts focussed on a simple approach, termed Epidemic Routing or Flooding [31], in which data delivery resembles a process similar to a disease spreading. Packets are replicated and copied to nearby neighbours in order to flood the network until all messages are delivered to their destination/s. The performance of this approach, from the point of view of delivery delay and resource

(energy) consumption, has been investigated analytically [32]–[34], later with application to marine wildlife [35] and with particular focus on energy efficiency [36]. Note that in a resource-unlimited network, flooding will always result in minimum delay, since the shortest path is guaranteed to be traversed by at least one of the copies in all packet instances. Modifications to flooding have been made, in an attempt to reduce resource consumption, for example by imposing limitations on the number of replications [37]–[40]. Social Group Based Routing (SGBR), introduced in [41], aims to offer increased energy efficiency through reduced levels of replication by means of social grouping. An example is the regular visits made by taxis to taxi-ranks, where messages can be transferred more readily than when they are distributed while travelling.

Perhaps the most common approach for routing in ad-hoc DTNs is based on historical contact information, giving rise to a delivery probability (DP). In [42], the Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) algorithm is introduced, which exploits information pertaining to previous meetings between nodes, including transitivity*. PRoPHET is extended in [43] to exploit contact duration, as opposed to contact frequency, as input to the DP calculation. Meetings between nodes and visits to destinations are used in Meetings and Visits (MV) Routing [44], which builds on previous work by the same group in [45], to provide information for DP, such that messages are transferred to a neighbour if the DP is greater than that offered by itself. MaxProp [46], builds on MV Routing by addressing issues associated with bias toward short distance destinations and removal of stale messages through integration of buffer management schemes and packet prioritisation. A similar DP-based routing strategy, combined with a buffer management approach using packet-specific delivery likelihood, is presented in [34]. This protocol, called Message Fault Tolerance-Based Adaptive Data Delivery Scheme (FAD), uses a form of erasure coding [47]–[49] for efficient replication of packets in order to minimise resource consumption but ensure low delivery delay.

An alternative to DP is used in the Resource Allocation Protocol for Intentional DTNs (RAPID) [50][51], whereby a packet is selected for replication and forwarding to a node in view if, and only if, doing so offers increased marginal utility at the network level. Further attention is paid to utility-based methods in [52] and [53], in the case of

---

* Transitivity is the passing of information of not only one's contact history, but also the history of neighbouring nodes with whom one has been in contact

single and multiple message copies, respectively. Interestingly, it is shown here that transitivity can lead to an increased hop-count in certain scenarios, which is undesirable from the point of view of energy and bandwidth consumption. Utility-based replication and forwarding is investigated with application to heterogeneous nodes in [54], where properties of nodes are not uniform throughout the network.

A modified version of Link State Routing* (LSR) is presented in [55], which uses historical contact information to derive the Minimum Estimated Expected Delay (MEED) for message path evaluation. In [56], the assumption is made that while the contact schedule may not be deterministic, in many cases it follows patterns such that the network state at some future time can be described probabilistically. To this end, the Trajectory Prediction DTN Routing algorithm is introduced, which uses a Markov model [4] to define the likelihood of a node transitioning from one area to another and thus having some probability of forming a contact opportunity. Other probabilistic-based routing strategies, in which cyclic properties of the network play a major part, are described by the same authors in [57]–[61]. Application of probabilistic path planning to disaster situations is presented in [62], [63], in which messages are replicated and sent along the most promising paths according to meeting distributions built from collected historical data.

2.3.2   Deterministic Schedule

Routing strategies that rely on a contact schedule as input to decision making either assume deterministic scheduling or make the necessary assumptions to that effect. Bus, train and aircraft networks, for example, run to a schedule that if maintained rigidly represents a Fixed Schedule Dynamic Network (FSDN). However, delays and disruptions do occur in these scenarios, which introduce a certain level of uncertainty. Satellite networks offer perhaps the most deterministic mobility pattern due to predictable orbit parameters over the long-term, however the availability of platforms is not guaranteed, which may have the effect of expected contact events not being realised. FSDNs offer the fundamental benefit over stochastic schedule networks, with the applicability of graph theoretic methods for path searching between node-pairs.

As early as 2002 [15], routing in FSDNs was addressed using a modified version of Dijkstra's Shortest Path algorithm to find shortest (minimum hop count), foremost

---

* a proactive routing strategy traditionally employed on connected ad-hoc networks

(earliest arrival) and fastest (shortest duration of travel) journeys through a generic network.

In [64], a number of algorithms are presented, with different levels of mobility knowledge. Also included is the availability of nodes, representing the potential for communication, irrespective of proximity. A full knowledge approach is compared with partial knowledge networks, for which mobility patterns are flooded through the network and used for routing table construction. A similar approach is taken during the same year in the seminal paper by Jain et al [27], which considers various levels of knowledge via oracles, including contact capacity, buffer queue and traffic arrivals. This work is extended later in [65] and [66] by considering buffer congestion. Another approach to the problem is found in [67] and [68], in which the consideration of separate storage and transmission cost is introduced as a way to improve generality during route finding. This approach allows one to tailor the cost function according to the network objective, for example a pure delay minimisation can be sought via zero transmission cost, while zero storage cost maps to a minimum hop-count objective. As an alternative to the frequently employed Evolving Graph (EG), which comprises a set of static graph topologies, each representing the network at a point in time, the Time Independent Graph is introduced in [69], which captures all topologies over a particular time horizon in a single static graph.

Some authors [70]–[72] have implemented algorithms based on full knowledge of the network and traffic properties, in order to identify the performance that could be obtained in a particular network, as an upper bound for network designers. A further full knowledge protocol is presented in [73], in which the variance in performance of minimum hop, minimum delay and maximum delivery ratio objectives are shown with changes in network properties such as packet time-to-live (TTL), traffic congestion and buffer capacity. Also exploiting full knowledge of the network state at future times, a comparison is made in [27] to a linear programming (LP) formulation of the problem aimed at maximising network-level objectives. While the LP method is shown to exhibit marginally better performance in terms of delay and delivery ratio, it is found to be computationally intractable for anything other than the most basic of models. Others, for example in [41], [74] and [75], have also turned to LP in an attempt to achieve optimal results in which the search space is reduced through use of dominance rules (e.g. introduction of additional constraints on variable space during optimisation). While complexity using this approach is generally overwhelming for even modest real-

world systems, an alternative LP method, using column generation techniques [76], offers up to a three orders of magnitude reduction in time complexity, but still requires global knowledge of network-wide attributes.

Attempts to improve knowledge about the state of other nodes in the network using adapted forms of Link State Routing (LSR) exist. Positional Link State (PLS) Routing [77] for deep-space applications and Delay Tolerant Link State Routing (DTLSR) [78] for ground-based disaster scenarios, achieve this by flooding information on future contact schedule and predicted link availability through the network. Modification is made to the Better Approach to Mobile Adhoc Networking (B.A.T.M.A.N) algorithm [79], another LSR-like method, termed Store-and-Forward BATMAN (SF-BATMAN) in [80]. This strategy has recently been investigated for use as a routing protocol on federated satellite systems (FSS) [81]. An alternative approach to identifying states as well as resource limitations, in networks with periodic contact schedules, is presented in [82], where discovery messages are flooded through the network in order to identify the maximum amount of data that can be delivered without exceeding bandwidth, buffer or energy constraints. The problem is formulated into an integer linear programming (ILP) problem, which is known to be NP-hard[*] [83], and could be considered limited in the case of a continuously changing network.

Much attention has been paid to routing through satellite networks, including low Earth orbit (LEO) satellite constellations as well as deep-space networks. In [84] and [85], minimal delay routing between ground-based users is achieved via message relay between satellites and gateway ground stations. It is noteworthy that neither inter-satellite nor inter-gateway communication is considered possible in either case. Perhaps one of the most developed methods of routing in FSDNs, originally designed for deep-space networks [86] and later applied to LEO satellite networks [87], is Contact Graph Routing (CGR), which in its original form follows much of the literature in using Dijkstra's Algorithm to identify the next hop node along a journey, to which data is forwarded. Extensions include the consideration of an earliest arrival time objective [88], multiple destinations [89], timeliness effects of potential data delivery failure due to lossy signal [90], effects on delay from downstream buffer queues

---

[*] NP (non-deterministic polynomial-time) hard problems are those that "at least as hard (to solve) as the most difficult problems in the NP set". It is widely accepted that NP-hard problems cannot be solved in polynomial time, and thus quickly become intractable as the scale of the problem increases.

[91][92] and an approach to overcoming the effects of high priority data preventing transmission of bulk packets during a scheduled contact [93][94]. Also examined is the effect of probabilistic failures on satellites during their lifetime, modelled in [95] as an exponential distribution. Temporary failure of a node is used to update the contact plan which defines possible journeys through the network, which is questionable in reality since failure is stochastic and therefore should not be used explicitly for journey evaluation in this way.

A multi-objective approach, where a weighted combination of delay and link reliability (as a measure of packet loss over the link), is used as the utility metric in [96]. The Dynamic Graph Quality-of-Service (QoS) based Resource Allocation Model (DG Q-RAM) is introduced here, which selects journeys based on earliest arrival at the destination considering resource constraints including buffer and bandwidth, but not on a packet-by-packet basis.

An attempt at reducing the computational overhead associated with route calculation and next-hop selection is introduced in [97], called DTN Hierarchical Routing (DHR). DHR relies on a network-wide mobility repeat cycle to exist, whereby contact schedule properties are similar over any two of these cycles. This would be of particular significance to say a bus, train or air transport network, which repeats daily or weekly. It is shown that information required by individual nodes is significantly reduced, with results comparable to the traditional Dijkstra's algorithm. Another approach to reducing the network journey search space is through use of Maximal Relevant Journey Classes (MRJCs) in [98], where nodes maintain a table of suitable journeys through the network. Again, this is suited to cyclic or persistently connected networks, which limits generality.

2.3.3   Buffer Management & Congestion Control

Buffer Management (BM) can be thought of as a node-specific task of manipulating packets within the buffer. Typically it refers to the drop policy [99]–[101], i.e. the process taken to decide which packets to discard in the event of reaching buffer capacity, however it may also refer to forwarding- or replication-policy, i.e. which packet to forward/copy next during a transmission opportunity. Common drop-policies include first-in-first-out (FIFO), last-in-first-out (LIFO) and evict shortest lifetime first, with similar strategies being available for forwarding. The approach can often be trivial

in the case of single destination networks, however selection of the next packet for processing, from a buffer containing data destined for different end points, can become complex. It is often tackled by introducing destination-independent utility metrics such as expected contribution to the network objectives [51] or packet-specific priority [102].

Some cross-over exists between BM and Congestion Control (CC), where the latter can be considered a term to describe how a network copes with general resource limitations. Examples of CC are avoidance of delivery to congested neighbouring nodes [103], packet refusal based on long-term net flow of data to the buffer [104] and financial pricing model-based methods for establishing bundle value [105]. Many other CC strategies exist and the reader is directed to an excellent review in [22] for more information. BM and CC can be considered either dependent on, or independent of routing. In the latter case, the routing strategy takes no interest in resource limitations and congestion, but uses an external mechanism to provide this information. For the work presented in this dissertation, a routing-dependent approach is taken such that CC & BM is inherent in the routing strategy.

2.4   Modification of Dijkstra's Shortest Path Algorithm

Dijkstra's shortest path algorithm [106], [107] serves as the most efficient method of finding the shortest path between two nodes, S and D, in a positively weighted directed graph (digraph), G. It works by maintaining two sets of nodes, one containing only those that are considered to have been *visited* ($\mathcal{V}_v$), and one with those that are considered unvisited ($\mathcal{V}_u$). Note that $\mathcal{V}_v + \mathcal{V}_u = V$, where $V = (v_1, v_2, \dots, v_n)$ is the complete set of n nodes, and initially, $\mathcal{V}_v = \emptyset$ and therefore by definition $\mathcal{V}_u = V$. The distance (weight) between nodes S and i, d_{Si}, for $i = (1, 2, \dots, n)$ is maintained throughout the process, beginning with $d_{Si} = \infty$ for all i. From the current node, which will begin as the source S, the distance from the source to each of its neighbours, which are also in $\mathcal{V}_u$, is identified. If this distance is less than the currently marked distance, d is updated. The current node is then moved into $\mathcal{V}_v$ and the node in $\mathcal{V}_u$ with the shortest distance from S becomes the current node.

This process is repeated until either

- at least one path to the destination node, D is found, and all other nodes in $\mathcal{V}_u$ are at a greater distance, or
- following evaluation of the current node, the distance to all nodes in $\mathcal{V}_u$ remains infinity, thus indicating no available journey from source to destination.

With the restriction of no negative edge weights being enforced, it is impossible for a path to reduce its cost through additional node traversals, thus preventing endless loops. The process is illustrated in Figure 5 and Table 2.
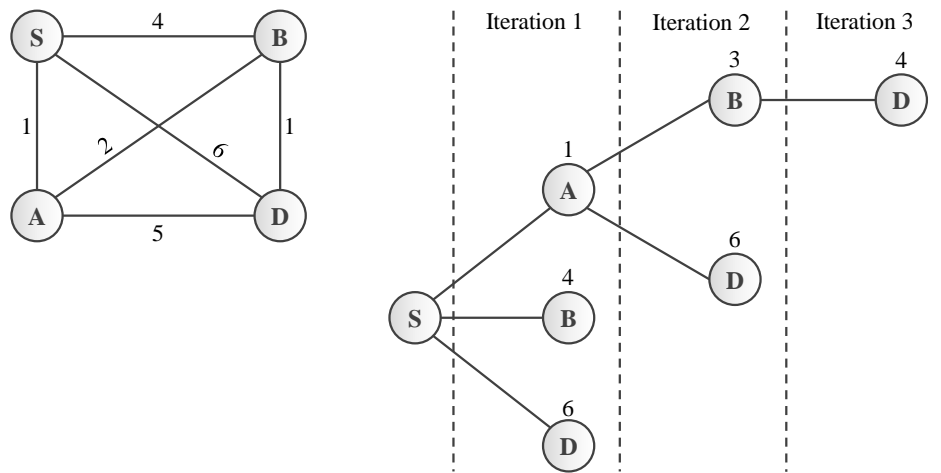
Figure 5 – Illustration of Dijkstra's shortest path algorithm (numbers on edges represent distance)

Iteration	From node	Distance to node			Shortest path to node		
		A	B	D	A	B	D
1	S	1	4	6	1 (SA)	4 (SB)	6 (SD)
2	A	-	3	6	-	3 (SAB)	6 (SAD)
3	B	-	-	4	-	-	**4 (SABD)**

Table 2 – Procedure of Dijkstra's algorithm on graph in Figure 5

At best, executing Dijkstra's algorithm based on a minimum-priority queue, implemented by a Fibonacci heap, has time complexity of $O(E + V \log V)$ [108], where E is the number of edges in the graph and V is the number of vertices/nodes.

For application of Dijkstra's algorithm in the framework of DTNs, certain modifications are required. Most notably, a sequence of edges that form a path must be traversed over time, demanding an additional dimension to the algorithm. Furthermore, an edge

between a particular node-pair may exist multiple times, at different times, which must be considered separately since the potential downstream paths from one instance may differ from those emanating from a later instance. One approach to achieving the required modification is to repeat the graph at each topology change, and include an edge between adjacent instances of the same node, representing storage of a packet during that time step. Ultimately, so long as the time-dimension is considered such that an edge forming part of a journey is never traversed before those upstream of it, the implementation should be successful.

Replacing distance with some other measure such as cost or value can have implications on the algorithm execution, such that the edge quantity may no longer be time-invariant. For example, consider that storage on a node may be cheap during a period of low activity, but expensive during a period of high activity, it may be of value to transfer away from this node during a period of high activity and then return to it at a later date, which would not be possible in the traditional algorithm due to the application of the visited and unvisited sets.

Chapter 3

Routing in Delay-Tolerant Networks

One of the most important characteristics in data delivery performance of a network is the protocol employed to route data from its source to destination. This has been discussed in the previous chapter, and a novel strategy is introduced in the following. Here, a method is described for efficient routing of data, specifically through delay and disruption tolerant networks (DTNs), where an end-to-end connection between source and destination is rarely available. Development has been carried out with generality in mind, not only in terms of data types, but node types also, and while satellite networks are an obvious application of the protocol, the algorithm can be thought of as network-independent. It is posed that this new protocol, called Spae[*], provides a network resource considerate approach to data routing, with effectiveness that exceeds both traditional and current examples from the literature.

Spae exploits a deterministic contact schedule and knowledge of expected downstream resource availability, in the form of communication bandwidth, energy and buffer storage and provides packet-wise forwarding of data. Forwarding decisions are made on a hop-by-hop basis each time contact is made with another node in the network, ensuring robustness to stochastic network changes. Upon making contact, two nodes must share information relating to their buffer contents via some handshake interaction and packets are routed[†] one-by-one, highest priority first, until all packets have been dealt with. The journey along which a packet is routed is selected as the one with highest expected value, measured as a function of network objectives and downstream link availability. The expected resources of edges and nodes along this highest value journey are then reduced by an amount equal to that demanded by traversal of the packet, such that an apparent resource is suitably lower when routing subsequent packets. Edges along which resources are sufficiently low to prevent

---

[*] Scottish word – to predict/foretell
[†] The term "routed" shall be used throughout to represent the virtual assignment of a packet over a particular journey. The packet may or may not traverse the particular journey in reality, however it enables identification of the ideal *next hop*.

traversal of a packet are invisible when routing remaining packets, thus avoiding delivery along journeys with little or no hope of success. Feasible journeys are considered those along which the packet would reach the destination before its expiration. Finally, packets are physically transferred to the neighbouring (local) node if, and only if, it is the next hop along the selected journey.

For clarity, some of the terminology used within this chapter is introduced next. In terms of network performance, three metrics are considered; *delivery delay* is the duration between a packet entering the network and being delivered to a destination, *delivery ratio* is the ratio of the total number of delivered packets to the total number of generated packets and *hop count* is the number of nodes on which a packet is stored before reaching the destination. With respect to data traffic, the term *contention* is used for the ratio of total number of generated packets to the total number that could be downloaded, given 100% utilisation of destination contact events, and *time to live* (TTL), which is the duration since generation before a packet expires (and dropped from the network). Finally, *network connectivity* (C) is used as described in [27], as a measure of inter-node communication range representing the fraction of total time that a node pair are in contact, averaged over all node pairs in the network. E.g. $C_{ij} = 0.1$ would imply that node i is in contact with node j, 10% of the time and a value of $C = 1$ implies that all nodes are connected at all times.

3.1 Network Model

In the following sections, the network model is formally described in order to provide an explicit definition of Spae routing. Spae offers a scale and time-independent network routing strategy, applicable to any fixed schedule dynamic network (FSDN), in which information pertaining to expected start time and duration of node-pair contacts is available.

3.1.1 Evolving Graph Configuration

A graph $G(V,E)$ is made up of a set of nodes $V = \{v_1, v_2, \dots, v_n\}$ and edges $E = \{e_1, e_2, \dots, e_m\}^*$ and represents the union of an ordered sequence of G's sub-graphs $S_G = G_1, G_2, \dots, G_\tau$, which occur at times specified by the sequence $S_t = t_1, t_2, \dots, t_\tau$ respectively, such that $G = \bigcup_{i=1}^{\tau} G_i$. The evolving graph $\mathcal{G} = (G, S_G, S_t)$ is used to define the dynamic network that exists over the time horizon $T = [t_1, t_\tau]$. Formation of a new sub-graph occurs at each change to the set of edges $E_i(G_i)$, i.e. whenever one or more edges are created or destroyed, $E_{i+1} \neq E_i, \forall E_i \in E$. For clarity, graph G_i is in place during the interval $[t_i, t_{i+1}]$. An edge $e(u,v) \in E_i(G_i)$ is considered directional and uniquely characterised by its sending node u and receiving node v. Indeed, it follows that identical edges present across one or more adjacent graphs $e(u,v) \in \{E_i, E_{i+1}, \dots, E_{i+k}\}$ represent edges that remain in place while k other edge formations/removals occur elsewhere in the network. The cause of edge formation is arbitrary so long as a future contact schedule is known, e.g. edge formation might be the result of either node proximity in a mobile network or intermittent operation of nodes in a static wireless sensor network.

For each edge e in the evolving graph $\mathcal{G}$, there exists a nominal capacity $q(e)$, which is the total volume of data able to be sent over e. The set Q_i represents the capacity of all edges at time i, $|Q_i| = |E_i|, \forall\, i = [1, \tau]$. Capacity of the edge between sending node u and receiving node v, from time t_i to t_{i+1} is

$$q(t_i) = \int_{t_i}^{t_{i+1}} r_{uv}(t)\, dt, \qquad \forall i \in [1, \tau], \qquad\qquad 3.1$$

where $r_{uv}(t)$ is the data-rate (bandwidth) at time t from node u to node v. For clarity, $\boldsymbol{R}(t) = \mathbb{R}^{n \times n}$ is defined as the matrix representing the data rate for each node pair at time t, where r_{ij} is the ijth element in $\boldsymbol{R}$.

---

* Often, the set of edges E is captured in the form of an adjacency matrix of size $\mathbb{R}^{n \times n}$, whereby a non-zero entry in the ith row and jth column, $e_{ij} \neq 0$, represents a directed link from node i to node j. This is a plausible approach here and would naturally follow through to the evolving graph instances $E_i \, \forall\, i = [1, \tau]$.

3.1.2 Data Traffic

Generally, data is defined as a packet $p(\rho, \alpha, \varphi, v_d)$ with a unique volume ρ, time of creation (birth-date) $\alpha \in T$, time to live (TTL) φ (i.e. time since α until expiration) and destination $v_d \in V$. The objective of any routing algorithm is to deliver packets to their destinations in such a way that best satisfies the network objectives, e.g. minimum end-to-end delay (latency), minimum hop count and/or total volume delivered. In some networks, packets may be required to pass through specific nodes in the network, en-route to the destination, for security purposes or other reasons. Whilst it would be possible to include this in Spae routing by replacing the modified version of Dijkstra's algorithm [106] that finds the optimal journey through the evolving graph with an algorithm that finds the optimal journey in a Steiner Graph problem [89], it is considered beyond the scope of this work.

In most terrestrial systems, the arrival rate of data to the network is stochastic, but often can be defined by some probability distribution, which may indeed vary with time or node location. In the case of deterministic data generation*, it is argued that an alternative, bespoke routing algorithm could be developed, that exploits this information to best maximise the network objectives.

3.1.3 Journeys

A journey $\mathcal{J}$ through the network, between two nodes (u, v), is defined by a start time (t_0), a path $\mathcal{P}$, made up of an ordered sequence of $H + 1$ nodes, $\mathcal{P} = i_0, i_1, \ldots, i_H$, where $i_h \in V, \forall h = 1, 2, \ldots, H$ and an ordered sequence of times $\sigma = t_1, t_2, \ldots, t_H$, where $t_h \in S_t, \forall h = 1, 2, \ldots, H$ at which the edges between nodes exist. The total number of hops (transfer of custody) is H, $i_0 \in \mathcal{P}(\mathcal{J})$ is the source node and $i_H \in \mathcal{P}(\mathcal{J})$ is the final node. It is clear that $t_h \geq t_{h-1}$ and the receiving node in hop h must be the sending node in hop $h + 1$. Finally, it follows that t_H represents the start time of the contact with the destination.

---

* The deterministic data generation case shall be used as an upper bound on performance for comparative purposes in this work, achieved by applying a traffic oracle that knows the priority and location of all traffic in the network.

The cost of a journey is generalised as the sum of the storage cost $\mathcal{S}_i$ on board each node i and the transmission cost $\mathcal{T}_{ij}$ between each node pair ij, along a journey $\mathcal{J}$. The total journey cost is therefore

$$\mathcal{C}(\mathcal{J}) = \sum_{h=1}^{H} \int_{t_{h-1}}^{t_h} \mathcal{S}_{i_{h-1}}(t)dt + \sum_{h=1}^{H} \mathcal{T}_{i_{h-1}i_h}(t_h), \qquad 3.2$$

where H is the total number of hops and $t_h \in S_t$ is the time at which hop h occurs. The storage and transmission costs are defined according specific mission objectives, such that for example a transmission cost of $\mathcal{T}_{ij} = 0$ for all node pairs would imply that a journey with minimal duration would be the one of least cost.

Availability (γ) of a node is the probability that the node will be available for data sharing at a scheduled contact. This may be either constant or time-dependent, but must be known/predictable for implementation into Spae. This can be considered a reasonable assumption for a satellite network, whereby the availability of a satellite could be either published by the operator, or calculated as some moving-point average from historical availability data. The total availability of a journey is therefore formulated as

$$\gamma(\mathcal{J}) = \prod_{h=1}^{H} \gamma_{i_h}(t_h), \qquad 3.3$$

where $\gamma_{i_h}(t_h)$ is the availability of the hth hop receiving node, at the time of that hop. It is clear that a journey with one or more nodes displaying low availability will have a low value. For completeness, the set $\Gamma(t) = \{\gamma_1, \gamma_2, \dots, \gamma_n\}$ represents the availability of each node at time t, where $|\Gamma| = |V|$. The value $\mathcal{V}$ of a journey is considered to be the inverse of the risk-adjusted cost, $\mathcal{V} = \gamma/\mathcal{C}$, such that a low cost journey with high availability (low risk) is of high value. Naturally, in a completely available network the value reduces to the inverse of the cost.

The highest value journey $\mathcal{J}^*$ from the set of feasible journeys $J = \{\mathcal{J}_1, \mathcal{J}_2, \ldots, \mathcal{J}_Y\}$, is

$$\mathcal{J}^* = \mathcal{J}_i | \mathcal{V}(\mathcal{J}_i) = \max_{\mathcal{J} \in J} \mathcal{V}(\mathcal{J}). \qquad\qquad 3.4$$

3.2   Spae Routing

Given the network and data traffic descriptions provided in the previous sections, it is now possible to formally define Spae. At the start of a contact between nodes u and v, their respective buffers $B_u = \{p_{u1}, p_{u2}, \ldots, p_{uI}\}, I = |B_u|$ & $B_v = \{p_{v1}, p_{v2}, \ldots, p_{vK}\}, K = |B_v|$ are shared over a handshake interaction during which the complete set of packets resident on board both nodes are placed into virtual combined buffer (VCB), with $B_{uv} = \{p_{uv1}, p_{uv2}, \ldots, p_{uv(I+K)}\}$. Packets are routed, one-by-one, typically highest priority first, but not necessarily, until the VCB is empty or until no capacity remains in either direction of the current contact event. The journey identified in this process is used to identify the next hop for a packet, from its current custodian, and thus whether custody should be transferred during the current contact. In other words if the next hop along the journey is to the current neighbour during the current contact, then custody is transferred. Prior to routing of the 1st packet in the VCB, a virtual resource set, Q_i' is created, which is a duplicate of the nominal set, Q_i, for each graph that could potentially be part of the journey for a packet within B_{uv}. This resource set is generic for Spae, such that it could include bandwidth, node buffer storage, energy or other resources, and can be described formally as

$$Q_i' = Q_i, \qquad i = [k, k + y(p_{uvX})], \qquad\qquad 3.5$$

where k is the graph in $\mathcal{G}$ at the current time t_k, and $y(p_{uvX})$ is the number of graphs until expiry of the packet with the longest remaining time before expiry (p_{uvX}). It follows therefore that $t_{(k+y(p_{uvX}))} \leq t_k + \varphi(p_{uvX}) - \alpha(p_{uvX})$ and $t_{(k+y(p_{uvX})+1)} > t_k + \varphi(p_{uvX}) - \alpha(p_{uvX})$. The complete set can be defined as $Q' = \{Q_k', Q_{k+1}', \ldots, Q'_{k+\max(y(p_{uvX}))}\}$. Worth noting is that exactly how the resources are measured/estimated is independent from how Spae operates, such that they could arrive from an oracle-layer that is able to transmit information between all nodes at all times, or from on board predictions considering previous contacts with others in the network. Here, nominal resources are considered a reasonable first step, i.e. the effect

of traffic arriving at a downstream node from elsewhere in the network is ignored. While this approach is recognised to potentially result in unexpected congestion, it is shown later to provide significant benefits over no resource consideration whatsoever. Following routing of packet p along a journey $\mathcal{J}$, the entries in Q' that correspond to the edges and nodes in $\mathcal{J}$ are reduced by the magnitude required to transfer custody of packet p during that contact, which in the case of edge capacity (bandwidth) is

$$q'_{h-1,h,k} = q'_{h-1,h,k} - \rho_p, \qquad k = t_h(\sigma), \forall h = 1,2, \dots, H, \qquad 3.6$$

where H is again the number of hops in $\mathcal{J}$ and k is the sub-graph number in G in which the hop takes place. While it would be possible to include other resources such as energy and buffer storage, only edge bandwidth is considered here in order to illustrate the concept, with energy and buffer storage resources considered infinite.

Journey evaluation is achieved by searching a subset of the evolving graph using a modified version of Dijkstra's algorithm (Section 2.4), adapted for use on time-varying networks. The subset for packet p is defined as the set of graphs that occur prior to expiration of p, i.e. $[G_k, G_{k+y(p)}]$, following the notation used above in equation 3.5. A journey may only be selected with the modified Dijkstra's algorithm if it is considered feasible, with respect to packet p. A feasible journey is one on which a sufficient level of expected resource exists and which can be completed before expiration of the packet. Therefore, feasibility of journey $\mathcal{J}$, considering bandwidth as the resource that is potentially limited, can be defined as

$$\chi(\mathcal{J}) = \begin{cases} 1 & \left| q'_{h-1,h,k} \geq \rho_p \wedge t_H - \alpha \leq \varphi, \qquad k = t_h(\sigma), \forall h = 1,2, \dots, H \right. \\ 0 & \text{otherwise} \end{cases}. \qquad 3.7$$

After either routing all data in the VCB, or exploiting all available capacity in both directions of the current contact event, each packet should be transferred to the neighbouring node, retained on board the current custodian, or dropped due to a lack of feasible journeys. At this point, the routing process for this contact is complete.

Regarding application and manipulation of the evolving graph described in Section 0, the current implementation assumes that a satellite-based network has single-point transmission (send) and multi-point reception (receive) capabilities, exhibiting full-duplex operation when a two-way link exists. This is not a necessary condition for Spae

to function correctly, but is considered a realistic configuration assuming multiple antennas capable of electronic beam-forming, while simultaneously being considerate of energy usage.

An example of Spae routing is shown in Figure 6, in which three packets in node A's buffer and two packets in node B's buffer are routed, to destination D, along journeys described in the table on the right of the figure. An evolving graph is shown on the left, where the terms in parentheses (t,q), alongside each edge, indicate the time (t) and expected available resources (q). For simplification, all nodes are considered available for transfer, cost of storage is 1 per time step and cost of transmission is 1 everywhere, except for transfer to the destination D, which has zero cost. Each packet $(p_{Ni(j,k)})$ on node N is described as having a resource demand (j) and priority (k) within the combined buffer.
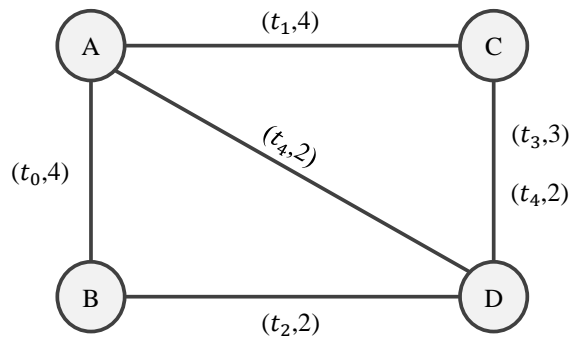
$$B_A = \{p_{A1(1,5)}, p_{A2(2,2)}, p_{A3(1,1)}\} \quad B_B = \{p_{B1(3,4)}, p_{B2(1,3)}\}$$

Packet	Resource required	Path ($\mathcal{P}$)	Times (σ)	Edge resource before	Edge resource after	Journey cost ($\mathcal{C}$)
p_{A1}	1	A-B-D	t_0, t_2	4, 2	3, 1	3
p_{B1}	3	B-A-C-D	t_0, t_1, t_3	3, 4, 3	0, 1, 0	5
p_{B2}	1	B-D	t_2	1	0	2
p_{A2}	2	A-D	t_4	2	0	4
p_{A3}	1	A-C-D	t_1, t_4	1, 2	0, 1	5

Figure 6 – Schematic example of Spae routing

Even in this simple case, the routing approach found by Spae is not trivial, with preferred routes exhibiting insufficient resources in a number of cases. For example, the relatively large resource demand from packet p_{B1} requires it to traverse three nodes before reaching the destination, with lower cost journeys being unavailable due

to resource limitations. The result of this routing event would be a transfer of p_{A1} from A to B and p_{B1} from B to A, while all other packets remain on their current custodian for transfer at future contacts.

3.2.1 Pseudocode

The pseudocode in Table 3 describes the Spae routing procedure as implemented in software, which is executed at the start of each node-pair contact event:

Spae – procedure at each node to node contact event

Input: $\mathcal{G}, u, w, \mathbf{R}, B_u, B_w, \mathcal{S}, \mathcal{T}, \Gamma$

Output: Custodian of each packet in combined buffer post-routing

---

$\quad B_{uw} = \cup \{B_u, B_w\}$

$\quad Q' = Q$

$\quad$**for** $k = 1\ to\ |B_{uw}|$ do

$\quad\quad v = v(p_k) \in \{u, w\}$

$\quad\quad$Find $\mathcal{J}^* = \mathcal{J}_i | \mathcal{V}(\mathcal{J}_i) = \max\limits_{\mathcal{J} \in J:\ \chi(\mathcal{J})=1} \mathcal{V}(\mathcal{J})$ with *modified DIJKSTRA* $(\mathcal{G}, u, \mathbf{R}, \mathcal{S}, \mathcal{T}, \Gamma)$

$\quad\quad$**if** $\chi(\mathcal{J}^*) = 1$

$\quad\quad\quad$**for** $h = 1\ to\ H$, where $H = |\sigma|$ and $\sigma \in \mathcal{J}^*$ do

$\quad\quad\quad\quad q'_h = q'_h - \rho_{p_k}\ ;\ q' \in Q'$

$\quad\quad\quad$**end**

$\quad\quad\quad$**if** $i_1 = \{u, w\} \neq v\ \wedge\ t_1(\sigma) = t_0(\sigma)$, where $i_1 = \mathcal{P}^*(1), \mathcal{P}^* \in \mathcal{J}$, do

$\quad\quad\quad\quad v(p_k) = i_1$

$\quad\quad\quad$**end**

$\quad\quad$**else**

$\quad\quad\quad$drop p_k from buffer

$\quad\quad$**end**

$\quad$**end**

Table 3 – Pseudocode for Spae Routing

The above pseudocode is also described informally by the flow diagram in Figure 7:
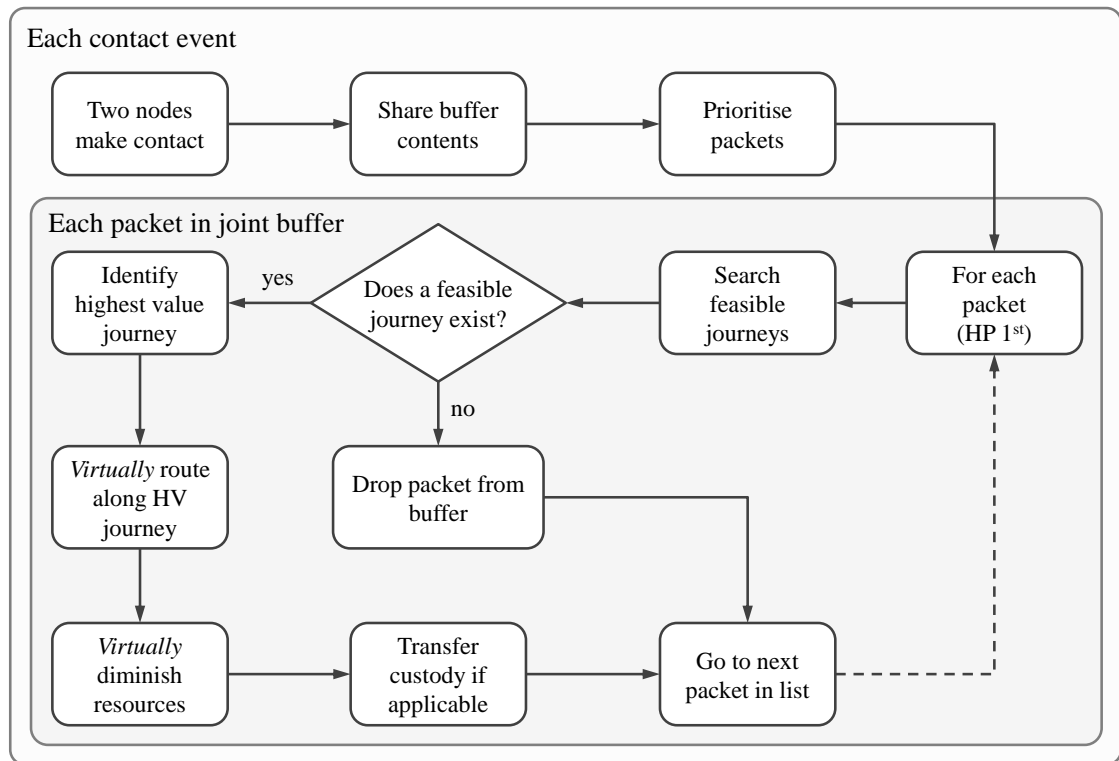
Figure 7 – Spae routing process

3.2.2   Complexity

During each contact event, Spae can be seen to have a worst case complexity equivalent* to running a modified Dijkstra's algorithm, for multiple graphs extending over the period of a packet's remaining time to live, for each packet on board the node pair's combined buffer. In *big-O* notation, this is $O\big(|B|(|E| + |V|log|V|)\big)$, where $|B|$ is the number of packets in the combined buffer, $|E|$ is the number of downstream edges (i.e. the number of edges in all future graphs) and $|V|$ is the number of nodes in all downstream graphs. In reality, the complexity is generally not this great, since computation can be reduced given that *a)* once the resources of each graph (in both directions) of the current contact have diminished below a level required for routing of a packet, routing is terminated since no additional transfer can take place during that

---

* While execution of the modified Dijkstra's algorithm only represents the "search feasible journeys" and "identify highest value journey" steps in Figure 7, the other steps consist of trivial operations that have either constant ($O(1)$) or linear time complexity with respect to buffer size ($O(B)$). As such, their complexities can be neglected when considering the overall algorithm complexity.

contact and *b)* a packet with the same destination node as some previously routed packet can be routed along the same journey, if sufficient resources are expected to remain along that journey. To achieve this, a feasible journey table is maintained, which can be checked at the start of routing each packet.

3.2.3   Performance Attributes

Performance of a routing protocol for DTNs is typically measured as some function of delivery delay, hop count and delivery ratio. Delay is often critical for information such as Earth observation data, machine-to-machine messaging and satellite-based automatic identification system (S-AIS) data. Hop count is often considered a good measure of energy usage, since transmission of data requires power from the satellite bus, but in a federated satellite system (FSS) [*] hop count is also a good measure of the potential financial implications of utilising a relay satellite within the federation. Delivery ratio is effectively a measure of payload operation efficiency, where a low result would indicate wasted resource. Indeed, for important and critical data, a high delivery ratio might be considered necessary.

3.3   Simulation Environment

In order to evaluate the performance of Spae, and compare it to other routing strategies, two scenarios are simulated. In the first, nodes exhibit random mobility, which illustrates Spae's independence of a topology repetition requirement. The second scenario emulates a federated satellite system (FSS), in which satellites orbit the in Earth on trajectories that are not necessarily related and have agreed to join a federation. Nodes within the federation agree to share data storage and download resources by transferring data through the network over inter-satellite communication links, along preferred journeys towards its respective destination.

A simulation environment has been developed that provides a means of evaluating the performance of Spae compared with other routing strategies. The environment provides the necessary mobility propagation of each node, resulting in the relative positions that are exploited for building the evolving graph. The arrival of data is modelled as a stochastic process following a geometric distribution for inter-packet

---

[*] A FSS consists of a satellite network of platforms operated by multiple stakeholders. Satellites are expected to share resources in order to be part of the federation, but interactions may not necessarily be for the good of the whole network, due to conflicting stakeholder objectives.

arrival time, which has been shown to accurately represent the arrival of satellite image requests from customers [109]. The probability of packet generation is defined as a function of network contention and total network capacity of contacts with ground stations. This approach ensures a comparative level of data in the network, given different network attributes, allowing for comparison. Performance attributes of delay, hop count and delivery ratio are considered throughout, each shown with respect to varying data TTL, inter-node link range and bandwidth. Values shown are averages for all packets, delivered by all nodes, over the entire simulation.

Spae routing is compared against three other protocols, which are described below in order of sophistication and complexity. The first approach, *next2see*, is a strategy whereby packets are transferred to a neighbour if it is expected to make contact with a ground station before the current custodian. The second protocol, *nextHop*, exploits Dijkstra's algorithm to identify the highest value journey, along which packets are transferred should the neighbour represent the next hop in that journey. It differs from *Spae* in that downstream resources are not considered. Finally, in *fullKnow*, an upper bound on performance is evaluated, where full knowledge of all current and future network properties exist, including traffic generation, future node availability and resources. As discussed earlier, this approach is considered unattainable in most real-world scenarios. In all of the protocols, there exists a limitation on the number of packets transferred over an edge due to finite bandwidth, but no limitations on buffer or energy are considered. The properties of each strategy are summarised in Table 4.

Name	Description	Knowledge			
		D	M	R	T
next2see	Data transferred to if neighbour is sooner to be in contact with destination node	✓	✗	✗	✗
nextHop	Data transferred if neighbour is next along the highest value journey (not considering downstream resources)	✓	✓	✗	✗
Spae	Data transferred if neighbour is next along highest value journey, (considering downstream resources)	✓	✓	✓	✗
fullKnow	End-to-end routing carried out at source with knowledge of all network attributes, including data traffic	✓	✓	✓	✓

Table 4 – Routing strategy comparison ("knowledge" parameters are: D = destination contact time, M = network mobility, R = nominal resources, T = data traffic)

3.4 Case Study 1: Random Mobility Network

In this random mobility network, nodes move according to a random waypoint mobility model, as described in [110], which allows assessment of routing protocol performance in a scenario without cyclic mobility. Direction and velocity of each mobile node is defined at every time step in the simulation, with area boundaries acting as perfect reflectors should contact be made with them (Figure 8).
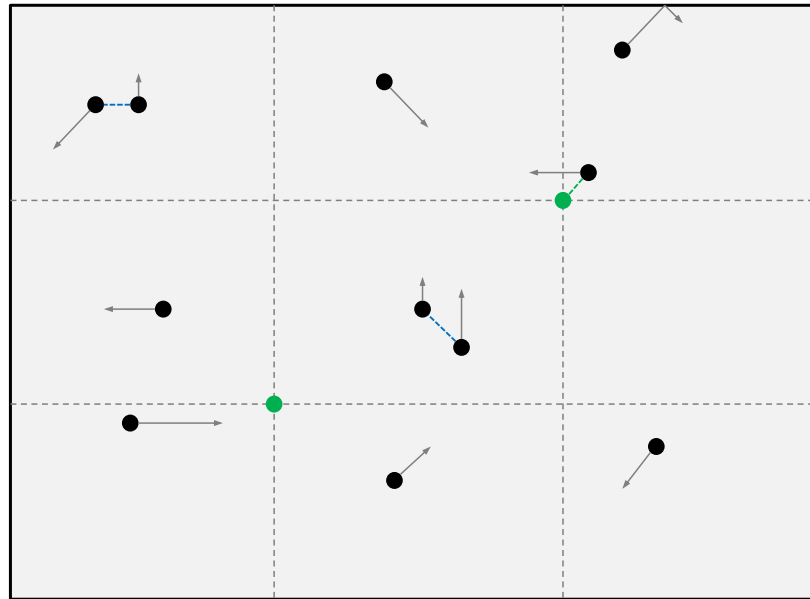
Figure 8 – Random mobility network. Black nodes represent mobile nodes (sources), green nodes represent destinations (sinks)

This type of network is of particular interest for more stochastic systems such as traffic or social networks. In all of the simulations here, destination nodes remain stationary throughout and data may be transferred when node separation is below a pre-defined link range threshold.

Properties of the network are defined in Table 5.

Property	Value	Comment
Number of nodes	10	Mobile nodes on which data is acquired according to a geometrical distribution
Number of destinations	2	Stationary nodes to which data is delivered
Link range between nodes & destinations	3	Node separation distance below which data can be transferred
Data rate between nodes & destinations	2	Number of data packets transferable per unit time
Node availability	100%	Nodes are considered always available for data transfer
Grid size	50 x 50	Area in which nodes move
Simulation length	50,000	Duration of each simulation in arbitrary units of time
Simulation time step	1	Size of simulation step
Storage cost ($\mathcal{S}$)	1	Cost per unit time
Transmission cost ($\mathcal{T}$)	0[*]	Cost per hop

Table 5 – Random mobility network model properties

A number of simulations have been carried out in order to obtain results that illustrate the relationship between a range of design variables (Table 6) and the performance attributes of delay, hop count and delivery ratio (finite TTL only).

Design Variable	Range	Description
Node to node link range	{0, 1, 2, 3, 4}	Inter-node distance below which data can be transferred
Node to node data rate	{1, 2, 4}	Rate (packets per time step) at which data is transferred
Contention	{0.1, 0.5, 0.9}	Ratio between volume of generated data and download capacity
Time to live	{200, ∞}	Time from data generation to expiry

Table 6 – Random mobility network design space definition

---

[*] A transmission cost of 0 results in delay minimisation being the only routing objective.

A sub-set of the results are included here for clarity, with the full set included in Appendix A. The first and final 5,000 time steps were ignored in all simulations in order to avoid transient delivery dynamics in the results. The results in the following sections show analysis of infinite (∞) and finite (200) TTL, each at low (0.1) and high (0.9) contention, for various node to node link ranges and a node to node data rate of 2 packets per time step.

3.4.1   Infinite Time To Live

The following two sections show results of simulations carried out using data with an infinite TTL. Owing to the fact that contention is <1 throughout, all data is delivered such that the delivery ratio is 1 for all protocols in all cases and is therefore not shown.

3.4.1.1   Low Contention

At low contention (0.1), it is of little surprise to see that the strategies employing journey search (*nextHop*, *Spae* and *fullKnow*) achieve similar performance at all link ranges, since it is unlikely that a path will become congested and consequently prevent anticipated delivery of packets. The *next2see* approach does however suffer from a longer delay due to a lack of intelligence, which ttl be evident throughout the results.
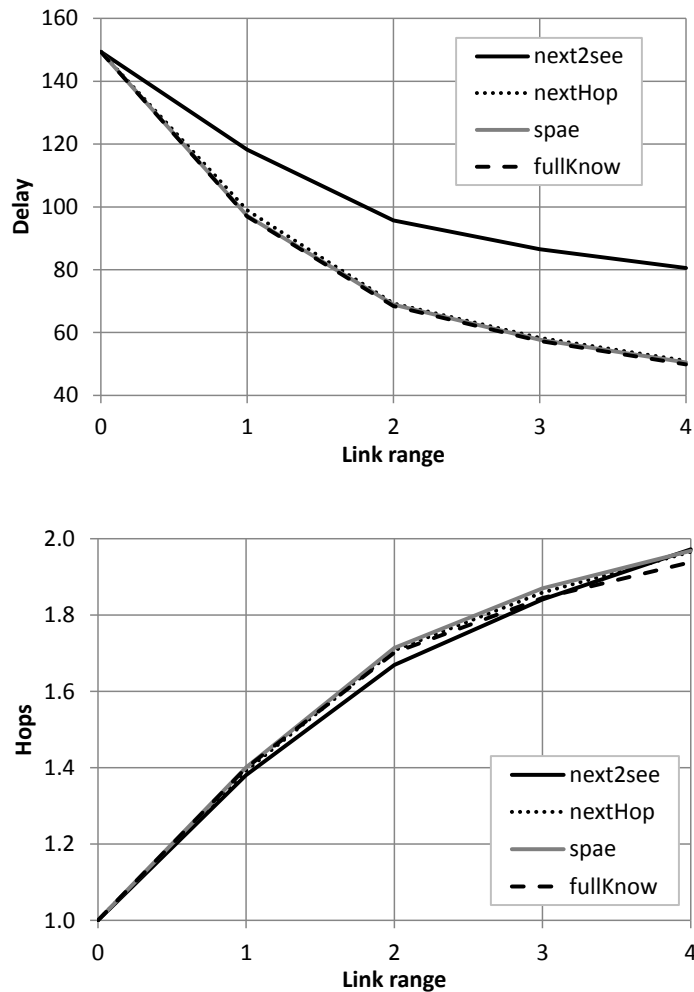
Figure 9 – Delay (top) and hop count (bottom) vs. node to node link range for low contention and infinite TTL

3.4.1.2   High Contention

At high contention (0.9), the benefits of resource consideration are clear, with *nextHop* showing a clear increase in delay, while Spae is able to more closely match the full knowledge approach. This can be attributed to Spae's ability to effectively route around congestion and reduce the frequency at which packets fail to traverse their originally

planned journey. Interestingly, *next2see* suffers an increase in delay with an increase in link range, which can be attributed to an increasing number of packets being sent to nodes expecting imminent contact with the destination, despite no increase in their download capacity. This will inevitably result in a greater number of packets remaining in the buffer, which are transferred on to the next unwilling recipient. Also noteworthy is the magnitude of the delay, which is 4-5 times greater than that seen at low contention (for Spae), which illustrates the importance of this characteristic on network performance.
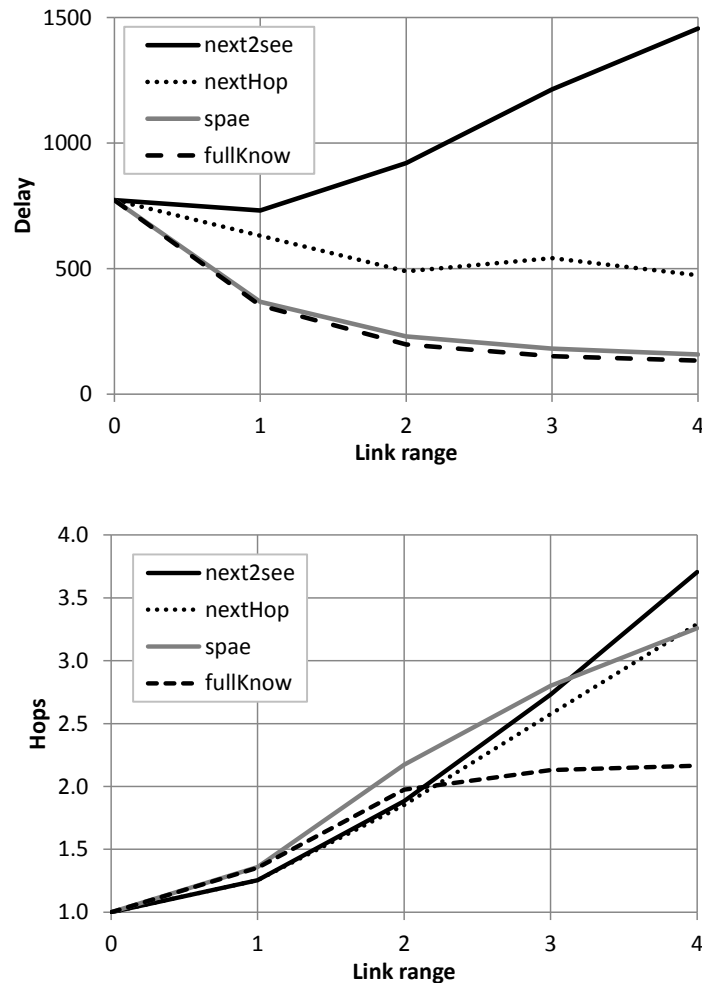
Figure 10 – Delay (top) and hop count (bottom) vs. node to node link range for high contention and infinite TTL

In terms of hop count, all protocols exhibit a slight increase over their low contention counterparts up to a link range of ∼2, however the *fullKnow* method is able to curtail this increase due to its ability to avoid real zones of congestion. Spae can only predict this congestion, which becomes more difficult as the link range, and thus the number of

potential journeys, goes up. Spae appears to show a reduction in the rate of hop count increase, with increased link range, while both *next2see* and *nextHop* both indicate a generally increasing rate.

3.4.2   Finite Time To Live

The following sections show results of simulations carried out using data that expires after 200 time steps. This value has been selected since it will allow for complete data delivery at low contention, high link range (best case), but <100% delivery in most other situations, for this particular network scenario. A finite TTL results in data being dropped from the buffer upon expiry, or, in the case of *nextHop*, Spae and *fullKnow*, at the moment it is known that no feasible journey exists.

3.4.2.1   Low Contention

In the case of low contention (0.1), a similar result in terms of hop count is seen to that at infinite TTL, while delay is significantly lower at low link range. This is reflected in the delivery ratio result being <100%, and should indeed be expected, since those packets that require longer than the allowable time will be dropped from the network. The effect is in fact two-fold, since dropping of packets from the network also frees up capacity for younger data to be delivered, resulting in its delay being reduced further.

The lower rate of reduction in delay from low to high link range, compared to the infinite TTL equivalent, is a result of the respective increase in delivery ratio, such that while a greater number of timely journeys are available at increased levels of networking, there is more demand over those journeys.

Once again, at low contention it is evident that only *next2see* lacks the ability to track performance of the full knowledge protocol, since unsuccessful journey traversal is rare.
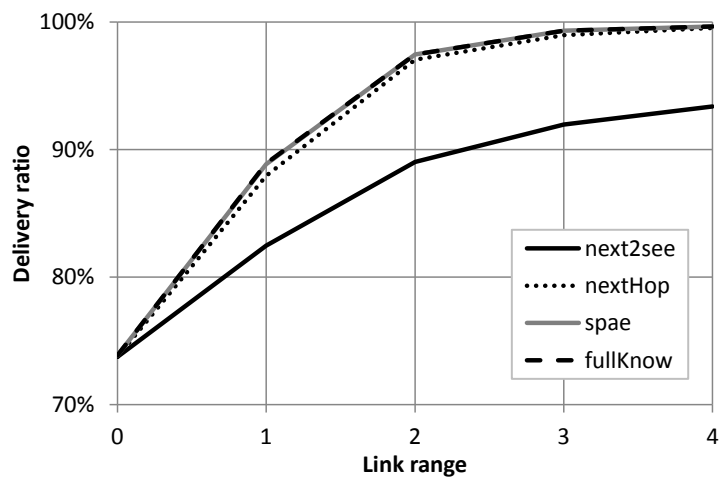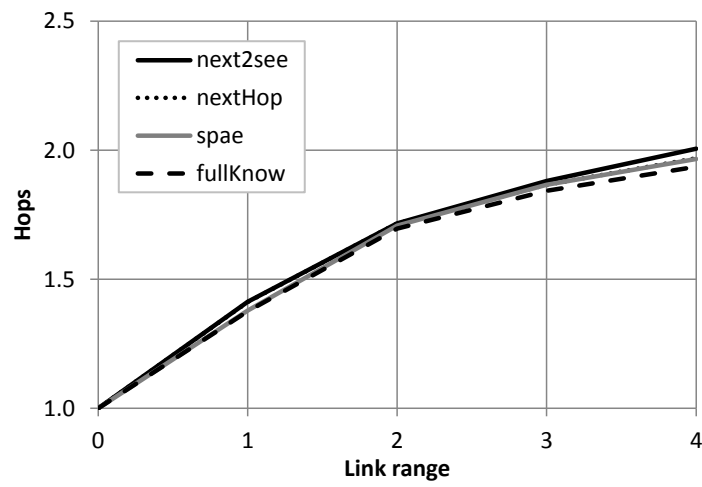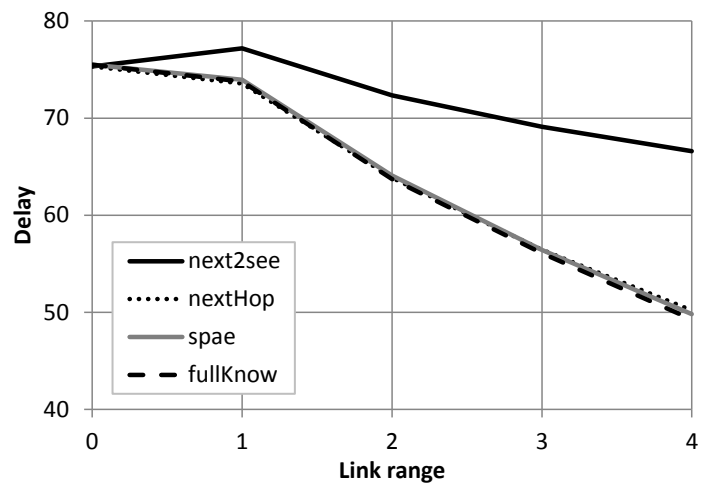
Figure 11 – Delay (top), hop count (middle) and delivery ratio (bottom) vs. node to node link range for low contention and finite TTL

3.4.2.2   High Contention

The results of high contention are significantly more complex than those in the previous sections, since there exists the coupling of packet dropping and high demand for journeys. Spae performs well in terms of closely tracking the achievements of its full knowledge rival, however once again it relies on a greater hop count at high link range due to an increasing number of incorrect journey selections as the link range increases.

One might be tempted to suggest that *nextHop* offers better performance than both Spae and *fullKnow* based on delay results alone, however it is clear from the generally lower delivery ratio that this achievement is falsely represented. Indeed, one could perhaps argue that should a low delivery ratio be considered acceptable for the mission then this could be a situation for selecting this strategy, however it is posed that this is an inefficient approach to data acquisition and would be better achieved by simply acquiring less data.
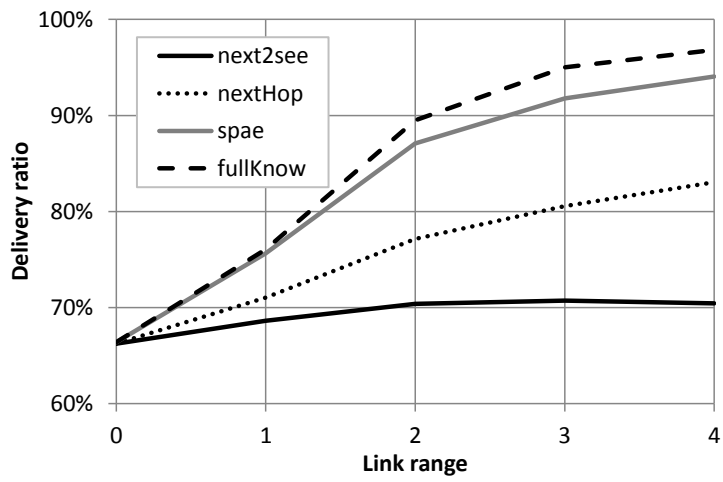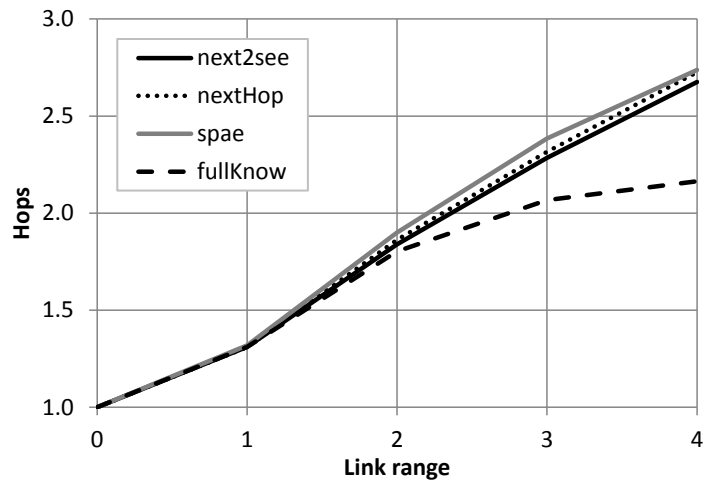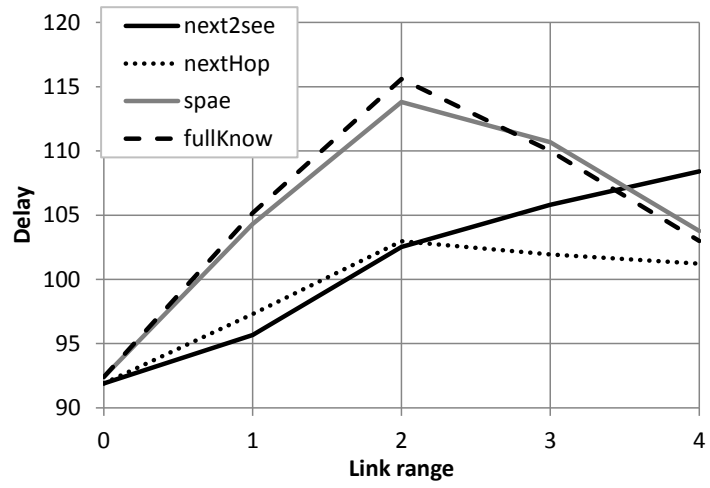
Figure 12 – Delay (top), hop count (middle) and delivery ratio (bottom) vs. node to node link range for high contention and finite TTL

3.4.3   Spae behaviour

Whilst the above plots illustrate how Spae compares to other protocols in terms of data routing performance in discrete simulation examples, it does not make clear the effect of coupled design variables, in particular link range and contention. Again, it is seen that inter-node data rate has a minor effect on performance in comparison to these metrics, such that it is omitted from the results shown below.

3.4.3.1   Infinite Time To Live

The plots in Figure 13 show the effects of link range and contention on delay (top) and hop count (bottom). It is evident that at low contention an increase in link range from 0 to 4 has the effect of a reduced delay by ∼50%, while at high contention the benefits are significantly greater. Here, delay is reduced by almost 80%, from 798 to 174, which is greater at higher node to node data-rate. Regarding hop count, as would be expected, a higher link range results in higher hop count throughout, owing to the greater number of potential journeys available to each packet. At high contention, hop count increases significantly with an increase in link range, which can be attributed to the need for packets to be routed around congested areas by taking longer, more convoluted journeys.
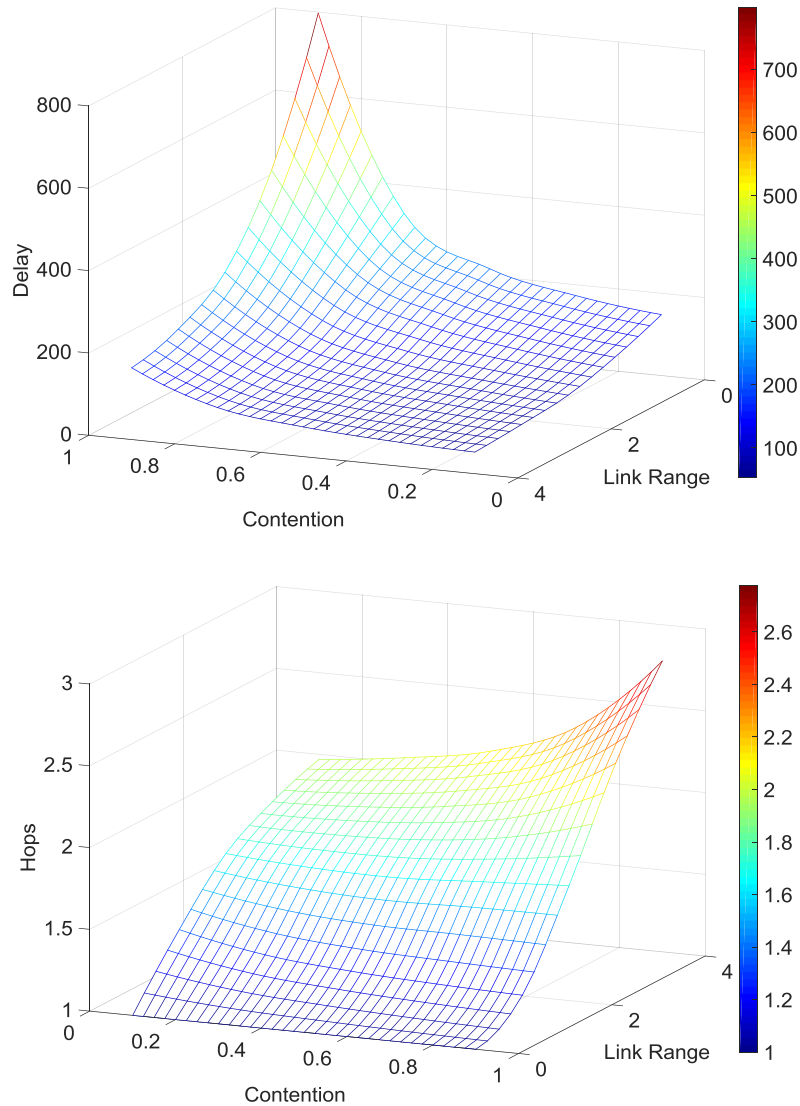
Figure 13 – Delay (top) and hop count (bottom) vs. contention and link range for Spae routing at node to node data rate = 1

3.4.3.2   Finite Time To Live

The delay performance of Spae for a network delivering data with finite TTL (200 time units) must be considered in tandem with the delivery ratio, where a lower value for the latter allows for an artificial reduction in the former because of reduced demand over edges. This is clear from the drop in delay at low link range, which does not align with the same dataset in the infinite TTL case, but does match where there is also a significant drop in delivery ratio. There is a less prominent correlation between contention and delivery ratio, although some relationship does exist, where a greater fraction of the packets are dropped in congested networks. The hop count performance is similar to that seen in the infinite TTL case, in terms of both quantity and trends.
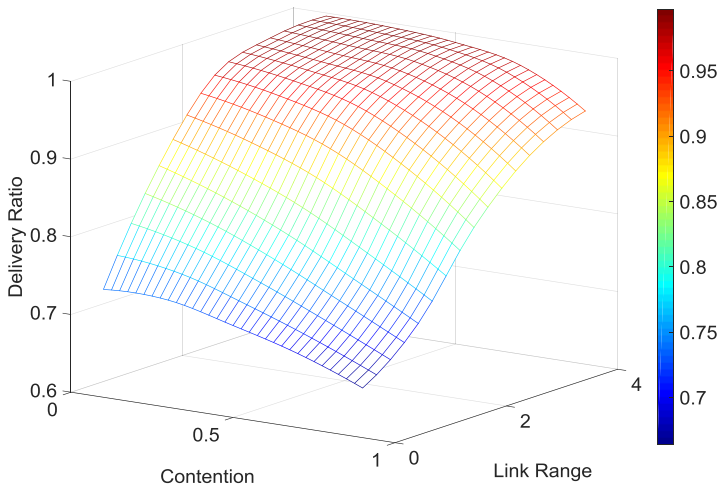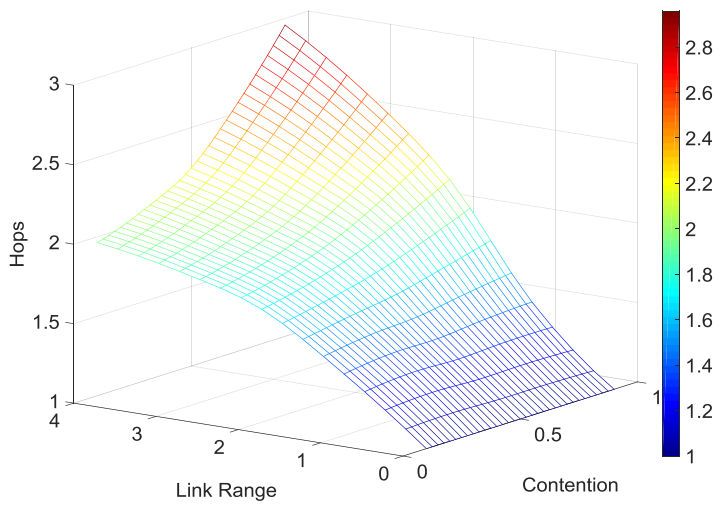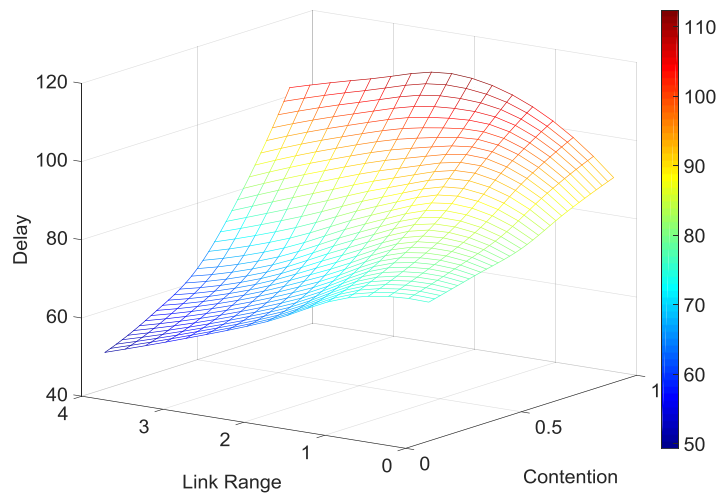
Figure 14 – Delay (top), hop count (middle) and delivery ratio (bottom) vs. contention and link range for Spae routing at node to node data rate = 4

3.5 Case Study 2: Federated Satellite System

In this scenario, a federated satellite system (FSS) is simulated, in which 10 platforms in low Earth orbit agree to form part of a federation such that data is shared in order to reduce delivery delay from that achievable by each member operating in isolation. Each platform can be considered to be owned and operated by a different set of stakeholders, making this a simplification of what would likely be a complex supplier-customer interaction [111], however it can be argued that these results show the theoretically achievable performance that could be expected should systems partake in a federation. The analysis goes some way to representing a multi-stakeholder interaction in applying a node availability of <100%, resulting in certain data transfer windows being refused by the intended recipient. As above, node unavailability is not known by other nodes in the network *a priori* (except for in the *fullKnow* case), however a node's long term expected availability is known throughout the network such that journeys can be selected with this information in mind. The satellite orbits are generated at random in order to best represent a FSS, such that the federation does not exhibit characteristic topology repetition associated with dedicated constellations, such as Walker-delta patterns [112]. While Spae could indeed be applied to such a constellation, it would perhaps be beneficial to develop a bespoke routing algorithm that can best exploit this repetition, especially in the case of deterministic data acquisition.

Details of the simulation parameters are presented in Table 7.

Property	Value	Comment
Number of satellites	10	Satellites on which data is acquired according to a geometrical distribution
Number of ground stations	3	Kourou, French Guiana Villafranca, Spain Perth, Australia
Orbit altitude	400km – 800km	Uniform distribution
Orbit inclination	0° - 100°	Uniform distribution
Minimum elevation	10°	Elevation above the horizon for communication with ground station
Data rate to ground station	150Mbps	
Node availability	50% - 100%	Probability of a node being available for data transfer (node availability constant during simulation)
Simulation length	5 days	Duration of orbit propagation
Simulation time step	10 seconds	Size of simulation step
No. simulations	25	Number of simulations executed
Contention	0.5	Variable parameter defining the ratio between generated and deliverable data
Storage cost ($\mathcal{S}$)	1 per second	Cost per unit time
Transmission cost ($\mathcal{T}$)	1800 seconds	Cost per hop (equivalent to 30 minutes of storage)

Table 7 – Federated satellite system model properties

Unlike in the random mobility case, there exists a non-zero transmission cost, which equates to 30 minutes of storage, which is to represent the financial cost that would likely be demanded by the recipient of a transfer event, in order to offset the burden of having to store and forward data. This has the effect of delay not being the exclusive performance metric, such that a long journey with few hops might be favoured over a short, many-hop journey.

The results that follow illustrate the effects of varying link range, data rate and TTL on the performance metrics of delay, hop count and delivery ratio (finite TTL only), at a constant contention of 0.5.

3.5.1   Infinite Time To Live

In satellite systems that exploit high capacity solid-state storage, buffer resource is likely to be cheap and plentiful such that dropping of packets due to a full buffer could be considered unlikely. Therefore, to ensure complete data delivery, a high TTL is required to avoid dropping due to expiry. The following two sections show simulation results with infinite packet TTL.

3.5.1.1   Effect of ISL Range

Figure 15 shows the effect of an inter-satellite link range between 0km and 2500km, with a constant ISL data rate of 150Mbps, on both delay and hop count. *Spae* can be seen to offer comparable delay performance to the full knowledge upper bound, while *nextHop* follows the same trend but at an almost constant additional delay from the two more sophisticated protocols (at ranges greater than 1000km). The delay-effectiveness of *Spae* can be seen to diminish relative to *fullKnow*, at higher ISL range (i.e. at higher connectivity), which can be attributed to the greater number of journey options to a promising penultimate node and hence a greater likelihood of a congested path being taken. That being said, the difference in delay between the *fullKnow* results and those from *spae*, at a link range of 2500km, is 37% of the difference in delay between *fullKnow* and *nextHop* at the same link range, which is evidence of its superiority.

The hop count for the three best performing strategies is comparable, while that of *next2see* suffers from inefficient routing along sub-optimal journeys.
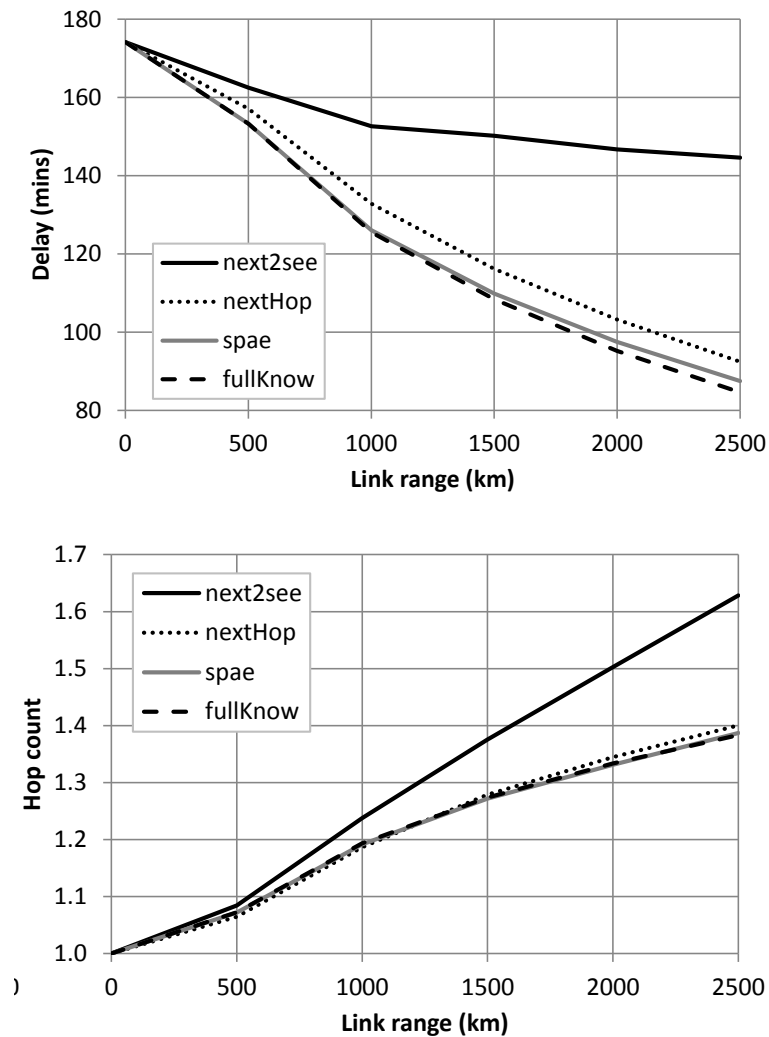
Figure 15 - Effect of ISL range on delay (top) and hop count (bottom)

As would be expected, an increase in ISL range has value in terms of delay, since contacts that were perhaps not available with a shorter range are realised. This is effectively a measure of the network connectivity. Statistics of network connectivity are shown in Figure 16, with respect to ISL range. Noteworthy is that calculation of connectivity here considers all connections between satellite pairs, i.e. an effective node availability of 1 everywhere, and no connections to ground stations.

Figure 16 tells us that, for example, at a link range of 2000km, on average each satellite pair are in contact for approximately 2% of the time.
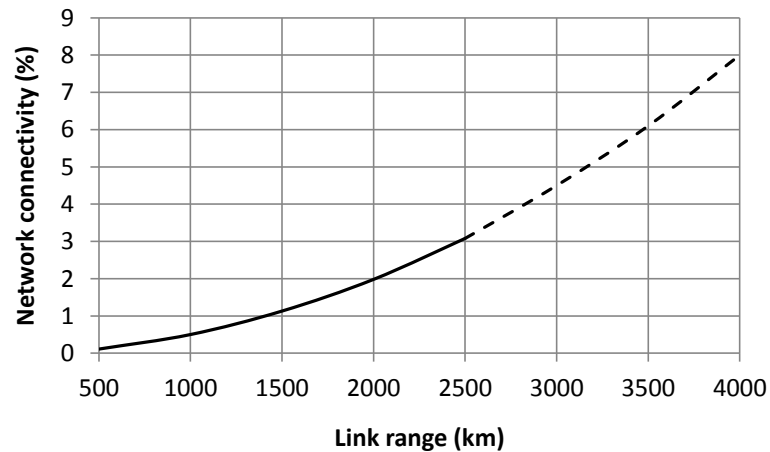
Figure 16 - Network connectivity at various ISL range (analysis includes range up to 2500km, but plot extended to 4000km* for illustrative purposes)

3.5.1.2   Effect of ISL Data Rate

Results for various ISL data rates are shown in Figure 17, with a constant link range of 1500km [113]. It is clear that a data rate above ~100Mbps is of limited value in this scenario, which suggests that the majority of data intended for transfer between nodes is achievable at this rate.

---

* Satellites in the Iridium constellation have an intra-plane separation of ~4088km

The hop count trend is similar to that seen with varying levels of link range (Figure 15), where by the three most sophisticated algorithms are comparable.
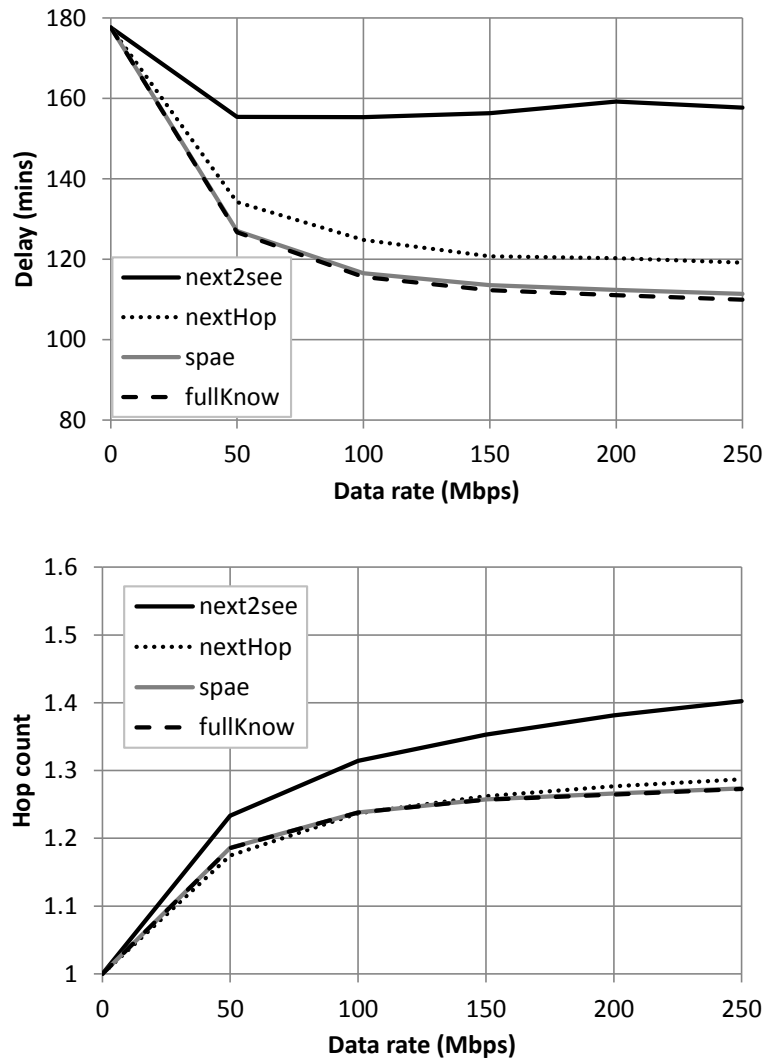
Figure 17 - Effect of ISL data rate on delay (top) and hop count (bottom)

3.5.2   Finite Time To Live

In the following sections, a finite TTL is imposed on the data traffic, which reflects the situation in which a rapid response, or low delay, is required, such as for weather now-casting and high value financial transaction applications.

3.5.2.1   Effect of Finite Time To Live

Given an ISL range of 1500km and data rate of 150Mbps, performance at different TTL values is shown in Figure 18. As would be expected, the average delay increases with

increasing TTL, which itself represents the maximum possible delay, since the packet would be dropped should it exceed this value. *Spae* tracks *fullKnow* closely in terms of delay performance and delivery ratio, with *nextHop* exhibiting a consistently lower performance in the case of the latter metric. This can be attributed to the greater frequency of situations where packets are routed along overly congested paths, which then need re-routing, and therefore expire before reaching the destination. This hypothesis is reinforced in the greater hop count exhibited by both *next2see* and *nextHop*, especially at lower TTL.
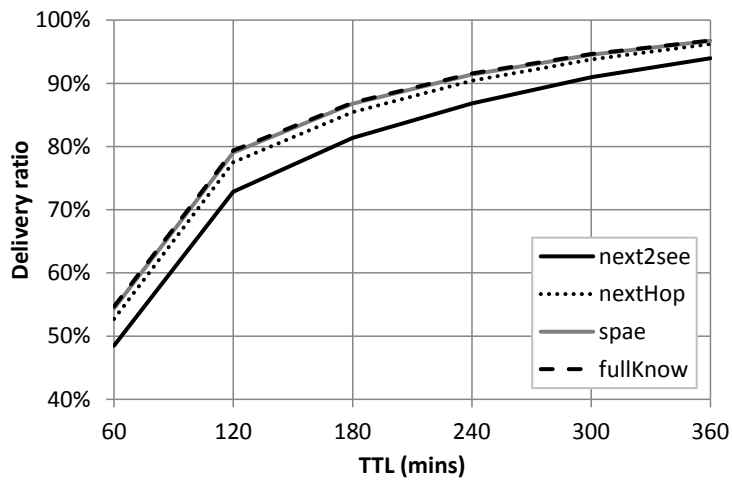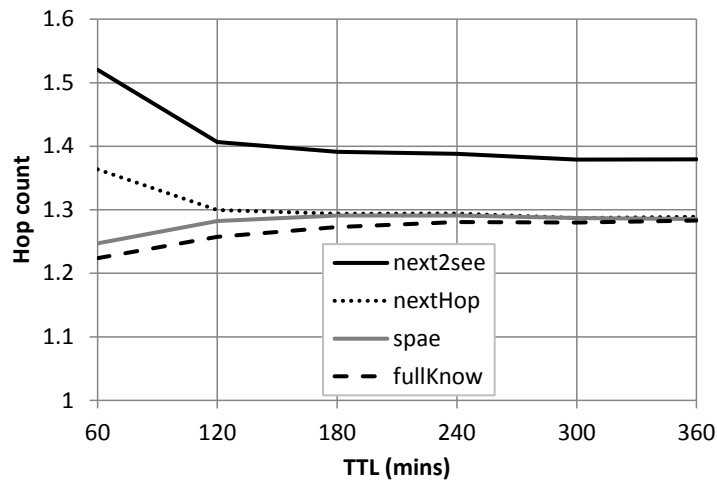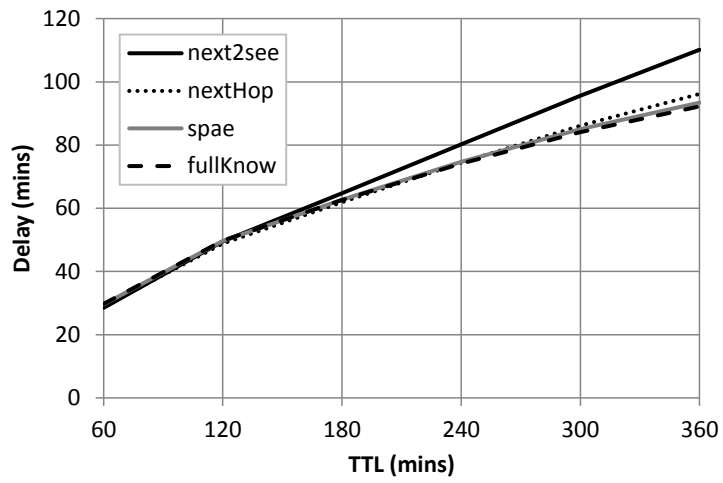
Figure 18 - Effect of TTL on delay (top, hop count (middle) and delivery ratio (bottom)

3.5.2.2   Effect of ISL Range

The effect of ISL range is shown below (Figure 19), with an ISL data rate of 150Mbps and TTL of 180 minutes. Note that the axis scale in the plot showing delay (top left) is significantly smaller than that of the plot in the previous section (Figure 18 – top left), such that ISL range appears to have a relatively small effect on delay in general for a fixed, finite TTL. It is clear however that a trend towards lower delay is emerging at greater connectivity for all protocols except *next2see*. Interesting is the turning point in the delay plot, for *nextHop*, Spae and *fullKnow*. This is due to contact events up to this point being generally saturated with data, which offers little scope for routing packets effectively around congestion. Beyond this point however, additional connections appear between nodes that were previously out of view, such that less often are the edges saturated, allowing the most favourable journey to be taken by more packets, resulting in a delay reduction. For hop count, an increase in ISL range results in an increase in overall performance, but at a decreasing rate. The delivery ratio would be expected to increase beyond the range shown here, until the network reaches a level of connectivity that enables delivery of all data prior to its expiry.
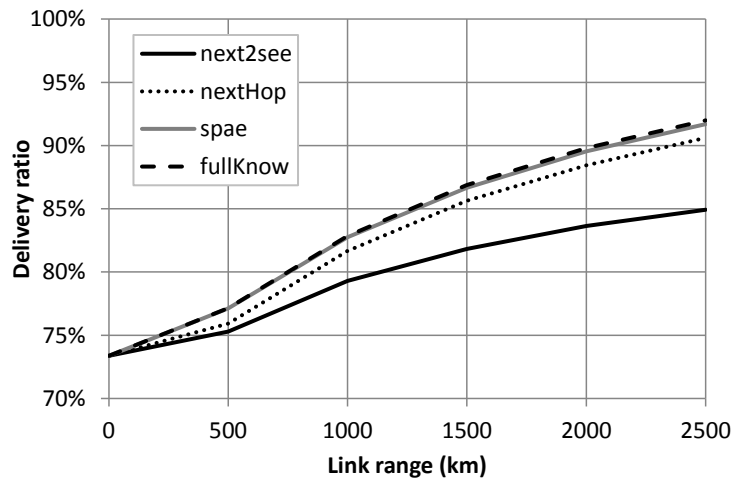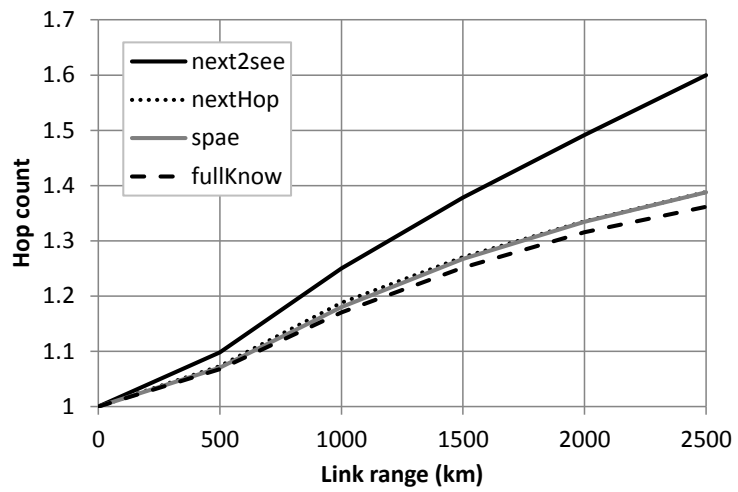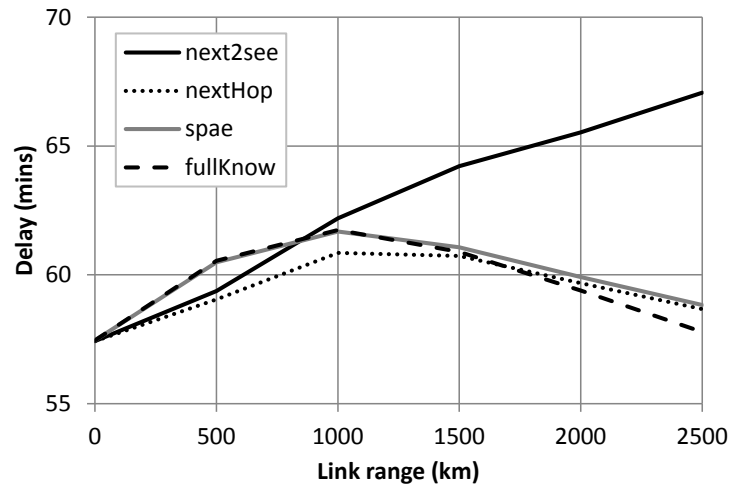
Figure 19 - Effect of ISL range on delay (top), hop count (middle) and delivery ratio (bottom)

3.5.2.3   Effect of ISL Bandwidth

The final set of results (Figure 20) show the effect of varying ISL data rate, while maintaining constant range of 1500km and a TTL of 180 minutes. As also seen in the infinite TTL case (Figure 17), the benefit of data rate at levels >100Mbps is limited, again indicating that the majority of packets that want to be transferred during a contact, are able to be transferred. However, it is considered probable that a more congested network would benefit from greater data rate, since the data traffic level will be higher. *Spae* continues to exhibit comparable results to the packet-optimal *fullKnow* strategy, except for in hop count performance where it more closely tracks *nextHop*. Delivery ratio from the *nextHop* protocol remains lower than both Spae and *fullKnow*, throughout.
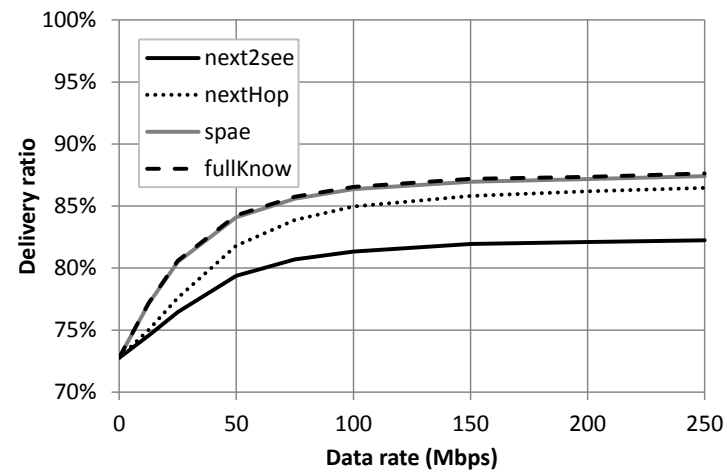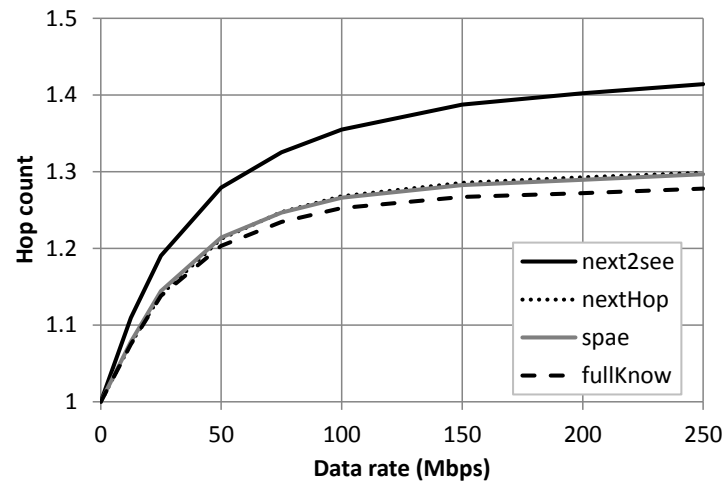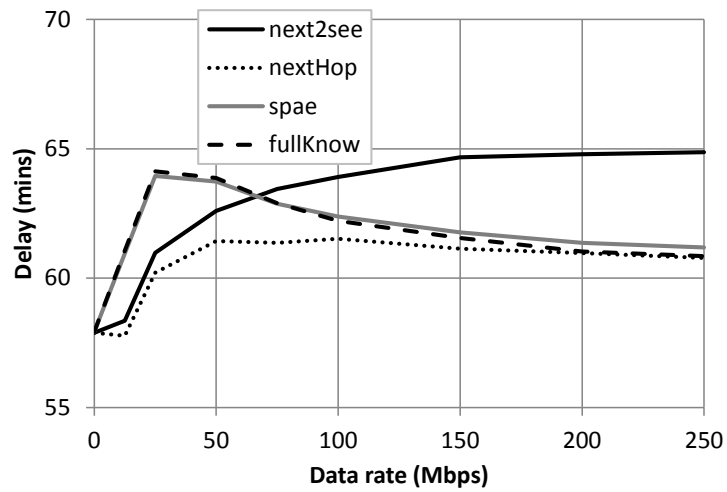
Figure 20 - Effect of ISL bandwidth on delay (top), hop count (middle) and delivery ratio (bottom)

3.6 Discussion

It is evident throughout all of the simulations conducted that Spae offers network performance that is closer to the optimal full knowledge strategy than either the *next2see* or *nextHop* protocols. The cost of this increase in performance is an increase in computational time, driven by the packet-by-packet routing that Spae provides. The exact increase in computation is dependent on the type of network and data traffic being routed. Spae's increase in performance is testament to the fact that more information is available during the routing process than for the other protocols, however the benefits increase considerably at higher levels of contention (i.e. a more congested network), where the other protocols would often route packets along journeys that do not then materialise.

Chapter 4

Satellite Network Design

The evolution in both the approach to design, and the application, of satellite networks has been vast over the past 60 years, motivating a resurgence in the popularity of multi-agent space systems since the downfall of a number of ambitious satellite constellations; Iridium, Teledesic and Globalstar [114], of which only Teledesic never recovered. From the purely analytical solution of a polar constellation for continuous Earth coverage [115] to the concept of a distributed, multi-institutional Federated Satellite System (FSS) [116], the concept of a satellite network has evolved a great deal, but their appeal remains as an effective alternative to traditional space solutions.

The term satellite network, or space network, is broad and could be analogous to a generic collection of space-based systems working toward a set of shared goals. Indeed, an even more general description could include a single space-based asset operating in conjunction with a set of ground-based assets, however since inter-satellite networking is the focus of this dissertation, this scenario will be omitted.

It is thought that the concept of satellite networks was posed as early as 1945, when a 3-satellite constellation in Geostationary orbit (GEO) was presented for the purpose of global radio coverage [117]. Somewhat later, interest turned to regional or global coverage from low Earth orbit (LEO) [115], with the aim being able to achieve a particular level of coverage using the minimum number of platforms. This work was extended in [112], [118], in which the now well-established Walker Delta and Walker Star patterns are introduced. Other improvements and developments were made over the following years, focussing on orbital plane positioning [119], higher-order coverage [120], closed form solutions to the continuous single and multiple coverage problem [121], optimal phasing of satellites between planes [122], computation efficiency for large constellation topology determination [123] and optimisation methods for constellations with discontinuous coverage [124].

A summary of the most common examples of satellite networks is presented in Table 8.
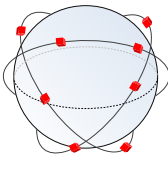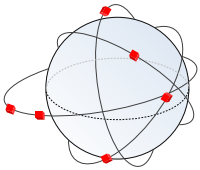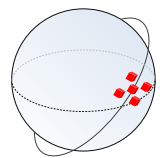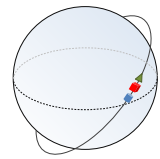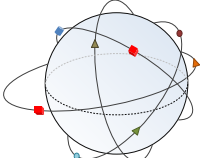
Type	Connectivity*	Graphic	Examples
Dedicated constellation	Topology dependent† Periodic or continuous		NAVSTAR (GPS) Iridium Globalstar O3b TDRSS & EDRS
Ad-hoc constellation	Topology dependent Non-periodic		Flock (Planet) Lemurs (Spire)
Swarm or Cluster	Continuous		SWARM (A&B) Cluster THEMIS
Fractionated System	Continuous		F6 (DARPA) Proba-3 A-Train
Federated Satellite System	Intermittent Non-periodic		ISS with TDRSS Sentinel 1&2 with EDRS

Table 8 – Satellite networks types (note: nodes of similar shape & colour share design parameters)

While the investigations referenced in the previous paragraph are interesting from an academic perspective, exploiting the minimum number of satellites is not necessarily the optimal configuration. For example, deploying a greater number of satellites from fewer launch vehicles might achieve comparable or better performance to that of some base-line mission, in addition to more graceful degradation in the event of off-nominal operations, but have a lower overall cost. It is this cost vs. performance, or utility trade-off that shall be the focus of the following chapters and has been applied previously, such as the cost per function metric used in [3]. Other value-based strategies have been applied, such as a coverage per mass figure of merit [125], in which vastly different architectures are compared using mass as a representation of cost.

---

* Refers to the potential for communication between nodes in the network, as defined by standard graph theory terminology

† A dense constellation may have connectivity due to proximity of nodes, however a sparse constellation may be intermittently connected.

4.1 Inter-Satellite Networking

The concept of inter-satellite networking (ISN) is not new, beginning with the Apollo programme in 1969, which used a VHF link for communication between the Lunar Module and Command Module. Since then, we have witnessed the launch of the first Tracking and Data Relay Satellite (TDRS) in 1983, which uses a multi-access S-band antenna for inter-satellite communication. Fifteen years later, the 66-satellite Iridium constellation became operational, offering low-latency point-to-point communication between any two users on the Earth using Ka-band inter-satellite links in order to move data through the constellation. Optical (laser) inter-satellite link (ISL) technology was first used in 2001 [126], between the European Space Agency's Artemis satellite in GEO and the French space agency's (CNES) SPOT-4 Earth observation platform in LEO. Recently [127], an optical inter-satellite link was shown to be an effective solution to achieving near real-time images from the Sentinel-1A satellite via the Alphasat[*] platform, demonstrating another LEO-GEO link. A similar laser-based ISL technology is due to fly on the European Data Relay System (EDRS).

Despite the above examples, the concept of ISN has not been exploited extensively in LEO, in particular between satellites with non-negligible relative velocity and/or different mission objectives. This application has the potential to open up new markets, providing low latency data delivery from both large, and modest sized constellations and federations.

4.1.1 Inter-Satellite Link Technology

The presence of an ISL provides the possibility for data transfer, at a particular data rate[†], between two satellites that are within a certain range of each other. There may also be a requirement for the platforms to exhibit a specific relative attitude in the case of directional antennas or optical systems. Transmission and reception of data will require power for operation of the sending and receiving antennas, respectively, which must not exceed some upper bound defined by the platform-specific constraints. Finally, delivery of data, unless maintaining a copy on board, will free-up storage

---

[*] Noteworthy is that the Alphasat platform is a precursor to the satellites that make up the EDRS constellation.

[†] It is common for the term *bandwidth*, used in the context as done so in computing terminology, to replace data-rate. They shall be used interchangeably herein.

capacity, while reception of data will consume storage capacity, which cannot exceed the constraints of the storage device on board.

4.1.2   Impact on System

Incorporation of an ISN sub-system has implications beyond the cost of its own development and manufacture, in the form of the effects from additional system-level mass, volume and power required to support the capability. Furthermore, tertiary effects exist such as higher launch cost due to the larger platform. During the conceptual design phase, analytical methods could be used to consider these system- and mission-level impacts, thus separating the direct development cost of the ISN capability.

Unless defined otherwise, for example where an off the shelf device may already be fully defined, evaluation of the link budget will allow power demand to be calculated such that relationship between power, range, data rate, bit error rate (BER), frequency and bandwidth of an ISL device can be found. Following this, and given a certain power demand, a mass can be estimated by either analysing previously flown systems with similar performance or carrying out a bottom-up design exercise.

4.2   The Design Process

There exists a plethora of good practice methods when referring to design of space (or other) systems, however much of the literature shares common elements. These are described in Figure 21, where the first three plots are performed almost universally, with the final plot, mapping to the value space [128], being applied to problems with large numbers of objectives/attributes.
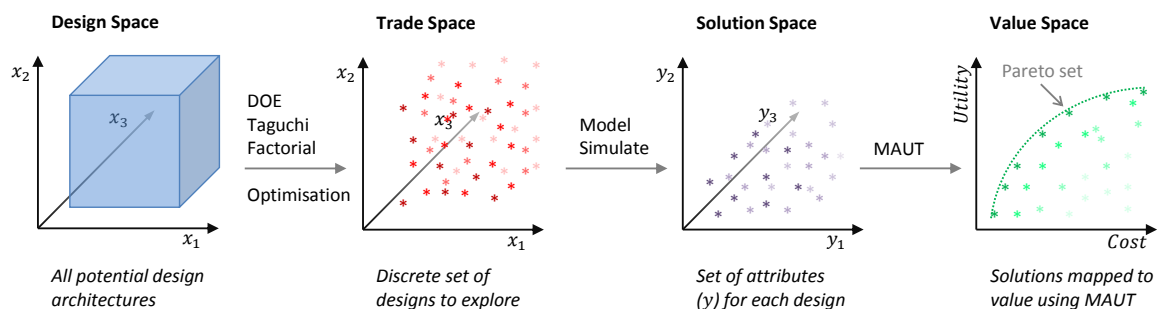
Figure 21 – Typical design process

In addition to the processes shown in Figure 21, it is also commonplace to explore how a design behaves over its lifecycle [129], [130], which is critical in the understanding of the true mission value [128]. This is often a result of a desire to understand how uncertain change-parameters affect a design's utility, such as a change in market demand, system failure or technology obsolescence. Consideration of this phenomenon will be made in this dissertation, with emphasis on stochastic failure, defined further in Section 4.3.

4.2.1   Design Space

The majority of mission studies begin with generation of a design space, which is typically described by a set of discrete or continuous design variables, with upper and lower bounds. A finite number of design architectures, each comprising a unique set of design variables, are then carried forward to form the trade space. This process can be accomplished using a variety of methods such as full factorial analysis (where all possible design architectures are selected from a discrete set), Taguchi's methods [131], random selection, or some informed selection as found in optimisation routines.

4.2.2   Trade Space

A trade space consists of the set of design architectures that have been selected for evaluation to identify their respective attributes. The analysis can be via analytical and/or numerical methods, but typically involves simulation to emulate operational behaviour. Outputs from this analysis form the solution space, which comprises a set of equal size to the trade space, described on a set of stake-holder-defined attributes.

4.2.3   Solution Space

The solution space is a mapping of each design point in the trade space to its location with respect to the set of attributes considered important by stake-holders. A promising design is one that satisfies the project requirements and performs well over many/all attributes, or indeed simply exhibits high value in the case of combined multi-attribute utility (MAU) function (see Section 4.2.4.2). Note that often, attributes may be conflicting in terms of their performance measure, e.g. it is common to desire high spatial resolution (narrow beam width and low altitude) and high revisit rate (wide beam-width and high altitude) from an observation payload, such that trade-off can be

challenging. Multi-attribute utility functions can prove useful in such situations, shifting the decision making to true stake-holder preferences, as opposed to results-based intuition [132].

4.2.4   Value Space

The value space provides the decision maker with a simple representation of a potentially complex solution-set, through the use of a value proposition. Simply put, value allows for the mapping of attributes to a metric which concisely expresses the preferences of stake-holders such that a design of higher value is always preferred to one of lower value. This can be formally represented as the binary relation value-function

$$A \succcurlyeq B \Leftrightarrow V_A \geq V_B,$$

4.1

where $A = \{a_1, a_2, \dots, a_n\}$ and $B = \{b_1, b_2, \dots, b_n\}$ are different architectures made up of n design variables. In other words, design A is always preferred over design B if the value of A exceeds that of B [133]. By formulating the proposition in this way, it is possible to remove much of the difficulty faced when presented with a solution set of high dimensionality. For the majority of missions, the amount of function one gets per unit cost is a convenient measure of value [125], [3], but there are rare cases where cost is not a factor, in which case value would be measured against some other parameter. The work in this dissertation will focus on the former, with cost being a factor in the value proposition.

4.2.4.1   Value

Value has many definitions and uses, but can be universally described as the measure of worth, be it monetary or otherwise. In the case of mission value, as used here, the value function must be constructed according to the specific mission needs. For a mission in which financial revenues are the exclusive measure of utility (U), the value (V) can be generally described as

$$V = \frac{U - C}{C},$$

4.2

where C is the total cost.

More generally, where utility is perhaps a combination of performance- and financial-based attributes, the value proposition often takes the form

$$V = \frac{U}{C},$$

4.3

which applies well to missions with scientific or military objectives that cannot easily be quantified financially. This is effectively the inverse of a cost per function metric introduced in [3].

4.2.4.2 Utility

Utility can be defined as the measure of effectiveness held by a single attribute, or set of attributes, such that a high utility is preferred from the point of view of performance. While attributes are described in terms of their associated units and quantity, utility offers a normalised quantification between attributes that can be used for direct comparison and/or combination. For example, in the case of Earth observation, spatial resolution of an image is measured in meters and is generally minimised for better performance, while revisit rate is measured in frequency, and is generally maximised. By mapping these attributes to a normalised utility, which has a pre-defined maximum and minimum bound, they can be compared directly, by each having a value of between 0 (worst) and 1 (best). Furthermore, by scaling and combining multiple attributes, using methods from multi-attribute utility theory (MAUT) [132], it is possible to exploit a single utility metric that describes the overall performance of the system. This approach has been used to good effect in the aerospace community as a way to combine attributes and better understand stake-holder preferences during trade-space exploration exercises [134].

There exist a number of underlying axioms and assumptions that must be understood before implementing MAUT. These include attribute transitivity, e.g. if $X \succ Y$ and $Y \succ Z$, then $X \succ Z^*$, and the level of independence (preferential, utility and additive) with which attributes can be treated with respect to each other.

---

* the symbol $\succ$ denotes preference of the term on the left hand side over that on the right hand side

On the independence parameters:

i. *Preferential independence* is where an attribute's preference is unchanged when considered alongside any other attribute. If X and Y are two attribute sets, then if $(X_1, Y_1) \succ (X_2, Y_1)$, preferential independence would dictate that $(X_1, Y_2) \succ (X_2, Y_2)$ always.

ii. *Utility independence* is where the utility of an attribute in X remains unchanged for varying attributes in Y.

iii. *Additive independence* is a characteristic whereby the decision maker expresses indifference to combinations of different attributes of a particular utility. If X and Y are two attribute sets and if $(X_1, Y_1) = (X_2, Y_2)$, then additive independence would dictate that $(X_1, Y_2) = (X_2, Y_1)$. In other words, if we are indifferent to choosing X_1 over X_2, when combined with Y_1 and Y_2 respectively, then we remain indifferent should they be paired with Y_2 and Y_1 respectively.

While a general form for multi-attribute utility (MAU) calculation does exist, the multiplicative form offers a simple and often acceptable approach, for which preferential and utility independence must be demonstrated. Moreover, the presence of additive independence allows use of the simpler, additive form. These independence criteria, along with decision-maker preferences, are typically derived through interviews, where lottery questions* are posed to identify the stakeholder preferences with respect to individual, and combinations of, attributes. From this information, it is possible to derive single-attribute utility (SAU) functions, which represent the utility (u) as a function of each attribute x, where $u(x) \in [0,1], \forall x$.

---

* Consider the simple case where one can either have a 50% chance of winning £100, or a 100% chance of winning £50. The answer to this, and follow on questions, would be used to identify one's preference as to how the performance (money) maps to utility.

The MAU function ($U(x)$) can be derived using the multiplicative form, as

$$U_{mult}(x) = \frac{1}{K}\left(\prod_{i=1}^{n}[Kk_iU_i(x_i) + 1] - 1\right),$$ *4.4*

where k_i is the weight applied to the attribute i, and K is a scaling constant that can be found through solving

$$1 - K = \prod_{i=1}^{n}(1 + Kk_i).$$ *4.5*

The simpler, additive form is defined by

$$U_{add}(x) = \sum_{i=1}^{n}k_iU_i(x_i).$$ *4.6*

Determination of the set k_i can be difficult, but again it is typically established using lottery questions during interview [132]. It is important to understand the relationship between the number of attributes and their relative influence on the overall utility. Consider the comparison of two systems, i and j whose multi-attribute utility is defined by two attributes, A and B, which for the sake of argument have equal weighting. For each particular instance of the system, its values of A and B will give rise to a utility, U_i and U_j. Say that the overall utility of j is twice that of i, at 0.8 and 0.4 respectively, such that one could feasibly spend twice the amount on j than i and retain equivalent value (based on the above definition in equation 4.3). Now introduce a third attribute, C, with equal weight to A and B, on which both i and j perform equally well, say 0.5. Now $U_i = 0.43$ and $U_j = 0.7$, such that we can spend only 1.6 times as much on j than we do on i in order to retain equal value, despite no changes to the design or their respective capabilities. The change in value has been artificially introduced from the additional attribute, which reduces the relative influence from each individual attribute. This effect can be ignored when one is only interested in whether design i is of greater or lesser value than design j, but becomes important when equating absolute monetary figures to relative value. This phenomenon is addressed in Section 5.2.2, through introduction of a Maximum Price Value, which differs from the previously introduced

Preference Value and allows conversion from the purely preference-based value space to one where financial decisions can be made.

There are occasions when attributes can/must remain separate and in their ordinal form, as found in the solution space. Two or three attributes can be evaluated effectively on traditional two- or three-dimensional plots, respectively, however this is generally where human analytical spatial cognitive ability ends [135] and alternative methods of visualisation for higher solution space dimension must be introduced, such as radar plots. It is here that MAUT presents most benefit. While other factors are often important, such as how a stake-holder's preference may change throughout the life of a mission, or how sensitive a particular design variable is to the overall system utility [128], evaluation of these factors are considered beyond the scope of this dissertation, but are recommended for future work on this subject.

4.2.4.3   Cost

The cost of a space mission includes contributions from hardware, software, labour (design, development, assembly, integration, test and verification) facility development, rental and usage, launch, operations and maintenance, all of which can vary significantly depending on the type of mission. In addition, owing to the long duration over which many missions are developed, inflation must not be neglected and risk may be significant for certain systems.

Cost modelling as a discipline is known to be complex, with a number of problems yet to be addressed [136], [137]. The method of cost modelling, be it a top-down, parametric approach using cost estimating relationships (CERs), a bottom-up approach exploiting a breakdown of unit costs, a heritage-based method building from similar past missions, or some hybrid method, is arbitrary for the analysis presented here, so long as a cost is obtained. Typically, CERs are exploited during the conceptual design phase since they offer an effective way of integrating cost estimation into a trade-study or optimisation routine. Available off the shelf tools include the NASA Air Force Cost Model (NAFCOM), Price's PRICE-H and Galorath's SEER-H for large (traditional) space industry [138] and the Aerospace Corporation's Small Satellite Cost Model (SSCM) for satellites under 500kg [139]. Note that there currently exists no formal or standard approach to cost modelling of nano-satellites (<10kg), however attempts have been made by the author in [140] and are described in detail in Section 5.6.4.

4.2.5   Optimisation

While the previous sections define a generalised approach to the mission design and analysis process, it is possible, and indeed preferred, in many instances to exploit optimisation algorithms for this purpose. Optimisation offers an automated approach to the process of finding optimal, or near-optimal, solutions based on objectives, constraints and selection rules that are prescribed by the analyst. The general strategy is described in Figure 22, in which the value space may or may not be included in the iterative process, depending on the protocol being applied.
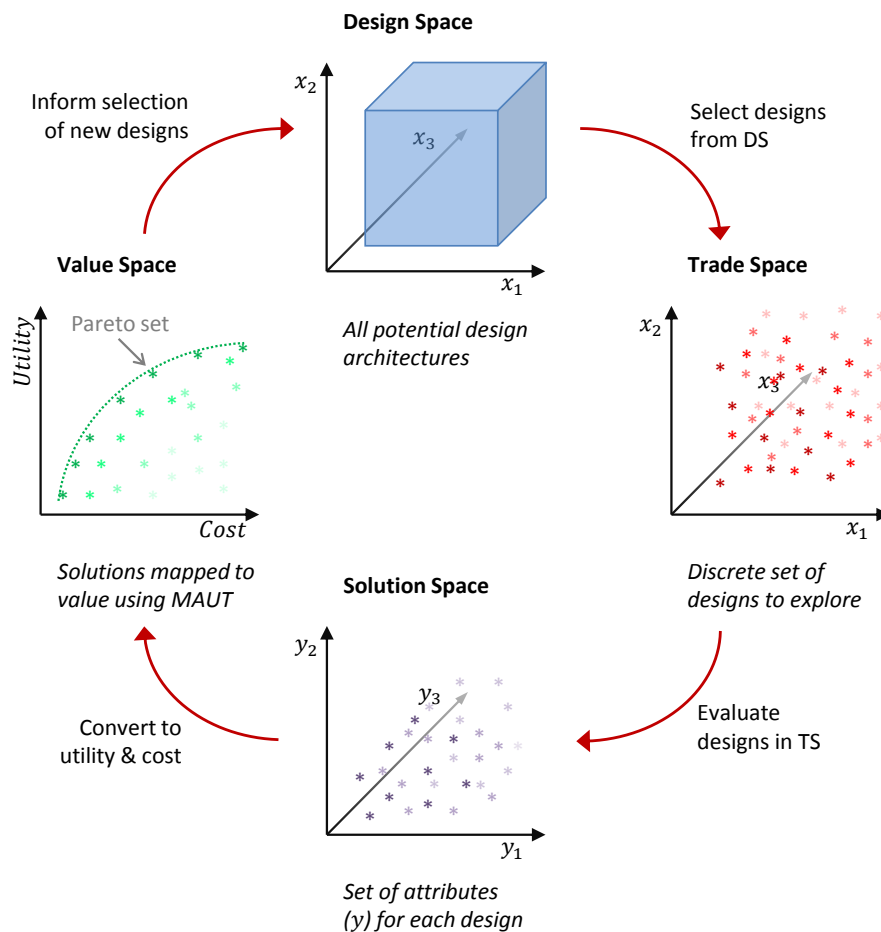
Figure 22 – Optimisation process for design

Depending on the shape of the solution space, either gradient-based methods or non-gradient-based (heuristic) methods are most effective, and aim to maximise/minimise some objective function subject to constraints on the design vector.

In the case of minimisation of multiple (n) objectives, the problem can be formulated as

$$\min[f_1(x), f_2(x), \dots, f_n(x)],$$

$$x \in X,$$

<div align="right">*4.7*</div>

where X is the set of design variables. A design x^* is said to be Pareto optimal if, and only if, there is no $x \in X$ such that

$$f_i(x) \leq f_i(x^*) \;\; \forall i \in \{1, 2, \dots, n\} \text{ and}$$

$$f_j(x) < f_j(x^*) \text{ for at least one index } j \in \{1, 2, \dots, n\}.$$

<div align="right">*4.8*</div>

The complete set of Pareto optimal designs form the Pareto set, which are often carried forward for further analysis. For complex design problems, the solution or value spaces are often not gradient-based, which demands use of a heuristic method, such as the Non-dominated Sorting Genetic Algorithm (NSGA-II) [141]. Of course, the method applied should be selected based on the particular problem at hand.

Optimisation has been used extensively in aerospace mission design, with genetic algorithms [142], particle swarm optimisation (PSO) [143], dynamic programming [144], collaborative optimisation [145], [146] and multi-disciplinary design optimisation (MDO) [147]–[150] receiving most attention. Furthermore, some authors have combined traditional trade-space exploration with optimisation in order to reduce the computational effort required in the latter stage, by identifying promising regions in the design space with the former [151]. Whether employing traditional methods, optimisation, or some hybrid approach, the objective is the same; to find the best design solutions.

4.3   Uncertainty Analysis

A point design, i.e. one that is rigid in its characteristics and operation, may be optimal in the nominal environment for which it was developed, but may be left exposed in the real world in which uncertainties exist. Consideration of uncertainties as part of the design process, in particular from the point of view of failure (reliability engineering), has been practiced since the early 1900s [152]. However, studies into the effects resulting from arbitrary uncertain events taking place has only recently come to the

fore [153]–[155]. For clarity, uncertainty can be separated into two forms; i) Aleatoric – that which cannot be known *a priori*, such as minor fluctuations in atmospheric conditions or the time of future solar activity, and ii) Epistemic – that which occurs due to lack of knowledge, such as the centre of mass of an instrument or the cost of a launch. Here, while both are acknowledged as important and valid, the former, aleatoric uncertainty, will be considered. This is because, as indicated by the definition, it is unavoidable and is therefore always present. Irrespective of the type of aleotoric uncertainty (e.g. failure, market change, environmental change, technology obsolescence), the over-arching effect is generally a drop in system utility, which we hope to measure in order to better understand utility.

The study into "-ilities" [153], i.e. attributes providing resilience to the effects of uncertainty, has been extensive. Flexibility was one of the first to be considered [156], and can be defined as a system's ability to respond to changes in its initial objectives, requirements or environment. Later, other characteristics such as survivability [153][157]; the system's ability to minimise the impact of a temporary change in requirements or environment and evolvability [158]; the ability of an architecture to be inherited and changed across generations, begun to receive attention. The focus in this dissertation is on the benefits offered by inter-satellite networking via additional robustness[*] or resilience, which has been defined as a system's ability to satisfy a fixed set of requirements, despite changes in the environment or within the system [159]. The change considered here is failure of on board systems, resulting in full or partial degradation.

In order to evaluate the impact change has on the value offered by a system, epoch era analysis (EEA) was introduced in [130] and [160], which requires evaluation of the system's utility at various points through the lifecycle. At these epochs, changes are imposed on the system, or its environment, given uncertainties identified *a priori*. These changes result in a different utility and cost result, improving our understanding of the through-life value proposition. While EEA offers a method of assessing a particular design's value during the lifetime, subject to stochastic effects, it would be computationally impractical for a complete understanding of the probabilistically expected value in a truly uncertain environment.

---

[*] While not an -ility by name, robustness and resilience can be grouped within this family of characteristics

4.3.1   Markov Chains

Markov chains offer a convenient way of modelling the stochastic process of component/system failure and allow evaluation of expected state evolution over time. A Markov process is a stochastic process defined by a set of random variables $\{X(t), t \in T\}$, where each $X(t)$ is a random variable, or "state", defined on some probability space, at time t. T is the time horizon and can be discrete in nature, whereby $T = \{0, 1, 2, \dots\}$ forms a discrete time Markov chain (DTMC), or continuous, where $T = \{t: 0 \leq t < \infty\}$ forming a continuous time Markov chain (CTMC). For the purposes of this work, the focus shall be on DTMCs due their applicability to systems for which the probability of transitioning between states may be time-varying [4].

The Markov property is analogous to the memoryless property, such that transition from one state to another depends only on the current state, and not on states in which the system resided during some previous time. Formally, this is

$$
\begin{aligned}
\text{Prob}\{X_{n+1} = x_{n+1} | \ X_n = x_n, X_{n-1} = x_{n-1}, \dots, X_0 = x_0\} \\
= \text{Prob}\{X_{n+1} = x_{n+1} | \ X_n = x_n\},
\end{aligned}
\tag{4.9}
$$

where x_i is the state of the system at time i. The above can be simplified to

$$
p_{ij}(n) = \text{Prob}\{X_{n+1} = j | \ X_n = i\}.
\tag{4.10}
$$

Given a discrete and finite set of states, the matrix $P(n)$, in which $p_{ij}(n)$ is the entry in the ith row and j^{th} column, is called the transition probability matrix, and contains the probability of transitioning between any two states at time n. $P(n)$ is written as

$$
P(n) = \begin{bmatrix} p_{11}(n) & p_{12}(n) & \cdots & p_{1K}(n) \\ p_{21}(n) & p_{22}(n) & \cdots & p_{2K}(n) \\ \vdots & \vdots & \ddots & \vdots \\ p_{K1}(n) & p_{K2}(n) & \cdots & p_{KK}(n) \end{bmatrix},
\tag{4.11}
$$

where K is the number of discrete states and thus $P(n) = \mathbb{R}^{K \times K}$. It follows that

$$
\sum_{all\ j} p_{ij}(n) = 1, \forall i,
\tag{4.12}
$$

where $i = j$ is a possible scenario and simply represents the absence of any transition from the current state. In a (time-) nonhomogeneous DTMC, $P(n)$ may vary with n, which is an important characteristic to capture for systems that suffer from either reliability decay over time, or a higher failure probability at the beginning of life. Analysis into the robustness/survivability achieved from swarm size [161] and system fractionation [162] has been conducted using a homogeneous DTMC, which results in this time-varying phenomena being missed. In [163] and [164] it is shown that this time-varying property is real for space systems and should be incorporated.

For systems that do not undergo servicing or repair, such as the majority of satellites, transition from a failed state back to an operational one is not possible, such that the 1st column in $P(n)$ is $[p_{11}(n) \quad 0 \quad 0 \quad 0]^T$ and the final row (fully failed) is $[0 \quad 0 \quad 0 \quad 1]$, which represents an *absorbing* state. As an example, Figure 23 illustrates a typical Markov chain with four states, in which state 2 always transitions (no arc to itself) and state 4 is absorbing. It is recognised that satellite systems generally feature redundancy on most sub-systems, in particular on those that are mission critical. This is considered throughout this work such that the failure probabilities used are derived from actual flight data, where redundant systems, if present, would have failed.

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 \\ 0 & 0 & p_{23} & p_{24} \\ 0 & p_{32} & p_{33} & p_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
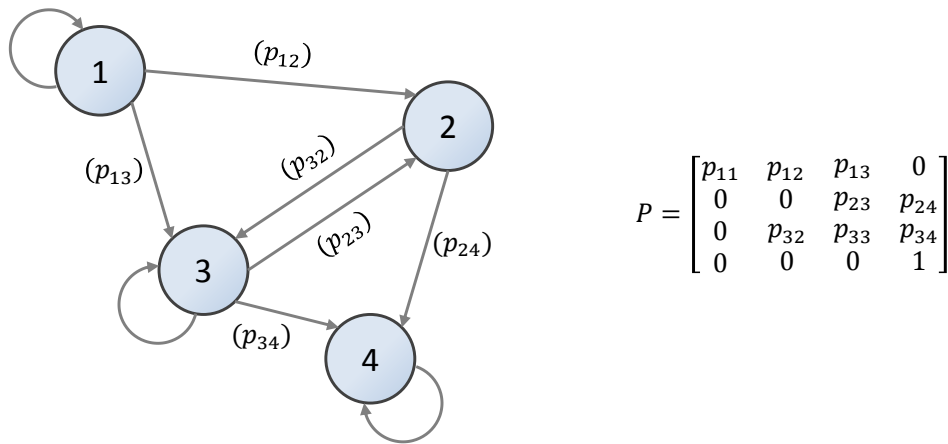
Figure 23 – Example of a DTMC with an absorbing state

In order to identify the expected utility being offered by a system subject to stochastic failures over its lifetime, transient analysis should be conducted, which provides an answer to the question "what is the probability that our system will be in state x after m time steps, given that it is in state y now"?.

The Chapman Kolmogorov equations offer such functionality, derived from the fact that the probability for a two-step transition from state i to k is

$$\text{Prob}\{X_{n+2} = k| X_{n+1} = j, X_n = i\} = p_{ij}p_{jk}, \forall j \in \{1, 2, \dots, K\},$$

$$\text{Prob}\{X_{n+2} = k| X_n = i\} = \sum_{all\ j} p_{ij}p_{jk}, \qquad \text{4.13}$$

which is the ik^{th} element in P^2 for a homogenous DTMC, or the ik^{th} element in $P^{(n+2)}(n, n+1, n+2) = P(n+2)P(n+1)P(n)$ in a nonhomogeneous DTMC. This can be generalised to obtain the Chapman Kolmogorov equation for nonhomogeneous DTMCs, as

$$P^{(m)}(n, n+1, \dots, n+m-1) = P(n)P(n+1)\dots P(n+m-1). \qquad \text{4.14}$$

Finally, given a row vector $\pi^{(n)} = \mathbb{R}^{1 \times K}$ describing the probability of being in state $i \in \{1,2,\dots,K\}$ at some time n, the probability of being in state i at time $(m+n)$ is

$$\pi^{(m+n)} = \pi^{(n)}P(n)P(n+1)\dots P(n+m-1). \qquad \text{4.15}$$

To the best of the author's knowledge, the work presented in this dissertation forms the first use of nonhomogeneous DTMCs to model failure and dynamic lifetime utility for distributed space networks that form a delay- or disruption-tolerant network.

4.3.2   Monte Carlo Analysis

While the analysis of Markov chains offers a useful and analytical method of finding the expected state in which a system will reside during its lifetime, it offers no information as to the variance of this data. Monte Carlo simulation (MCS) offers this, via the execution of potentially large numbers of performance evaluations, with random variables (corresponding to uncertain parameters) assigned according to some probability distribution that represents their expected range. The results of these simulations provide both the expected value, and variance, of system state/performance, which may influence the decision maker. For example, say system A has an expected value of 0.7 over its lifetime, while system B has an expected value of 0.69, we would instinctively select system A for our design. However, if it were shown that there was significant potential for A to have a value as low as 0.3, while system B's

lower bound is 0.5, we would perhaps question our original logic in an attempt to minimise risk.

An example of MCS can be found in [165], where orbit lifetime is analysed via 210,000 simulations, with normally distributed random variables for mass (m), cross sectional area (S), drag coefficient (C_d) and launch date (Dl), resulting in a probability distribution for orbit lifetime (Figure 24).
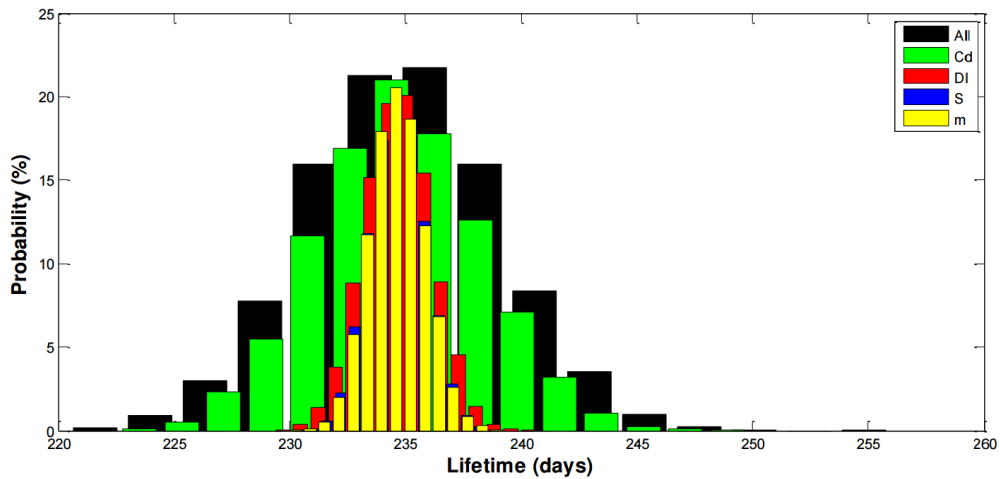
Figure 24 – MCS for orbit lifetime prediction [165]

By performing Monte Carlo simulation (MCS) with failure built in to numerical models, it is possible to extract information pertaining to both expected state of the system and variance of the state over the lifetime. This problem is tackled in [157] for the case of a space "tug" mission, by predefining a degraded utility to each failed state and running MCS over the life of the mission. The analysis in [157] also includes repair of the system once a failed state is reached, which is subject to some cost added to the overall mission cost. While the degraded utilities are defined in an attempt to capture general performance losses, their arbitrary nature leaves the analysis open to interpretation. Ideally, degraded utility should be calculated in the same way as nominal utility, i.e. via some analysis, although given a large number of potential states this could prove computationally challenging.

Another use of MCS to measure robustness to failure is applied to the case of fractionated systems, in which a satellite from the fractionation fails according to some mean time to failure and replaced according to some mean time to recovery [162]. The lifetime utility is evaluated using two scenarios; the first being where failure of a node in the fractionation results in an overall utility of 0 until it is repaired/replaced, while

75

the second being where failure results in partial degradation and a utility of 0.75 for the fractionation, until repaired/replaced. A fully operational system would exhibit a utility of 1. As intuition would suggest, it is shown that a monolithic architecture is more cost effective for shorter lifetimes, but a fractionated system shows benefits over longer lifetimes.

A typical procedure for identifying lifetime utility for a system subject to stochastic failures, using MCS, is illustrated in Figure 25. It comprises 6 main steps; definition of state transition probabilities, definition of a cost matrix (cost of transitioning from state i to state j), operational simulation for evaluating performance in each state, identification of utility in each state, discrete event Monte Carlo simulations and finally, calculation of expected utility & cost over the mission lifetime.
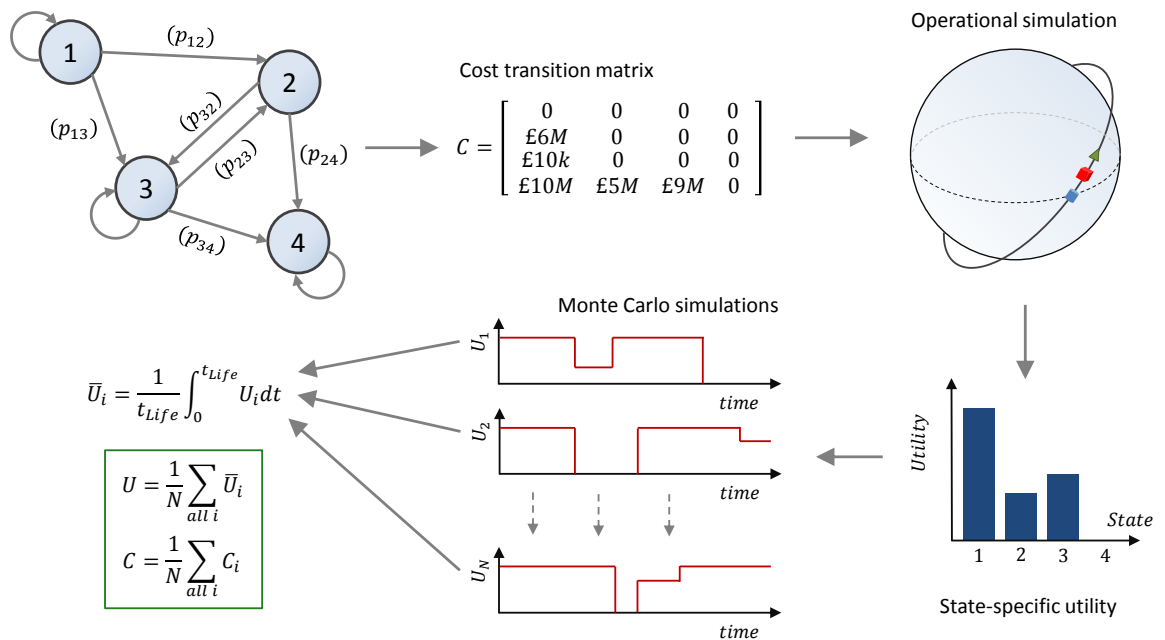
Figure 25 – Monte Carlo simulation procedure for uncertainty analysis

Since a full cycle of MCS analysis can be expensive in terms of both manual and computational effort, it is generally accepted that only promising designs that show high value in their nominal configuration are evaluated in this way. The process described here shall be used in order to validate Markov chain-derived results of value over the life of a mission that is subject to failures, in Chapter 6 of this dissertation.

Chapter 5

The Value of Inter-Satellite Networking

In this chapter, a methodology is presented that enables a decision maker to assess the value that an inter-satellite networking (ISN) capability would offer their mission, given some baseline design architecture. The objective is to decouple overall mission value from the often difficult task of calculating additional sub-system development cost, by identifying the price one would be willing to pay for such a capability. The capability, in this work, is ISN. In what follows, the underlying mission-level benefits of ISN, and how it affects a space system is discussed, followed by derivation of utility for a system with an ISN capability. This utility is then coupled with the indirect costs associated with including such a system, such as the implications from a greater system mass and power demand, thus allowing one to derive relative value between different systems. This value-difference is analogous to a financial quantity available for developing, manufacturing and integrating the ISN capability, while maintaining equal or better overall value.

The concepts introduced here are demonstrated via two case studies, the first being a single satellite operating as part of a multi-platform federated satellite system (FSS) and the second being a single-operator-led constellation of nano-satellites carrying out Earth observation

5.1 Necessary Conditions for a Networked System

Consider n satellites operating independently, i.e. without an inter-satellite networking capability, carrying out generic data acquisition and delivery functions. For each satellite, i, the long-term average rate of data delivery to the user (destination node) must be equal to or greater than the long-term average rate of data acquisition, accounting for any on-board compression and processing, in order to avoid either a build-up of data in the buffer, or the need for packet deletion/dropping.

In other words

$$\sum_{j=1}^{m} \int_{t_{j0}}^{t_{jf}} R_{i,j}dt \geq \sum_{k=1}^{p} \int_{t_{k0}}^{t_{kf}} R_{i,k}dt \quad \forall i = (1, 2, \ldots, n), \qquad 5.1$$

where m is the number of discrete data delivery events during some arbitrary period of time, p is the number of discrete data acquisition events over the same period and R is the rate of data transfer during those events. Equation 5.1 is a necessary condition that must be met for successful data provision from systems without networking capability, but is not strictly necessary otherwise. Indeed in the networked case, some proportion of a platform's data may potentially be delivered by a neighbour with spare delivery capacity. This is fundamental in the argument for fractionated satellite systems, where one or more nodes in the fractionation conduct data downlink, while other nodes focus on alternative tasks. In a generic network however, the problem is more complex, since each node may only be connected to a sub-set of other nodes, and only a sub-set of nodes may have the potential to download data, at specific rates, to specific destinations.

Now consider a network of n satellite nodes and a single destination node[*] T. Between each node pair there exists a link representing the long-term average data transfer capacity over some arbitrary period of time, τ. Each node i acquires an amount of data D_i during this period and can transfer an amount of data c_{ij} to node $j \in (1, 2, \ldots, n)$. It is possible to establish whether or not this network is capable of delivering all acquired data, given no time constraints on data lifetime, by representing it as a transport network on which a maximum flow problem [166] is solved.

---

[*] Even in networks with multiple ground-based destinations, it is generally the case that once delivered to *somewhere*, the terrestrial internet can be exploited to reach the final destination in a negligible period of time, such that all destination nodes can be treated as one global node.

To achieve this, for a multiple source network, it is useful to include a global source node, S, that features a link to each node i equal in capacity to D_i (Figure 26).
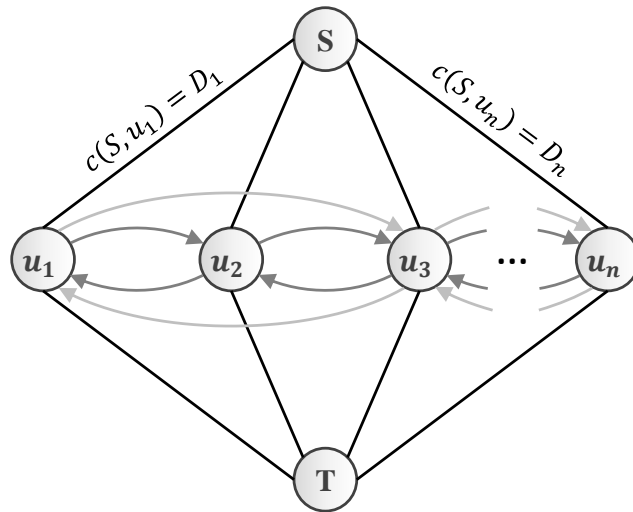
Figure 26 – Network with global source node for maximum-flow calculation

The maximum flow problem can be solved using either the Ford-Fulkerson algorithm [166], or some more recent, less time complex equivalent algorithm [167]–[171]. Using the Ford-Fulkerson algorithm on a connected, directed graph $G(V, E)$ with nominal flow capacity $c(u, v)$ along edge $(u, v) \in E$, the maximum flow between source S and destination D can be found using the procedure in (Table 9).

Ford-Fulkerson

Input: digraph $G(V, E)$, source S, destination T

Output: Maximum flow f_{max}

$f(u, v) = 0$ for all edges $(u, v) \in E$

$c_f(u, v) = c(u, v)$ for all edges $(u, v) \in E$ (residual edge capacity)

while path p from S to T: $c_f(u, v) > 0$ for all edges $(u, v) \in p$

$\quad f(p) = \min\{c_f(u, v)\}$ (flow over path p)

$\quad c_f(u, v) = c_f(u, v) - f(p)$ (decrease residual capacity in direction of flow)

$\quad c_f(v, u) = c_f(v, u) + f(p)$ (increase residual capacity in direction opposing flow)

end

$f_{max} = \sum f(u, T)$

Table 9 – Ford-Fulkerson maximum flow algorithm pseudocode

For a connected satellite network, delivery of all acquired data can be considered possible, assuming no delay-constraints on the data, if $f_{max} = \sum c(S, u)$ for all nodes u that acquire data. If this condition is not met, it is likely that some data must be dropped due to insufficient download capacity being available.

In the federated satellite system (FSS) scenario, it is likely that data of different types exist that must travel between specific source-destination pairs, which may be better treated as a multi-commodity flow [172]. Analysis by solving the multi-commodity flow is beyond the scope of this dissertation, but is recommended in the specific case of multiple unique source-sink pairs.

It is important to note that while the above analysis confirms the existence of sufficient capacity to deliver all acquired data over the long term, and thus offer a heuristic indication of success, it does not explicitly quantify network performance (e.g. delay or hop-count), or indeed whether complete delivery can actually be made. For example, consider data with a very short time to live in which the total capacity vastly exceeds that deemed necessary. The average delay may still be long relative to the time-to-live, thus resulting in significant amounts of data expiry and packet dropping. Furthermore, the capacities used above are long-term average values, such that while a path p may demonstrate a certain capacity, it may be unattractive during operations because of its long-delays and therefore avoided in reality. It is critical therefore, that one carries out more in-depth assessment, such as that illustrated in Chapter 3 and later in this chapter, in order to better understand the behaviour of the network and its effect on overall value.

5.2   Formal Definition of ISN Value

Value is the metric used to represent the absolute worth of a system, such that greater value should always correspond to a better system. It is therefore possible to use value to efficiently compare different systems, with potentially very different design architectures. In order to identify the value of inter-satellite networking (V_{net}), it is necessary to make a comparison between a system with ISN and a system without it. The utility of both can be found through modelling and analysis of the system's operation, however while the cost of the baseline system should be fully understood using traditional methods, the cost of the modified system may only be known less the development and manufacture of the ISN technology. Since the value of ISN could be

defined as the price (Pr_{net}) one would be willing to pay for it, it could be considered worth implementing so long as the overall mission value of a modified system (V') remains greater than that of a baseline system (V_0). The ISN value can thus be formalised as

$$V_{net} = \max_{V' > V_0}(Pr_{net}).$$
<div align="right">5.2</div>

Note that overall value here is assumed to capture any additional complexity, risk and time implications of such a modification, and it can be assumed that equal value is indeed equal from the point of view of the stakeholders. A real price less than this upper limit would equate to an effective increase in value and thus represent benefit to the stakeholder. Logically it follows that this maximum price is the difference in nominal mission cost, i.e. not including the ISN development, manufacture and integration, between the two systems such that given their respective utilities, the mission value is unchanged. Indeed, as we have seen earlier, value can be defined in a number of ways depending on the mission type and stakeholder requirements, but typically it is some relationship between the utility and cost.

5.2.1   Preference Value

Returning to the non-commercial value proposition introduced in Section 4.2.4.1, the preference value (V) of a system can be represented as the ratio between a system's utility (U) and cost (C),

$$V = \frac{U}{C}.$$
<div align="right">5.3</div>

From the perspective of networking, it follows that the value of a non-network capable system is

$$V_0 = \frac{U_0}{C_0},$$
<div align="right">5.4</div>

while that of a network capable system is

$$V' = \frac{U'}{C' + Pr_{net}},$$
<div align="right">5.5</div>

where Pr_{net} takes the definition from equation 5.2, which is the price of ISN capability development and integration. In order to ensure a higher system value, the price one should be willing to pay for such a modification is therefore

$$Pr_{net} \leq \frac{U'}{V_0} - C'. \qquad\qquad 5.6$$

Whether or not a design team considers it feasible to develop/buy the ISN capability for this price (or indeed less) must be decided on a case-by-case basis. To illustrate the concept graphically, it is necessary to first understand the relationship between the design variables and the value space (Section 4.2.4), the latter of which is typically defined by an architecture's cost and utility (Figure 27).
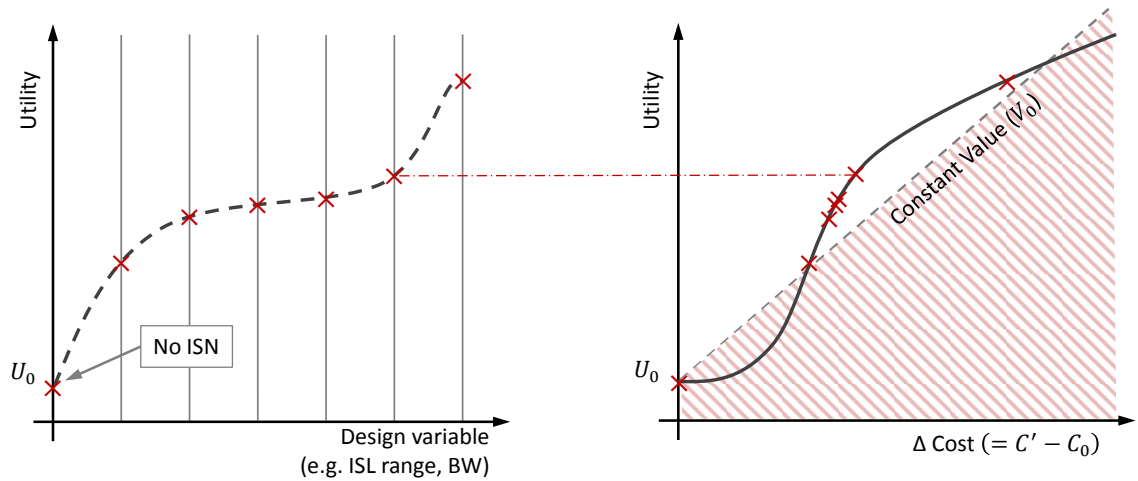
Figure 27 – Mapping from design variable utility to value space

The shaded area under the constant value line (defined by equation 5.4) contains architectures that are not of interest to the designer, since their increase in utility is not sufficient to outweigh the additional cost required to obtain it, in terms of value.

Those architectures above the shaded area however, offer higher value, where the distance along the horizontal axis from the constant value line defines the value of ISN, as defined in equation 5.2 (Figure 28).
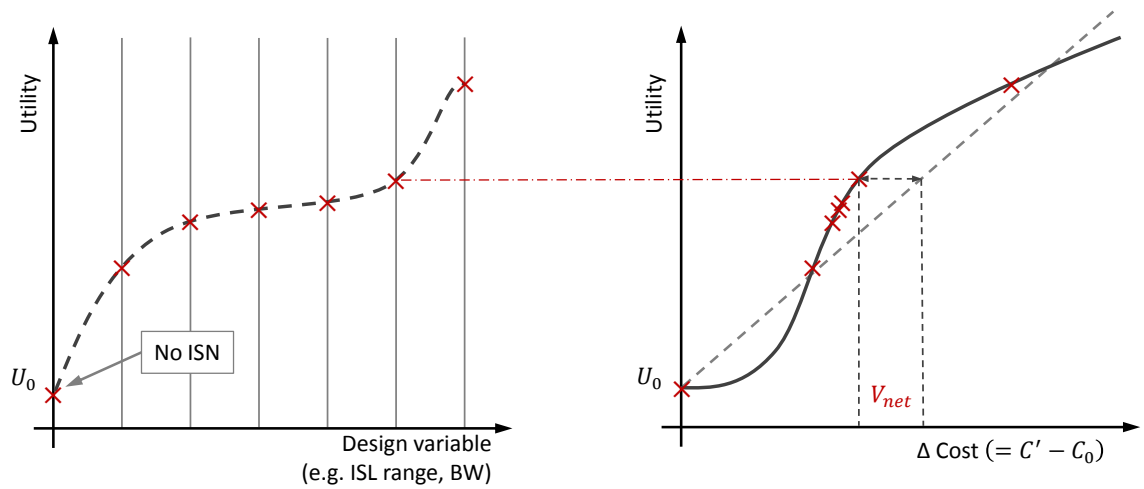
Figure 28 – Indication of ISN value for a specific design architecture

Effectively, the greater the horizontal (cost) separation between the point on the value space and the constant value line, the greater the price one should be willing to pay for such a capability.

5.2.2   Maximum Price Value

The preference value (V) used in the previous section is a convenient way to represent value in terms of purely technical preference, i.e. amount of function per unit cost. This however, is not always sufficient in capturing a stakeholder's view of what they would be willing to spend on a system. Consider the situation where one has a modified system that achieves twice the utility of its baseline counterpart, this does not necessarily mean one would be willing to spend twice the amount to obtain it.

For this purpose, the concept of a maximum price value (V_{MP}) is introduced, and illustrated in Figure 29.
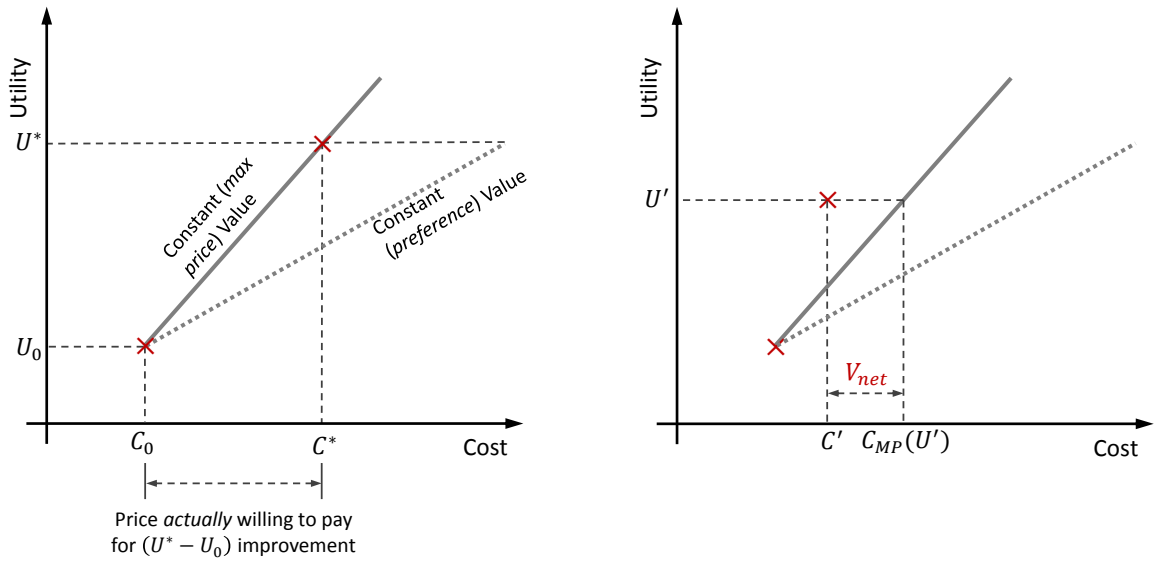
Figure 29 – Maximum price value (represented as a linear function as an example)

The maximum price value is, as the name suggests, the maximum price one would be willing to spend to obtain a certain utility. This differs from the preference value from the previous section, which is independent on spend willingness, except in the case of the utility being exclusively revenue, where the two will equate. A maximum price value can be derived in much the same way as the preference value, but arguably best carried out following calculation of a baseline system's cost and utility. At this point, a decision maker may make a judgement call as to how much they would spend in total (C_{MP}), to achieve greater levels of utility, which is represented by a function of the form

$$C_{MP} = f(U_0, U, C_0). \qquad 5.7$$

For example, the plot in Figure 29 shows a linear V_{MP}, starting at the baseline system (C_0, U_0). It follows therefore, that the value of ISN for a system achieving a utility of U' at a cost of C', with a value defined as a maximum price value, is

$$V_{net} = V_{MP} = C_{MP}(U') - C'. \qquad 5.8$$

In the case of a linear definition of V_{MP}, this equates to

$$V_{net} = V_{MP} = \frac{U' - U_0}{m} + C_0 - C', \qquad\qquad 5.9$$

where m represents the gradient of this function and for which equation 5.2, $V_{net} = \max_{V'=V_0}(Pr_{net})$, holds true. Should the maximum price value be defined by some other expression, e.g. an exponential function, equation 5.9 would be modified accordingly.

5.2.3   Implementation Price

The price that one is willing to pay for a commodity is an insightful metric, however the actual cost of implementing a capability, is subject to properties of the capability itself. For example, implementing an ISL system with a link range of 100km is likely to be significantly less costly than implementing one with a link range of 10,000km, which must be considered by the designer when evaluating the value of ISN in real-terms. Simply because there may be more excess value over the baseline for one design than another, does not necessarily mean it is a better modification to make. Two things must be noted about Figure 28:

i.   The "$\Delta\,Cost$" defines only the cost impact at the system and mission levels, and does not include the cost to develop and manufacture the ISN capability. This is what the designer must account for when they assess the ISN value (V_{net}), i.e. the expected price of the development and manufacture must be deducted from this amount in order to truly assess the benefit.

ii.   While some design parameters might be continuous, e.g. any ISL range could theoretically be achieved through appropriate transmit power, others may exhibit discontinuities such as the application of discrete frequency bands.

The price one is willing to pay for a particular capability can be translated back to the design space through a mapping of the separation from the constant value function (Figure 30).
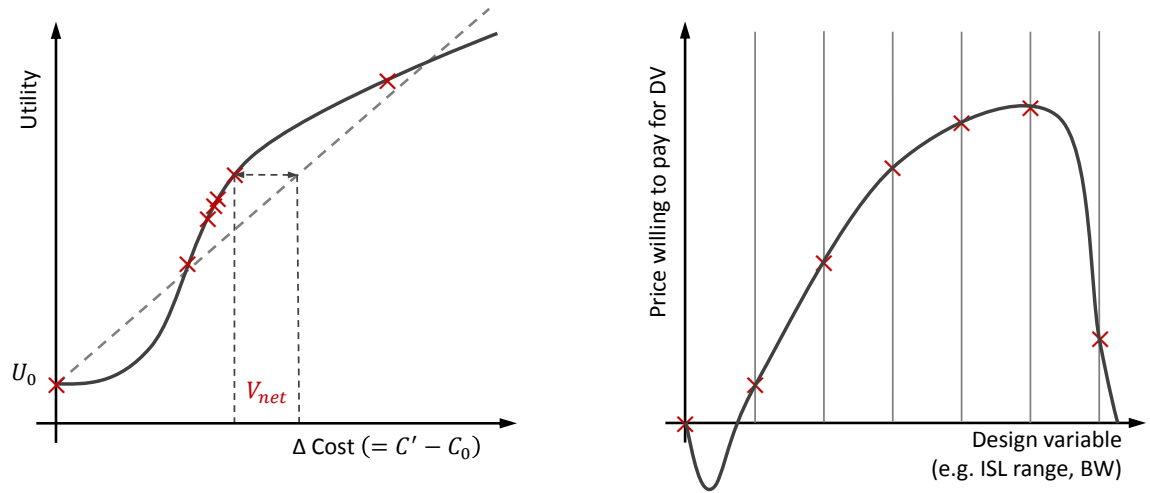
Figure 30 – Mapping from value space to price worth paying to achieve design modification ('DV' refers to design variable)

The estimated value of some specific modified system, i, based on the assumption that one can develop and manufacture the ISN capability for $C_{ISN,i}$ and where preference value and maximum price value are equal, is

$$V'_{est,i} = \frac{U'_i}{C'_i + C_{ISN,i}}. \qquad\qquad 5.10$$

Again, the above estimation would need to be made on a case by case basis and is therefore neglected from this work for reasons of generality. There is no universal method of saying what the addition of a particular inter-satellite communication sub-system to a spacecraft would cost.

5.2.4   Application to Trade-Space Analysis

While the previous sections serve to illustrate the process for modification of a single design architecture, from the addition of a single design variable, there are no barriers to the introduction of other design variables, and/or incorporation into a trade-space analysis in order to compare many architectures. For example, say design *A* features a large ground station network (GSN) while design *B* includes a small GSN, design *A* will therefore offer a greater nominal utility and, perhaps, greater value. However, the

addition of ISN may have very little effect on data latency and data throughput for design A, but a significant effect on design B, resulting in the modified system value of design B to be greatest (Figure 31).
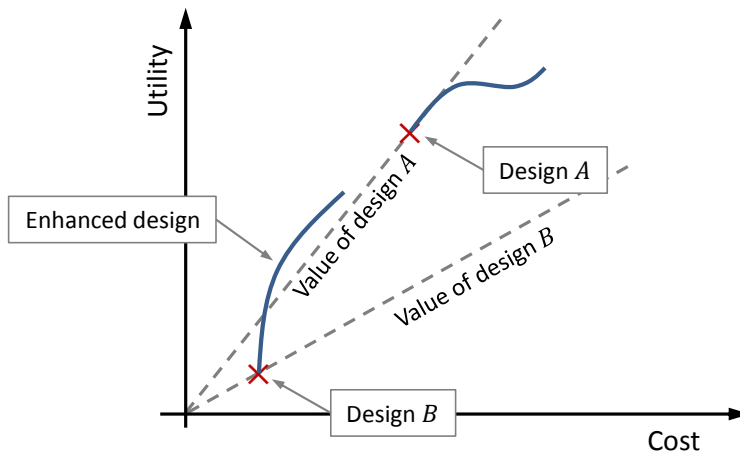
Figure 31 – Example of the effect of ISN on different design architectures – blue lines represents change in utility due to ISN modification

Formally, the scenario expressed by Figure 31 can be defined as

$$V_{B,j} > V_{A,i} \quad \forall i \in I, \hspace{4cm} 5.11$$

where $j \in I$ is a specific modified variant of design B, which is greater in value than any variant of design A, and I is the set (discrete or continuous) of possible ISN modifications.

5.3 Utility of Inter-Satellite Networking

Networking is known to have a generally positive impact on mission-level performance, through reduced delay and increased overall capacity, but can also have a detrimental effect on some attributes.

It is likely that if ISN is being considered as part of the design architecture, there exists some desire to improve one or more of the following:

i. Latency – time from data creation/acquisition to delivery

ii. Delivery ratio – proportion of acquired data that is delivered before expiry

iii. Data throughput – volume of data that is delivered (important when comparing design architectures in a trade-space analysis)

iv. Revisit rate* – frequency of visits to a particular target or region, for which acquired data is delivered

Indeed, there may be other attributes that are important to stakeholders, such as image resolution (Earth observation payloads) or quality of service (communication payloads) that are not affected significantly by an inter-satellite networking capability. Should these attributes also vary between design architectures that are being compared, then they should be included in the overall utility metric in order to correctly evaluate the impact of the networking in real-terms. Ultimately, the complete set of attributes that are of interest, and that have the potential to vary across the design space, should be considered.

In order to identify explicit value of the system, a multi-attribute utility function is recommended, derived using the procedure described in Section 4.2.4.2. While evaluating system performance across multiple separate attributes offers useful information to the analyst, comparison across a large solution space rapidly becomes difficult to manage at attribute numbers greater than two or three.

5.4   Mission Cost

Given the definition of value provided in the previous sections, it is apparent and indeed expected that stakeholders will want their mission to achieve the highest utility for the lowest cost. Therefore, it is necessary to estimate mission cost during an early concept phase in order to minimise the risk of carrying forward sub-optimal architectures into the detailed design phase.

---

* While a non-networked system would visit particular targets at the same time and with the same frequency as a networked system, data acquired during a visit may be less likely to get delivered, if say for example a short *time to live* were employed, in which case the visit would be neglected.

To do this effectively, one should consider as many elements of the mission as possible, including those associated with the space, ground, launch, operation and maintenance (replenishment) segments, i.e.

$$C = C_{space} + C_{ground} + C_{launch} + C_{ops} + C_{maint},$$ 5.12

The time-value of money should also be considered, where future cash flows are discounted according to some rate (derived from the expected rate of return on investment of the same amount), to obtain the net present value (NPV) of a system. If possible, uncertainties such as market volatility or maintenance cost variability should be accounted for through use of expected-NPV (eNPV) or Monte Carlo methods. This is of particular importance when comparing architectures that have different development times or design lifetimes.

The cost model used by the designer, which enables calculation of the cost for each element in equation 5.12, are arbitrary for the purposes of this work, but in general must be selected according to the mission-type and level of fidelity required. It is acknowledged that there exist other considerations for a thorough cost model to be implemented, such as specific payload development costs and disposal costs, however for the purposes of this work, the aforementioned elements are considered to provide sufficient fidelity for demonstration purposes.

5.5   Case Study 1: Federated Satellite System

This case study shows how the value of ISN on a single satellite within a federation can be derived, in which federation members interact through sharing of data storage and data download resources. The satellite in question must have the data that it collects delivered to the ground, either directly, or via other space assets.

5.5.1   Design Space

The design space for this analysis will comprise only networking properties (Table 10).

Variable	Units	Range
Inter-satellite link range	km	0, 1000, 2000, 3000, 4000, 5000
Inter-satellite data rate	Mb/s	2.5, 6.25, 12.5, 25, 37.5, 50

Table 10 – FSS design variables

All other design parameters, outlined in Table 11, remain constant throughout the analysis, such that the effects, and consequently the value, of including an inter-satellite networking capability can be evaluated.

Variable definition	Units	Quantity
Altitude	km	780
Inclination	degrees	98.8 (Sun synchronous)
Eccentricity	-	0.0
Local time of ascending node*	24 hour	12:00
Ground station 1	-	Svalbard, Norway
Ground station 2	-	New Norcia, Perth, Australia
Data rate to GS	Mb/s	50
Number of other satellites in federation†	-	9
Payload packet volume	Mb	250
Packet routing method	-	Spae
Spae storage cost	s^{-1}	1
Spae transmission cost	Per hop	600 (equivalent to 10 minutes of storage)
Simulation duration‡	days	3
Time step duration	s	10
Number of simulations for each design	#	30

Table 11 – Other FSS design parameters

---

* 12:00 chosen for illustrative purposes, but there are no restrictions on other times, such as 10:30, being defined during repeated analyses.

† Other satellites in the federation reside in other circular low Earth orbits between 600km - 1000km altitude, and 0° - 100° inclination. They are randomly assigned access to Kourou, South Point (USA) and Hartebeesthoek stations within the ESTRACK network.

‡ The first and last 10,000 seconds of each simulation were ignored in the results calculation, to ensure only steady state conditions were evaluated.

Regarding data acquisition, each satellite in the federation collects data at random, according to a Poisson process, at a rate dictated by their individual download potential and their associated level of contention. Contention, which is the ratio of acquired data volume to potential delivered data volume, shall be between 0.25 and 0.75 for each satellite in the federation, and is assigned randomly using a uniform distribution.

In order to capture the effects of interactions with other satellites on system demands, energy shall be required for transmission to other nodes in the network. The amount of energy required shall be a function of the volume of data being transferred over a link, and power derived from a link-budget assessment with the design parameters in Table 10 and Table 11. For this analysis, it shall be assumed that if an interaction takes place, all packets that are identified by Spae as being candidates for routing to the current contact, are routed, such that energy limitations are neglected. To account for this, the amount of power required by the bus, and consequently the sizing of the power sub-system, shall be carried out post-simulation, and incorporated in the cost estimation. In other words, a system undertaking significant amounts of satellite-satellite communication will require greater levels of energy generation (larger solar arrays and greater battery capacity) than one communicating only with the ground station network, and will thus suffer greater system cost. This effect is also considered in communication with the GSN, where a networked system will transmit packets generated by other nodes in the network, such that additional power may be required for this purpose[*].

To calculate performance in terms of data latency and hop count, a simulation environment is required that emulates the mission, which includes the behaviour of, and interaction with, other participating nodes in the federation and the ground station network. Information about data dynamics is collected from the moment it is acquired, to the time at which it is delivered, which includes the times of reception and transmission, and the nodes through which the data traverses. This is of particular importance for the platform of interest, since the data transmission events that take place over the course of the simulation are used in the derivation of mission cost

---

[*] It is recognised that within the federated satellite system concept, financial transactions are likely to play a key role in the data transmission process, such that while a node may benefit financially from agreeing to deliver another's data, it may also have to pay to have their own data delivered. Literature surrounding research into this interaction process is in its infancy, such that this analysis assumes all participating nodes agree to carry out data hosting and delivery for free, for the good of the federation.

(Section 5.5.3). The cost model exploits information obtained from the operations simulation, such as the number of data transmission events and whether they were carried out during a sunlit period or not, as input to the power demand calculation.

All modelling and simulation for this study is carried out in the Matlab modelling environment.

5.5.2   Mission Utility

The attributes deemed important in deriving the overall mission utility in this case study are outlined in Table 12, along with their respective weighting in the calculation of a multi-attribute utility function.

Attribute	Objective	Limits	Utility function	Weighting
Data latency (mean)	minimum	[10, 60] mins	Linear	0.6
Data latency (variance)	minimum	[20, 30] mins	Linear	0.2
Hop count (mean)	minimum	[1, 1.6]	Linear	0.2

Table 12 – Performance attributes of the FSS case study

Since only data networking parameters (link range and data rate) make up the design space, only attributes related to these variables need to be included in the utility definition. This is not to say that other attributes are not important to the stakeholders, however they will not vary across the different architectures, so are not considered explicitly.

5.5.3   Mission Cost

Total mission cost is calculated as the sum of the cost of the space, ground, launch, operations and maintenance segments, each of which are summarised in the following sections. Note that the cost of developing and/or implementing the ISL system is to be deducted from the difference in value identified, as defined in Section 5.2.3 as the implementation price.

5.5.3.1   Space Segment

For this example, the Aerospace Corporation's Small Satellite Cost Model (SSCM) (version 8.0) is used to derive an estimated cost for the space segment (C_{space}), where cost is calculated as a function of;

i.   dry mass (M_{dry})
ii.   power sub-system mass (M_{power})
iii.   end of life power (P_{EOL})
iv.   average power (P_{ave})
v.   solar array area (A_{SA})
vi.   propulsion subsystem dry mass
vii.   payload power
viii.   telemetry, tracking and command (TT&C) subsystem mass
ix.   downlink data rate
x.   pointing accuracy
xi.   attitude control system type

For illustrative purposes, models used to derive the first five of these shall be described here, since these are the elements impacted from the addition of an ISN capability to the platform. Furthermore, for clarity and brevity, only the difference in the elements are shown below, i.e. the components likely to be different for systems with differing levels of ISN capability. Note that a full SSCM exercise is carried out to generate the results shown.

The change in system mass (ΔM_{dry}) is a sum of the additional mass of the power system (ΔM_{power}), which is the sum of the addition from solar arrays (ΔM_{SA}) and batteries (ΔM_{bat}), the ISL sub-system (M_{ISL}) and structure (ΔM_{stru}) required to support the ISN sub-system. The mass of the solar arrays can be derived using

$$\Delta M_{SA} = \rho_{SA}\Delta A_{SA}, \qquad\qquad 5.13$$

where

$$\Delta A_{SA} = \frac{\Delta P_{EOL}}{\eta_{cell}n_{pack}S\gamma(1-d)^L}, \qquad\qquad 5.14$$

is the change in solar array area, ρ_{SA} is the solar array mass per unit area, ΔP_{EOL} is the end-of-life power required from the solar array during sunlit conditions to operate the ISN sub-system, η_{cell} is the solar cell energy conversion efficiency, n_{pack} is the cell packing efficiency, S is the solar flux at the satellite's location ($\sim 1360\text{W/m}^2$ [173]), γ is the orbit-average fraction of cells that are illuminated*, d is the cell degradation factor per year (2.75% [174]) and L is the operational lifetime. The additional power required can be calculated from

$$\Delta P_{EOL} = \frac{\Delta P_{sun}}{\eta_{sun}} + \frac{\Delta P_{ecl}\tau_{ecl}}{\eta_{ecl}\tau_{sun}}, \qquad\qquad 5.15$$

where ΔP_{sun} and ΔP_{ecl} are the average powers required by the ISN sub-system during daylight and eclipse plus additional ground station communication required to deal with differing download demands, τ_{sun} and τ_{ecl} are the periods of the orbit in sun and eclipse, and η_{sun} and η_{ecl} are efficiencies of power transfer to the sub-systems during sunlit (from arrays) and eclipse (from batteries) conditions. ΔP_{sun} and ΔP_{ecl} can be found via simulation results, where the ISL duty-cycle will depend on contacts with other satellites and volumes of data being transferred during these contacts, and to the ground, i.e.

$$\Delta P_{sun} = \int_{T_{dawn}}^{T_{dusk}} \left(P_{ISL} + \Delta P_{ground}\right)dt \qquad\qquad 5.16$$

and

$$\Delta P_{ecl} = \int_{T_{dusk}}^{T_{dawn}} \left(P_{ISL} + \Delta P_{ground}\right)dt. \qquad\qquad 5.17$$

Additional mass of the battery (ΔM_{bat}) is dependent on the additional energy required from the battery (ΔE_{bat}) and the energy density (ε_{bat}), such that

---

* Note that if the satellite has a solar panel configuration that cannot be kept sun-oriented, such as for body-mounted cells and non-gimballed deployed arrays, simulation will likely be required to establish the orbit-average fraction (γ) of the cell area being illuminated. Here, it is assumed that the solar arrays are gimbal-controlled and are maintained sun-pointing, while in reality it would most likely be a combination of gimballed and body-mounted (fixed) arrays.

$$\Delta M_{bat} = \frac{\Delta E_{bat}}{\varepsilon_{bat}}, \qquad\qquad 5.18$$

where

$$\Delta E_{bat} = \frac{\Delta P_{ecl}\tau_{ecl}}{\eta_{ch}DoD}, \qquad\qquad 5.19$$

where, η_{ch} is the battery charging efficiency and DoD is the depth of discharge for the battery type on board.

The mass of the inter-satellite link sub-system can be derived in the same way as for a conventional communication system, with sizing of the antenna, amplifiers and supporting systems calculated using a link budget analysis. The mass of the antenna (M_{ant}), in this case a parabolic dish antenna, is proportional to the diameter (D) squared or area (A), according to

$$M_{ant} = k_p D^2 = \frac{4}{\pi}k_p A, \qquad\qquad 5.20$$

where k_p is between 4.94 and 7.96 [175], depending on the antenna type. The area is a function of the gain (G), wavelength (λ) and antenna efficiency ($\eta_{ant} = 0.55$) [175] such that

$$A = \frac{\lambda^2 G}{4\pi\eta_{ant}}. \qquad\qquad 5.21$$

Mass of the transmission power amplifier is dependent on amplifier properties, which in this case study shall be a travelling wave tube (TWT) amplifier, the mass of which can be represented by the cubic function [175]

$$M_{TWT} = 0.00002P_T^3 - 0.0029P_T^2 + 0.1807P_T + 1.8432, \qquad\qquad 5.22$$

where P_T is radiated RF (output) power required for the signal. Note that full redundancy is generally expected at the amplifier level, such that total mass would be double. Finally, supporting systems, including distribution unit low noise amplifiers for the incoming signal, any electronics required for demodulation and decoding signals, and margin is approximated as an additional 30% of the total subsystem mass, thus

$$M_{ISN} = 1.3(M_{ant} + 2M_{TWT}).$$ <div style="text-align: right">*5.23*</div>

ISL communication transmission power is derived through a link budget assessment, as presented in equation 5.24 below, with the parameters on the right-hand side defined in Table 13.

$$P_T = 10\log_{10}k + 10\log_{10}T_s + 10\log_{10}R + \frac{E_b}{N_0} - G_T - G_R - L_S - L_A - L_L,$$ <div style="text-align: right">*5.24*</div>

Parameter	Definition	Unit	Quantity	Comment
G_T	Transmission antenna gain	dBi	38	Considers 3dBi of pointing losses
G_R	Receiver antenna gain	dBi	41	
E_b/N_0	Energy-per-bit to noise ratio	dB	12.6	9.6dB required for GMSK modulation + 3dB margin
k	Boltzmann's constant	J/K	1.38×10^{-23}	Constant
T_S	Receive system noise temperature	K	700	
R	Signal data rate	Mbps	TBC	Table 10
L_S	Free-space losses	dB	Link-range dependent	$L_S = \left(\frac{\lambda}{4\pi S}\right)^2$
λ	Signal wavelength	mm	50	60 GHz
S	Link range	km	TBC	Table 10
L_A	Attenuation loss (due to atmosphere)	dB	0	Space-space link so no atmospheric effects
L_L	Line losses (transmitter to antenna)	dB	0.6	

<div style="text-align: center">*Table 13 – Link budget parameters*</div>

A power conversion efficiency of 10% is assumed between that contributed by the platform and that achieved at the antenna output [176]. It is this input power that contributes toward the system design through equations 5.16 and 5.17. In order to integrate the inter-satellite link sub-system into the small satellite cost model, it is represented as an additional payload. This allows inclusion of the demands on the system in terms of mass, power and data, but keeps cost of development separate, as per the value proposition introduced in this work.

Finally, the structural mass (M_{stru}) is calculated as 10% of the total mass, which provides input to the cost model [177].

5.5.3.2   Ground Segment and Operations

Since no new ground infrastructure is to be built as part of this mission, the ground segment cost can be neglected (cost of ground software development is captured in the space segment cost using the SSCM), such that only operations cost need be included. In this example, it shall be assumed that a price of 150USD per active ground station pass is agreed with the operators.

5.5.3.3   Launch Segment

The cost of launch is a complex, often mission-specific quantity, that is negotiated between the payload provider and launch service provider, which makes it difficult for conceptual studies to predict launch cost with confidence. There are, however, some publically available figures, which are exploited in this analysis. At the time of writing, SpaceX quote a launch cost of 4654USD/kg to LEO (28.5° inclination) using their Falcon 9 launch vehicle (LV) [178], which shall be used here for illustrative purposes. The real cost one would expect to incur would likely be a function of LV capacity utilisation, inclination, altitude and customisation of the interface amongst other factors.

5.5.3.4   Maintenance

For this mission, it shall be assumed that the satellite has no on-orbit servicing capability and will de-orbit naturally, thus both maintenance and disposal cost will be zero.

5.5.4   Results and Discussion

Results of the federated satellite system case study are illustrated in the following, with the aim of capturing the value added from an ISN capability on the satellite of interest. Figure 32 shows the multi-attribute utility plotted against mission cost, with a linear maximum-price value implemented to indicate constant value as defined by the stakeholders. In other words, any design point that resides below the dashed line offers lower value than the baseline (no ISN) design, while anything above the line above offers greater value.
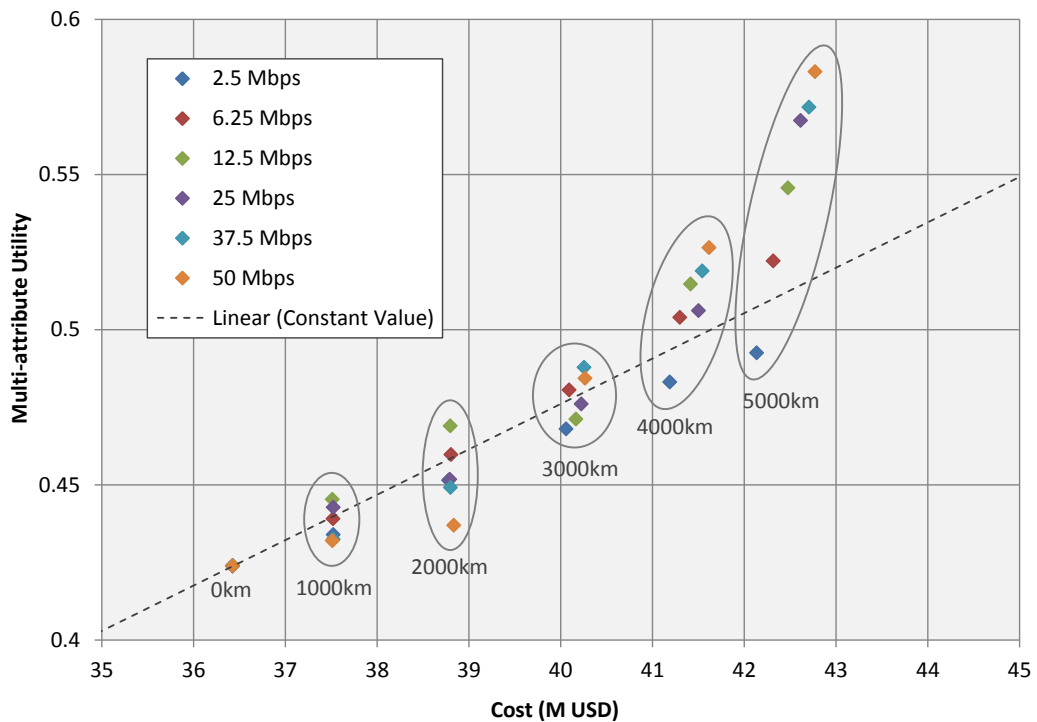
Figure 32 – Multi-attribute utility vs Cost, for the federated satellite system case study

The maximum price value has been selected for illustrative purposes, following calculation of the cost and utility offered by the baseline, non-networked system. In this case study, a linear constant value function, with a gradient of 0.01463, is used which passes through the baseline design point (0km ISL range).

Some interesting conclusions can be drawn from Figure 32 that arguably would not have been apparent without this analysis. Implementation of a high data rate ISL is of greater value at higher link range, whereas a very low ISL data rate system offers consistently lower value than the baseline. Also clear is that cost increase is driven by link range rather than data rate, which is a reflection of the increased number of

contact opportunities available at higher link range and thus greater power demand from the platform. This, in turn, requires both a larger electrical power system (greater mass) and more download requests from carrying other's data more often.

As discussed at the start of this chapter, understanding the utility-cost relationship is only the first step in appreciating potential added value to the mission. The designer must make a decision based on this information as to which design modification is most attractive, based on both the nominal value over and above the baseline, but also the complexity and cost of the modification itself. Recall that the cost derived during this analysis accounts for implementation of the modification to the platform, and the additional platform, launch and operations costs coming from that. However, the cost of developing/purchasing the technology is neglected. Figure 33, below, shows the price one should be willing to pay for developing the modification, accounting for the difference between actual value and constant value of the different design points.

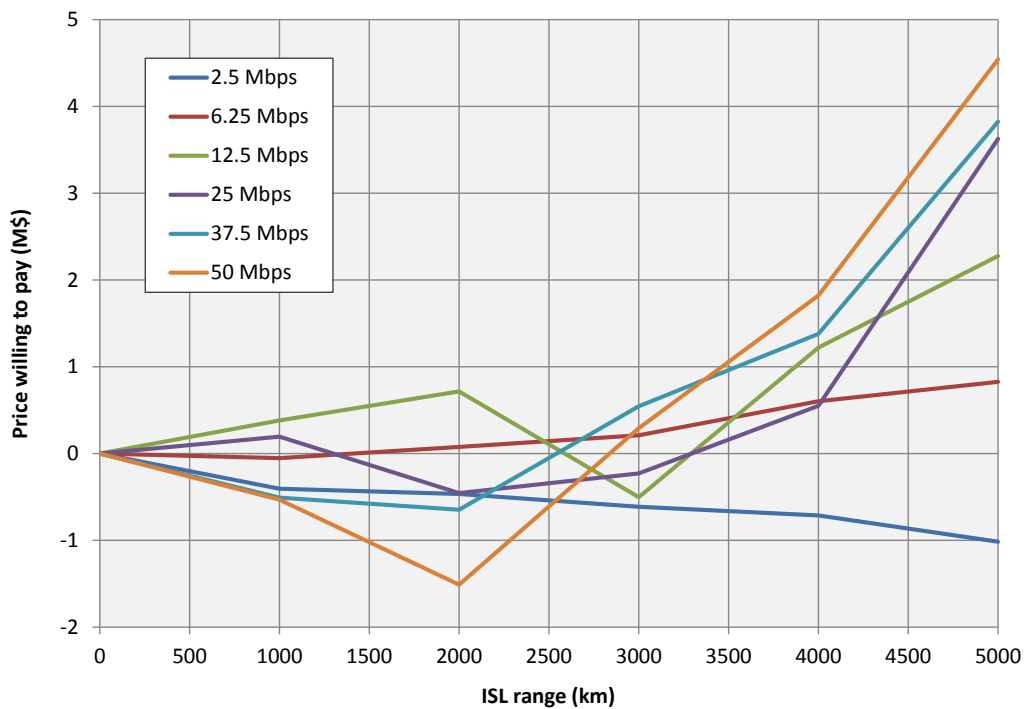*Figure 33 – Price willing to pay for ISN addition vs. ISL range, for different data rates*

From the results shown in Figure 33, it could be considered likely that the most attractive designs to take forward to a more detailed phase would be that with a 12.5 Mbps & 2000km ISL system, or those with 25-50 Mbps & 5000km ISL system. A greater data rate and link range would present a more challenging design, which would need to

be accounted for. Once the expected cost is subtracted from the willingness price, the remaining amount is the true value of ISN.

Looking in more detail at three of the designs that were evaluated, it is possible to extract a greater level of understanding of their performance (Table 14).

Attribute	DR = 12.5Mbps Range = 2000km	DR = 50.0Mbps Range = 2000km	DR = 50.0Mbps Range = 5000km
Utility metrics			
Delivery delay (min)	41.5	39.5	28.9
Delay variance (min)	27.9	27.5	21.2
Hop count	1.06	1.22	1.50
# packets delivered over ISL	197	736	1749
Cost metrics			
Platform cost (M$)	27.5	27.5	28.4
Launch cost (M$)	0.73	0.73	0.77
Operations cost (M$)	10.5	10.5	13.6
Design parameters			
Platform mass (kg)	155.8	157.7	165.4
ISN system mass (kg)	5.69	7.59	14.38
ISL transmit power (W)	0.64	5.09	31.8
Increase in Solar array area due to ISL demand (m^2)	0.018	0.026	0.157

Table 14 – Mission attributes for three promising design architectures

5.6 Case Study 2: Nano-Satellite Constellation

In this second case study, the value of ISN to a nano-satellite constellation, tasked with carrying out micro-wave radiometry for extreme weather detection, is investigated. The constellation must perform measurements on global tropospheric properties in support of severe weather now-casting, including the detection of convective thunderstorms, cyclones and hurricanes. To identify the optimal design point for this mission, from a pre-defined design space, a trade-space analysis is carried out. Spae routing (as defined in Chapter 3) is used as the data transfer mechanism, in the cases when ISN is included, offering an efficient means of routing data through the network.

Unlike in the previous case study, all satellites in this mission work toward a common set of objectives, such that there is no reliance on externally operated satellites for assistance. This is not to say that inclusion of other assets would not be of benefit, however a line must be drawn on the size of the design space for practical reasons. Requirements exist, with which the mission must comply, and are summarised in Table 15.

#	Requirement	Aim
1	Revisit rate at +/- 40° latitude	> 1 visit per hour
2	Latency at +/- 40° latitude	< 1 hour
3	Platform mass	< 20 kg
4	Platform volume	< 12 litres
5	De-orbit lifetime (post end of operations)	< 25 years

Table 15 – Mission requirements

In order to comply with requirements 3 & 4, limitations are imposed on the platform such that it must comply with the CubeSat standard [179]. For this, the structure must comprise an integer number of single unit (U) elements, of dimensions ~10cm x 10cm x 10cm. Compliance with requirement 5 is guaranteed by incorporating an AEOLDOS de-orbit device [180] onto those platforms that do not comply in their nominal configuration.

5.6.1   Payload Design

The instrument on board each platform is a passive microwave radiometer, which scans the surface of the Earth using a rotating (17RPM) offset parabolic antenna that reflects microwaves onto a detector in the main body of the satellite (Figure 34).
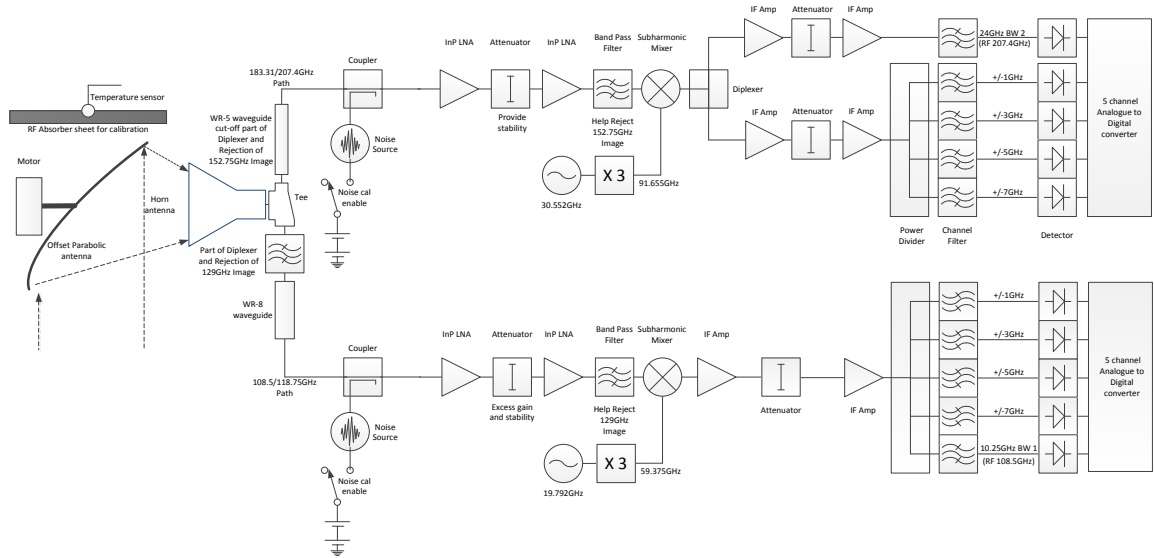

Figure 34 – Example design of a suitable payload instrument [181]

With the antenna rotation axis parallel to the direction of flight, a scanning motion is achieved such that data can be acquired by sweeping in the across-track direction while moving forward in the along-track direction (Figure 35).
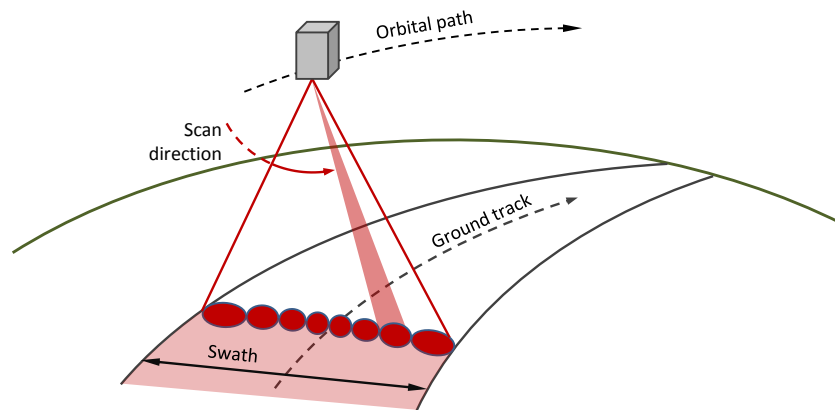

Figure 35 – Instrument data acquisition process

Data is collected at two discrete frequencies, 118GHz and 183GHz, which enables detection of temperature profiles and precipitation respectively, while also being manageable in terms of data acquisition volume, resolution and sensor geometry, given

the limitations on platform size. Assuming an aperture of 75mm diameter, beam widths of 4.1° and 3.5° can be achieved for measurements at 118GHz and 183GHz, respectively. The instrument collects data during the rotation arc that is between +/- 45° either side of nadir at a rate of 20 kbps, which includes telemetry and data overheads.

As for any Earth observation platform with a fixed-geometry payload design, resolution is a function of altitude and cannot exceed certain physical limits. For this specific design, pixel resolution is defined as the distance on the ground between two spot beam measurements. If the antenna is designed such that it achieves a beam-width (θ) close to the theoretical minimum with respect to each channel frequency, then pixel resolution (r) can be formulated as

$$r \approx H\theta, \qquad\qquad\qquad 5.25$$

where H is the satellite altitude above the Earth's surface. As the instrument scans away from the sub-satellite point and toward the fringes of the swath, the resolution decreases and the footprint of the spot-beam becomes elliptical (Figure 36).
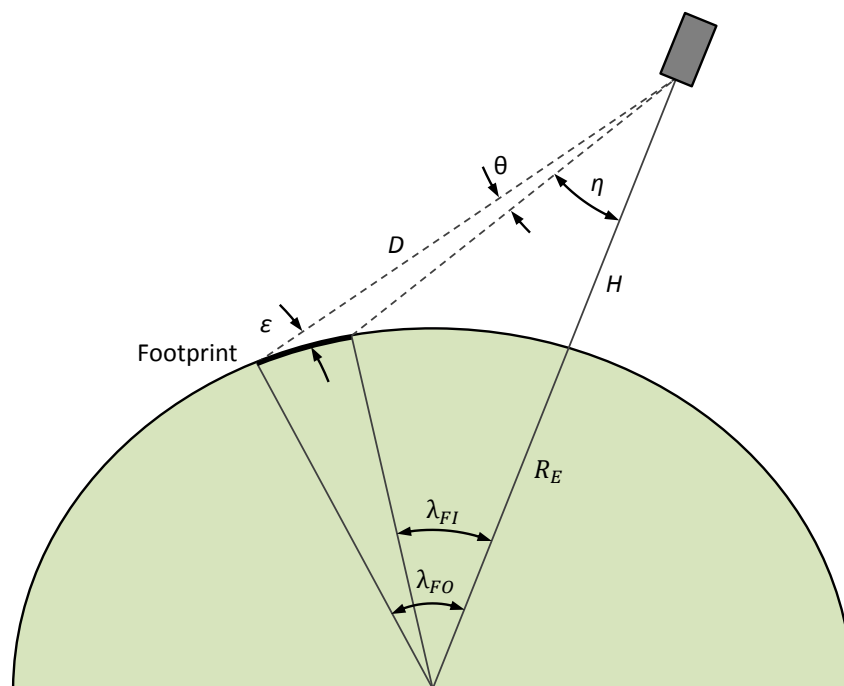

Figure 36 – Off-nadir sensor footprint (satellite altitude exaggerated for clarity)

This is an inevitable consequence of remote sensing from space and the footprint length (L_F) can be approximated by

$$L_F \approx \frac{D \sin \theta}{\sin(\varepsilon)},$$ 5.26

where

$$D = \frac{R_E \sin \lambda_{F0}}{\sin(\eta + \theta)},$$ 5.27

$$\varepsilon = \cos^{-1} \left\{ \frac{(R_E + H) \sin(\eta + \theta)}{R_E} \right\} \text{ and }$$ 5.28

$$\lambda_{F0} = \frac{\pi}{2} - \varepsilon - (\eta + \theta),$$ 5.29

where D is the slant range to the outer edge of the footprint, ε is the elevation above the ground, λ_{F0} is the Earth central angle to the outer edge of the footprint and η is the swath angle minus the beam-width (θ) (Figure 36).

The relationship between altitude and resolution, where resolution here is taken as the maximum diameter of the instrument spot footprint (this will equal the diameter at the sub-satellite point), defined by equations 5.25 and 5.26, is shown in Figure 37.
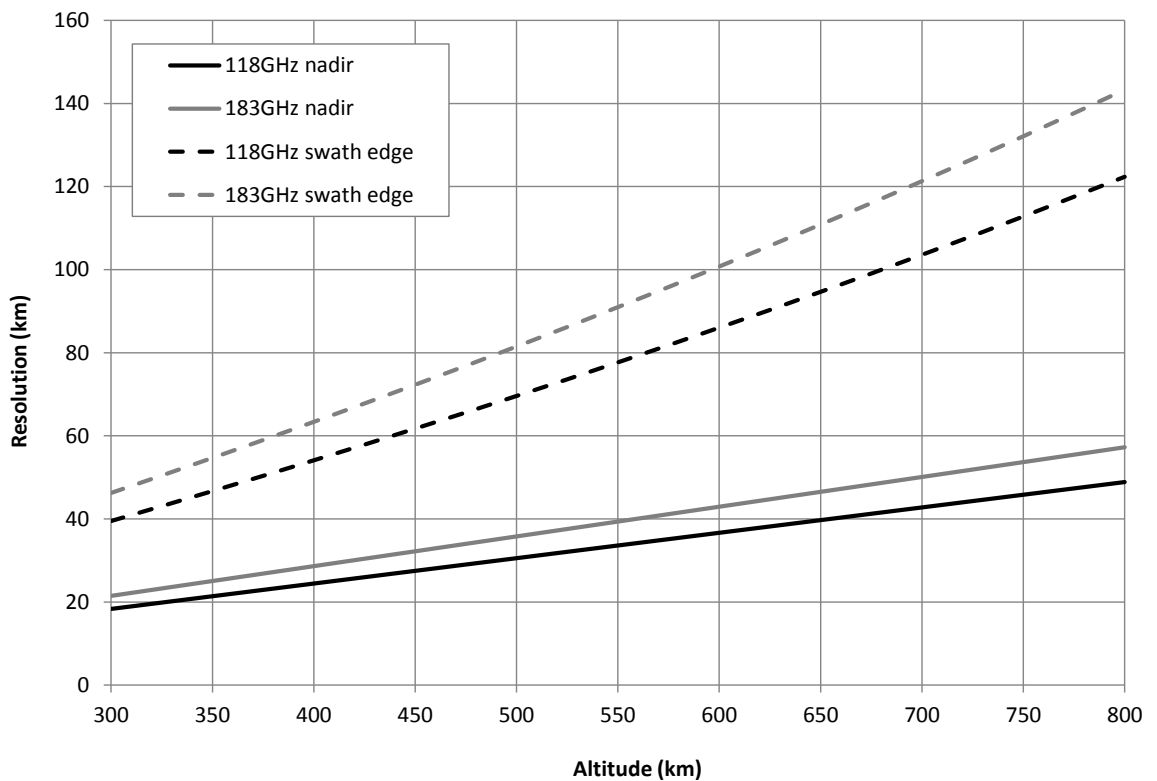

Figure 37 – Pixel resolution vs. altitude (for nadir and swath-edge measurements)

During the analysis, it is assumed that data for all points on the Earth's surface that fall within the swath of the instrument will be captured.

5.6.2   Design Space

A trade-space analysis approach is taken in this example, in which multiple design architectures are examined and compared. The following sections describe the set of variables that make up the design space.

5.6.2.1   Constellation Topology

A Walker Delta/Walker Star [112] hybrid constellation shall be employed for this mission in order to provide efficient global coverage. The Delta component, comprising satellites in medium inclination orbits, provides the necessary means of achieving a sufficient revisit rate over low and medium latitudes, while coverage over high

latitudes is achieved from the Walker Star constellation (at an inclination of 90°). Owing to the coupling between the Earth's rotation and the constellation, a more evenly distributed level of coverage can be obtained in this way, as opposed to a purely polar constellation, while maintaining a reasonable level of structured satellite placement. An example of this network is shown in Figure 38, where the ground tracks of a 3-plane Delta and a 2-plane Star constellation are imposed on the Earth-sphere. Note that while some areas may seem congested in this image, the satellites in the medium inclination Delta pattern will shift in right-ascension of ascending node during their lifetime due to effects of non-spherical Earth gravity. The polar satellites, however, will remain fixed in terms of this parameter, thus altering the relative positions of the orbit types over time.
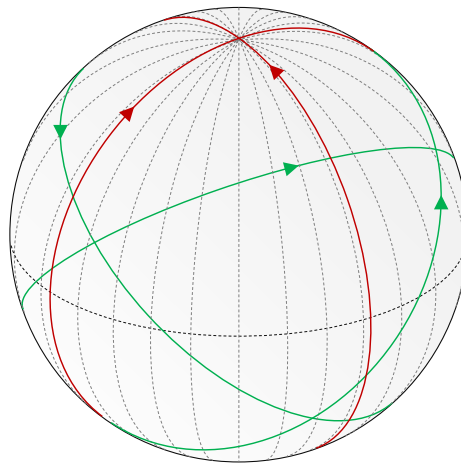
Figure 38 – Walker Delta (green) and Walker Star (red) hybrid constellation. Shows ground tracks for the 3-plane Delta and 2-plane Star networks

The Walker patterns, both Delta and Star, can be defined using the *i: T/P/F* scheme, where *i* is the orbit inclination, *T* is the total number of satellites in the constellation, *P* is the number of planes, around which the satellites are evenly distributed, and *F* is the phase, or relative spacing between satellites in adjacent planes. The phase can be defined as a function of the difference in true anomaly (v) between satellites in adjacent planes, as

$$F = \frac{vT}{2\pi}.$$
<div align="right">*5.30*</div>

Given the payload operational characteristics, the orbit altitude is a driver in defining spatial resolution (as discussed in Section 5.6.1). This, combined with the benefits

associated with uniform resolution over the entire data set (e.g. for image splicing and historical comparison), makes a consistent satellite altitude across the constellation attractive. That being said, a modest eccentricity would likely be considered acceptable in many cases, especially given the lack of consistency toward the swath edge, but for the purposes of this work it shall be assumed that all designs exhibit zero eccentricity and a similar altitude across the network.

5.6.2.2   Ground Station Network

Two ground station networks (GSNs) are considered in the design space, both comprising a sub-set of ground stations from the nominal and/or augment ESTRACK network. The properties of each are defined in Table 16 and shown in Figure 39.

GSN	No. GSs	Locations	Latitude (°)	Longitude (°)	Altitude (m above sea level)
1	5	Villafranca, Spain	40.4N	4.0W	665
		South Point, USA	19.0N	155.6W	367
		Kourou, French Guiana	5.3N	52.8W	-15
		Hartebeesthoek, S. Africa	26.0S	28.0E	50
		Perth, Austrailia	31.8S.	115.9E	22
2	8	GSN 1 +...	-	-	-
		Svalbard, Norway	78.2N;	15.4E	400
		Troll, Antarctica	72.0S;	2.5E	1270
		Santiago, Chile	33.1S	70.7W	723

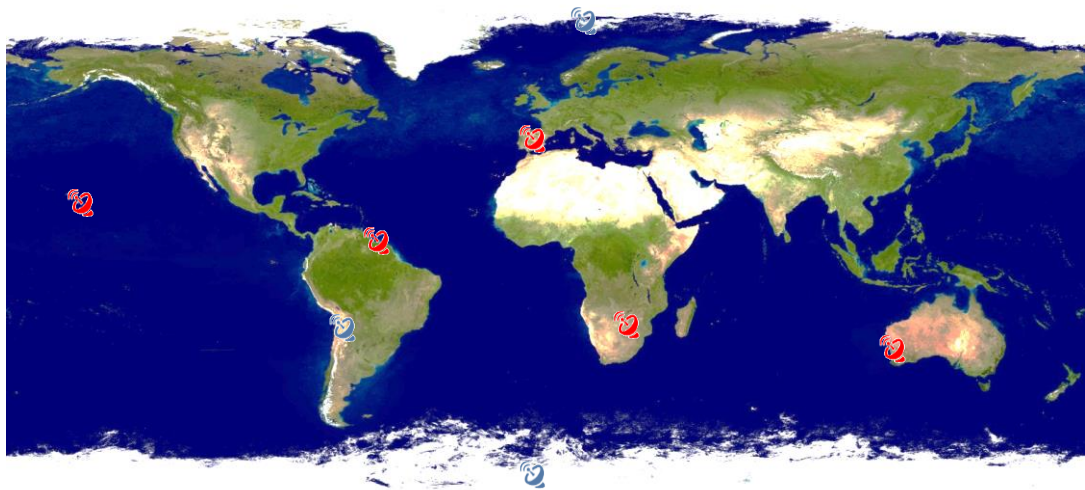*Table 16 – Ground station network properties*

Figure 39 – Ground station network map (those in GSN2 are in blue)

As would be expected, a larger GSN offers more capacity for data download and greater frequency of passes, which generally results in reduced delivery delay, higher delivery ratio and higher revisit rate, at the expense of additional operational cost.

5.6.2.3   Inter-Satellite Links

For half of the designs investigated in this case study, it is assumed that inter-satellite networking is possible, using an omni-directional communication system. Assuming a specific hardware design, the power demand from the ISN sub-system can be expressed as a function of the signal and hardware properties using a link budget analysis. The link budget parameters, from which transmission power is derived, are outlined in Table 17.

Parameter	Definition	Unit	Quantity	Comment
G_T	Transmission antenna gain	dBi	0	Omni-directional
G_R	Receiver antenna gain	dBi	0	Omni-directional
E_b/N_0	Energy-per-bit to noise ratio	dB	12.6	9.6dB required for GMSK modulation + 3dB margin
k	Boltzmann's constant	J/K	1.38×10^{-23}	Constant
T_S	Receive system noise temperature	K	340	
R	Signal data rate	kbps	200	
L_S	Free-space losses	dB	Link-range dependent	$L_S = \left(\dfrac{\lambda}{4\pi S}\right)^2$
λ	Signal wavelength	m	0.69	435MHz (UHF)
S	Link range	km	2000	See Figure 40 below
L_A	Attenuation loss (due to atmosphere)	dB	0	Space-space link so no atmospheric effects
L_L	Line losses (transmitter to antenna)	dB	0.5	

Table 17 – Link budget parameters

The relationship between RF output power and link range is presented in Figure 40, along with the input power required, assuming a system efficiency of 10%.
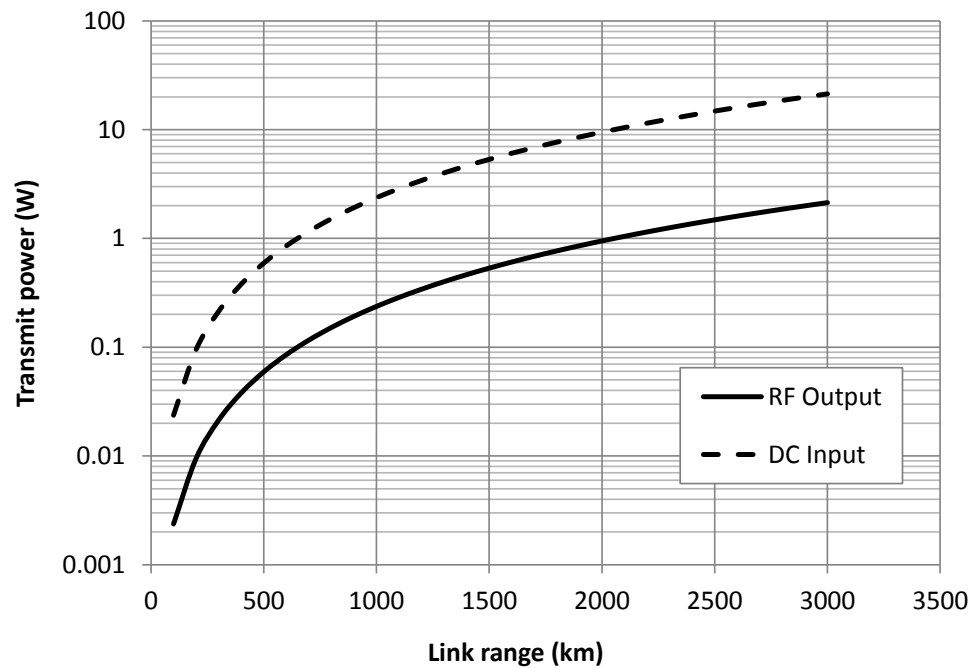

Figure 40 – Inter-satellite link input and RF output power vs. link range

It is clear from Figure 40 that a link range greater 2000km requires more than 10W input power during operation, which is a significant fraction of the nano-satellite's power budget. This will therefore be the maximum range over which inter-satellite communication is considered to take place.

5.6.2.4   Time To Live

Two different data life-times are included in the design space, in order to assess the impact of packet dropping on overall performance. Owing to the nature of severe weather monitoring and detection, low latency measurements are critical, such that data beyond a certain age could be considered redundant. In which case, it could be argued that delivering such data detracts from the potential to download new data, however it is often useful to have older data for validation and comparison purposes. Times to live of 60 minutes, 120 minutes and infinite lifetime are compared.

5.6.2.5 Design Space Summary

The design space that is explored is summarised in Table 18, from which a total of 576 designs are analysed.

Variable	Units	Range	Interval	No. variables
Altitude	km	[400, 600]	100	3
Inclination (Delta)	deg.	[40, 60]	10	3
T/P/F* (Delta)	-	[15/5/1], [16/4/1], [18/6/2], [20/4/2]	-	4
T/P/F (Star)	-	[6/2/1] & [9/3/1]	-	2
Data Time to Live	min	[60, 120]	-	2
Ground station network	-	GSN1, GSN2	-	2
ISL range	km	[0, 2000]	-	2
TOTAL				576

Table 18 – Design space

For each design point, 10 simulations are executed, each with the right ascension of ascending nodes for the two constellations defined at random. The average of each performance attribute across these executions is used for the utility calculation.

---

* Walker pattern definition, where T = total number of satellites, P = total number of planes and F = true anomaly phase

5.6.3   Mission Utility

To meet with the mission objectives and requirements, the following performance attributes are considered (Table 19).

Attribute	Units	Aim	Definition
Pixel resolution	km	Min.	Distance on ground distinguishable by a single pixel – average of nadir image at 118GHz and the 183GHz
Revisit rate* mean	Images per hour	Max.	Mean average of the mean revisit rate
Revisit rate variance	-	Min.	Difference between 1st and 3rd quartile of revisit rate
Data delay mean	minutes	Min.	Mean average of the mean data delay
Data delay variance	-	Min.	Difference between 1st and 3rd quartile of the delay
Delivery ratio mean	[0, 1]	Max.	Mean average of the mean DR
Delivery ratio variance	-	Min.	Difference between 1st and 3rd quartile of the delivery ratio

Table 19 – Nano-satellite performance attributes

For the purposes of this investigation, the single-attribute utility functions take a linear form (Section 4.2.4.2) and the ranges corresponding to a 0 – 1 utility score.

---

* For any particular target on the ground, there exists a set of sightings from satellites in the constellation, which are subsequently downloaded to users on the ground. The duration between these sightings is considered the revisit rate at the target. Over the lifetime of the mission, the average revisit rate at a particular target can be calculated as the total number of delivered images divided by the lifetime. The metric here is the average from all targets over the globe.

Their weightings, corresponding to their multi-attribute utility contribution, are outlined in Table 20.

Attribute	Range	Multi-attribute weighting (k)
Pixel resolution	20 – 50 km	2
Revisit rate mean	0 – 3 visits / hr	3
Revisit rate variance	0 – 1.5 visits / hr	2
Data latency mean	0 – 60 minutes	3
Data latency variance	0 – 30 mins	2
Delivery ratio mean	0 – 1	1
Delivery ratio variance	0 – 0.5	1

Table 20 – Single attribute utility properties

While pixel resolution is derived purely as a function of altitude, the other six attributes rely on simulation outputs, since all are inter-connected and dependent on other network parameters in a non-trivial manner. For example, a short time to live on the collected data is likely to guarantee a low delivery delay, since old data is discarded thus freeing up bandwidth for remaining data, but will also result in a low delivery ratio for the same reason. There is no analytical way of fully capturing these relationships.

5.6.4 Mission Cost

Owing to the lack of available cost models for nano-satellite missions, a bespoke model has been developed for this work, which is described in the following sections, and in [140]. The total mission cost is defined as the sum of the cost for each segment (space, ground, launch, operations and maintenance), such that

$$C_{total} = C_{space} + C_{ground} + C_{launch} + C_{ops} + C_{maint}. \qquad 5.31$$

5.6.4.1 Space Segment

The approach taken to costing the nano-satellite platform is summarised by the algorithm in Table 3, in which the system design takes an iterative form in order to best satisfy the requirements, for minimum cost. The design process includes sizing of the power system required for operations, inclusion of a de-orbit device for systems that are non-compliant with de-orbit regulations (re-enter within 25 years post-operations)

and consideration of discrete sizing outlined in the official CubeSat design specification [179].

Algorithm: Space segment cost	Definition/reference
Input: Mission aims, objectives and requirements	
Output: nano-satellite system definition and cost	
Select sub-systems based on mission requirements	
Sum nominal power for each sub-system (j) in sun & eclipse	Eqn. 5.16 & 5.17
Calculate battery capacity requirement	Eqn. 5.19
Calculate solar array area requirement	Eqn. 5.14[*]
for all discrete volume possibilities (V_i) **do**	1U, 2U, 3U, 4U or 6U
Sum volume of each sub-system ($V_{S/C} = \sum_{j=1}^{n} v_j$)	
if $V_{S/C} \leq V_i$ **then**	
Calculate number of solar arrays (incl. deployable)	Function of satellite geometry
Sum mass of components	
Calculate natural lifetime (t_n)	Eqn. 5.32 (below)
Mission lifetime = min(natural life, design life)	
Calculate de-orbit duration from end of life (t_d)	Eqn. 5.32
while $t_d > 25$ years	
Add deorbit device and recalculate t_d	
Update mass and volume	
Calculate de-orbit duration	Eqn. 5.32
if $V_{S/C} \leq V_i$ **then**	
Design is feasible	
Calculate cost of platform i as sum of sub-system cost plus margin	
else	
Design is infeasible	
else	
Design is infeasible	
Select design that is feasible and with minimum cost	

Table 21 – Algorithm for system design and costing

---

[*] Since solar arrays are assumed to be fixed in terms of orientation with the spacecraft body, it is assumed that on average, only 15% of the total solar array area is illuminated during sunlit conditions [140].

Platform selection can often be non-trivial, e.g. a fully laden 2U CubeSat in need of deployable solar panels and a de-orbit device may be more costly than a sparsely packed 3U CubeSat which can decay without such devices due to its lower ballistic coefficient. The lowest cost system is therefore selected as the one to take forward in the model. The following sections provide further detail into some of the more involved space segment design elements.

Development cost of the system, which is carried out by the platform provider, shall be approximated as a fixed-cost of five full-time employees, at £75k per year, for two years. Of course, the more complex systems exhibiting ISN capability will likely cost more to develop, however this cost shall be considered as part of the added value associated with ISN within the scope of this mission.

Mass, Volume, Power and Cost

The mass, volume, power and cost of the platform are calculated as a sum of their respective parameters for each sub-system on board. Sub-system selection is made according to mission requirements (e.g. a 3-axis attitude determination and control system is selected since a strict pointing demand from the payload/communication system requires it) and as a function of other dependent variables (e.g. battery hardware is selected such that it will satisfy the power demand from the sub-systems on board), from a database of components, taken from various online sources [182]–[184].

The power generation from satellites with fixed orientation solar panels and a time-varying attitude with respect to the Sun, has been investigated by the author in [140]. It was found that at various attitude schemes, and for different numbers of body mounted and deployed solar panels, energy collection efficiency (i.e. the amount of energy collected compared to the maximum amount that could be generated from all solar arrays) was between 15% - 27%. As such, a conservative approach is taken here, with 15% solar array utilisation being used.

De-orbit Device Selection

The requirement to include a de-orbit device is generally difficult to confirm, given large variability of atmospheric density with variation in both altitude and time [185], uncertainty in satellite drag coefficient and uncertainty in the body's uncontrolled

attitude relative to the incoming flow. However, in order to identify general trends, some assumptions are made which allow analytical approximations for the orbit decay time to be used. One approach is based on the fit of a power curve to the international standard atmosphere to form an analytical relationship between density and altitude between 150km and 1000km [186]. From this, it is possible to find the decay time (t_d) as a function of initial (r_i) and final (r_f) orbit radii, as

$$t_d = \frac{M}{C_D A \Lambda \sqrt{\mu R_E}} \frac{\left[(r_i - R_E)^{(1+\gamma)} - (r_f - R_E)^{(1+\gamma)} \right]}{1000^{\gamma}(1+\gamma)},$$ *5.32*

where M is the satellite mass, A is the average cross sectional area projected in the velocity direction, μ is the Earth gravitational parameter, R_E is the mean Earth volumetric radius (6371km), and γ and Λ are non-dimensional constants from the atmospheric density curve-fit, which take the values of 7.201 and 10^7 respectively. Given a beginning of life orbit radius and a minimum allowable orbit radius, considering acceptable limits for successful operations, the natural decay time can be calculated. The mission lifetime is thus the minimum of the natural decay time and the nominal lifetime. Note that solar flux effects have been neglected in this work, however general perturbations methods exist that could be incorporated should higher fidelity be required [165].

Compliance with the Inter-Agency Space Debris Coordination Committee 25 year de-orbit time recommendation [187] is imposed in this study, such that if non-compliant, a de-orbit device is added to the system. Here, the aerodynamic end of life de-orbit system (AEOLDOS) is used, which is a drag-sail device that can deploy at the end-of-life if required. Two variants exist, with either 1.5m² or 3.0m² cross-sectional areas, depending on system demand. A drag coefficient of 2.2 is used throughout this work [176].

Equation 5.32 is shown graphically in Figure 41 for a satellite of 5kg mass, with a nominal surface area of $0.02m^2$. Plots for the expected lifetime in the case of deploying a small or large AEOLDOS drag sail are also presented, to illustrate the effect of such a device.
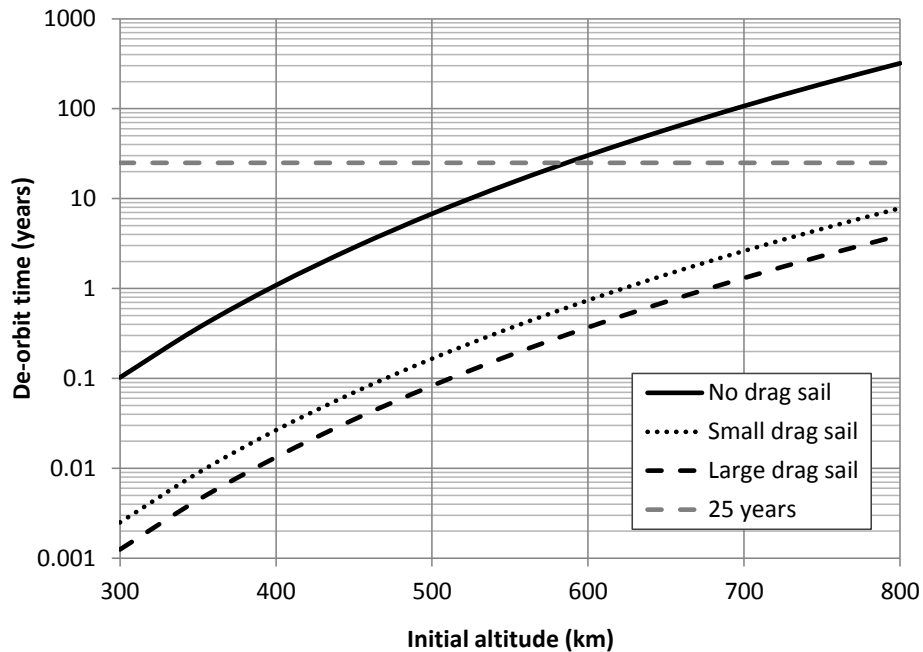

Figure 41 – De-orbit time vs. initial altitude

5.6.4.2   Ground Segment

Due to the large number of satellites being deployed for this mission, a traditional approach to ground station exploitation, whereby an arrangement is made with existing ground station operators to take care of data collection using existing facilities, would not be possible. It is assumed therefore, that dedicated tracking stations are installed at the sites indicated in Section 5.6.2.2, which exploit automation and remote access, without the need for on-going on-site human interaction. This strategy can be considered relevant at the time of writing, as it reflects the approach taken by Spire Global with respect to their ground station network.

The cost of setting up a suitable ground station, with S-band transmit and receive capability, capable of remote access operations is in the region of £80k [188] (including installation and maintenance), which shall be used as a one-off cost for each ground station in the design architecture's respective network.

5.6.4.3 Launch Segment

Launch cost is dependent on a number of factors, including the orbit to which the platforms are being delivered, the launch vehicle type that is commissioned and the number of launches required to build the constellation. Furthermore, the number of LVs required is dependent on the individual LV capacity and the number of satellites in each orbit plane[*]. The relationship is not continuous, for example three LVs capable of carrying 5 platforms each could populate a 15/3/1 Walker Delta constellation, whereas four would be required for a 16/4/2 constellation, despite the addition of only one satellite. In the latter case, unless another satellite is found to populate the remaining availability on each of the launches, the cost per platform would increase by a factor of 1.2.

Since published costs for the dedicated nano-satellite launch vehicles are not yet available, an estimation is made based on various articles, market predictions and current ride-share costs. It is noted that there is likely to exist a cost saving in the case of ride-share launches over dedicated ones, but a fully exploited nano-LV has the potential to be comparable in cost. A cost estimate is therefore made based on the number of CubeSats per LV, assuming one LV is required for each orbit plane in the constellation (Table 22).

No. satellites per LV	Cost per satellite (£k)
3	650
4	600
5	550

Table 22 – Estimated cost per satellite based on number per launch

5.6.4.4 Operations

For this mission, a remotely accessed ground segment shall be operated, allowing for a significant cost reduction over traditional methods of paying in-situ operators per pass, as demonstrated by Spire Global, with a network of ground stations in 21 locations at the time of writing. Given a significant reduction in on-site maintenance and human interaction, and in-house operations being carried out for the majority of contact

---

[*] While it is acknowledged that there are launch vehicle upper stages capable of multiple plane injections in operation (e.g. Fregat), for the purposes of this study, the cost model is restricted to LVs with single-plane deployment capability.

events, it is assumed that an order of magnitude reduction in cost is attainable, with £20 per ground station pass (C_{pass}) attainable. The operations cost, for the first three years, is therefore

$$C_{ops} = 3 \times 365 \times \sum_{i=1}^{m} \sum_{k=1}^{p} \beta_{ik} C_{pass,ik}, \qquad\qquad 5.33$$

where m is the number of satellites, p is the number of ground stations and β_{ik} is the average ground station pass rate (measured in passes per day) for each satellite-ground station pair. This cost is unique to each design architecture, increasing with the total number of GS passes. It is unlikely that 100% of GS passes would be utilised in reality, due to either unavailability or lack of demand, however the proposed cost presents a useful upper bound.

5.6.4.5   Maintenance

Replenishment of satellites in the constellation is required when a platform becomes non-operational, either through failure or through de-orbiting. The effects of failure are addressed in detail in Chapter 6 and as such only natural orbit decay is considered here. However, owing to the generally less stringent requirements on quality control and testing for nano-satellites, it is assumed that platforms in this case-study will either fail, or become obsolete, after 3 years of operations. Maintenance cost due to de-orbiting is therefore a function of the altitude, as orbit lifetime will dictate the frequency at which replacement must be carried out. While satellites in low orbits will achieve a better pixel resolution, they will decay in height at a greater rate compared to their higher altitude counterparts, and thus need more regular replacement.

Figure 42 shows the orbit lifetime (from equation 5.32) for a 5kg 6U CubeSat, as an example, with controlled attitude such that the smallest area face is always perpendicular to the satellite velocity vector. Where lifetime is found to be <3 years, a replacement system would need to be deployed to satisfy mission requirements. Note that it is possible for 3U CubeSats to de-orbit from an altitude of ~420km in <6 months, which highlights the variability in lifetime due to geometry, mass and solar flux activity.
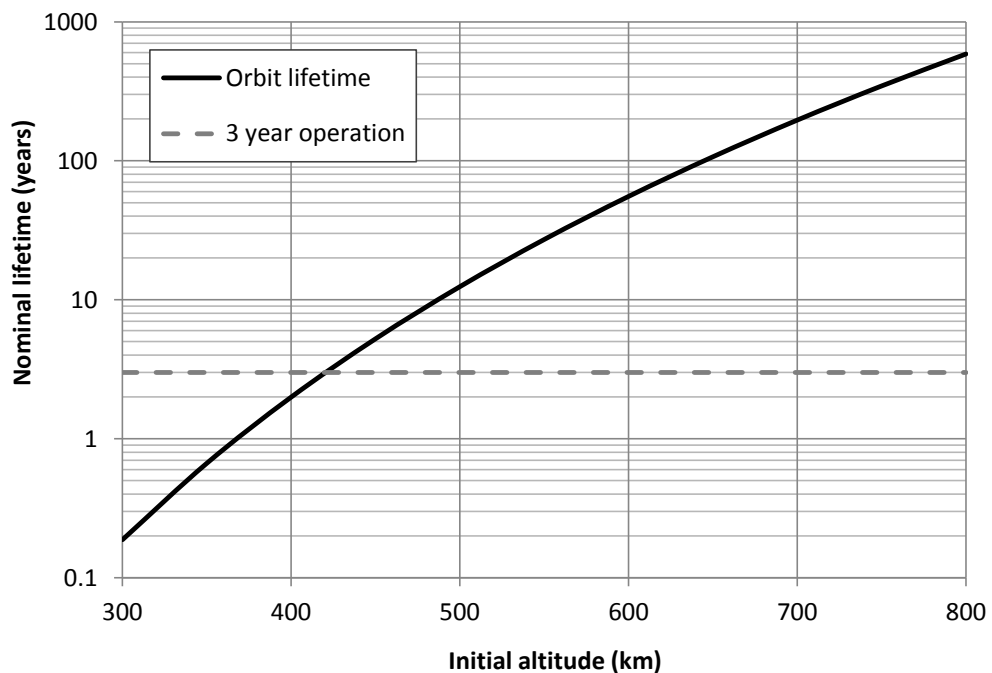
Figure 42 – Orbit lifetime vs. initial altitude

Since only the cost of the first three years are considered here, if a replacement is required within that timeframe, only the associated fraction of the replacement cost is considered. That is, if the lifetime is two years, only half the cost of replacing the fleet would be included in the maintenance cost for the first 3 years, since those platforms would have another year of life at the end of this period.

5.6.5   Analysis Environment

A model has been developed[*] in which satellites orbit over a network of targets and ground stations from which they collect data and to which they disseminate data respectively.

---

[*] Using the Matlab modelling environment

The targets, or sources, are distributed uniformly over the globe according to the location of vertices and edge mid-points on an icosahedron at level 1 (Figure 43).
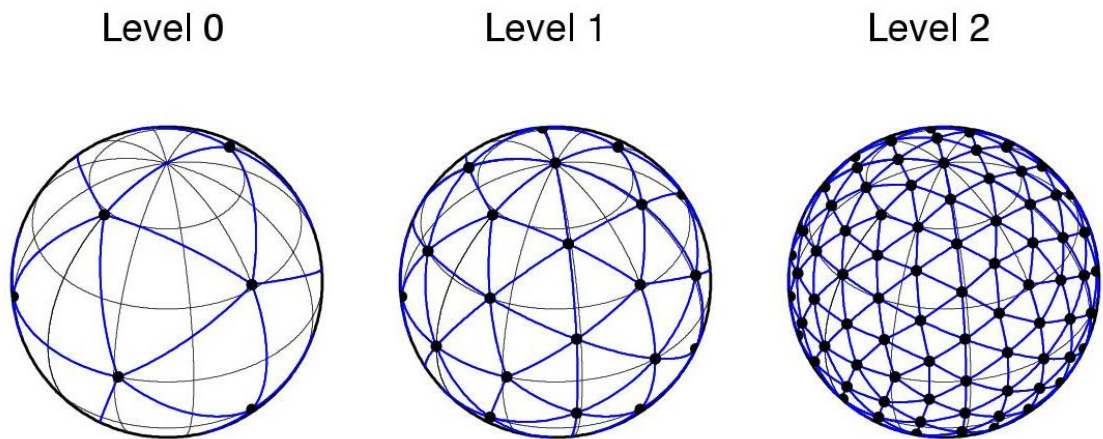
Figure 43 - Uniform distribution of points on a sphere using Icosahedron vertices [189]

In order to confirm compliance with the requirements of revisit rate at 40° latitude, 12 additional target nodes are distributed evenly around the +/-40° latitude lines. These are analysed separately from the other 42 nodes distributed evenly around the Earth. Noteworthy is that these source nodes are used in the performance assessment, but are not the only locations over which data is collected. Data collection is indeed assumed to happen continuously, providing an upper bound on network capacity and throughput, while ensuring global coverage.

Throughout this work, all data is considered to be of uniform priority, and delivery always follows a first-in-first-out (FIFO) scheme. In other words, data always joins the rear of the buffer queue and is delivered only when all older data has been delivered or deleted. This approach holds true also with the transfer of data via inter-satellite links (ISL), such that data arriving from a neighbouring satellite joins the queue behind its immediate chronological predecessor. I.e. the two buffer queues are merged.

Ten simulations of 1 day operation are executed on each design defined in (Table 18), with a time step of 60 seconds.

This parameter set offers an acceptable level of fidelity, while being considerate of computational requirements. Figure 44 illustrates the procedure for the trade-space analysis conducted for this case study.
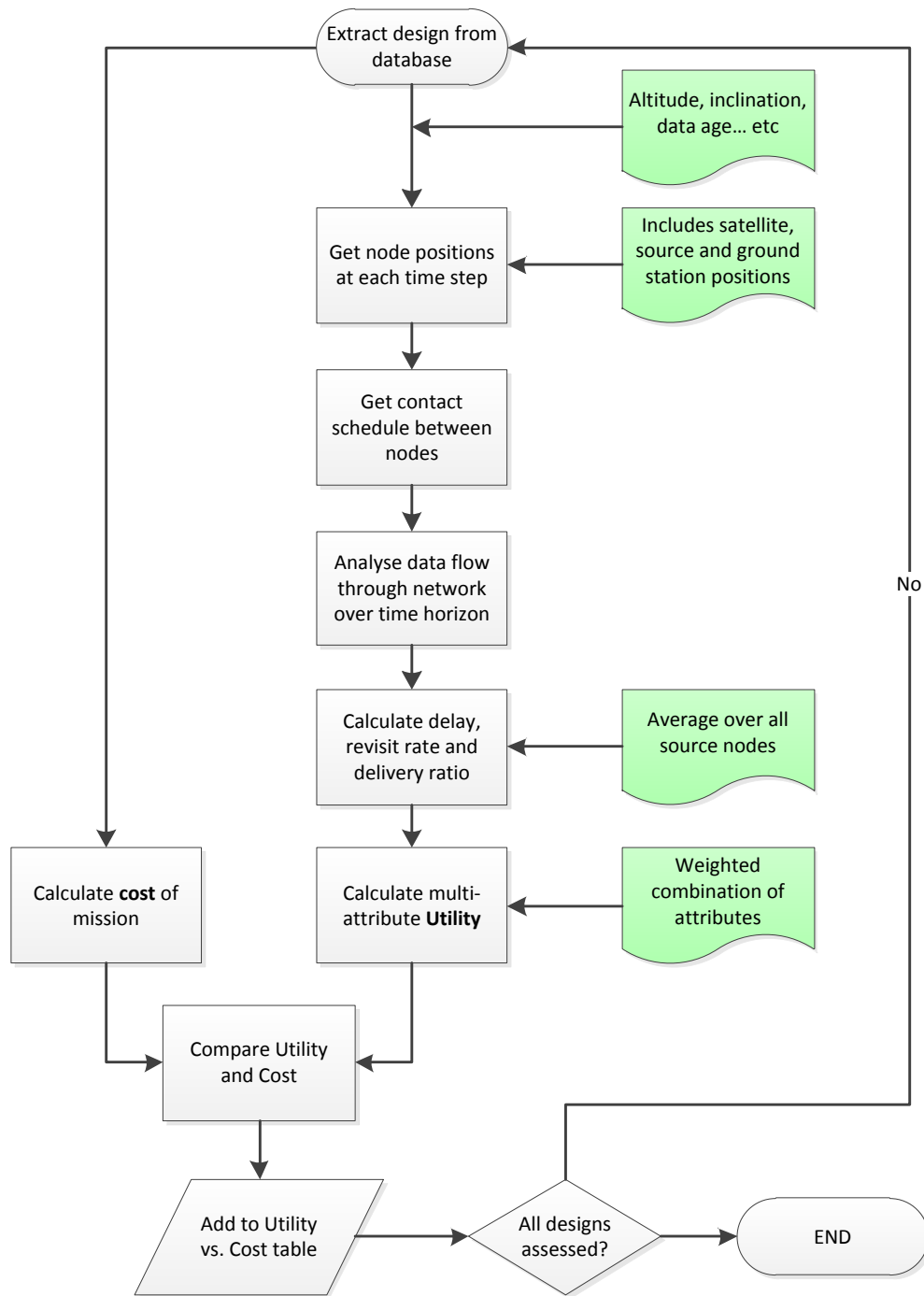

Figure 44 – Analysis procedure

5.6.6 Results and Discussion

A scatter-plot matrix showing the relationship between each of the single attribute utilities is presented in Figure 45. This sort of plot is useful in the case of multiple attributes, since it highlights, in this case, such characteristics as delivery ratio variance increasing with a decreasing delivery ratio and the apparent independence of latency with revisit rate. The diagonal entries (from top left to bottom right) include grouped histograms to show the distribution of each resolution (orbit altitude) set over that attribute. E.g. it is clear that only at 400km altitude is the variance in latency ~20 minutes.
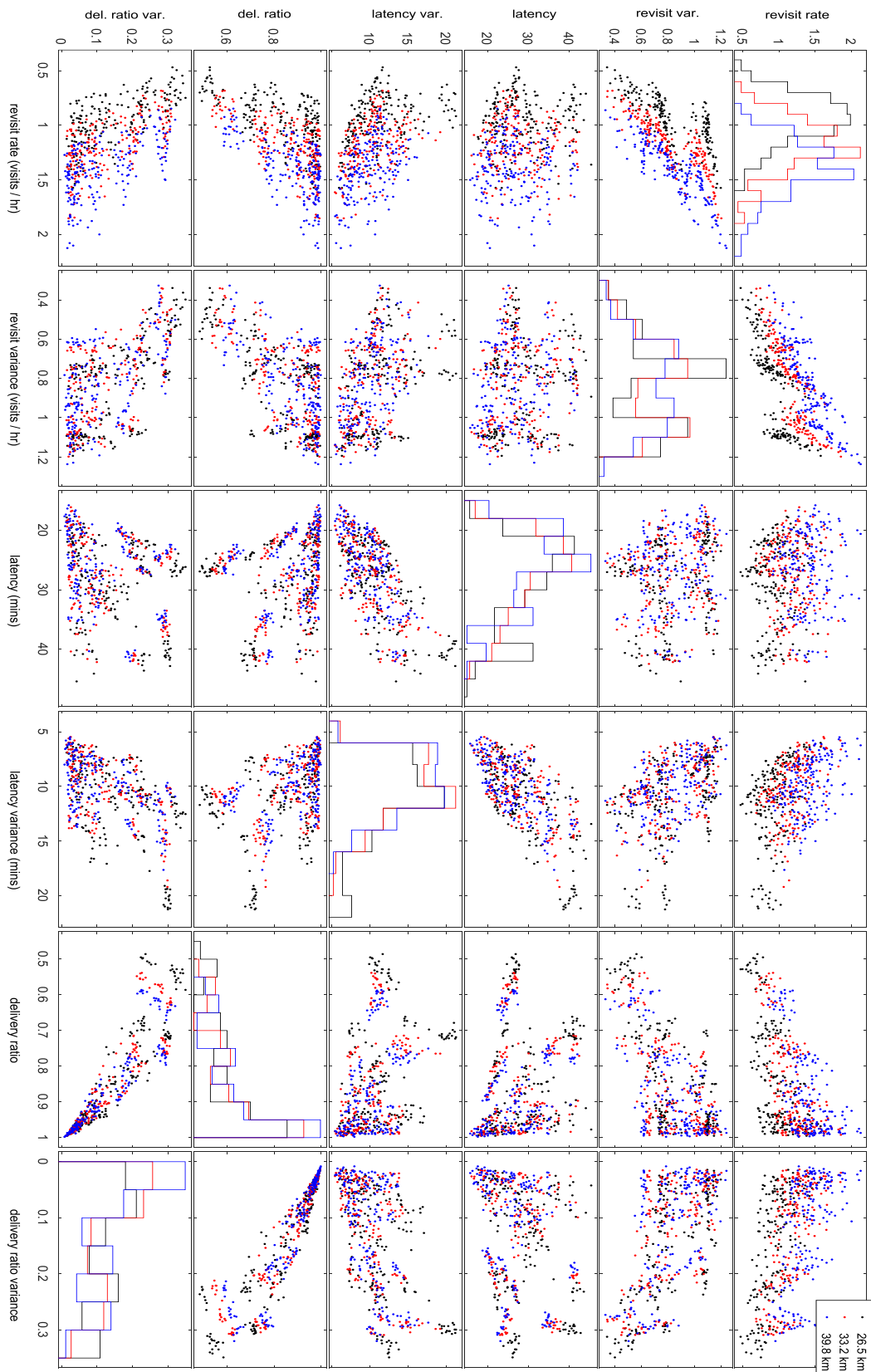
Figure 45 – Scatter-plot matrix showing relationship between each attribute – design architectures of different resolution (orbit altitude) grouped by colour

Results of the utility vs. cost data, grouped by ISL range, are presented below in Figure 46. Plots showing these results with the data grouped by the other design variables (altitude, inclination, network size, data time to live and GSN number) are shown in Appendix B. The design architectures which fail to comply with the mission requirements (Table 15) are represented by an '×' marker, while the compliant designs have a ' · ' marker. It is clear from Figure 46 that ISN generally results in a higher mission utility, which is to be expected, and in fact in some instances converts a non-compliant, non-networked, design, into one that meets the mission requirements. In contrast to the previous case study, in which only ISN parameters formed the design space, here it is possible to see how ISN is effected by other design variables, such as altitude, data time to live and ground station network topology.
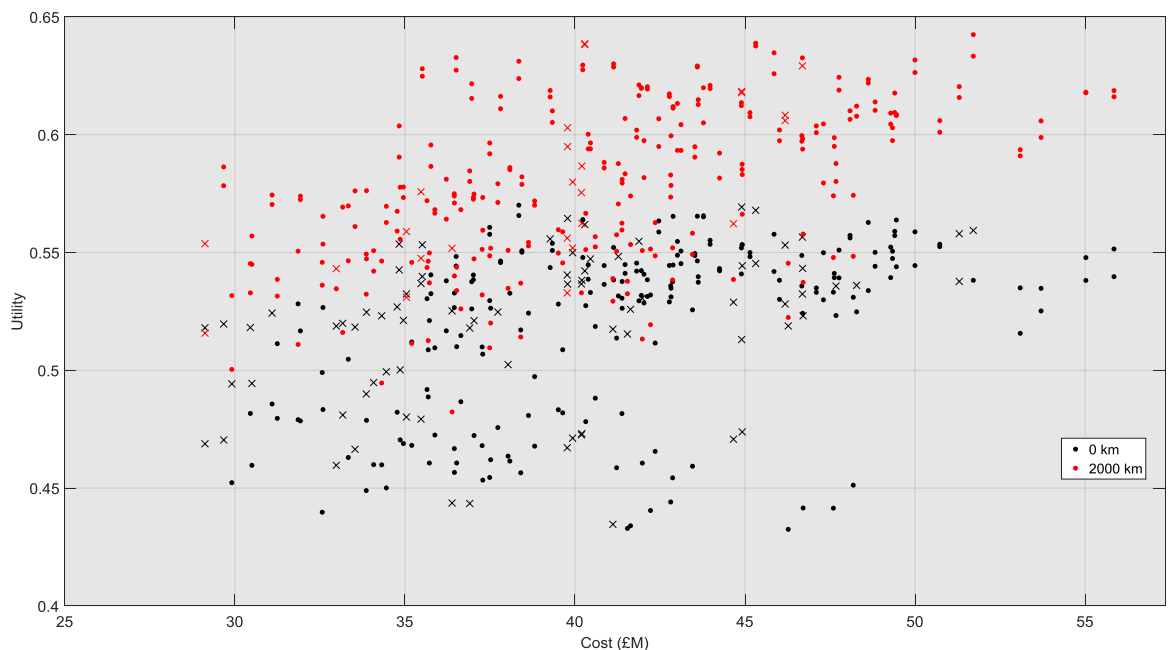
Figure 46 – Utility vs. cost of nano-satellite case study – results grouped by ISL range

Despite a general trend towards higher utility at higher cost, the preference value (U/C vs. C) tends to reduce with increasing cost, as shown in Figure 47, where the actual value of each design replaces the multi-attribute utility on the vertical axis. The highest value compliant design with ISN (#338) and the highest value compliant design without ISN (#193), are highlighted in Figure 47, with their design variables shown also. Note that these designs differ not only by their ISN capability, but also in inclination and Delta constellation topology. By comparing these designs, it is possible to derive the value of ISN for this design space, using equation 5.6. Recall that U' and C'

are the utility and cost of the ISN-capable design respectively, and V_0 is the value of the non-networked design. In this case, the value of ISN, or price one should be willing to pay for ISN, is 5.7M£. Whether or not this is worth it in reality, is a question that must be addressed by the design team responsible for this particular mission, as this value must be sufficient to design, develop and implement the ISN capability into each of the platforms.
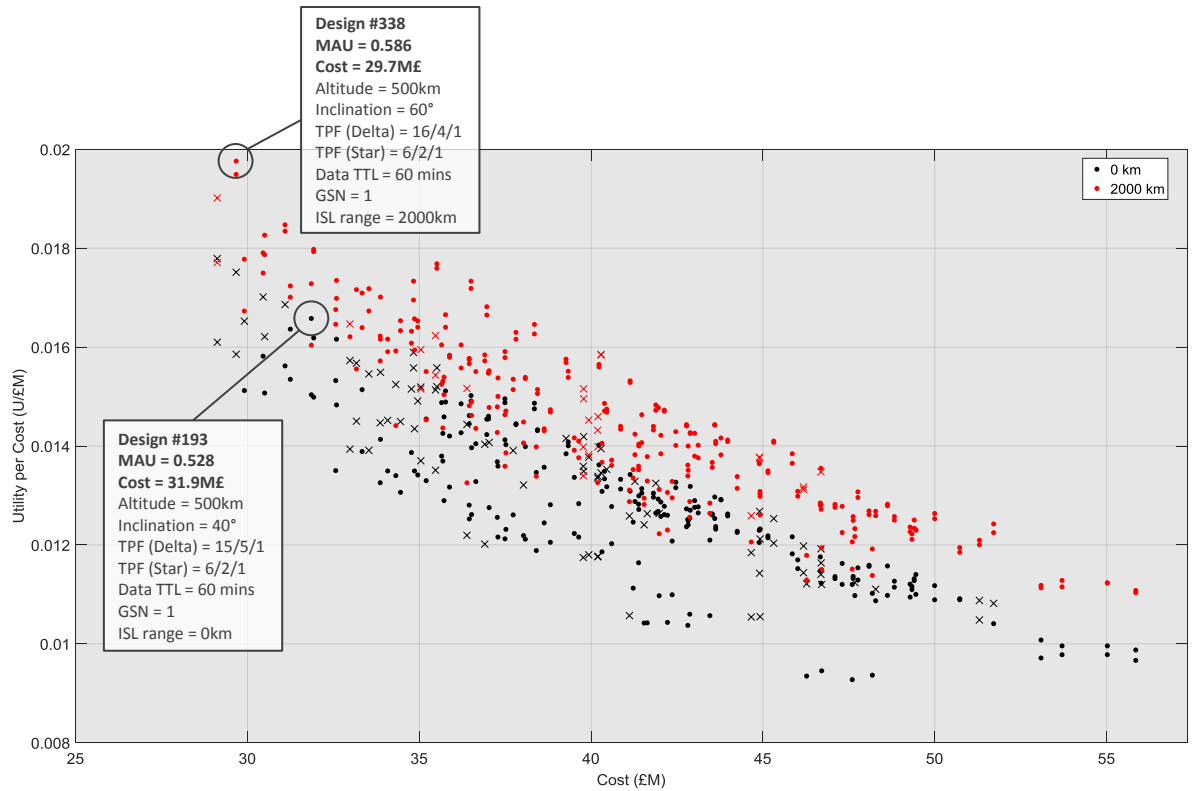

Figure 47 – Preference value (U/C) vs. cost of nano-satellite case study – highest value (compliant) design with and without ISN highlighted

A detailed analysis of the performance attributes including revisit rate, latency and delivery ratio, for each payload target latitude, are presented in Appendix B, for designs 193 and 338.

Chapter 6

Networking, Reliability and Resilience

In the previous chapter, methodologies for quantifying the value of inter-satellite networking during periods of certainty, were introduced and demonstrated. Whilst this is applicable to the majority of space systems for most of their lifetime, it is important to understand how a system may respond to failure and how that may affect performance. In this chapter the potential impact of failure is introduced, the way in which inter-satellite networking can help diminish the effects is presented and the value of inter-satellite networking for a space system that is subject to on-board failure, is identified. It will be shown that ISN has a generally positive effect on a system's survivability, since there exists additional functionality that is otherwise not present. Namely, this is the ability to relay information, an activity that has value both in terms of sending one's own data toward its final destination, but also in sending other's data along its respective journey.

6.1 Failure, its Sources, Types and Effects

As discussed in Section 4.3, aleatoric uncertainty shall be incorporated into this analysis, in particular with respect to system failure. Complete failure affects over 6% of satellites within their first 7 years and almost 9% of satellites within their first 12 years [163]. It is therefore not something that can be ignored, and both the likelihood of its arrival and its effects on the mission utility, should be understood. Exactly how a particular sub-system fails can be considered arbitrary for the purposes of this analysis, however it is important to understand the likelihood of failure occurring over a particular timeframe. It has been shown, in [163] and [190], that satellite sub-system failure can be approximated to a Weibull distribution, capturing the higher probability of failure at beginning of life that is neglected when using other functions, such as a linear regression.

The reliability of a traditional space platform, or in other words its probability of residing in a non-failed state, is shown in Figure 48.
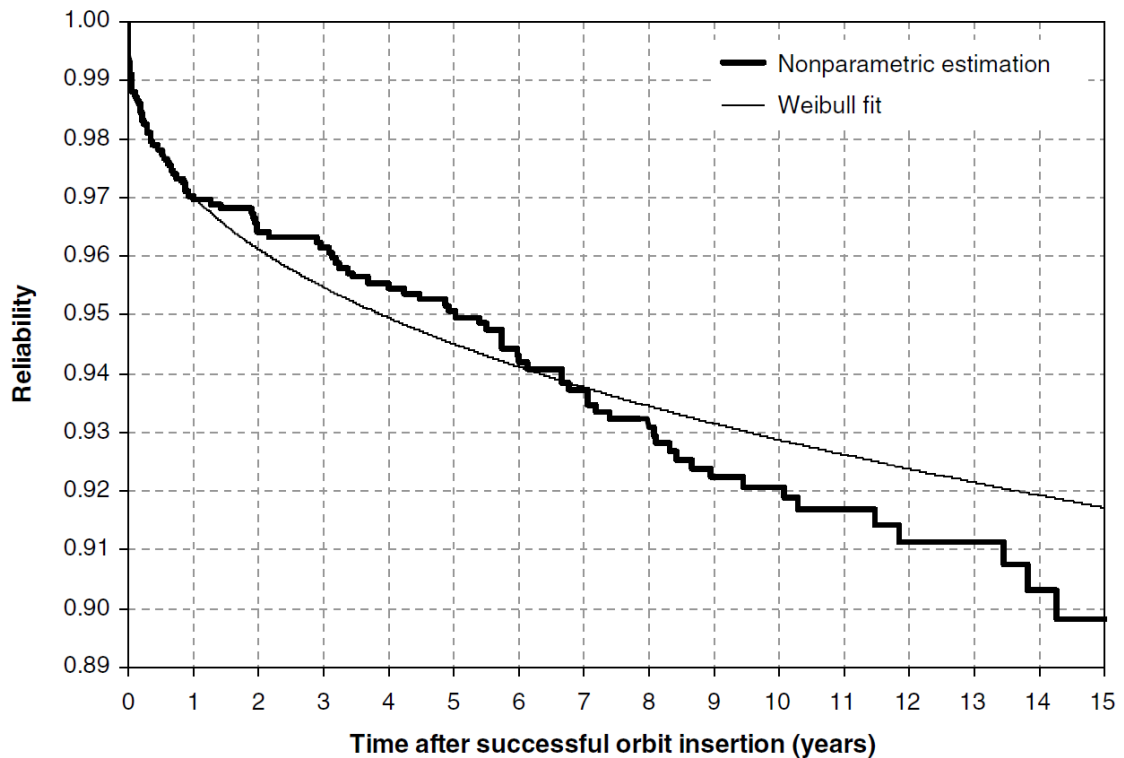

Figure 48 – Weibull distribution correlation to actual satellite reliability data [163]

6.2 State Transition Using Markov Chains

The operation of a traditional, monolithic satellite can typically be reduced to that of data collection via its payload/s, followed by data delivery to the end user via its communication sub-system/s. Failure of either of these sub-systems, or a critical supporting element (e.g. power system or on-board computer), would result in this functionality not being possible and thus transition into a failed state. In reality, there exist various intermediate, partially-failed states following failure of some non-critical component[*], but for the purposes of this work, and to effectively illustrate the effect of a networking capability, these are omitted. For a satellite with an inter-satellite networking capability, there exists another function in addition to the nominal collect and deliver capabilities. This is its ability to relay information, which not only provides a higher nominal utility via increased functionality, but introduces four degraded

---

[*] Consider, for example, failure of a string of solar cells or degraded reaction wheel mobility. The result would likely be a reduction in payload duty-cycle, such that some utility can still be maintained, but at a lower level than if fully operational

operational states that are not available to a non-networked system. This is described in Figure 49, where the assumption is made that additional hardware is required for ISN, thus offering additional resilience*.
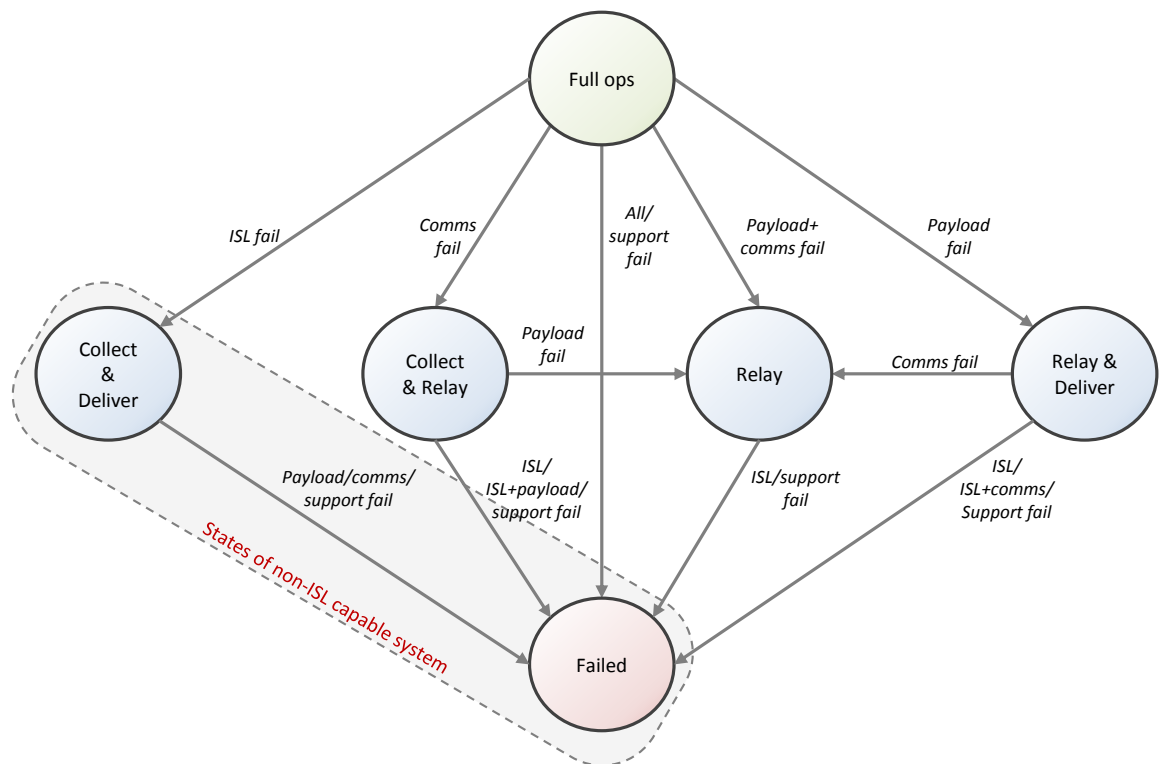

Figure 49 – State transition Markov chain diagram for a networked and non-networked system

Figure 49 shows the state possibilities being considered in this chapter, including the partially failed states and the potential transitions to and from each state, summarised in Table 23.

---

* While it is technically feasible that an ISN capability is implemented using the same hardware as used between the satellite and ground station, for the purposes of this work an ISL link is assumed to require additional hardware. This is considered a reasonable assumption given the likelihood of differences in link frequency and potential pointing demands.

No.	State	Transition from	to	Comment
1	Full Ops	None	All	All systems operational
2	Collect & Deliver	1	6	ISL fail (full ops for traditional system)
3	Collect & Relay	1	5, 6	Communications fail
4	Relay & Deliver	1	5, 6	Payload fail
5	Relay	1, 3, 4	6	Payload & communications fail
6	Failed	All	None	1. Critical support system failure, or 2. Combined failure of ISL and either payload or comunications systems

Table 23 – State definition and transition (transitions from/to self not included)

Indeed, all partially- or fully-failed states could transition to a more operational state if a repair/replace service is available. This concern is not addressed in this work, but is a logical next step, and is recommended as a future work topic. The Markov chain in Figure 49 can also be represented as a probability state-transition matrix of the form

$$P(t_k) = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{15} & p_{16} \\ 0 & p_{22} & 0 & 0 & 0 & p_{26} \\ 0 & 0 & p_{33} & 0 & P_{35} & p_{36} \\ 0 & 0 & 0 & p_{44} & p_{45} & p_{46} \\ 0 & 0 & 0 & 0 & p_{55} & p_{56} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad 6.1$$

where $P(t_k)$ is the transition matrix at time t_k, and p_{ij} is the probability that the system will transition from state i to state j during time period $t_{k+1} - t_k$. Note that the number of non-zero entries along the row direction indicates the number of states to which a state (defined by row number) can transition, and along the column direction the number of states from which a state (defined by column number) can be transitioned. Owing to the fact that the probability of transitioning between states, in the case of space platforms, is variable with time (time-nonhomogeneous), the discrete-time form of Markov chains shall be employed throughout, whereby the matrix P may differ over the lifetime. Recall from Section 4.3.1, that the probability of being in each state at time t_k, given the likelihood of being in each initial state as defined by the vector $\pi^{(0)}$, is

$$\pi^{(t_k)} = \pi^{(0)} P(0) P(1) \dots P(t_k - 1). \qquad 6.2$$

For illustrative purposes, consider a satellite with failure probabilities for the payload, ISL system, communication system and critical support systems, of 5%, 4%, 3% and 5% respectively, in any given year. The state probability plot is shown for both a non-networked system (Figure 50) and a networked system (Figure 51).
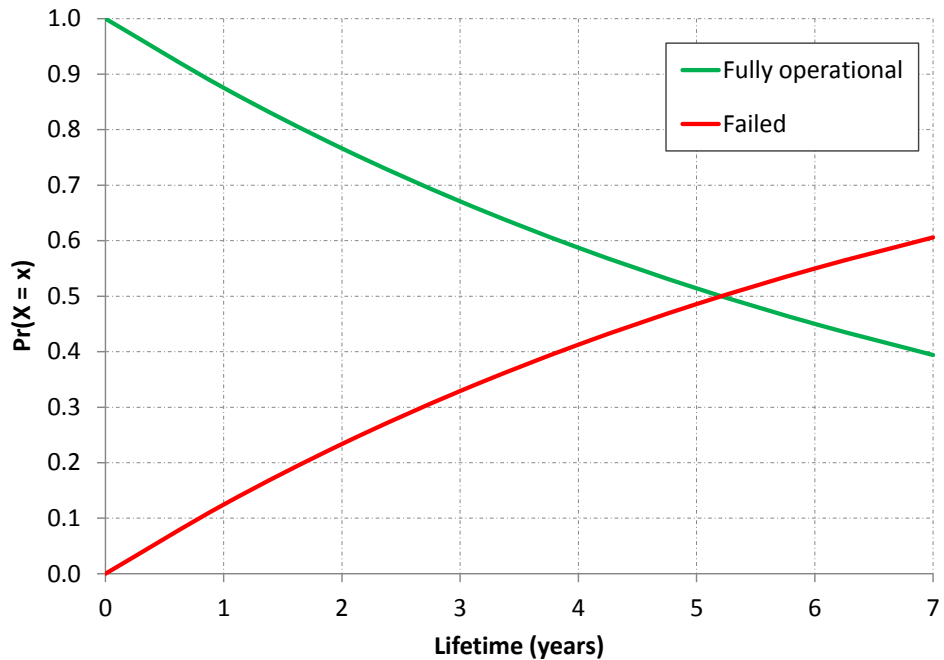

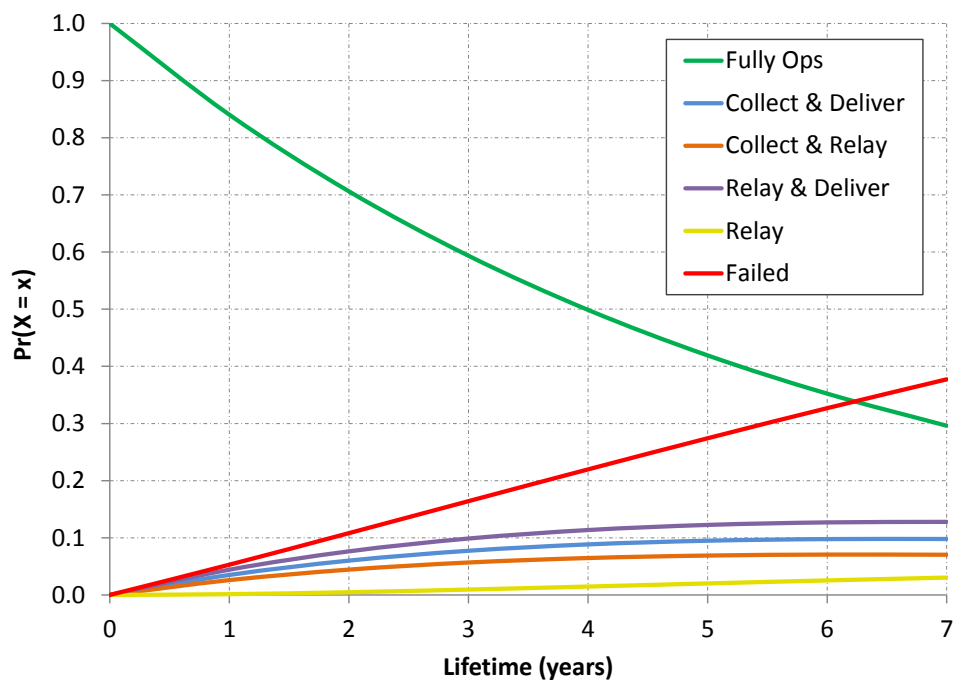*Figure 50 – State probability of a system without networking capability*

Figure 51 – State probability of a system with a networking capability

What Figure 50 and Figure 51 tell us is that, for this specific set of transition probabilities;

    i.   A network-capable system is equally likely to remain in either a fully operational state or a collect & deliver state, as a non-networked system is to remain in its fully operational state (which is, by definition, a collect & deliver state).

    ii.   A network-capable system is less likely to enter into a fully failed state, since there are intermediate, degraded states that are reachable. In this example, after 7 years, the networked system is ~23% less likely to be in a fully failed state, than a non-networked system.

While the failure likelihood of the above figures is perhaps exaggerated for illustrative effect, it is clear that there may be value in incorporating inter-satellite networking, with regards to lifetime utility, but it will come at a cost. Whether this additional financial and complexity burden is worth taking must be considered on a case-by-case basis and will be the focus of the later part of this chapter.

6.3   Effect of Failure on Mission Value

Determination of a system's expected state probability over its lifetime may be of interest from an academic perspective, but it provides limited insight into how the mission's value will be affected by failure. This problem can be tackled either semi-analytically or numerically, the choice of which is dependent on factors including network size, state complexity and available computational resource for the analysis. While the discussion so far in this chapter has surrounded failure effects on a single node and its various states, when considering a network, it is the state taken by each node combined to form a network-state that is perhaps of greater importance. A network-state can be described as a particular set of states in which each node that comprises the network resides, e.g. the fully operational network-state will constitute all nodes in the network being without defect, while other network-states may feature specific nodes with some level of failure present.

6.3.1   Network Types

Regarding the composition of a network in terms of node properties, it is of course a continuous spectrum. However, introduced here are three explicit families of network

that can be said to exist at the extremes of this scale, between which any degree of crossover is possible:

i. *Homogeneous* – a set of similar nodes that exhibit the same possible state transitions, with the same probability of transition. The influence on the network, i.e. the effect of a state transition in one node, is the same as that from any other node. E.g. a constellation of satellites of similar design, evenly distributed around the Earth.

ii. *Non-homogeneous* – as for a homogeneous network, but with each node exhibiting a different level of influence over the network. E.g. a constellation of satellites of similar design in an equatorial orbit, but with one in a polar orbit offering exclusive communication with, and observation of, high-latitude regions.

iii. *Heterogeneous* – a network of dissimilar nodes, exhibiting different state transition possibilities and having different levels of influence over the network.

6.3.2   Permutations vs. Combinations

In order to evaluate how network states evolve over their lifetime, we must understand the different states in which the network could reside at any particular time. In the case of non-homogeneous networks, the set of network states can be described using permutations (Section 6.3.2.1), while for homogeneous networks, the network states can be described by combinations (Section 6.3.2.2). Heterogeneous networks require evaluation of each set of individual node states in order to define the full set of network states, which cannot be simplified further.

6.3.2.1   Permutations

Consider a network of N nodes, where $\mathcal{N} = \{n_1, n_2, \dots, n_N\}$ is the set of nodes, to which the set of states $\mathcal{M} = \{m_1, m_2, \dots, m_M\}$ are attributed, such that a node can reside in any one of M states. A network-state S_i is defined as a unique set of node-state pairs for all nodes, where $S_i = \{s_1, s_2, \dots, s_N\}$, and $s_j \in \mathcal{M}$ is the state of node j. This can be defined formally as an N-permutation of the multi-set $\mathcal{M}$ [191], where the number of permutations, and thus the number of network-states, is equal to M^N.

A simple two-node, three state (a, b, c) network can be used to illustrate this concept:

Network State #	1	2	3	4	5	6	7	8	9
Node 1's state	a	b	c	a	b	c	a	b	c
Node 2's state	a	a	a	b	b	b	c	c	c

Table 24 – Network states for a 2-node, 3-state network with permutations

While it is useful to understand the utility achieved by the network in each network-state, for semi-analytical derivation of the expected utility over the lifetime, this can quickly become impractical as the number of nodes and states increase. E.g. the number of unique network-states for 10 nodes, each with 6 state possibilities, is over 60 million. To overcome this limitation, it may be possible to exploit combinations, instead of permutations.

6.3.2.2 Combinations

For networks that exhibit some degree of uniformity amongst the nodes, in terms of their influence on network performance[*], a fundamental simplification can often be made. This is to represent the network-states as a non-repeated, un-ordered set of node-states. For example, in a 3-node, 2-state (a, b) system, the network states $\{a, b, b\}$, $\{b, a, b\}$ and $\{b, b, a\}$ would all be considered equivalent under this simplification, since they each exhibit one node in state a and two in state b. Using the same notation as in the previous section, the number of network-state combinations, for a network made up of n nodes, each with m node-states, can be found through application of the combinatorial problem, multiset counting [191]. Formally, an N-combination of the multiset $\mathcal{M}$, as defined above, is

$$\binom{N + M - 1}{N} = \frac{(N + M - 1)!}{N!\,(M - 1)!},$$

6.3

where the left-hand side of equation 6.3 uses standard binomial coefficient notation. In other words, this can be defined as the number of N-combinations of M distinct objects, each with unlimited supply.

---

[*] For example, a homogenous Walker constellation of identical satellites, all carrying out similar functions, could be considered a uniform network, while one comprising the international space station (ISS), a sun-synchronous Earth observation platform and a geo-stationary communication satellite, each with a very different function, could not.

To illustrate the problem with using permutations in large networks, the number of permutations (repeated) and combinations (non-repeated), for a 6-state network of various node-counts, is shown in Figure 52.
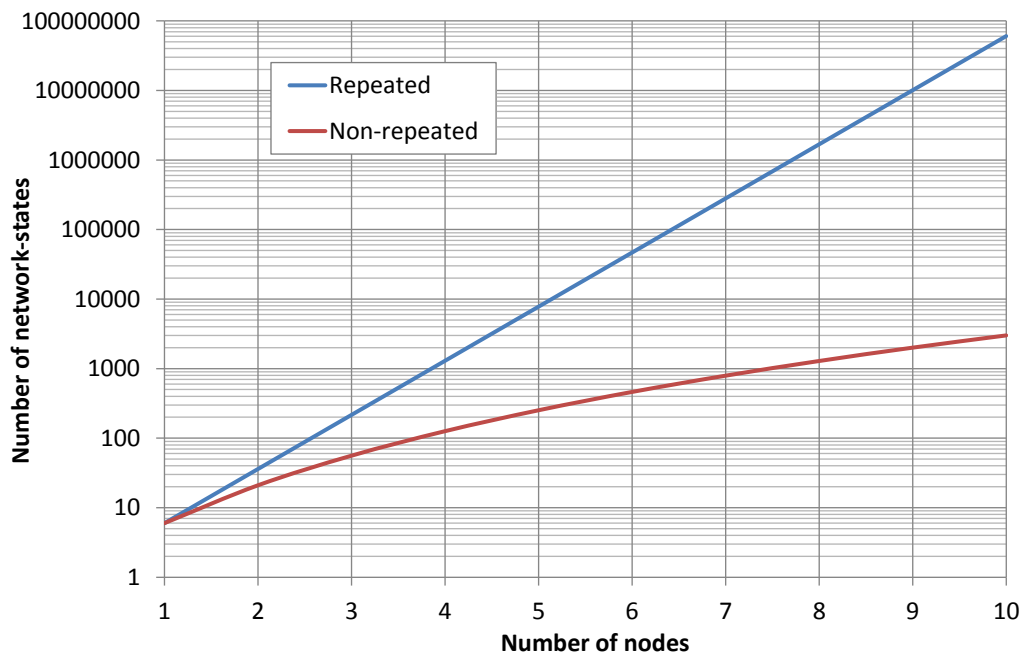

Figure 52 – Number of permutations (blue) and combinations (red) of network-states for a six state network

It is clear that for large networks, exploiting non-repeated network-states is not only beneficial, but necessary in the case of desktop computation facilities, for semi-analytical assessment of life-cycle utility. It must, however, be reiterated that this is a simplification, and should be used with caution on networks that are not strictly homogeneous in the sense of node-influence. In the event of a highly heterogeneous network, a numerical approach is recommended, whereby Monte-Carlo simulations are executed in order to derive an estimate for the expected utility at different times over the mission lifecycle.

6.4   Analysis Methodologies

6.4.1   Markov Chain Assessment

The lifetime value of inter-satellite networking can be found in a semi-analytical fashion by comparing, to an equivalent non-networked system, the utility associated with each network-state over the lifetime multiplied by the likelihood of that state

existing. This lifetime utility (U_L) can then be measured against lifetime cost (C_L), to find the lifetime value (V_L). Formally, this is

$$V_L = f(U_L, C_L), \qquad\qquad 6.4$$

where

$$U_L = \frac{1}{n}\sum_{k=1}^{n} U_k, \qquad\qquad 6.5$$

with U_k as the expected utility at time k, defined as

$$U_k = \sum_{j=1}^{m} P_{kj} U_j . \qquad\qquad 6.6$$

P_{kj} is the probability of being in state j at time k and U_j is the utility of the system in state j. Given a set of states in which the network can reside, it is necessary to identify the state-specific utilities *a priori*, using either permutations or combinations, as appropriate. For large state-spaces, this is perhaps not possible, such that a sample might instead be analysed, and the utility for those neglected from this analysis, estimated via interpolation.

Separately from the calculation of network-state utility, it is necessary to derive the probability of being in each network state, over the lifetime of the mission. This is achieved using Markov chains, as described in Sections 4.3.1 and 6.2, and in this work a time-discrete approach is taken to enable assessment of nonhomogeneous networks, where reliability/survivability may not be constant over time.

6.4.2   Monte Carlo Method

A semi-analytical assessment can provide an effective means of evaluating the survivability and expected value of a system over its life-cycle, however it can fail to offer both a solution for larger, more complex systems and insight into the variance around the expected lifetime value. Numerical assessment, in the form of Monte Carlo analysis, fills this void and can offer an attractive alternative. The objective is ultimately the same, to identify lifetime value of a system that is subject to failure, but achieves

this via a number of numerical simulations in which failure events occur according to a suitable probability distribution. In contrast to the semi-analytical method described above, it is not necessary, although is of course feasible, to compute/estimate the utility for each network-state, *a priori*. Instead, the utility could be calculated as necessary according to the failures that occur during the analysis. It would be deemed sensible to store the utility of network configurations for re-use later should they arise during a different simulation case.[*]

Each discrete-event simulation, many of which make up the Monte Carlo analysis, should incorporate random failures of sub-systems according to their probability. The result is a set of simulations that feature random failures that transition the network into degraded states over time. As mentioned previously, one of the benefits of a Monte Carlo approach, over the semi-analytical approach, is the availability of variance data of the lifetime utility around the expected value. It is for the analyst to assess the importance of such a metric, however the result can be revealing nonetheless.

6.5   Case Study: Opportunistic Satellite Network

An example mission is evaluated, which comprises a constellation of 7 satellites of similar design (such that combinations can be used to describe the network states) that collect data stochastically over their orbit and download it to a number of ground stations distributed around the globe. This type of mission could be considered representative of an Earth observation system, or a machine to machine (M2M) messaging service. Comparison is made between a traditional, non-networked version of the system, and one with ISN capability, to illustrate the potential lifetime value of ISN. First, a semi-analytical assessment is made, using Markov chains to estimate lifetime utility of the systems subject to failures, followed by a numerical Monte Carlo assessment, to verify the previous result and offer a greater depth of understanding.

---

[*] It is recognised that in the case of a Monte Carlo analysis being run in parallel on a large number of computational threads, it would perhaps be beneficial to evaluate the utility beforehand, in order to avoid repeated simulations being executed.

Failures are assumed to occur according to a Weibull distribution, with the parameters β^* and $\theta^\dagger$ outlined in Table 25, as defined in [190].

Sub-system	Weibull parameters	
	β	θ (years)
Payload	0.8874	7983
Communications/ISL	0.3939	400,982
Other support sub-system‡	0.668	2236

Table 25 – Weibull parameters for sub-system failure [190]

The reliability, i.e. the probability of being operational, of each sub-system over a 12 year lifetime, according to the parameters in Table 25, is shown in Figure 53. The problem of beginning of life failure on the communications, ISL and critical support systems is clear, whereas payloads tend to exhibit a more steady decay in reliability over the lifetime.
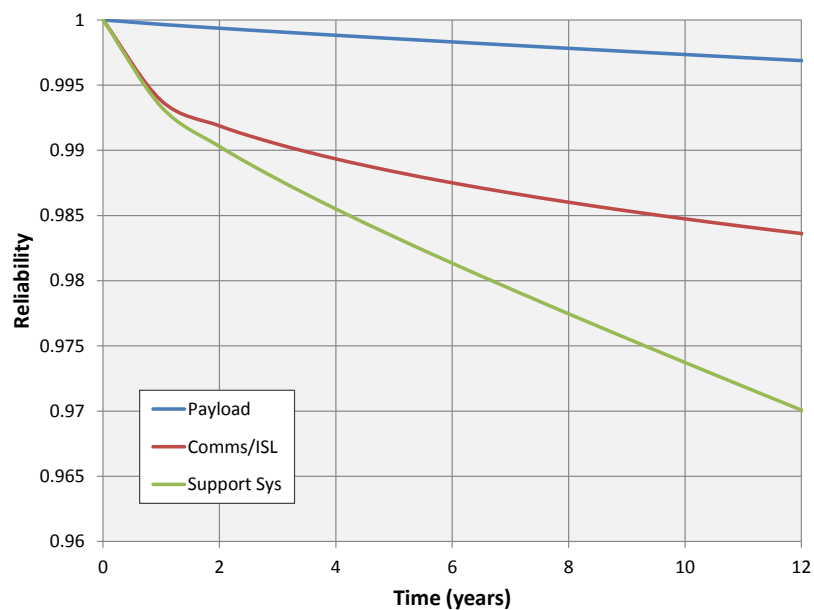
Figure 53 – Reliability of case study sub-systems

---

* β is the shape parameter, where a value of < 1 indicates a decreasing likelihood of failure over time, a characteristic reflecting higher failure probability at beginning of life.
† θ is the scale parameter, which dictates how quickly the reliability tends to zero once any infant mortality effects have been experienced.
‡ The Weibull parameters for the support system are calculated as the combined failure probability of the control processor, battery, electrical distribution and solar array sub-systems [190].

The satellite network being analysed here is one that is assumed to exploit opportunistic launch, something that has become increasingly popular over the past decade due to the relatively high launch cost savings compared with dedicated orbit placement. This approach is of course not appropriate for certain systems that demand specific orbital parameters, but it is assumed that for this mission, relative orbital position between each asset is not important. As per the analysis conducted in the previous chapter, it is considered possible to transfer data between satellites when their respective distance is below some threshold. Data routing shall be carried out using the *Spae* algorithm defined in Chapter 3, to ensure low data latency through the network. The simulation parameters are summarised in Table 18.

Variable	Units	Quantity	Comment
Altitude	km	[400, 800]	Random for each satellite in each simulation
Inclination (Delta)	deg.	[20, 100]	Random for each satellite in each simulation
Data time to live	min	120	Time from acquisition to expiry
ISL link range	km	5000	Threshold distance below which ISN can occur
ISL data rate	Mb/s	25	
Space-ground data rate	Mb/s	50	
Simulation duration	days	1	Operational scenario duration
Simulation time step	s	10	Numerical simulation parameter
No. simulations per state	#	30	For each network state possible

Table 26 – Design space

The constellation is considered to have access to a ground station network comprising assets in Kourou (Lat 5.3N, Lon 52.8W), Perth (Lat 31.8S, Lon 115.9E), Svalbard (Lat 78.2N, Lon 15.4E), South Point (Lat 19.0N, Lon 155.6W) and Hartebeesthoek (Lat 25.9S, Lon 27.7E). Visibility of the ground stations to each satellite is dependent on their respective orbital inclination.

6.5.1   Mission Utility

The success of a mission such as this would likely be judged on some combination of the amount of data provided, the amount of data lost and the latency associated with the delivered data. A multi-attribute utility, as described in Section 4.2.4.2, is used to measure the performance of the systems and directly compare their respective values.

The attributes considered important in this example are summarised in Table 27, along with their relative contribution to the MAU, for which a linear form is used.

Attribute	Objective	Limits	Weighting
Data latency (mean)	minimum	[20, 120] mins	2
Data latency (variance)	minimum	[10, 40] mins	1
Delivery ratio (mean)	maximum	[0, 1]	1
Data volume (mean)	maximum	[0, 7] Gb / day	2

Table 27 – Attributes considered in the multi-attribute utility calculation

Noteworthy is the data volume, which is an important consideration when failures are involved. If we consider the independent (non-networked) system in question, none of the other parameters (latency or delivery ratio) would be affected by a payload or communications system failure, since in both instances the culprit node is simply considered failed and thus removed from the network. The latency and delivery ratio of the other nodes would not change, since they were not relying on the failed node anyway.

6.5.2   Mission Cost

Cost for this mission is calculated using the Aerospace Corporation's Small Satellite Cost model, in the same way as described in Section 5.5.3. In the interest of brevity, the details are not repeated here. Indeed, there do exist some differences, which are applicable to the differences in the mission profile. Namely:

i.   Owing to the fact that this is not a federated satellite system and thus a single stakeholder would not be relying on other organisations for download tasks, the operations costs are considered lower, at 50 USD per ground station pass. This is further justified by the fact that a greater number of passes will be experienced (seven satellites as opposed to one) and thus a discount would likely be negotiated.

ii. The space and launch segment costs are considered for seven platforms instead of one.

6.5.3   Markov Chain Assessment

The results presented in this section are derived using the semi-analytical approach outlined in Section 6.4.1. Here, the utility for both the networked and independent systems, for each of its network states, is derived through numerical simulation. At discrete intervals over the lifetime (1 year in this case), the expected utility is calculated using equation 6.6, for each of the systems as shown in Figure 54.
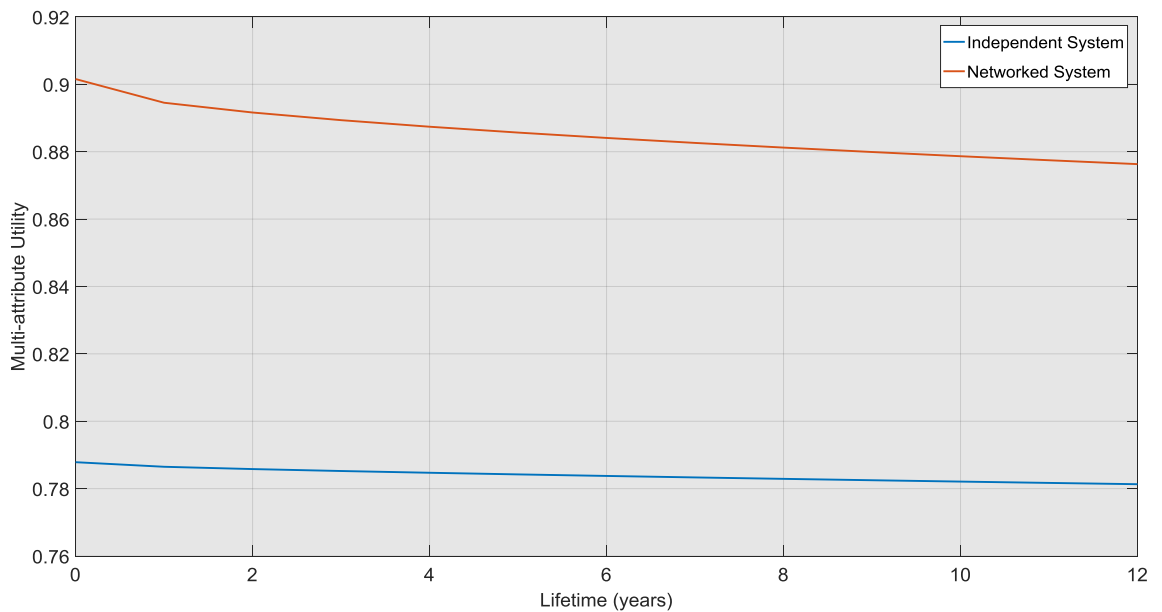
Figure 54 – Multi-attribute utility (MAU) of the independent and networked systems over the lifetime

Perhaps surprisingly, the reduction in utility over the lifetime is greater in the case of the networked system (from 0.902 at the beginning of life, to 0.876 at end of life), than for the non-networked system (form 0.788 at beginning of life to 0.781 at end of life), despite the increased likelihood of each node remaining in a non-failed state due to the additional relay functionality. This can be explained by a number of factors:

i. According to the reliabilities of the payload, communications system and ISL system (Table 25), it is clear that the likelihood of payload failure is significantly lower than for the communications and ISL systems. This means that in the networked case, it is more likely that nodes will be relied upon to download other's data (whose communication system has failed), than for a reduced amount of data to be collected (in the case of a payload failure). Consequently, the networked system suffers a reduction in delivery ratio and an increase in data latency as failures occur. In the independent system,

however, neither of these attributes are affected by a communications system or payload failure, since in both cases the offending node is simply removed from the network completely (fully failed).

ii.   The most significant contribution to the drop in utility for the independent system is a reduction in data volume, which can ultimately be mapped linearly with the expected number of failed platforms. In the networked case, it happens to be that the reduction in data volume is even greater (Figure 55), despite the fact that the additional relay functionality exists. The reason for this is due to the fact that ISN results in a higher delivery ratio (more of the collected data is being delivered in the first place) and thus the contacts with the ground are significantly closer to capacity. Thus, any failure in communications systems throughout the network results in more strain being put on this parameter, resulting in a reduced delivery ratio and thus a corresponding drop in data volume. What is clear, however, is that despite the greater reduction in delivered data, the volume remains significantly higher throughout in the ISN case and would continue to do so regardless of lifetime.
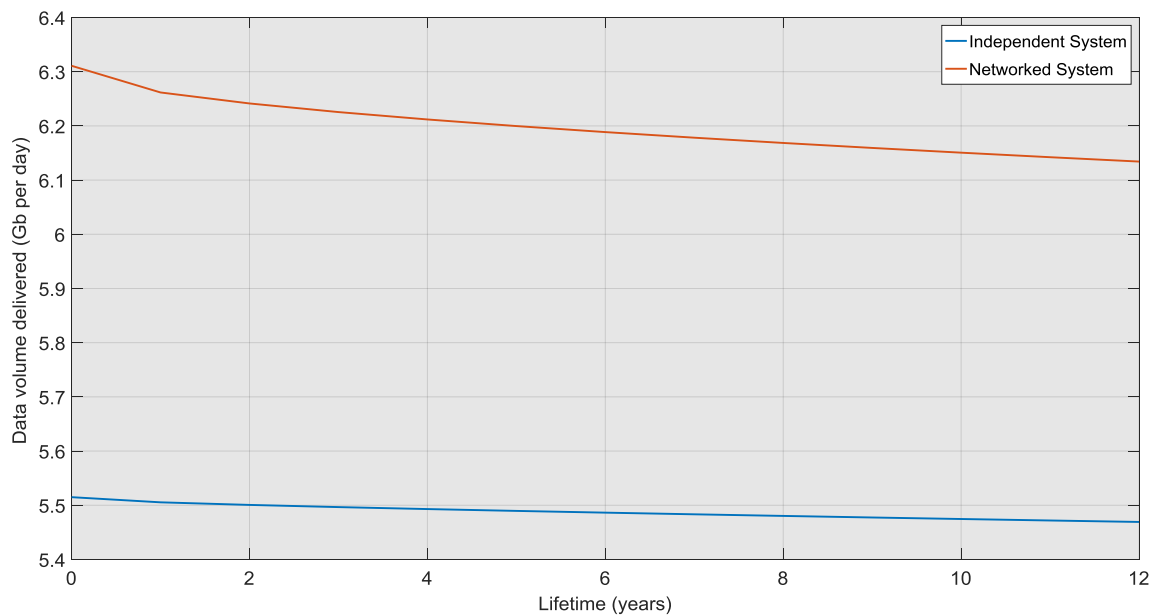
Figure 55 – Expected volume of delivered data over mission lifetime

Finally, the fully operational value and expected lifetime value (average over the entire 12 year lifetime) are shown in Table 28, as defined in equations 6.4 and 6.5.

System	Nominal (fully operational) value (U per bnUSD)	Lifetime value (U per bnUSD)
Independent	3.217	3.201
Networked	3.541	3.478

Table 28 – Nominal and lifetime value

The expected value of each system, plotted over the lifetime, is shown in Figure 56.
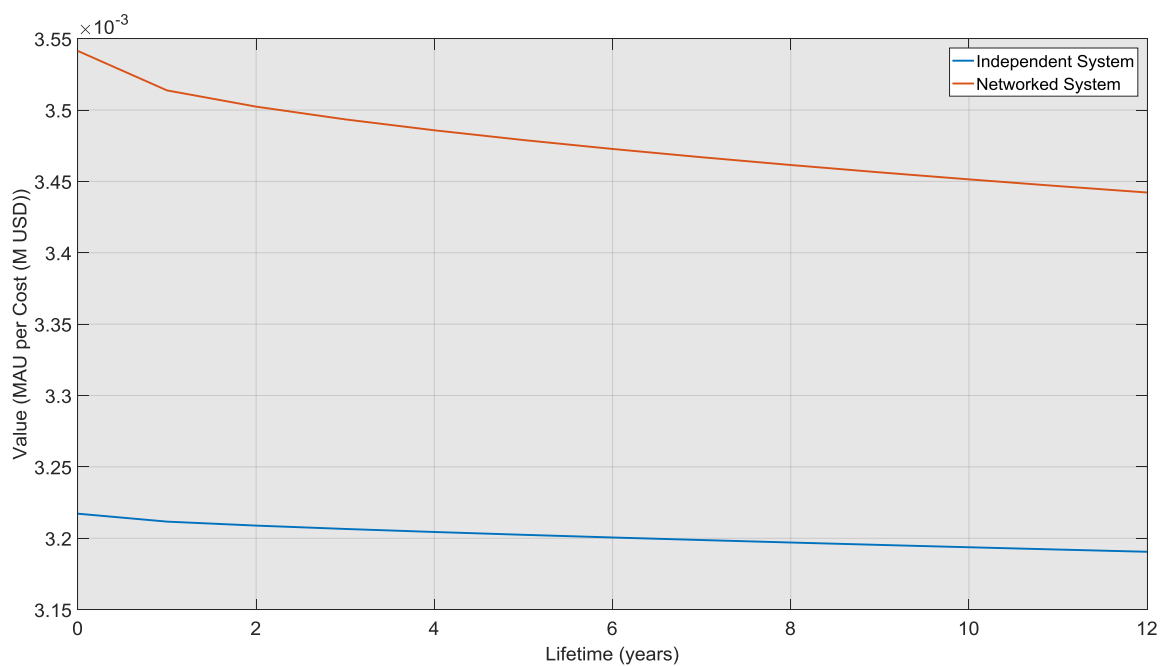
Figure 56 – Lifetime value

What can be extracted from this, and the above information, is that while ISN offers an in increased probability of remaining in a fully- or partially-operational mode, it may not in turn always offer a reduced magnitude loss in value over the lifetime. More importantly however, rather than the loss in value, it is likely that a networked system will continue to offer higher value over the lifetime of a mission that is subject to failure. One could argue that while more value may be lost when considering uncertainty, there was more value to begin with.

If we are to assume that a preference value (Section 5.2.1) is appropriate (i.e. the value of the non-networked system can be mapped to a constant value for any system cost), then based on the lifetime value (Table 28) it is reasonable to suggest that the price a stakeholder should be willing to pay for the implementation of ISN for this mission, is 21.99M USD. Without considering uncertainty at all, as per the analysis in Chapter 5, the ISN value would be 25.65M USD, which illustrates the risk associated with neglecting this important phenomenon.

6.5.4   Validation Using Monte Carlo Simulation

In order to validate the Markov chain assessment and provide insight into the variance of the lifetime value parameter, a Monte Carlo analysis is executed. Here, 100,000 simulations are conducted, for each of the networked and independent systems, in which transitions between network states are experienced over the lifetime according to probability of sub-system failures as defined previously.

The general process that was executed during the Monte Carlo analysis is illustrated by the algorithm in Table 3, where Y is the number of MC simulations, T is the number of time intervals (each equal to 1 year), N is the number of nodes in the network, $\mathcal{S} = \{s_0, s_1, \dots, s_{|\mathcal{S}-1|}, s_{|\mathcal{S}|}\}$ is the set of possible network states, $\mathcal{M} = \{m_0, m_1, \dots, m_{|\mathcal{M}|-1}, m_{|\mathcal{M}|}\}$ is the set of node states, $P = \mathbb{R}^{|\mathcal{M}| \times |\mathcal{M}| \times T}$ is the node state transition matrix with entry P_{ijk} being the probability of a node in state i transitioning to state j at time k, S_{ij} is the state in which the network resides at time i and simulation j, with the initial state being fully operational in both the independent and networked system examples and m_{jk} is the state in which node j resides in at time k, which is dictated by S.

Monte Carlo analysis
Input: $Y, T, N, \mathcal{S}, \mathcal{M}, P$ and $S_{1,all} = 1$
Output: S
$\quad$ **for** $i = 1$ to Y **do**
$\quad\quad$ **for** $j = 1$ to T **do**
$\quad\quad\quad$ **for** $k = 1$ to N **do**
$\quad\quad\quad\quad X = [0,1] \quad \lvert Prob(X = x) = Prob(X = y)\ \forall x, y \in [0,1]$ *
$\quad\quad\quad\quad y = 0$
$\quad\quad\quad\quad$ **for** $a = 1$ to $\lvert \mathcal{M} \rvert$ **do**
$\quad\quad\quad\quad\quad$ **if** $X < y + P_{m_{kj}aj}$
$\quad\quad\quad\quad\quad\quad z_{kj+1} = a,$ **break**
$\quad\quad\quad\quad\quad$ **else** $y = y + P_{z_{kj}aj}$
$\quad\quad\quad S_{j+1i} = s_b,$ where $z_{kj+1} \in s_b, \forall\, k$

Table 29 – Algorithm for Monte Carlo analysis

The output from this analysis, the array N, defines the state of the network at the start of each time interval, for each of the Monte Carlo simulations. The multi-attribute utilities associated with these states can then be applied to identify the utility over the lifetime, for each simulation.

---

* In other words, X is a uniformly distributed continuous random variable between 0 and 1

144

The mean utility, with the 95th percentile illustrated by the error bar, is shown in Figure 57, which closely matches the results in Figure 54 and thus provides validation.
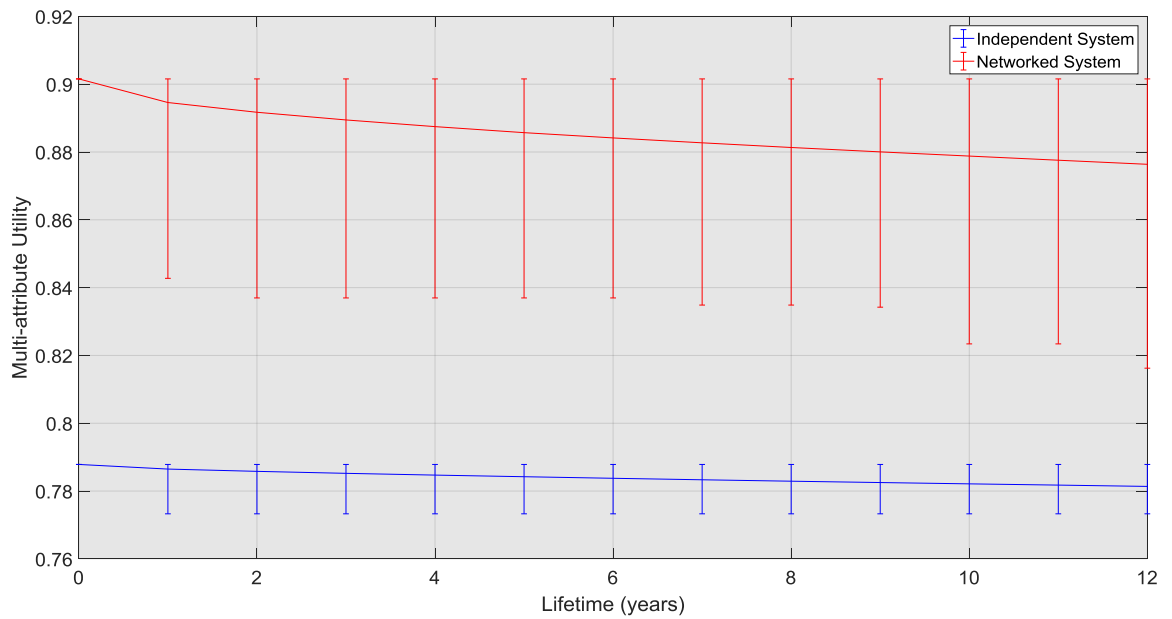
Figure 57 – Multi-attribute utility vs. lifetime, with 95th percentile error bar, from Monte Carlo

It is apparent that the range of MAU exhibited by the networked system is significantly greater than that of the independent system, which is further evidence of the network operating at closer to its full potential in the former case. Furthermore, while a larger range exists the utility remains greater throughout showing that the networked system will continue to achieve a higher utility over the entire lifetime.

6.6 Conclusions and Discussion

In this chapter, a methodology for evaluating lifetime value, that is value of a system that is subject to uncertainty in the form of on-board failures, has been described and applied. First it was shown how a space system capable of inter-satellite networking, which as a consequence exhibits the additional capability of data relay over its non-network-capable counterpart, is more likely to remain in a functional state, but less likely to remain in a fully functional state. The former is a result of the additional partially failed states of collect & relay, deliver & relay and relay available to the system, while the latter is due to the existence of an additional sub-system that may fail during the lifetime.

It was shown how multi-node networks and their network-states can be defined using either combinations or permutations depending on the relative influence of each node. The states of a network comprising only nodes with a similar design and similar operational characteristics could be represented by combinations, such that the states are an ordered set without repetitions. This is of particular importance for large networks, or those with a large number of node-states, since the number of network-states grows exponentially when described by permutations.

The increase in resilience, offered by an inter-satellite networking capability, suggests that such a system would offer an increased retention of utility, and thus value, over the lifetime of a mission compared to an equivalent independent system. While this may be true in some circumstances, the analysis carried out in this chapter shows that this is not necessarily the case, in particular when the networked system is operating closer to its maximum limitations than its independent counterpart. A case study, in which a network of (seven) Earth observation satellites aiming to collect and deliver as much data as possible with minimum delay was analysed for both a network-capable and independent scenario subject to sub-system failure. While the networked system consistently offered higher value over the lifetime of the 12 year mission, the reduction in value was greater, which can be attributed to both the way in which the particular performance attributes were affected by system degradation, and the fact that the networked system was operating at closer to its capacity and thus had more value to lose. Furthermore, the variance of utility over the lifetime was seen to be greater in the networked case, which is evidence of the greater potential drop in utility due to failure.

Chapter 7

Concluding Remarks

Throughout this dissertation, the objective has not been to justify the use of inter-satellite networking, but to explore its effect on space missions and offer approaches that are suitable for its evaluation. As such this work has aimed to provide a formalised approach to answering the question; "is the addition of an inter-satellite networking capability of value to a mission?" The following sections provide a summary of the work carried out, a recap of the knowledge contributions, conclusions from the analysis carried out, and recommendations for future work.

7.1   Summary of the Research

It was first identified that almost all space missions can be reduced in definition to systems that carry out data collection and dissemination, such that characteristics relating to routing of information often constitute important performance attributes. In response to this, an effective method of routing data through intermittently-connected delay-tolerant networks was introduced in Chapter 3, which overcomes limitations found in the currently available approaches. The new protocol, called Spae, offers a resource-considerate solution that is indifferent to data type, network topology, network scale, mobility patterns and resource type. Spae relies on the need for a deterministic (or at least predictable) mobility pattern due to its reliance on a modified version of Dijkstra's shortest path algorithm. Here, Spae was applied to a random mobility network and a federated satellite system network, and compared against other routing methods including a packet-optimal oracle-based strategy. The performance of these strategies, including data latency, hop count and delivery ratio (the volume of data that was delivered compared to the amount that entered the network), was evaluated for networks with various data routing characteristics such as link range, data rate and data time to live.

In Chapter 5, mission value propositions were introduced that offer a formalised way of identifying the value of inter-satellite networking for space missions. These were used, in combination with multi-attribute utility theory, to assess the value of various design architectures in a nano-satellite constellation and a federated satellite system. Simulations were developed that explored the mission performance and compared against mission cost. For each case study, the price one should be willing to pay for an inter-satellite networking capability was derived as a function of value, utility and cost and specific design architectures were identified as those offering greatest value potential.

Finally, in Chapter 6, uncertainties in the form of sub-system failure were introduced, having the effect of reducing utility over the lifetime of a system. It was shown that an inter-satellite networking capability decreases the likelihood of entering a fully-failed state, due to the existence of three additional, partially-failed states that are not otherwise available, such as in a non-network-capable system. Lifetime utility, and consequently lifetime value was derived, for a small satellite constellation, using a semi-analytical approach that exploited nonhomogeneous discrete-time Markov chains to find the probability of residing in different network states over the lifetime. Validation of these results was achieved using Monte Carlo methods, which also offered a deeper understanding through variance around the expected mission utility and value.

7.2   Contributions and Conclusions

Within the field of delay- and disruption-tolerant networking, a gap was identified in the sense that a generalised approach to data routing with consideration of downstream resource limitations was missing from the literature. A solution to this has been developed in the form of a resource-considerate data routing protocol, *Spae*, which exploits the deterministic mobility of satellite networks in order to route data between intermittently connected nodes. Analysis carried out in Chapter 3 shows that *Spae* offers a noticeable improvement over other, traditional routing schemes, and closely matches performance of a packet-optimal approach. This optimal method, while acting as a useful upper bound for comparative purposes, is considered unachievable in reality due to the level of knowledge required. While Spae consistently tracks the performance of a packet-optimal approach in terms of delay, it tends to suffer in terms of hop-count as the network becomes congested. This is due to an increased number of

intended journeys along which packets are sent, but fail to traverse, an effect that could be eased through introduction of a more sophisticated downstream resource prediction mechanism. The drawback of this, of course, is greater computation and complexity in selecting the journey. What can be certain is that consideration of resources improves the performance of data routing, and as networks become larger and more congested, this will become increasingly important. Ultimately, this follows the concept that with more information being available to the decision maker, a better decision can be made.

The use of inter-satellite networking (ISN) has the potential to add value to a mission, by offering an improved data delivery mechanism. Fundamentally, this may include an increase in delivery capacity, which may be a limiting factor in an equivalent mission without ISN. A formal definition of this condition was introduced in Chapter 6, represented as the solution to maximal flow problem with multiple sources. Also in this chapter, a methodology for quantifying the value of ISN is introduced. The approach proposes the use of preferential or maximum-price value in order to directly compare different design architectures, where value is a function of utility (derived from one or more attributes) and cost. Importantly, the value is formulated in such a way that does not require consideration of the development and manufacturing cost for the ISN technology, as this is something that can be difficult to justify at the conceptual phase of design. This problem can then be tackled at a later stage should an ISN-capable design be taken forward. The cost of ISN implementation, including the effects of additional mass on the system and changes in operational behaviour, is incorporated, which enables a comparison of designs to be made with confidence. The result is a value parameter that represents the maximum price one would be willing to pay for the ISN technology, in order to achieve a mission of greater value than a non-networked equivalent. It has been shown that this approach offers a convenient way of comparing the value offered by different designs, with ISN generally resulting in greater value in cases where the timely delivery of data is considered important. Indeed, ISN does not always offer greater value, e.g. where network connectivity is particularly low, or where significant data delivery through large ground station networks is possible, and it is evident that identifying the design points for which increased value does exist is not necessarily trivial.

Definitions of the dominant operating states for ISN-capable and ISN-incapable platforms have been, for the first time, formally defined in Chapter 6. In the non-ISN

case, this includes a system that is able to collect (via the payload) and deliver (via the communications system) information while operational, but offers nothing to the mission following failure of the payload, communications system or critical supporting sub-system. For an ISN-capable system on the other hand, the fully operational state also includes information relay (from the ISN sub-system). Four partially failed states exist as a result of the addition of ISN; collect & deliver, collect & relay, relay & deliver, and relay-only, resulting from the different combinations of sub-system failures. Transition between states was represented in matrix form and shown to decrease the likelihood of reaching a fully failed state during a system's lifetime. Also in this chapter, analysis of the effects of sub-system failure on space mission value over the lifetime was investigated semi-analytically for the first time, considering time-variant failure probability. This was achieved through the application of nonhomogeneous discrete-time Markov chains (DTMCs), with a failure probability represented using Weibull distribution to capture infant mortality effects. It was shown that while ISN generally offers a greater utility over the life of a mission due to the additional relay functionality, it does not necessarily result in a system that is more robust to failure in terms of value consistency. Indeed the value of a networked system was seen to reduce by a greater amount than its independent counterpart, which can be attributed to the greater amount of value that the networked system has to lose. Regardless of this fact, a networked system that offers greater utility at the start of a mission will continue to do so for the duration, but the additional lifetime value offered by ISN may be less when failures are considered.

7.3 Future Work

While every effort has been made to offer completeness within this dissertation, it is inevitable that some of the concepts might benefit from more attention. Recommendations for future work are provided in the following, based on the 3 main contributing themes.

7.3.1 Data Routing

As seen in the analyses in Chapter 3, there were cases where Spae failed to track the performance of a full-knowledge routing approach because of its inability to accurately predict success of downstream journeys when the network became congested. A potential solution to this would be better estimates of downstream resource

availability. This could be achieved using moving point averages or traffic-based estimations that consider the expected amounts of data to be travelling along each journey, at the expense of additional computational burden.

The implementation of Spae presented here lacks the ability to deal with a sending node out-degree of >1, i.e. if more than one node is available for data transfer, the link with highest expected value only is considered. Extending the capability to consider higher out-degree as part of the routing algorithm would increase the generality of the method.

As with the full knowledge approach against which Spae has been compared, journeys are selected on a packet-by-packet basis, based on the value associated with it reaching the destination. While this could be considered packet-optimal, other routes may offer higher overall network performance. Consider the case where minimising average latency across all packet instances was the objective, there will be an optimal way of achieving this where some packets may be deliberately sent over longer journeys in order to free up space for others. This problem has been tackled using linear programming techniques, which have been shown to become computationally intractable for even modest scenario complexity, but it is recommended that heuristic approaches are investigated as an alternative.

While the definition of Spae is such that resources are generic and could include data-rate, data storage capacity and on-board energy, only data-rate was incorporated in this work, for illustrative purposes. Imposing limitations on other resources would offer greater levels of completeness, and illustrate the full scope of the protocol.

As with all shortest path-centric algorithms, a large number of potential journeys results in a large number of searches being required to find those with the highest value. For Spae, the problem is amplified in the case of large packet count, in particular when journeys routinely become full during a routing exercise, since new journeys must then be searched more often. It is proposed therefore to introduce a check at the beginning of each contact event that identifies the level of change in the network condition, which may offer heuristic insights into how the journey scope can be reduced.

Application of Spae to delay-critical missions, such as those for high-value financial transactions and time-sensitive machine-to-machine messaging, is recommended.

Furthermore, application to scenarios that have demands on the upload of data from the ground to the space assets should be considered. There are no limitations on Spae being applied in this way, and the potential benefits from disseminating command and control data through the network using Spae could be significant.

7.3.2   Value Analysis

A more robust definition of mission value could be employed in future work to include stake-holder preference as a function of cost. One tends to be risk-seeking in a low cost domain, but risk-averse in a high cost domain, such that a small probability to gain lots, while spending a little is generally preferred over a small risk of losing lots when spending a lot, even when the theoretical value is the same. This is partially achieved through the implementation of a maximum price value, however this assumes knowledge of the maximum-price constant value function, which could instead be defined as a function of the stakeholder's willingness to accept risk.

At the system-level, it is proposed that the effects of other parameters that may affect the inter-satellite link being available are incorporated into the model. This might include, but is not limited to, Doppler shift, relative attitude of communicating platforms (in the case of directional communications) and slew manoeuvres (if deemed necessary). The latter would have implications on simulation complexity, requiring higher fidelity models that not only consider satellite attitude, but the operational logic associated with action-based decision making, e.g. should the satellite remain in its current attitude in order to effectively capture data, or track a neighbouring satellite to transfer high priority data?

While not considered necessary in order to illustrate the methodologies posed in this dissertation, it would nonetheless be interesting to incorporate communication sub-system models for optical/laser communication, phased array antennas, electronic beam-steering and software-defined radio-based designs.

In the case of the federated satellite system, it is assumed that all nodes in the network partake in data transfer for the good of the network, albeit with some unavailability imposed according to a probability distribution. In reality, should a FSS be deployed, transactions between nodes would likely require a negotiation phase in order to establish a price for data relay based on demands on the relay platform's resources, risk and downstream journey attributes. Indeed, the problem is further complicated

when considering what happens beyond the first hop, when the originator's data may need to be re-routed by other federates in the case of the preferred journey being unavailable.

7.3.3 Reliability

While Markov chains offer an effective method of defining system states and transitions between those states, they suffer from the curse of dimensionality, e.g. a system with m sub-systems, where each can reside in n sub-system states, would require n^m nodes in the Markov chain. Stochastic petri-nets offer a more efficient approach to capturing this state transition environment, should a higher fidelity system model be required than used here. Noteworthy is that the mathematics do not become less burdensome, but the graphical and matrix representation of the network would be simplified.

The analysis conducted in Chapter 6 does not consider options for repair or replacement of failed systems, but this is a real consideration for many missions. The options available to a designer in terms of replacement strategies and on-orbit servicing schedules are vast, almost limitless, but it is recommended that further investigation is carried out to identify optimal strategies for certain mission types. Replacement/repair cost should be considered, such that a true indication of lifetime value is captured.

The simulations executed in this work neglect orbital decay, such that results are idealised to some extent. While this might be considered acceptable in some cases, e.g. high altitudes or platforms with orbit control capabilities, it is often the case that orbit decay will have a detrimental effect on the mission and further justifies the above comment on needing a replacement strategy in place.

7.4 Closing Statement

This work provides insight and guidance for someone investigating the impact of inter-satellite networking, be it from a technical or programmatic view-point. From the introduction of a novel data routing algorithm, to new methods of value proposition and uncertainty consideration, a number of findings are presented that will help one to appreciate the non-trivial effects brought about by inter-connectivity within a network. As we move toward a more connected space environment, be that in the form of federated satellite systems, or large, mission-focussed constellations, these methodologies will grow in importance and provide a platform upon which our understanding can be built.

"To struggle and to understand - never this last without the other; such is the law"

– George Mallory

References

[1]   A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols and Applications," *IEEE Commun. Surv. Tutorials*, no. 99, pp. 1–1, 2015.

[2]   H. Ward, "A Global Internet: The Next Four Billion Users," *New Sp.*, vol. 3, no. 3, pp. 204–207, 2015.

[3]   G. Shaw, "The generalized information network analysis methodology for distributed satellite systems," 1998.

[4]   W. Stewart, "Section II - Markov Chains," in *Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling*, 1st ed., Princeton University Press, 2009, pp. 191–375.

[5]   C. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed. Springer, 2004.

[6]   P. Baran, "On Distributed Communications (I - XI)," 1964.

[7]   L. Kleinrock, "Information flow in large communication nets," *RLE Quarterly Progress Report*. 1961.

[8]   G. Prescott, S. Smith, and K. Moe, "Real-time information system technology challenges for NASA's earth science enterprise," *Proc. 20th IEEE Real-Time …*, 1999.

[9]   V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, E. Travis, and H. Weiss, "Interplanetary Internet (IPN): Architectural Definition," Internet Engineering Task Force (IETF), 2001.

[10]   V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-Tolerant Network Architecture: The Evolving Interplanetary Internet," 2002.

[11]   K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications - SIGCOMM '03*, 2003, pp. 27–34.

[12] A. Ferreira, "On models and algorithms for dynamic communication networks: The case for evolving graphs," *In Proc. ALGOTEL*, 2002.

[13] A. Ferreira and L. Viennot, "A Note on Models, Algorithms, and Data Structures for Dynamic Communication Networks. [Research Report] RR-4403," 2002.

[14] S. Bhadra and A. Ferreira, "Computing multicast trees in dynamic networks using evolving graphs," 2002.

[15] B. Xuan, A. Ferreira, and A. Jarry, "Computing Shortest, Fastest, and Foremost Journeys in Dynamic Networks," *Int. J. Found. Comput. Sci.*, vol. 14, no. 2, 2002.

[16] S. Bhadra and A. Ferreira, "Complexity of connected components in evolving graphs and the computation of multicast trees in dynamic networks," in *Ad-Hoc, Mobile, and Wireless Networks*, 2003, pp. 259–270.

[17] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro, "Time-Varying Graphs and Dynamic Networks," in *Ad-hoc, Mobile, and Wireless Networks*, Springer Berlin Heidelberg, 2011, pp. 346–359.

[18] Various, "Overview of Space Communications Protocols," 2014.

[19] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-tolerant networking architecture," 2007.

[20] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges," *Commun. Surv. Tutorials, IEEE*, vol. 8, no. 1, pp. 24–37, 2006.

[21] I. Cardei, C. Liu, J. Wu, and B. Raton, "Routing in Wireless Networks With Intermittent Connectivity," 2007.

[22] A. Silva, S. Burleigh, C. M. Hirata, and K. Obraczka, "A survey on congestion control for delay and disruption tolerant networks," *Ad Hoc Networks*, Aug. 2014.

[23] A. Voyiatzis, "A Survey of Delay-and Disruption-Tolerant Networking Applications," *J. Internet Eng.*, vol. 5, no. 1, pp. 331–344, 2012.

[24] I. Psaras, L. Wood, and R. Tafazolli, "Delay-/disruption-tolerant networking:

State of the art and future challenges," 2009.

[25] V. Borrel, M. Ammar, and E. Zegura, "Understanding the wireless and mobile network space: a routing-centered classification," *... Second ACM Work. ...*, 2007.

[26] R. Ramanathan, P. Basu, and R. Krishnan, "Towards a formalism for routing in challenged networks," *... Second ACM Work. ...*, 2007.

[27] S. Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant Network," in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications - SIGCOMM '04*, 2004, pp. 145–158.

[28] H. Dubois-Ferriere, "Space-time routing in ad hoc networks," in *Ad-Hoc, Mobile, and Wireless Networks*, no. 5005, 2003, pp. 1–12.

[29] A. Farago and V. Syrotiuk, "MERIT : A Unified Framework for Routing Protocol Assessment in Mobile Ad Hoc Networks," in *ACM SIGMOBILE*, 2001, pp. 53–60.

[30] A. Ferreira, "Building a reference combinatorial model for MANETs," *Network, IEEE*, no. October, pp. 24–29, 2004.

[31] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," *Tech. Rep. number CS-200006, Duke Univ.*, no. CS-200006, pp. 1–14, 2000.

[32] G. Neglia and X. Zhang, "Optimal delay-power tradeoff in sparse delay tolerant networks: a preliminary study," in *Proceedings of the 2006 SIGCOMM workshop on ...*, 2006, pp. 237–244.

[33] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance modeling of epidemic routing," *Comput. Networks*, vol. 51, no. 10, pp. 2867–2891, Jul. 2007.

[34] Y. Wang and H. Wu, "Delay/Fault-Tolerant Mobile Sensor Network (DFT-MSN): A New Paradigm for Pervasive Information Gathering," *IEEE Trans. Mob. Comput.*, vol. 6, no. 9, pp. 1021–1034, Sep. 2007.

[35] T. Small and Z. J. Haas, "The shared wireless infostation model - A New Ad Hoc Networking Paradigm (or Where there is a Whale, there is a Way)," *Proc. 4th ACM Int. Symp. Mob. ad hoc Netw. Comput. - MobiHoc '03*, pp. 233–244, 2003.

[36] Y. Li, Y. Jiang, D. Jin, and L. Su, "Energy-efficient optimal opportunistic forwarding for delay-tolerant networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4500–4512, 2010.

[37] T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking - WDTN '05*, 2005, pp. 260–267.

[38] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *SIGCOMM '05 Workshops*, 2005.

[39] T. Matsuda and T. Takine, "(p, q)-Epidemic routing for sparsely populated mobile ad hoc networks," *Sel. Areas Commun. IEEE ...*, vol. 26, no. 5, pp. 783–793, 2008.

[40] K. Harras and K. Almeroth, "Controlled flooding in disconnected sparse mobile networks," *Wirel. Commun. Mob. Comput.*, no. July 2008, pp. 21–33, 2009.

[41] T. Abdelkader, K. Naik, A. Nayak, N. Goel, and V. Srivastava, "SGBR: A Routing Protocol for Delay Tolerant Networks Using Social Grouping," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 12, pp. 2472–2481, Dec. 2013.

[42] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic Routing in Intermittently Connected Networks," in *Proceedings of The First International Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR 2004)*, 2004.

[43] C.-S. Lin, W.-S. Chang, L.-J. Chen, and C.-F. Chou, "Performance Study of Routing Schemes in Delay Tolerant Networks," in *22nd International Conference on Advanced Information Networking and Applications - Workshops (aina workshops 2008)*, 2008, pp. 1702–1707.

[44] B. Burns, O. Brock, and B. N. Levine, "MV routing and capacity building in disruption tolerant networks," in *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, 2005, vol. 1, pp. 398–408.

[45] J. a. Davis, a. H. Fagg, and B. N. Levine, "Wearable computers as packet transport mechanisms inhighly-partitioned ad-hoc networks," *Proc. Fifth Int. Symp.*

Wearable Comput., 2001.

[46] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks.," in *INFOCOM*, 2006.

[47] D. Patterson, G. Gibson, and R. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in *Proceedings of the 1988 ACM SIGMOD international conference on Management of data*, 1988, pp. 109–116.

[48] H. Weatherspoon and J. Kubiatowicz, "Erasure coding vs. replication: A quantitative comparison," *Peer-to-Peer Syst.*, pp. 1–6, 2002.

[49] Y. Wang, S. Jain, M. Martonosi, and K. Fall, "Erasure-coding based routing for opportunistic networks," in *Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking - WDTN '05*, 2005, pp. 229–236.

[50] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," in *ACM SIGCOMM*, 2007.

[51] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "Replication Routing in DTNs: A Resource Allocation Approach," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 596–609, Apr. 2010.

[52] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: The single-copy case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 63–76, 2008.

[53] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 77–90, 2008.

[54] T. Spyropoulos, T. Turletti, and K. Obraczka, "Routing in Delay-Tolerant Networks Comprising Heterogeneous Node Populations," *IEEE Trans. Mob. Comput.*, vol. 8, no. 8, pp. 1132–1147, Aug. 2009.

[55] E. P. C. Jones, L. Li, J. K. Schmidtke, and P. a. S. Ward, "Practical Routing in Delay-Tolerant Networks," *IEEE Trans. Mob. Comput.*, vol. 6, no. 8, pp. 943–959, Aug. 2007.

[56] I. Cardei, C. Liu, J. Wu, and Q. Yuan, "DTN Routing with Probabilistic Trajectory,"

in *Wireless Algorithms, Systems and Applications*, Springer Berlin Heidelberg, 2008, pp. 40–51.

[57] C. Liu and J. Wu, "An optimal probabilistic forwarding protocol in delay tolerant networks," *Proc. tenth ACM Int. Symp. ...*, pp. 105–114, 2009.

[58] C. Liu and J. Wu, "Scalable Routing in Cyclic Mobile Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 9, pp. 1325–1338, Sep. 2009.

[59] C. Liu and J. Wu, "Routing in a cyclic mobispace," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '08*, 2008, p. 351.

[60] C. Liu and J. Wu, "Practical Routing in a Cyclic MobiSpace," *IEEE/ACM Trans. Netw.*, vol. 19, no. 2, pp. 369–382, Apr. 2011.

[61] C. Liu, J. Wu, and I. Cardei, "Message forwarding in Cyclic MobiSpace: the multi-copy case," *2009 IEEE 6th Int. Conf. Mob. Adhoc Sens. Syst.*, pp. 413–422, Oct. 2009.

[62] M. Y. S. Uddin, H. Ahmadi, T. Abdelzaher, and R. Kravets, "A Low-energy, Multi-copy Inter-contact Routing Protocol for Disaster Response Networks," in *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2009, pp. 1–9.

[63] M. Y. S. Uddin, H. Ahmadi, T. Abdelzaher, and R. Kravets, "Intercontact Routing for Energy Constrained Disaster Response Networks," *IEEE Trans. Mob. Comput.*, vol. 12, no. 10, pp. 1986–1998, Oct. 2013.

[64] R. Handorean, C. Gill, and G. Roman, "Accommodating Transient Connectivity in Ad Hoc and Mobile Settings," in *Pervasive Computing*, Springer Berlin Heidelberg, 2004, pp. 305–322.

[65] M. Seligman, K. Fall, and P. Mundur, "Alternative custodians for congestion control in delay tolerant networks," in *Proceedings of the 2006 SIGCOMM workshop on Challenged networks - CHANTS '06*, 2006, pp. 229–236.

[66] M. Seligman, K. Fall, and P. Mundur, "Storage routing for DTN congestion control," *Wirel. Commun. Mob. Comput.*, vol. 7, no. 10, pp. 1183–1196, Dec. 2007.

[67] S. Merugu, M. Ammar, and E. Zegura, "Routing in Space and Time in Networks with Predictable Mobility," 2004.

[68] A. G. Tasiopoulos, C. Tsiaras, and S. Toumpis, "Optimal and Achievable Cost/Delay Tradeoffs in Delay-Tolerant Networks," *Comput. Networks*, vol. 70, pp. 59–74, Sep. 2014.

[69] D. Hay and P. Giaccone, "Optimal routing and scheduling for deterministic delay tolerant networks," in *Wireless On-Demand Network Systems and Services (WONS)*, 2009, pp. 27–34.

[70] A. Di Nicolo and P. Giaccone, "Performance Limits of Real Delay Tolerant Networks," in *Wireless On-Demand Network Systems and Services (WONS)*, 2008, pp. 149–155.

[71] M. Garetto, P. Giaccone, and E. Leonardi, "On the Capacity of Ad Hoc Wireless Networks Under General Node Mobility," in *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, 2007, pp. 357–365.

[72] M. Garetto, P. Giaccone, and E. Leonardi, "On the effectiveness of the 2-hop routing strategy in mobile ad hoc networks," in *IEEE International Conference on Communications*, 2007, pp. 3108–3113.

[73] T. Abdelkader, K. Naik, and A. Nayak, "Choosing the objective of optimal routing protocols in delay tolerant networks," in *International Computer Engineering Conference (ICENCO), 2010*, 2010, pp. 16–21.

[74] J. Alonso and K. Fall, "A Linear Programming Formulation of Flows Over Time with Piecewise Constant Capacity and Transit Times," Intel, 2003.

[75] R. Bocquillon and A. Jouglet, "Data Transfer in Delay-Tolerant Networks," in *2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications*, 2013, pp. 355–359.

[76] G. Amantea, H. Rivano, and A. Goldman, "A Delay-Tolerant Network Routing Algorithm Based on Column Generation," *2013 IEEE 12th Int. Symp. Netw. Comput. Appl.*, pp. 89–96, 2013.

[77] O. Gnawali, M. Polyakov, P. Bose, and R. Govindan, "Data Centric, Position-Based

Routing in Space Networks," in *IEEE Aerospace Conference*, 2005.

[78]   M. Demmer and K. Fall, "DTLSR: Delay Tolerant Routing for Developing Regions," in *Workshop on Networked Systems for Developing Regions*, 2007.

[79]   D. Johnson, C. Aichele, and N. Ntlatlapa, "A simple pragmatic approach to mesh routing using BATMAN," *2nd IFIP Int. Symp. Wirel. Commun. Inf. Technol. Dev. Ctries.*, p. 10, 2008.

[80]   L. Delosières and S. Nadjm-Tehrani, "BATMAN Store-and-Forward: The Best of the Two Worlds," in *2012 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2012*, 2012, no. March, pp. 721–727.

[81]   I. Lluch, P. T. Grogan, U. Pica, and A. Golkar, "Simulating a Proactive Ad-Hoc Network Protocol for Federated Satellite Systems," in *IEEE Aerospace Conference*, 2015.

[82]   G. Sandulescu, A. Niruntasukrat, and C. Charnsripinyo, "Resource-aware capacity evaluation for heterogeneous, disruption-tolerant networks," in *Proceedings of the 9th Asian Internet Engineering Conference on - AINTEC '13*, 2013, pp. 57–64.

[83]   S. Arora and B. Barak, *Computational Complexity: A Modern Approach*, 1st ed. Cambridge University Press, 2009.

[84]   H. Cruz-Sánchez, L. Franck, and B. Andre-Luc, "Use of Store and Forward Metrics for Service-Oriented Routing in Satellite Constellations," in *Satellite and Space Communications, IWSSC*, 2008, pp. 80–84.

[85]   R. Diana, E. Lochin, L. Franck, C. Baudoin, E. Dubois, and P. Gelard, "A DTN routing scheme for quasi-deterministic networks with application to LEO satellites topology," in *IEEE Vehicular Technology Conference (VTC)*, 2012.

[86]   S. Burleigh, "Contact Graph Routing," Internet Engineering Task Force (IETF), 2009.

[87]   C. Caini and R. Firrincieli, "Application of Contact Graph Routing to LEO satellite DTN communications," in *2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 3301–3305.

[88]   J. Seguí, E. Jennings, and S. Burleigh, "Enhancing Contact Graph Routing for Delay Tolerant Space Networking," in *IEEE Global Communications Conference*, 2011.

[89]   E. Birrane, S. Burleigh, and N. Kasch, "Analysis of the Contact Graph Routing Algorithm: Bounding Interplanetary Paths," *Acta Astronaut.*, vol. 75, pp. 108–119, Jun. 2012.

[90]   N. Bezirgiannidis, S. Burleigh, and V. Tsaoussidis, "Delivery Time Estimation for Space Bundles," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 3, pp. 1897–1910, 2013.

[91]   N. Bezirgiannidis and V. Tsaoussidis, "Predicting Queueing Delays in Delay Tolerant Networks with Application in Space," in *Proc. of 12th International Conference on Wired & Wireless Internet Communications (WWIC 2014)*, 2014, pp. 228–242.

[92]   N. Bezirgiannidis, F. Tsapeli, S. Diamantopoulos, and V. Tsaoussidis, "Towards Flexibility and Accuracy in Space DTN Communications," in *Proceedings of the 8th ACM MobiCom Workshop on Challenged Networks - CHANTS '13*, 2013, pp. 43–48.

[93]   N. Bezirgiannidis, C. Caini, D. D. Padalino Montenero, M. Ruggieri, and V. Tsaoussidis, "Contact Graph Routing Enhancements for Delay Tolerant Space Communications," in *2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, 2014, pp. 17–23.

[94]   N. Bezirgiannidis, C. Caini, and V. Tsaoussidis, "Analysis of Contact Graph Routing Enhancements for DTN Space Communications," *Int. J. Satell. Commun. Netw.*, 2015.

[95]   J. a. Fraire and P. a. Ferreyra, "Assessing DTN architecture reliability for distributed satellite constellations: Preliminary results from a case study," in *2014 IEEE Biennial Congress of Argentina (ARGENCON)*, 2014, pp. 564–569.

[96]   A. Kandhalu and R. Rajkumar, "QoS-Based Resource Allocation for Next-Generation Spacecraft Networks," *2012 IEEE 33rd Real-Time Syst. Symp.*, vol. 6, pp. 163–172, Dec. 2012.

[97]   C. Liu and J. Wu, "Scalable routing in delay tolerant networks," in *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '07*, 2007, pp. 51–60.

[98]   Y. Shao and J. Wu, "Understanding the Tolerance of Dynamic Networks: A Routing-Oriented Approach," in *2008 The 28th International Conference on Distributed Computing Systems Workshops*, 2008, pp. 180–185.

[99]   A. Lindgren and K. Phanse, "Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks," in *Communication System Software and Middleware*, 2006.

[100] A. Krifa, C. Barakat, and T. Spyropoulos, "Optimal Buffer Management Policies for Delay Tolerant Networks," in *2008 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2008, pp. 260–268.

[101] A. Krifa, C. Barakat, and T. Spyropoulos, "An optimal joint scheduling and drop policy for Delay Tolerant Networks," in *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2008, pp. 1–6.

[102] G. Fathima and R. Wahidabanu, "Buffer management for preferential delivery in opportunistic delay tolerant networks," *Int. J. Wirel. Mob. Networks*, vol. 3, no. 5, pp. 15–28, 2011.

[103] D. Hua, X. Du, L. Cao, G. Xu, and Y. Qian, "A DTN congestion avoidance strategy based on path avoidance," in *2010 2nd International Conference on Future Computer and Communication*, 2010, pp. V1-855-V1-860.

[104] Y. An and X. Luo, "MACRE: a Novel Distributed Congestion Control Algorithm in DTN," *Adv. Eng. Forum*, vol. 1, pp. 71–75, 2011.

[105] S. Burleigh, E. Jennings, and J. Schoolcraft, "Autonomous congestion control in delay-tolerant networks," in *Proceedings of the 9th AIAA International Conference on Space Operations (SpaceOps '06)*, 2006, pp. 1–10.

[106] E. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, 1959.

[107] M. Pioro and D. Medhi, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, 1st ed. Elsevier, 2004.

[108] M. L. Fredman and R. E. Tarjan, "Fibonacci Heaps And Their Uses In Improved Network Optimization Algorithms," *J. ofthe Assoc. Comput. Mach.*, vol. 34, no. 3, pp. 596–615, 1987.

[109] W. Chen, P. Palmer, S. Mackin, and G. Crowley, "Queuing Theory Application in Imaging Service Analysis for Small Earth Observation Satellites," *Acta Astronaut.*, vol. 62, no. 10–11, pp. 623–631, 2008.

[110] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wirel. Commun. Mob. Comput.*, vol. 2, no. 5, pp. 483–502, 2002.

[111] P. T. Grogan and O. L. De Weck, "Interactive simulation games to assess federated satellite system concepts," in *IEEE Aerospace Conference Proceedings*, 2015.

[112] J. Walker, "Circular orbit patterns providing continuous whole earth coverage," 1970.

[113] I. Lluch and A. Golkar, "Satellite-to-Satellite Coverage Optimization Approach for Opportunistic Inter-Satellite Links," in *Proceedings of the IEEE Aerospace Conference*, 2014.

[114] J. Beesemyer, A. Ross, and D. Rhodes, "Case Studies of Historical Epoch Shifts: Impacts on Space Systems and their Responses," in *AIAA Space 2012*, 2012, pp. 1–13.

[115] D. R. Luders, "Satellite Networks for Continuous Zonal Coverage," *ARS J.*, vol. 31, no. 2, pp. 179–184, 1961.

[116] A. Golkar and I. Lluch, "The Federated Satellite Systems Paradigm: Concept and Business Case Evaluation," *Acta Astronaut.*, vol. 111, pp. 230–248, 2015.

[117] A. C. Clarke, "Extra-terrestrial relays: Can Rocket Stations Give World-wide Radio Coverage?," *Wireless World*, no. October 1945, pp. 305–308, 1945.

[118] J. Walker, "Continuous whole-earth coverage by circular-orbit satellite patterns,"

1977.

[119] D. Beste, "Design of Satellite Constellations for Optimal Continuous Coverage," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-14, no. 3, pp. 466–473, May 1978.

[120] A. Ballard, "Rosette constellations of earth satellites," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-16, no. 5, pp. 656–637, 1980.

[121] L. Rider, "Analytic design of satellite constellations for zonal earth coverage using inclined circular orbits," *J. Astronaut. Sci.*, vol. 34, no. 1, pp. 31–64, 1986.

[122] W. Adams and L. Rider, "Circular polar constellations providing continuous single or multiple coverage above a specified latitude," *J. Astronaut. Sci.*, vol. 35, no. 2, pp. 155–192, 1987.

[123] T. Lang, "Symmetric circular orbit satellite constellations for continuous global coverage," *Astrodyn. 1987*, 1987.

[124] M. Matossian, "Improved candidate generation and coverage analysis methods for design optimization of symmetric multisatellite constellations," *Acta Astronaut.*, vol. 40, no. 2, pp. 561–571, 1997.

[125] S. Eves, "A Figure of Merit for Satellite Constellation Design," Cranfield University, 2002.

[126] F. Heine, H. Kämpfner, R. Lange, R. Czichy, R. Meyer, and M. Lutzer, "Optical inter-satellite communication operational," *Proc. - IEEE Mil. Commun. Conf. MILCOM*, vol. 16, no. 5, pp. 1583–1587, 2010.

[127] D. Troendle, C. Rochcow, P. Pimentel, F. Heine, R. Meyer, M. Lutzer, E. Benzi, P. Sivac, M. Krassenburg, and I. Shurmer, "Optical LEO-GEO Data Relay: The In-Orbit Experience," in *AIAA Space 2015*, 2015.

[128] G. Richardson, J. Penn, and P. Collopy, "Value-Centric Analysis and Value-Centric Design," in *AIAA SPACE 2010 Conference & Exposition*, 2010, no. September, pp. 1–11.

[129] O. De Weck, R. Neufville, and M. Chaize, "Staged deployment of communications satellite constellations in low earth orbit," *J. Aerosp. Comput. Information, Commun.*, vol. 1, no. March, pp. 119–136, 2004.

[130] A. Ross and D. Rhodes, "Using natural value-centric time scales for conceptualizing system timelines through epoch-era analysis," in *INCOSE International Symposium*, 2008.

[131] R. Kackar, "Taguchi's quality philosophy: analysis and commentary," in *Quality Control, Robust Design, and the Taguchi Method*, K. Dehnad, Ed. Springer US, 1989, pp. 3–22.

[132] R. Keeney and H. Raiffa, *Decisions with multiple objectives: preferences and value tradeoffs*, 1st ed. Cambridge University Press, 1993.

[133] D. Bouyssou and M. Pirlot, *Multiple criteria decision analysis: state of the art surveys*, 1st ed. Springer, 2005.

[134] A. Ross and D. Hastings, "Multi-attribute tradespace exploration as front end for effective space system design," *J. Spacecr. ...*, vol. 41, no. 1, 2004.

[135] E. Abbott, *Flatland*. Malachite Quills Publishing, 2012.

[136] P. D. Collopy, "The Challenge of Modeling Cost: The Problem," in *1st International Conference on Innovation and Integration in Aerospace Sciences*, 2005.

[137] P. D. Collopy, "The Challenge of Modeling Cost: Steps Toward a Solution," in *1st International Conference on Innovation and Integration in Aerospace Sciences*, 2005.

[138] S. Keller, P. Collopy, and P. Componation, "What is wrong with space system cost models? A survey and assessment of cost estimating approaches," *Acta Astronaut.*, vol. 93, pp. 345–351, 2014.

[139] E. Mahr and G. Richardson, "Development of the Small Satellite Cost Model (SSCM) Edition 2002," *2003 IEEE Aerosp. Conf.*, vol. 8, pp. 3831–3841, 2003.

[140] C. J. Lowe and M. Macdonald, "Rapid model-based inter-disciplinary design of a CubeSat mission," *Acta Astronaut.*, vol. 105, no. 1, pp. 321–332, 2014.

[141] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[142] T. Mosher, "Improving spacecraft design using a multidisciplinary design optimization methodology," University of Colorado, 2000.

[143] R. Odegard, "Increased confidence in concept design through trade space exploration and multiobjective optimization," Massachusettes Institute of Technology, 2008.

[144] E. Taylor, "Evaluation of multidisciplinary design optimization techniques as applied to spacecraft design," in *Aerospace Conference Proceedings, 2000 IEEE*, 2000.

[145] I. Budianto, "A collaborative optimization approach to improve the design and deployment of satellite constellations," Ph.D. Dissertation, Georgia Inst. of Tech., Atlanta, GA, 2000.

[146] I. a. Budianto and J. R. Olds, "Design and Deployment of a Satellite Constellation Using Collaborative Optimization," *J. Spacecr. Rockets*, vol. 41, no. 6, pp. 956–963, Nov. 2004.

[147] C. Jilla and D. Miller, "A Reliability Model for the Design and Optimization of Separated Spacecraft Interferometer Arrays," in *11th AIAA/USU Conference on Small Satellites*, 1997.

[148] C. Jilla, "A Multiobjective, Multidisciplinary Design Optimization Methodology for the Conceptual Design of Distributed Satellite Systems," American Institute of Aeronautics and Astronautics, Reston, Virigina, 2002.

[149] J. T. Hwang, D. Y. Lee, J. W. Cutler, and J. R. Martins, "Large-Scale Multidisciplinary Optimization of a Small Satellite's Design and Operation," *J. Spacecr. Rockets*, 2013.

[150] G. Amata, G. Fasano, L. Arcaro, F. Della Croce, M. Norese, S. Palamara, R. Tadei, and F. Fragnelli, "Multidisciplinary optimisation in mission analysis and design process," GSP 03/N16, 2004.

[151] J. Guo, W. Yao, and E. Gill, "Uncertainty Multidisciplinary Design Optimization of Distributed Space Systems," in *6th International workshop on satellite ...*, 2010.

[152] J. H. Saleh and K. Marais, "Highlights from the early (and pre-) history of

reliability engineering," *Reliab. Eng. Syst. Saf.*, vol. 91, no. 2, pp. 249–256, Feb. 2006.

[153] H. McManus, M. Richards, A. Ross, and D. Hastings, "A Framework for Incorporating 'ilities' in Tradespace Studies," in *AIAA Space*, 2007, pp. 1–14.

[154] J. Saleh, E. Lamassoure, and D. Hastings, "Space systems flexibility provided by on-orbit servicing: Part 1," *J. Spacecr. …*, vol. 39, no. 4, 2002.

[155] E. Lamassoure, J. Saleh, and D. Hastings, "Space systems flexibility provided by on-orbit servicing: Part 2," *J. Spacecr. …*, vol. 39, no. 4, 2002.

[156] J. Saleh, "Weaving time into system architecture: new perspectives on flexibility, spacecraft design lifetime, and on-orbit servicing," Massachusetts Institute of Technology, 2001.

[157] M. G. Richards, A. M. Ross, N. B. Shah, and D. E. Hastings, "Metrics for Evaluating Survivability in Dynamic Multi-Attribute Tradespace Exploration," *J. Spacecr. Rockets*, vol. 46, no. 5, pp. 1049–1064, Sep. 2009.

[158] J. Beesemyer, D. Fulcoly, A. M. Ross, and D. H. Rhodes, "Developing methods to design for evolvability: research approach and preliminary design principles," *9th Conf. Syst. Eng. Res.*, no. April, pp. 1–12, 2011.

[159] J. H. Saleh, G. Mark, and N. C. Jordan, "Flexibility: a multi-disciplinary literature review and a research agenda for designing flexible engineering systems," *J. Eng. Des.*, vol. 20, no. 3, pp. 307–323, Jun. 2009.

[160] A. Ross, H. McManus, and D. Rhodes, "Responsive systems comparison method: Dynamic insights into designing a satellite radar system," in *AIAA SPACE 2009 Conference & Exposition*, 2009.

[161] C. D. Jilla, D. W. Miller, and R. J. Sedwick, "Application of Multidisciplinary Design Optimization Techniques to Distributed Satellite Systems," *J. Spacecr. Rockets*, vol. 37, no. 4, pp. 481–490, 2000.

[162] G. F. Dubos and J. H. Saleh, "Comparative cost and utility analysis of monolith and fractionated spacecraft using failure and replacement Markov models," *Acta Astronaut.*, vol. 68, no. 1–2, pp. 172–184, Jan. 2011.

[163] J.-F. Castet and J. H. Saleh, "Satellite Reliability: Statistical Data Analysis and Modeling," *J. Spacecr. Rockets*, vol. 46, no. 5, pp. 1065–1076, Sep. 2009.

[164] J.-F. Castet and J. H. Saleh, "On the concept of survivability, with application to spacecraft and space-based networks," *Reliab. Eng. Syst. Saf.*, vol. 99, pp. 123–138, Mar. 2012.

[165] E. Kerr and M. Macdonald, "A general perturbations method for spacecraft lifetime analysis," in *Proceedings of the 25th AAS/AIAA Space Flight Mechanics Meeting*, 2015.

[166] L. R. Ford and D. R. Fulkerson, "Maximal Flow Through A Network," *Can. J. Math.*, vol. 8, pp. 399–404, 1956.

[167] E. a. a Dinic, "Algorithm for solution of a problem of maximum flow in networks with power estimation," *Sov. Math. Dokl*, vol. 11, no. 5, pp. 1277–1280, 1970.

[168] J. Edmonds and R. M. Karp, "Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems," *J. ACM*, vol. 19, no. 2, pp. 248–264, 1972.

[169] V. M. Malhotra, M. P. Kumar, and S. N. Maheshwari, "An $O(|V|^3)$ algorithm for finding maximum flows in networks," *Inf. Process. Lett.*, vol. 7, no. 6, pp. 277–278, 1978.

[170] Z. Galil and A. Naamad, "An $O(EVlog^2V)$ algorithm for the maximal flow problem," *J. Comput. Syst. Sci.*, vol. 21, no. 2, pp. 203–217, 1980.

[171] J. Orlin, "Max flows in O(nm) time, or better," in *Proceedings of the 45th annual ACM symposium on Theory of computing*, 2013, pp. 765–774.

[172] L. R. Ford and D. R. Fulkerson, "A suggested computational for maximal multi-commodity network flows," *Manage. Sci.*, vol. 51, no. April 2016, pp. 97–101, 1958.

[173] M. Macdonald and V. Badescu, *The international handbook of space technology*. Springer, 2014.

[174] W. Larson and J. Wertz, *Space mission analysis and design*, 3rd ed. Microcosm Press, 2005.

[175] F. Dietrich and R. Davies, "Communications Architecture," in *Space Mission Analysis and Design*, 3rd ed., J. Wertz and W. Larson, Eds. Springer, 1999, pp. 533–586.

[176] J. Wertz and W. Larson, *Space Mission Analysis and Design*, 3rd ed. Microcosm, 1999.

[177] E. Reeves, "Spacecraft Design and Sizing," in *Space Mission Analysis and Design*, 2nd ed., W. Larson and J. Wertz, Eds. Springer, 2005, pp. 285–337.

[178] "SpaceX: Capabilities and services," 2016. [Online]. Available: http://www.spacex.com/about/capabilities. [Accessed: 03-Apr-2016].

[179] A. Mehrparvar, "Cubesat design specification," 2014.

[180] P. Harkness, M. McRobb, P. Lutzkendorf, R. Milligan, A. Feeney, and C. Clark, "Development status of AEOLDOS - A deorbit module for small satellites," *Adv. Sp. Res.*, vol. 54, no. 1, pp. 82–91, 2014.

[181] M. Jessup, S. Wakelam, M. Braybrook, M. Macdonald, C. Lowe, C. Rogers, R. Thompson, and R. Ward, "A Novel Approach to Severe Weather Nowcasting Using RF Sensors onboard Nano-Satellite Constellations," in *4th IET Enterprise Workshop: RF technology for Aerospace: Trends, Challenges and Opportunities*, 2015.

[182] Clyde Space Ltd, "CubeSat shop." [Online]. Available: http://www.clyde-space.com/cubesat_shop.

[183] Pumpkin Inc., "Homepage." [Online]. Available: http://www.cubesatkit.com/.

[184] Innovative Solutions In Space, "CubeSat Shop." [Online]. Available: http://www.cubesatshop.com/.

[185] D. King-Hele, *Satellite orbits in an atmosphere: theory and application*, 1st ed. Springer, 1987.

[186] M. Macdonald, C. McInnes, C. Bewick (Lücking), L. Visagie, V. Lappas, and S. Erb, "Needs assessment of gossamer structures in communications platform end-of-life disposal," in *Proc. AIAA GNC Conference, Boston, MA*, 2013.

[187] Inter-Agency Space Debris Coordination Committee, "IADC-02-01 - Space debris mitigation guidelines," 2007.

[188] Unknown, "CubeSatShop.com - Full Ground Station Kit for VHF/UHF/S-band." [Online]. Available: http://www.cubesatshop.com/index.php?page=shop.product_details&flypage= flypage.tpl&product_id=118&category_id=3&option=com_virtuemart&Itemid= 72. [Accessed: 08-Apr-2016].

[189] N. Teanby, "Nick Teanby: Software." [Online]. Available: http://www1.gly.bris.ac.uk/~teanby/software_nocol.html. [Accessed: 12-Feb-2015].

[190] J.-F. Castet and J. H. Saleh, "Satellite and satellite subsystems reliability: Statistical data analysis and modeling," *Reliab. Eng. Syst. Saf.*, vol. 94, no. 11, pp. 1718–1728, Nov. 2009.

[191] R. Brualdi, "Permutations and Combinations," in *Introductory Combinatorics*, 5th ed., Pearson Education, 2009, pp. 46–55.

Appendix A

This appendix contains results from simulations carried out on a random mobility delay-tolerant network, which were not published in Section 3.4. The design space that outlines which designs were analysed is repeated below for clarity.

Design Variable	Range	Description
Node-node link range	{0, 1, 2, 3, 4}	Inter-node distance below which data can be transferred
Node-node data rate	{1, 2, 3, 4}	Rate (packets per time step) at which data is transferred
Contention	{0.1, 0.5, 0.9}	Ratio between volume of generated data and download capacity
Time-to-live	{200, ∞}	Time from data generation to expiry

Table 30 – Design space definition

Results are grouped by similar contention and data rate, with the delay and hop-count vs. link range shown for infinite time to live (left hand side of the page), and delay, hop-count and delivery ratio vs. link range shown for a time to live of 200 units (right side of the page).

Contention = 0.1, Data rate = 1
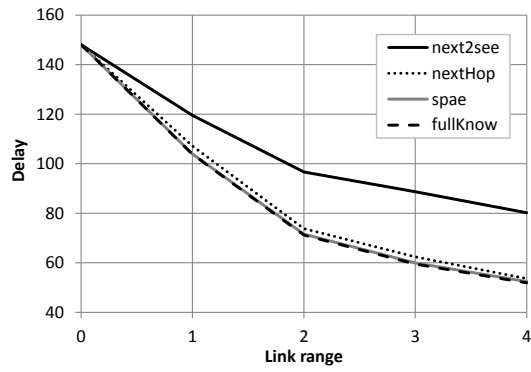
TTL = ∞   TTL = 200
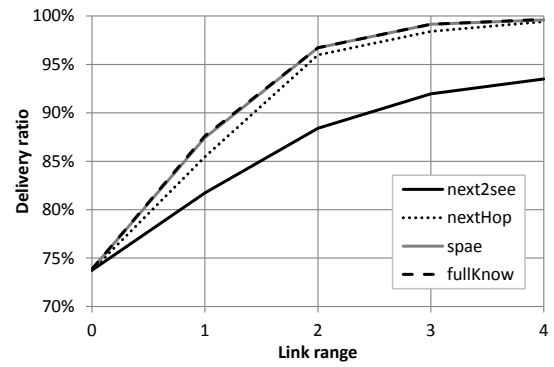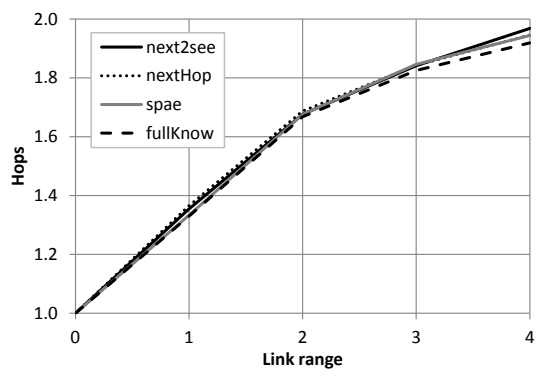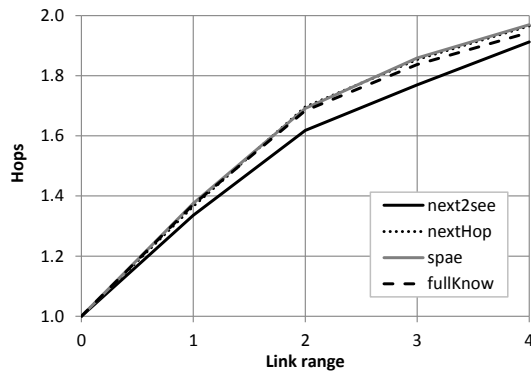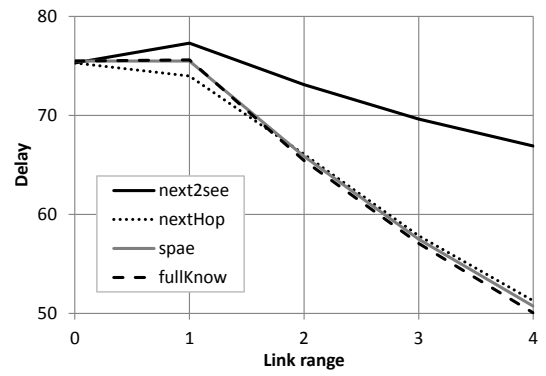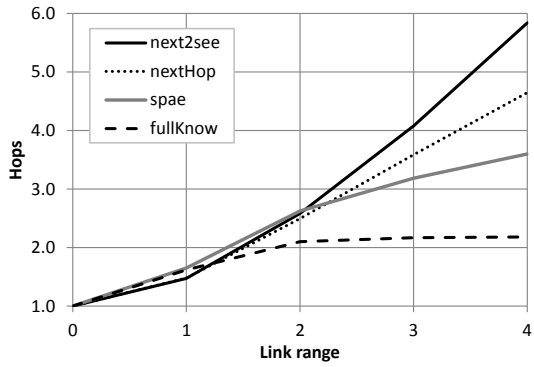
Contention = 0.1, Data rate = 4

TTL = ∞   TTL = 200
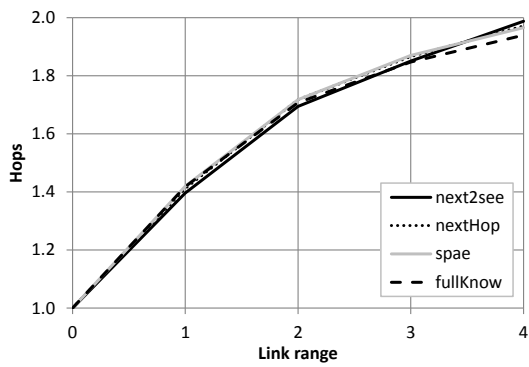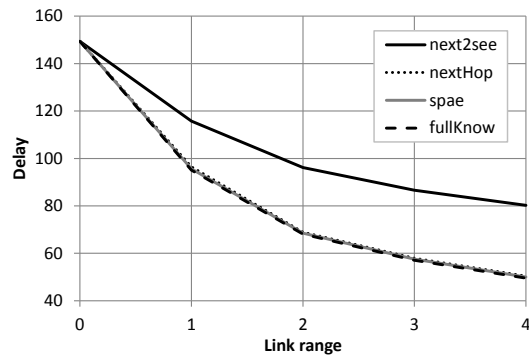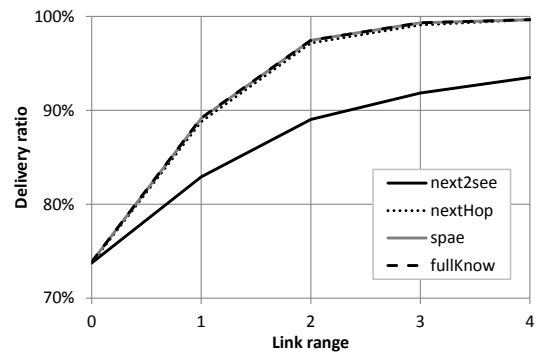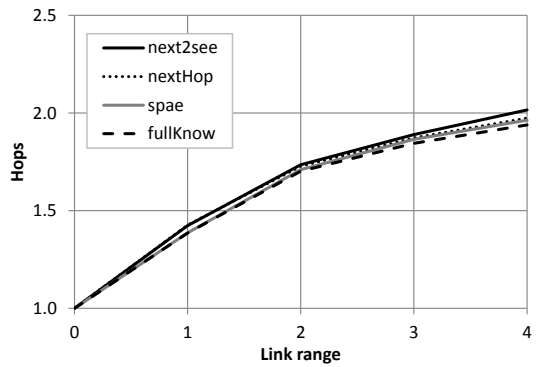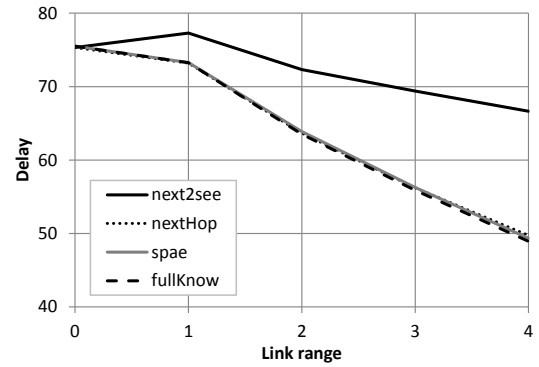
Contention = 0.5, Data rate = 1

TTL = ∞
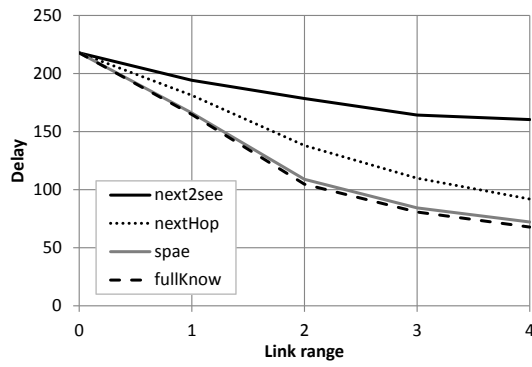
TTL = 200
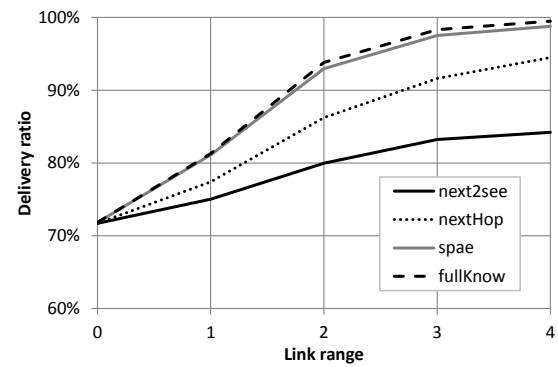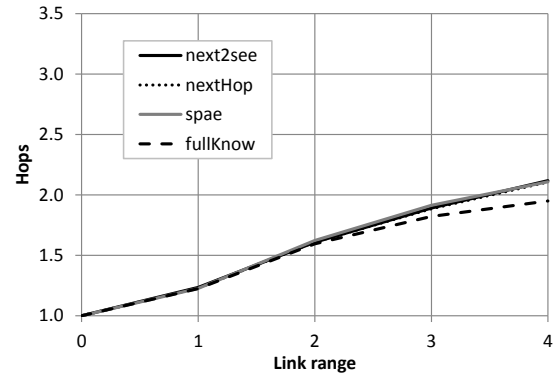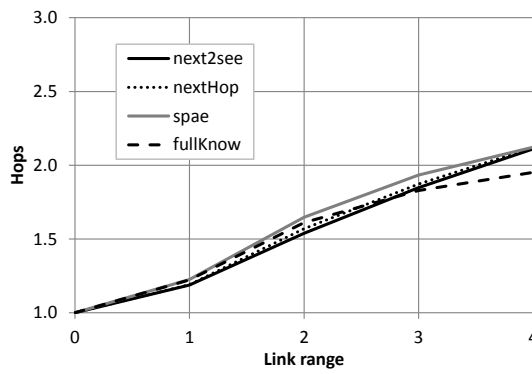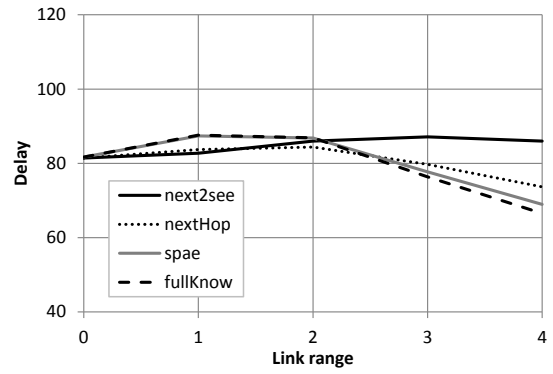
Contention = 0.5, Data rate = 2

TTL = ∞
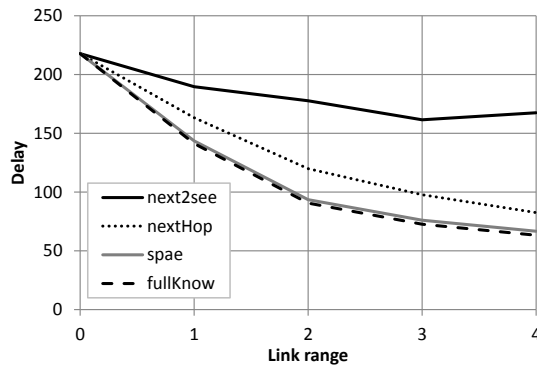
TTL = 200
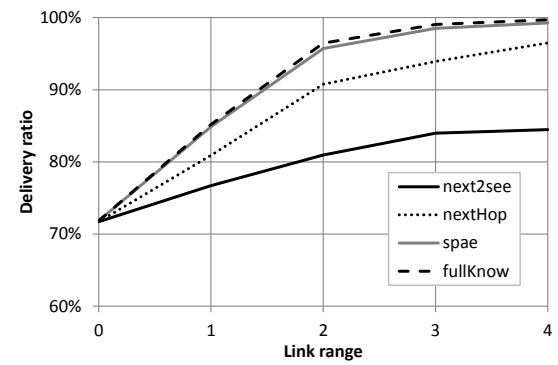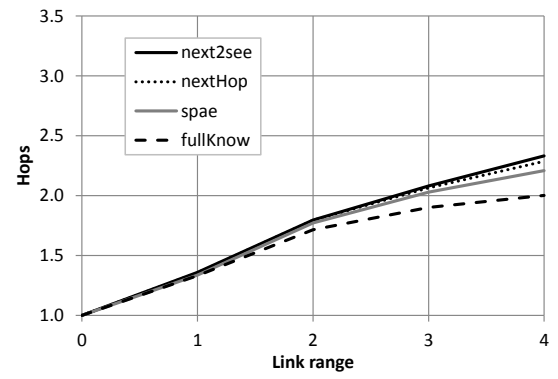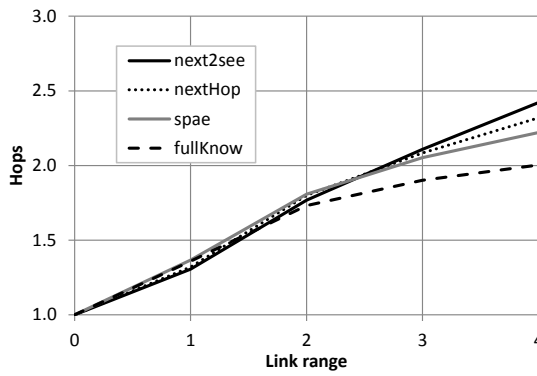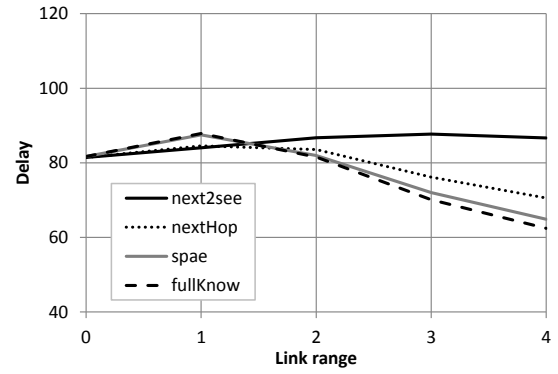
Contention = 0.5, Data rate = 4

TTL = ∞
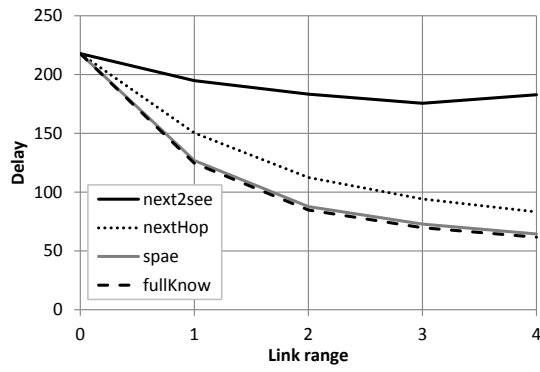
TTL = 200
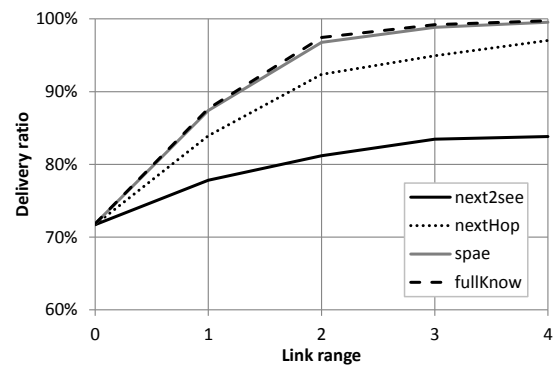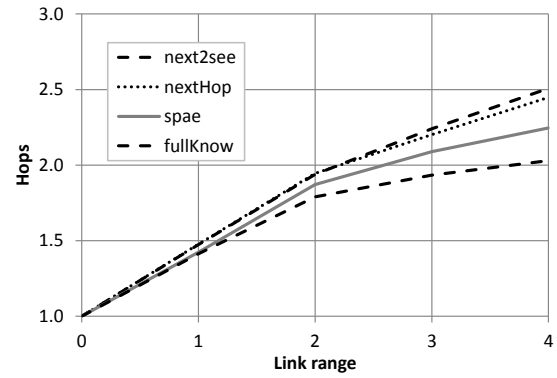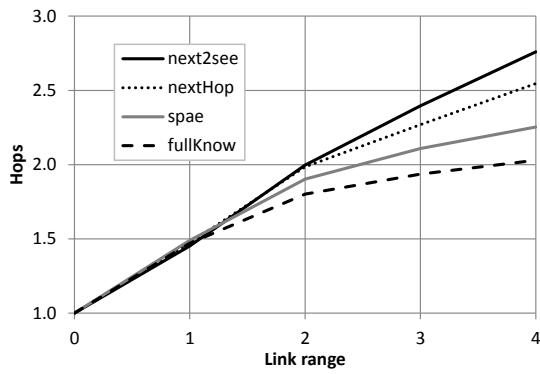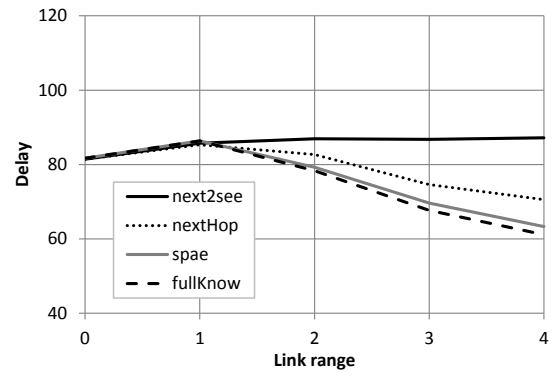
Contention = 0.9, Data rate = 1

TTL = ∞
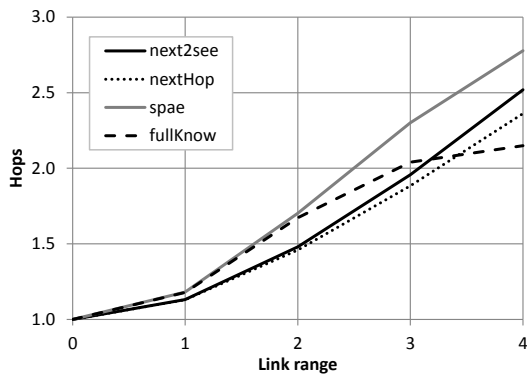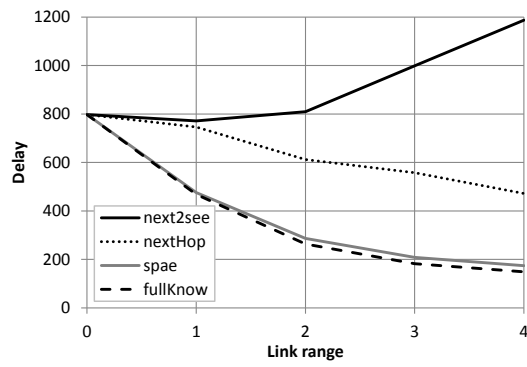
TTL = 200
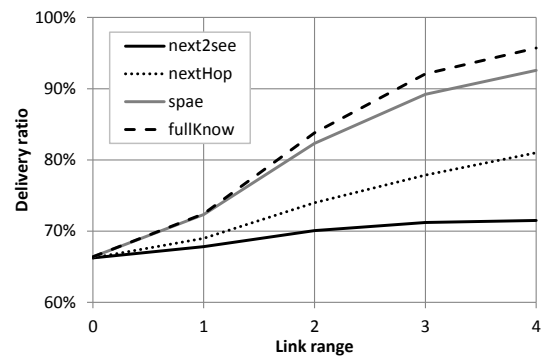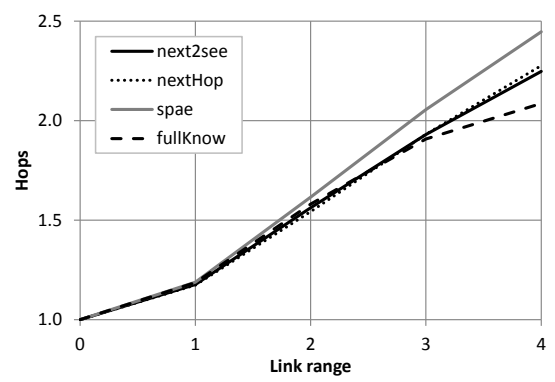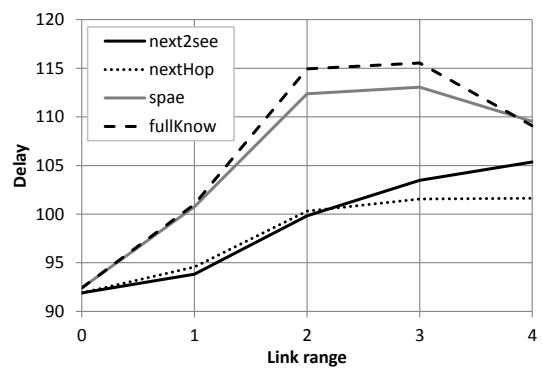
Contention = 0.9, Data rate = 4

TTL = ∞
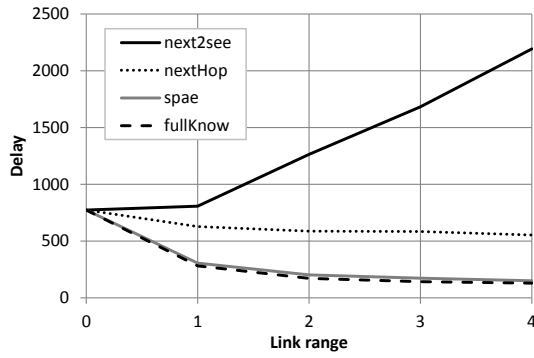
TTL = 200
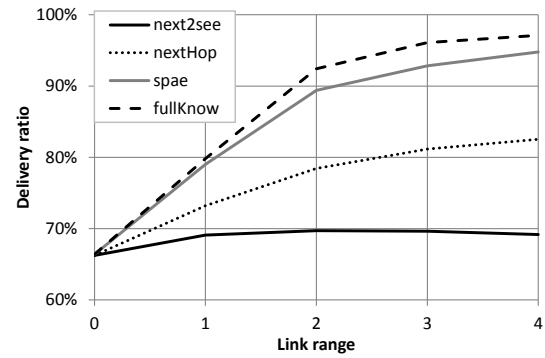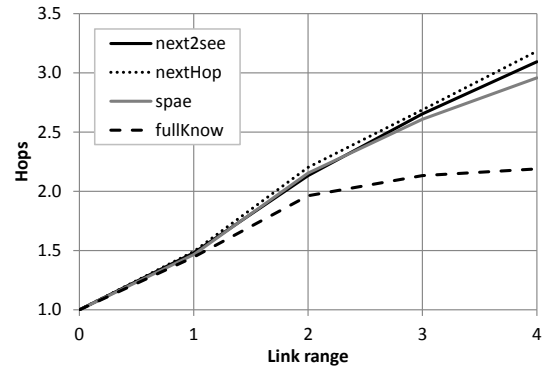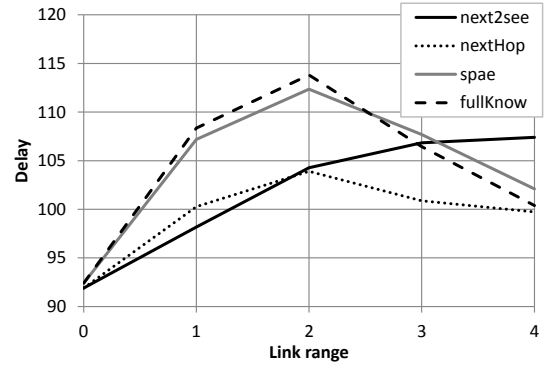
Appendix B

The following five figures show the value space (utility vs. cost) for results from the nano-satellite constellation case study evaluated in Section 5.6. Results are grouped by various design parameters; orbit altitude (Figure 58), orbit inclination (Figure 59), satellites number (Figure 60), data time to live (Figure 61) and ground station network (Figure 62), with dots indicating compliant designs (with respect to the mission requirements) and crosses indicating non-compliant designs.
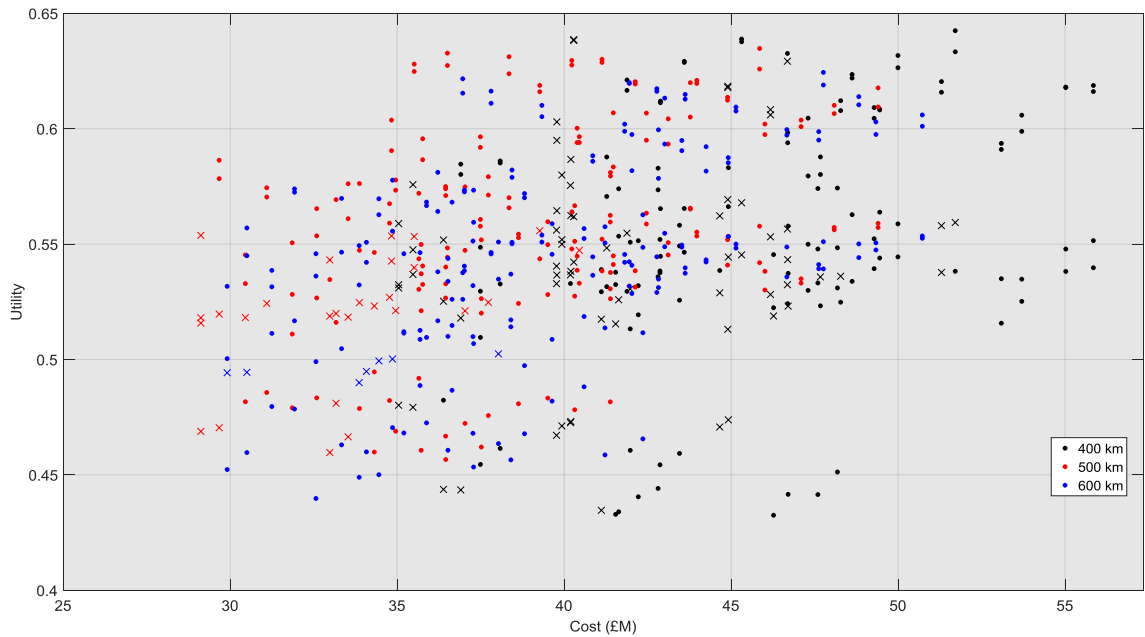
Figure 58 – Utility vs. cost of nano-satellite case study – results grouped by orbit altitude
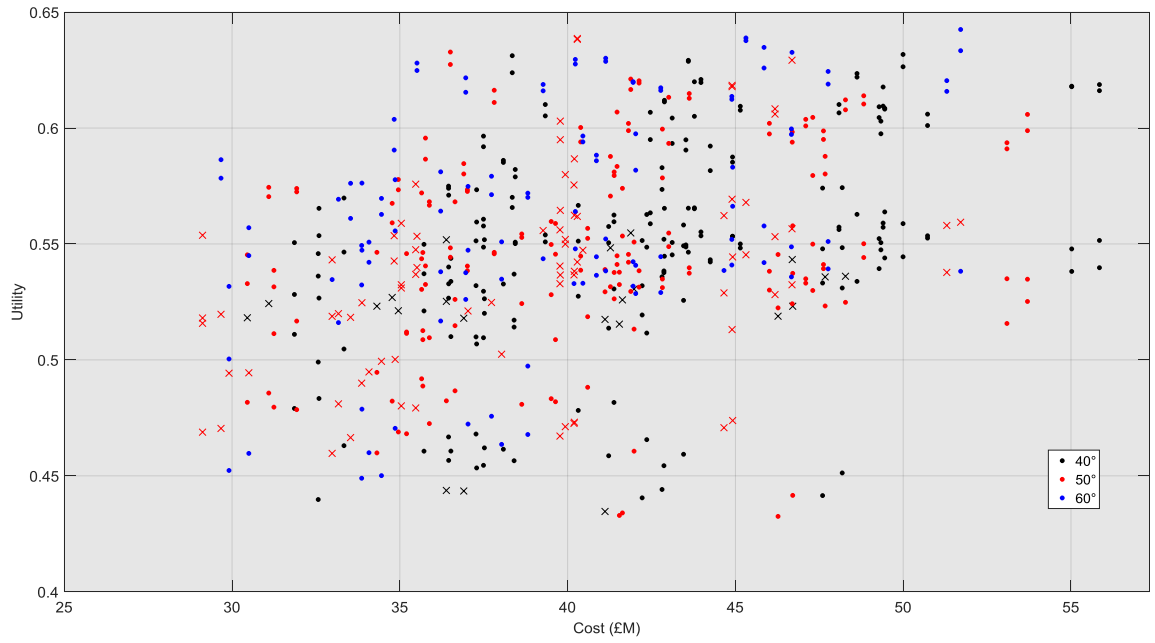
Figure 59 – Utility vs. cost of nano-satellite case study – results grouped by orbit inclination
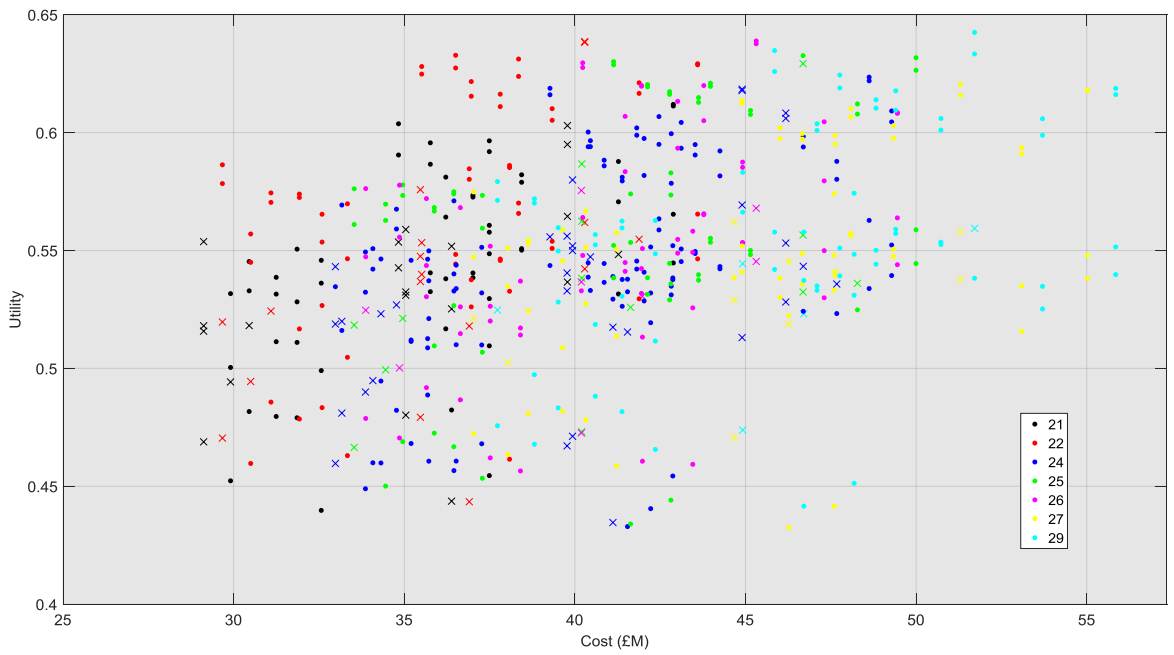
Figure 60 – Utility vs. cost of nano-satellite case study – results grouped by number of satellites
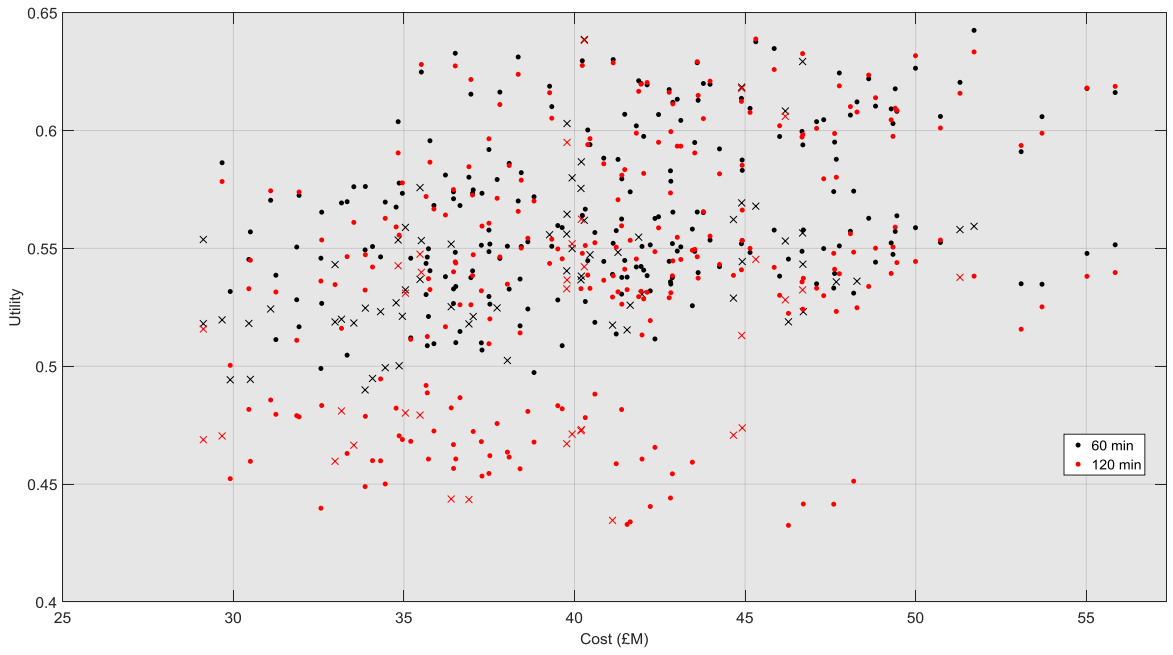
183

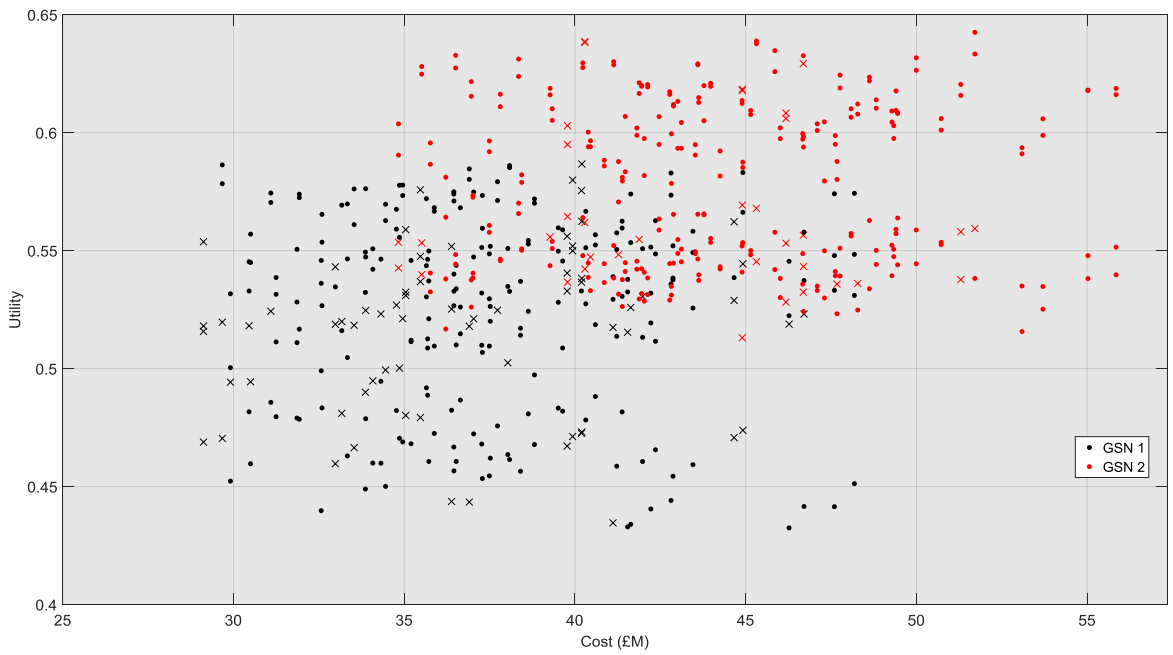*Figure 61 – Utility vs. cost of nano-satellite case study – results grouped by data time-to-live*


Figure 62 – Utility vs. cost of nano-satellite case study – results grouped by ground station network

184

The following two figures illustrate detailed results from the analysis carried out on the highest value non-ISN capable design (#193, Figure 63) and ISN-capable design (#338, Figure 64). The plots show total revisit rate, revisit rate of targets from which data was delivered, latency and delivery ratio, for points at different latitude.
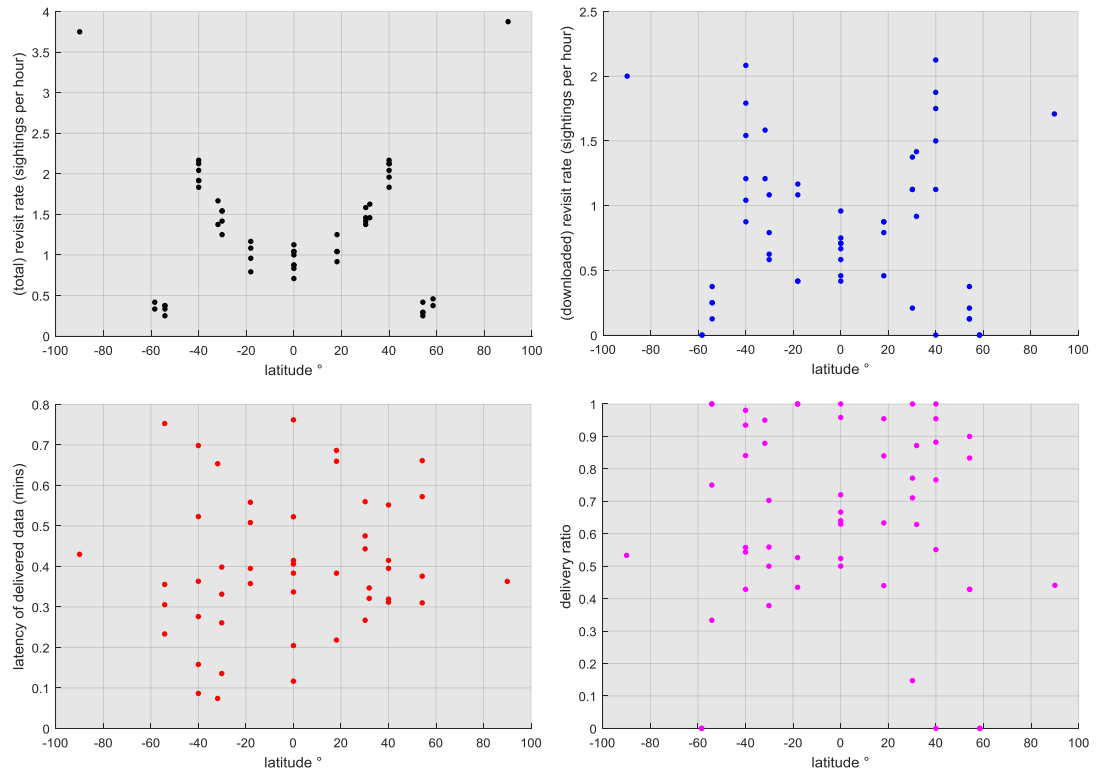
Figure 63 – Performance results of design #193, showing, clockwise from top left, revisit rate (total), revisit rate (downloaded), delivery ratio and latency, plotted against latitude, for each source node
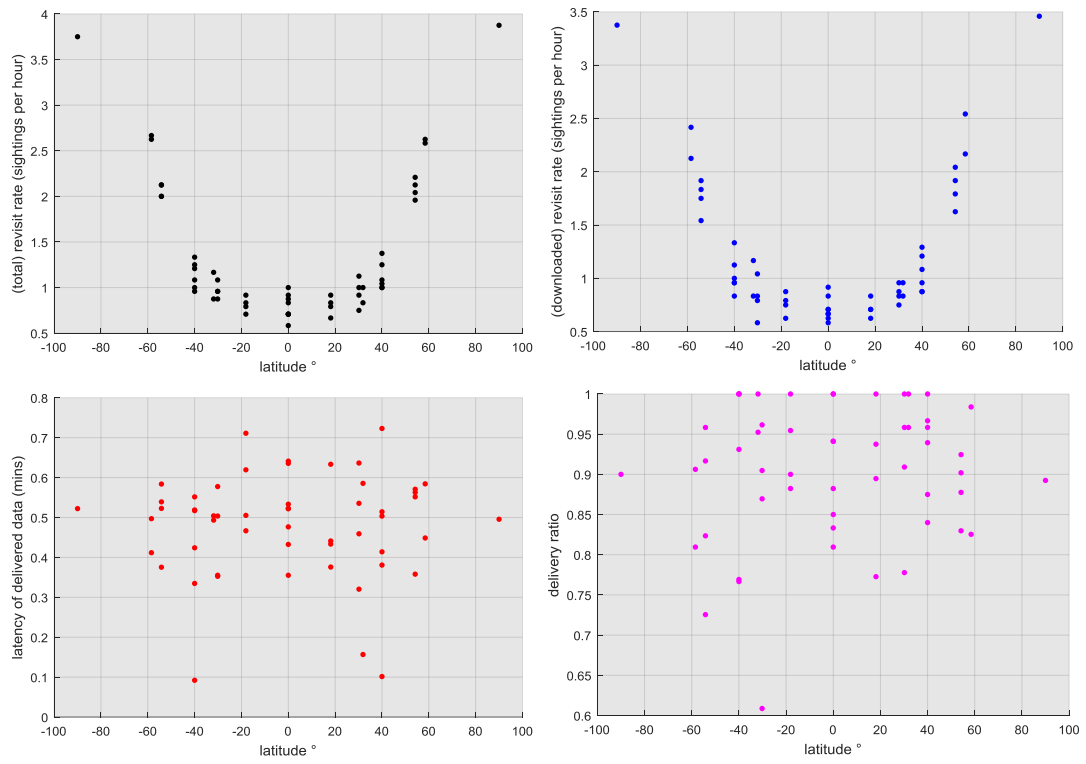
Figure 64 – Performance results of design #338, showing, clockwise from top left, revisit rate (total), revisit rate (downloaded), delivery ratio and latency, plotted against latitude, for each source node