

... CubeSat Networks ...



Stephen Ennis

Supervisor: Jonathan Dukes

School of Computer Science and Statistics

Trinity College Dublin

M.A.I. Computer Engineering

--/--/2017

Acknowledgements

...

Abstract

...

Contents

Acknowledgements.....	ii
Abstract.....	iii
Figures.....	vii
Tables.....	ix
Abbreviations.....	x
Chapter 1: Introduction	1
1.1 Objectives	5
1.2 Thesis Structure.....	8
Chapter 2: State of the Art.....	10
2.1 CubeSats.....	10
2.1.1 Capabilities.....	14
Space-to-Ground Communication Systems	15
Satellite-to-Satellite Communication Systems.....	16
Power capabilities	18
Other Capabilities.....	18
2.1.2 Applications	20
Sensing Missions	20
CubeSat Network Missions.....	23
2.2 Terrestrial Communications.....	30
2.2.1 Wireless Sensor Networks.....	31
2.2.2 Mobile Ad-Hoc Networks.....	38
2.3 CubeSat Communications	44
2.3.1 Physical Layer	45
2.3.2 Data Link Layer	46
2.3.3 Network Layer.....	52

2.3.4 Other Works	54
2.4 Other Areas of Note	54
2.4.1 Energy Aware Scheduling.....	55
2.4.2 Delay Tolerant Networking	55
Chapter 3: Proposed Protocols	57
3.1 CubeMac.....	59
3.1.1 TDMA.....	59
3.1.2 Cluster Formation.....	60
3.1.3 CDMA	61
3.1.4 Frame Structure	62
3.1.5 Energy Conservation.....	63
3.1.6 Drawbacks	65
3.2 D3	67
3.2.1 RERRs.....	68
3.2.2 Sequence Numbers.....	69
3.2.3 Route Costs.....	70
3.2.4 Discovery and Maintenance Behaviour	71
3.2.5 D3 Modifications	74
3.2.6 Drawbacks	75
Chapter 4: Simulation	76
4.1 OMNeT++.....	Error! Bookmark not defined.
4.2 Simulation Design.....	Error! Bookmark not defined.
4.2.1 Protocol Implementation.....	Error! Bookmark not defined.
4.2.2 Space-to-ground Implementation	78
4.2.3 Scenarios	78
Scenario 0.....	Error! Bookmark not defined.
Scenario 1.....	78
Scenario 2.....	79

Scenario 3.....	79
4.3 Implementation Issues	79
4.3.1 NS-3	79
4.3.2 CDMA	79
4.3.3 AODV.....	79
4.3.4 DYMO	79
4.3.5 Multiple Interfaces.....	79
Chapter 5: Results.....	81
5.1 Key Metrics	81
5.2 Simulation Results	81
5.2.1 Scenario 0.....	Error! Bookmark not defined.
5.2.2 Scenario 1	81
5.2.3 Scenario 2.....	81
5.2.4 Scenario 3.....	81
5.3 Discussion	81
Chapter 6: Conclusions.....	82
6.1 Discussion	82
6.1.1 Contributions	82
CubeMac.....	82
D3	82
CSN Simulation	82
6.1.2 Space Junk	82
6.1.3 Mission Design.....	82
6.2 Future Work.....	83

Figures

Figure 1 A projection of unit costs to LEO as the number of launches of a particular vehicle type increases. Image Credit: ARK Investment Management LLC.	2
Figure 2 CONTACTED An illustration of the CubeSat form factor and a deployment approach. Here CubeSats are deployed prior to delivery of the primary payload using a dispenser attached to the final stage of the launch vehicle. Image Credit: United Launch Alliance.....	3
Figure 3 An illustration of EDSN CubeSats in orbit forming a star (hub-and-spoke) topology CSN. Each CubeSat houses a radio for S2S and S2G communication. Communication to ground is perform only be the current elected Captain CubeSat.	5
Figure 5 Three 1U CubeSats beside a 3U (Poly Picosatellite Orbital Deployer) P-POD developed at CalPoly. The spring mechanism used by P-PODs to deploy CubeSats can be seen within the main housing. Image Credit: California Polytechnic State University	13
Figure 6 An illustration of the dedicated secondary payload deployers built into the “Orion Stage Adapter” of NASA’s upcoming Space Launch System. Existing launch vehicles are generally retroactively fitted with such deployers. Image Credit: NASA.	14
Figure 7 Unlike NASA’s EDSN approach, Gamalink seeks to establish multi-hop CubeSat networks capable of communicating with multiple ground stations. Gamalink designers refer to such networks as GAMANETs.	17
Figure 8 CONTACTED An off-the-shelf CubeSat attitude control unit. The rotational velocity of the three reaction wheels shown above can be altered in order to adjust attitude. Image Credit: Clyde Space.	19
Figure 9 CONTACTED (<i>Free constellation images available</i>) A conceptual illustration of the proposed RAVAN constellation. Image Credit: John Hopkins University Applied Physics Laboratory.	22
Figure 10 The Cpt/Lt protocol. EDSN designers refer to S2S communication as crosslinking. The Captain pings a Lieutenant before receiving state-of-health and science data packets. Image Credit: NASA Ames Research Centre	25
Figure 11 A timeline of the Captaincy negotiation process carried out between the two Nodes spacecraft. Image Credit: NASA Ames Research Centre	28

Figure 12 REDRAW An illustration of information that may be passed between layers. This is a departure for the treatment of each layer as an isolated black box. Image Credit: [70]	32
Figure 13 REDRAW A mobile peer architecture which bears clear similarities to CubeSats in orbiting passing over a ground station. Image Credit: [73]	34
Figure 14 REDRAW An example of network cluster forming with the election of cluster heads. With the introduction of a mobile sink clustering can be adapted to insure election of optimal cluster heads in order to maximize throughput [81]. Image Credit: [84]	37
Figure 15 REDRAW An illustration of the broadcast RREQ unicast RREP approaches used by DSR and AODV. An added feature above allows nodes to add routes to neighbours when receiving route packets. Image Credit: [91]	41
Figure 16 DARPA fractionate satellite concept. It can be seen that the mission payload exists independently of other core systems such as compute and communications. Image Credit: DARPA	45
Figure 17 REDRAW A comparison of common contention free MAC schemes. In CDMA a 'chip' is used to insure that signal on the medium are orthogonal and therefore cannot collide. Image Credit: IEEE Wireless Projects.....	47
Figure 18 IEEE RightsLink Channel utilization (Vertical axis), measured from 1.0 (100%) to 0, compared to network size (Horizontal axis) for LDMA, pure CSMA and pure TDMA. Image Credit: [111].....	50
Figure 19 ??? Throughput here is measured as the amount of time spent transmitting data divided by the amount of time available to transmit data. Considering the findings in Figure 18 It is likely that the experimental setup and the traffic simulated naturally favoured the T/CDMA protocol. Image Credit: [113]	52
Figure 20 An illustration of several NASA Tracking and Data Relay Satellites (TDRS) which make the primary backbone of NASA's deep space network. Some of the most advanced space bound radios enable the relay of signals from spacecraft throughout the solar system. Image Credit: NASA	56

Tables

Table 1 The requirements of several WSN application domain. RT – Round Trip, QoS – Quality of Service. Credit: [66].....	Error! Bookmark not defined.
Table 2 An extension of Rault et al.'s table shown in Table 1 to include a number of CSN applications. ET – Extra-Terrestrial. ET Science examples: measuring solar radiation, performing astronomical measurements etc.	31

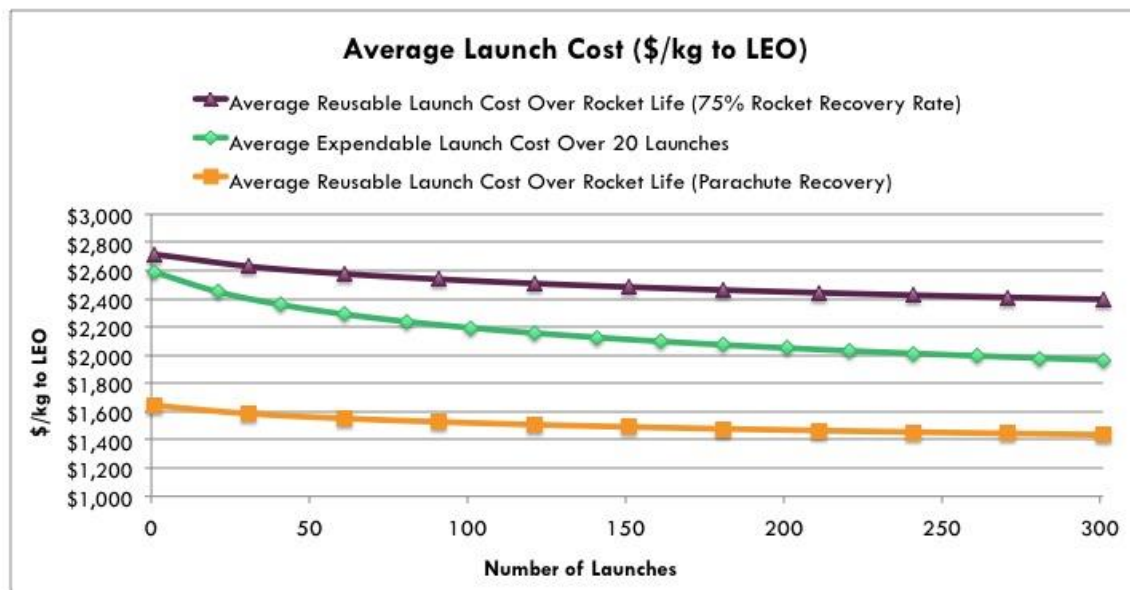
Abbreviations

Will populate and format last

LEO	Low Earth Orbit
SMB	Small to Medium Business
...	...

Chapter 1: Introduction

Access to Low Earth Orbit (160 – 2,000km) (LEO) has typically been restricted to military, government and large corporate institutions [1]. Over the past decade, two factors have disrupted this status quo and opened access to LEO for academic intuitions and SMBs alike. The first factor is the advent of the “private space race”. Greater competition has caused a dramatic drop in the cost of launching one kilogram into LEO i.e. the “unit cost to LEO”. In 2001 the NASA’s Space Transport System’s space shuttle’s unit cost to LEO, with a fully loaded cargo bay, was approximately \$60,000. Thanks in large part to the competitive prices of SpaceX, the minimum unit cost to LEO in 2017 is in the region of \$4,000 [2]. Analysis of launch vehicles currently under development has led to predictions of further drops in this cost as a result of increased reusability (Figure 1).



Source: United Launch Alliance, NASA, SpaceX, ARK Investment Management LLC

Figure 1 A projection of unit costs to LEO as the number of launches of a particular vehicle type increases. Image Credit: ARK Investment Management LLC.

The second, and perhaps most influential factor influencing affordable access to LEO, is the introduction of new commonly accepted small satellites classes. This work focuses on the capabilities and applications of CubeSats which, almost always, fall into the Nanosatellite (NanoSat) class. NanoSats have a wet mass of between 1kg and 10kg. The wet mass refers to the mass of the satellite along with the mass of the propellant required to 'lift' the satellite to its desired orbit. Like almost all satellites, the form factor of CubeSats is tailored to match the utilized launch vehicle or deployment mechanism. However, unlike many larger classes, there is considerable open-sourcing of the design and implementation of CubeSat components [3]. Such open-sourcing is historically rare in the satellite industry.

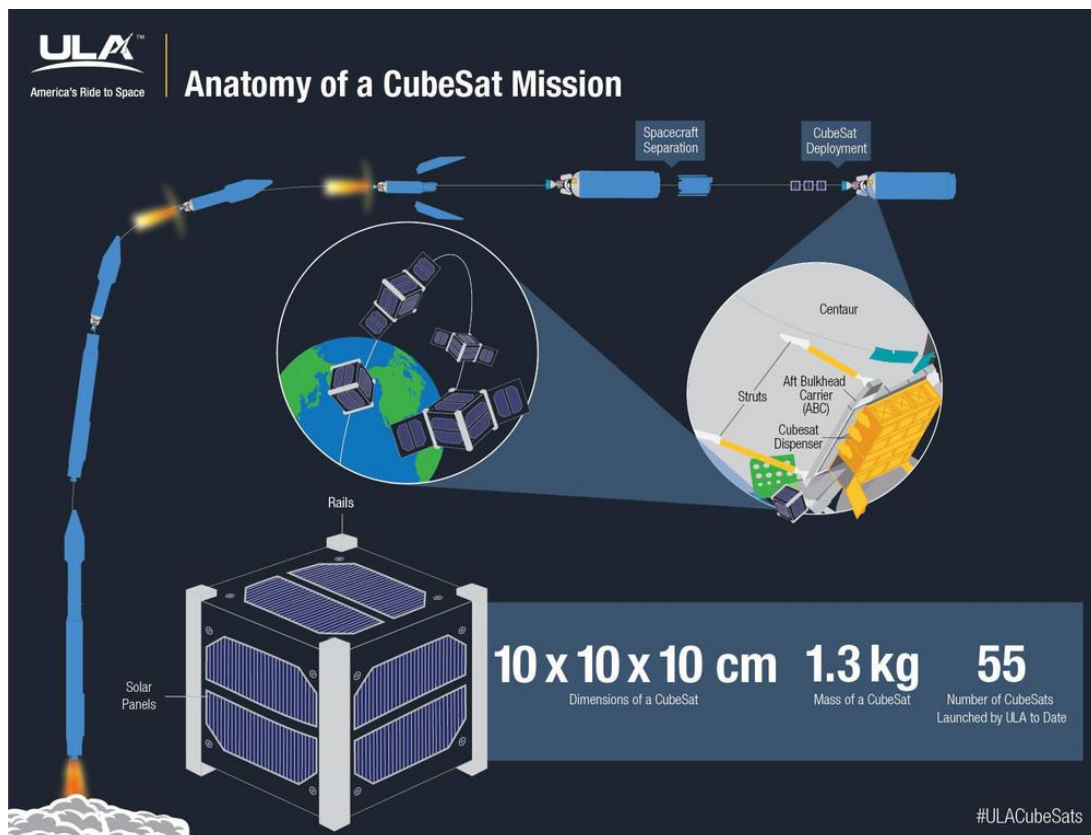


Figure 2 An illustration of the CubeSat form factor and a deployment approach. Here CubeSats are deployed prior to delivery of the primary payload using a dispenser attached to the final stage of the launch vehicle.
Image Credit: United Launch Alliance

CubeSats, as the name suggests, adopt a cube form factor. Each Cube, often referred to as a 'unit', is 10cm to a side (Figure 2). Multiple units are often combined in order to form larger CubeSats. Six unit configurations are typically the largest form factor used [4, 5]. CubeSats are generally constructed solely of commercial off-the-shelf components (COTS) components instead of those designed specifically for the extremes of space environments.

Single unit CubeSats have been shown to be capable of supporting many of the standard sub-systems that one may find on larger class satellites such as: orbital control [6], attitude determination and control (ADCS) [7], communications [8-10], and command and data handling (C&DH) [11, 12]. Along with several sub-systems, a CubeSat may carry a small 'payload' which is often a scientific instrument or some previously 'unflown' component such as an experimental antenna [13]. CubeSats have become increasingly popular within the space industry both for testing new technologies and for commercial applications. However, the primary applications for CubeSats remain within the educational and academic domains [14].

CubeSats, and other small satellites, have an advantage over other larger satellites in their ability to 'hitch' a ride alongside larger launch payloads. Effectively all modern large payloads are designed to match the capabilities of the launch vehicle. Frequently, launch vehicles will have some spare volume and lift capacity. Multiple CubeSat deployers have been developed which can make use of this spare volume and lift capacity [15, 16]. Such deployers can often carry multiple CubeSats. In cases where cargo and/or personnel are

being delivered to the International Space Station (ISS), CubeSats often hitch a ride. These CubeSats are then launched from the ISS's dedicated CubeSat deployer.

As a result of the lowering unit costs to LEO and the increasing affordability and capabilities of COTS CubeSat components, CubeSat missions have become increasingly ambitious [17-19]. This project focuses on a particular subset of emerging CubeSat missions which involve networked swarms of CubeSats; these will simply be referred to as CubeSat networks (CSNs). Multi-CubeSat missions offer greater redundancy which addresses the platform's limited power and durability. Missions involving CSNs seek to advance the platform by introducing varying degrees of autonomous cooperation and coordination between CubeSats. It is this cooperation and coordination that presents various new CubeSat mission applications. CSNs have the capacity to enable the collection of greater volumes of scientific data, novel interferometry [20], high fidelity sensory data, inexpensive low-data rate terrestrial communications and improved air traffic monitoring [21]. The space industry has taken the first crucial steps into designing and testing CSNs with missions such as EDSN (Edison Demonstration of Smallsat Networks) [22] (Figure 3), Nodes (Network & Operation Demonstration Satellites) [12] and Tianwang-1 [23].

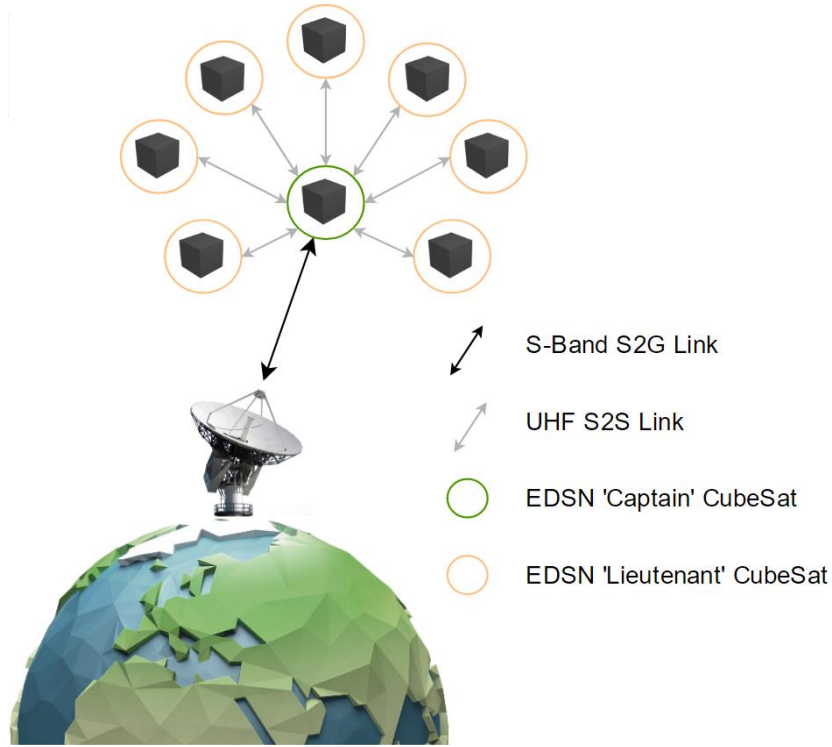


Figure 3 An illustration of EDSN CubeSats in orbit forming a star (hub-and-spoke) topology CSN. Each CubeSat houses a radio for S2S and S2G communication. Communication to ground is performed only by the current elected Captain CubeSat.

This work seeks to build upon data from the aforementioned missions. The overall aim being the exploration of fundamental aspects of the communication approaches employed in CSNs. In particular, this work attempts to identify how CSN based missions may approach communication in order to optimize space to ground (S2G) data throughput while remaining sensitive to CubeSat energy consumption.

1.1 Objectives

CSNs share many similarities with terrestrial concepts such as wireless sensor networks (WSNs) and mobile ad-hoc networks (MANETs). The state of the art in CSNs has its basis in works from these fields. Much of the existing work relating directly to CSNs was

published prior the design and launch of the first CSN mission. As a result, authors were often forced to make several assumptions as to the capabilities and constraints of CSNs.

The general motivation of this work is to assess CSN network and MAC layer protocol design. This assessment requires the analysis of existing work relating to WSNs, MANETs, CSNs, and the design and implementation of CubeSat missions. As mentioned, there are numerous and varied applications of CSNs. This work narrows the scope of interest to a hypothetical scientific mission. This chosen mission employs a number of CubeSats in trailing formation. As will be discussed further in section 2.3 and chapter 4, the chosen formation involves a number of CubeSats sharing approximately the same orbit. CubeSat's are assumed to be able to hold formation meaning they do not move relative to one another. The remaining motion is that of the CubeSat's relative to a ground station.

In the hypothetical mission, each CubeSat carries a scientific instrument which produces some data which must be communicated to ground. It is the objective of the CSN to efficiently route this data to ground. For the scientist on the ground the core concerns are the quality, quantity, and timeliness of the data received. It is assumed that issues relating to data quality are fully addressed. This leaves the quantity and timeliness of data received as metrics for the success of this hypothetical mission. The timeliness of data reception is important in applications such as communications and real-time Earth Observation (EO). This work makes an assumption that the value of science data is not affected by the time taken for said data to reach ground. As such this work's core objective is to explore approaches to maximizing the quantity of data received at ground i.e. S2G data throughput.

In terms of energy consumption, satellite-to-satellite S2S is generally considerably less expensive than S2G communication. S2S data rates are more likely to lie within the region of Mbps whereas S2G data rates are frequently as low as 12kbps. These imbalances differentiate CSN networks from many similar terrestrial networks. Ultimately, these imbalances present an optimization problem. Increasing the amount of S2G communication will increase S2G throughput but, it will also consume more energy overall and reduce the mission's lifetime. S2S may be used to communicate data to a CubeSat which has more available battery capacity and/or a better window of opportunity for S2G communications. Of course, too much S2S communication may prove wasteful when all CubeSats have enough battery and suitable S2G communication windows. Any solutions proposed by this work intend to address such fundamental challenges which arise when attempting to balance power consumption with S2G throughput.

This work presents simulations of modified communications protocols carried out using the discrete event network simulator OMNeT++ [24]. These protocols were chosen and implemented with consideration to the aforementioned power versus throughput (PvTP) trade-off for CSNs. The optimal approach to the PvTP trade-off is which enables the largest quantity of data to be received at ground over the duration of the mission. Protocols proposed in this work are assessed on their ability to increase throughput without compromising on overall energy consumption. Through examining the PvTP trade-off, this work intends to inform future CSN mission design choices.

The PvTP trade-off is affected by numerous mission aspects such as a CubeSat's power, recharging, and communications capabilities. These capabilities will vary from mission to mission and are liable to develop significantly over the coming years. As far as is possible, CubeSat capabilities are modelled on the current state of the art. Advancements in

CubeSat technology may alter the context of the PvTP trade-off. However, the core trade-off of throughput at the cost of energy consumption will remain.

1.2 Hypothetical Mission

...

1.3 Thesis Structure

The remaining five chapters of this dissertation are structured as follows. Chapter 2, “State of the Art”, examines relevant prior art relating to CubeSat and terrestrial communications as well as detailing the current capabilities and applications of CubeSats. This chapter serves as an extension to the background of PvTP trade-off and as the rationale behind many of the choices made during protocol and simulation design.

Chapter 3, “Proposed Protocols”, presents this work’s proposed protocols as informed by the state of the art. In some cases, references are made to restrictions placed upon protocol design due to the practicalities of OMNeT++. These restrictions are discussed further in the Chapter 4. This chapter presents in-depth explanations and illustrations of the proposed protocols alongside justifications of relevant design choices.

Chapter 4, “Simulation”, deals primarily with the experimental setup, implementation and analysis of the protocols simulated using OMNeT++. Detailing the experimental setup involves discussing several assumptions made regarding CubeSat and space-bound wireless communications. This chapter also places a focus on the challenges relating to the implementation and analysis of the proposed protocols.

Chapter 5, “Results”, introduces the key metrics chosen for the analysis of the performance of the simulated protocols. Results for several simulation scenarios are presented graphically in a number of figures. Each scenario represents a particular parameterization or configuration of the base simulation described in Chapter 4. Discussion is provided for each scenario. A final short summary of the results across all scenarios concludes the chapter.

Finally, Chapter 6, “Conclusions”, presents several areas of discussion and closing thoughts relating to the work reviewed and carried out. The intention of the chapter is to present the findings of this work in the larger contexts of CubeSats, satellites and the space industry. The chapter concludes by detailing potential future work in the development of the proposed protocols and in simulating CubeSat communications.

Chapter 2: State of the Art

The major literature informing this work can be roughly divided into three broad sections: CubeSats, terrestrial communications, and CubeSat communications. The first of these sections provides an in-depth exploration into the CubeSat platform along with the relevant capabilities and applications thereof. This section is followed by an examination of relevant terrestrial communication technologies which focuses on Wireless Sensor Networks (WSNs) and Mobile Ad-Hoc Networks (MANETs). Detailing relevant terrestrial technology provides important context for the following section on CubeSat communications. This final primary section seeks to explore, in depth, examples of the latest proposed approaches to CubeSat communications. Finally several notable secondary areas of research are discussed in brief. These areas fall outside of the scope of this work but are nonetheless influential in the greater context of space-bound communications.

This chapter covers several areas which provide a fundamental background to CSNs, the PvTP trade-off, and this work's proposed solutions. This chapter is not intended as an exhaustive review of all potentially relevant materials. Rather, this chapter is concerned with works which may clarify the chosen problem, detail potential solutions and justify or challenge this work's design decisions.

2.1 CubeSats

CubeSats typically fall within the 'Nano' satellite weight class (1-10kg). CubeSats are further classified by the number of 'units' which they contain, where a unit is a 10cm cube.

A one unit CubeSat is referred to as a '1U'. Configurations of 1U, 2U, 2.5U, 3U and 6U are all common.

CubeSats were first proposed by Bob Twiggs of Stanford University and Jordi Puig-Suari of California Polytechnic State University in 1999 [25]. In 2000 the first work detailing a new CubeSat standard was published [26]. The platform was intended to be an answer to the prohibitive costs and challenges involved in low-resource satellite development for academic purposes. At the time, there were effectively no standard approaches or components for the design and implementation of small satellites. Researchers relied almost entirely on placing instruments alongside primary payloads on larger satellites or pursuing the development of research satellites as lengthy collaborations across multiple research institutions. This was despite the fact that, frequently, research only required satellites with basic capabilities.

2003 saw the first launch of a CubeSat on-board a Russian Eurorocket [25]. At the time of writing there have been 487 CubeSats successfully launched or deployed into orbit [27] since 2003. Spread across 14 years this number may seem unimpressive, however, approximately 75% of all these launches have taken place between 2012 and the time of writing in 2017. This is due largely to the recent boom in the private space industry which has greatly lowered the cost of access to LEO [1].

Thanks in large part to a San Francisco based company, named Planet Labs [28], roughly 40% of all CubeSats were developed by commercial entities. Academic and research institutions have developed approximately 40% and the remaining 20% is divided between civilian and military institutions. In terms of CubeSat applications, roughly 60% of all missions are dedicated to Earth imaging, 20% to technology demonstration, 10% to

education, and the remaining 10% is divided between various commercial, military and scientific applications [27].

Unsurprisingly, the core motivation behind the recent popularity of CubeSats is their cost. Costs are driven down by three factors, the use of COTS components, open sourcing, and low launch costs. Effectively every component of a modern CubeSat is available in COTS form. Retailers such as Clyde Space offer a wide range of products from batteries to attitude determination and control systems [29]. COTS components reduce costs significantly by removing the need to develop or source custom components from third parties.

Combined with COTS components, open sourcing lowers costs further by reducing development time and the need for external expertise. Open and often proven approaches for CubeSat systems are becoming increasingly widespread as the platform develops [3]. Although this may seem intuitive, such sharing and open-sourcing of work in the satellite industry has been historically rare. To date, there is no accepted standards body for the domain. Researchers, such as Puig-Suari at the California Polytechnic State University (CalPoly), have driven the domain forward since its inception by creating a number of pseudo-standards. Crucially, CalPoly led the development and design of standard CubeSat deployers [30] (Figure 4). Such deployers have come to be defined as the de-facto standards for the domain. A similar pattern can be seen elsewhere in the domain such as with the development of a proposed standard satellite bus design [31]. Recently, “OpenOrbiter” by Straub et al. from the University of North Dakota is a prime example of open pseudo-standard framework for CubeSat development [32].



Figure 4 Three 1U CubeSats beside a 3U (Poly Picosatellite Orbital Deployer) P-POD developed at CalPoly. The spring mechanism used by P-PODs to deploy CubeSats can be seen within the main housing. Image Credit: California Polytechnic State University

Depending on the complexity of the CubeSat, development costs may range anywhere from \$50,000 to \$250,000 [33]. This can be compared to a development cost in the order of millions of dollars for larger satellites weighing over 100kg. A similar gap has emerged in terms of launch costs. Satellites over 100kg may see launch costs in the order of hundreds of thousands if not millions of dollars. This of course depends heavily on the launch vehicle used and the orbital requirements (Low, High, Medium, Solar etc.). CubeSats avoid these costs by ‘hitching’ a ride as secondary payloads by using volume and lift capacity not required by the primary payload. Providers such as SpaceX have disrupted the satellite industry by offering greatly reduced cost access to LEO [34]. These factors have led to CubeSat launch costs as low as \$10,000 [33]. With recent development towards multi-CubeSat and CSN missions, multiple CubeSats may take the place of their larger counterparts at a fraction of the cost. CubeSats are also making the move beyond LEO with new developments towards deep space and lunar applications [16, 18] (Figure 5).

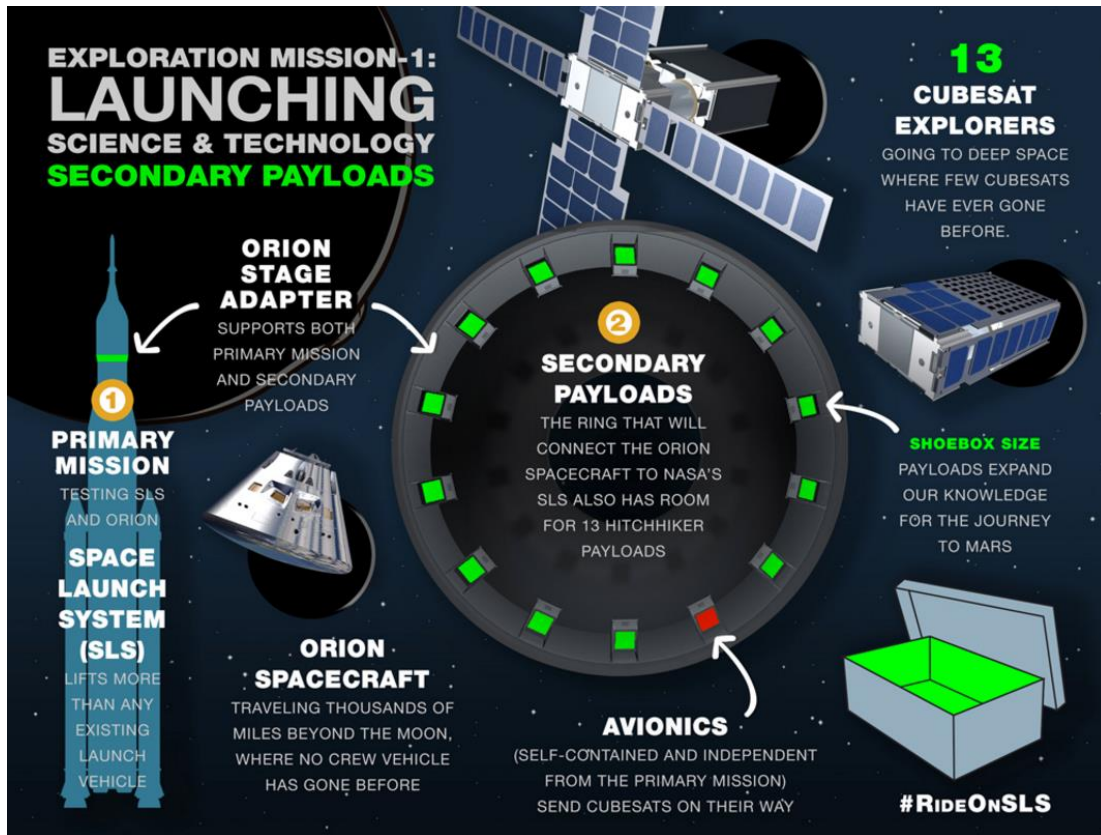


Figure 5 An illustration of the dedicated secondary payload deployers built into the “Orion Stage Adapter” of NASA’s upcoming Space Launch System. Existing launch vehicles are generally retroactively fitted with such deployers. Image Credit: NASA.

2.1.1 Capabilities

This section provides an overview of the state of the art technical capabilities of CubeSats. In line with this work’s objectives, a focus is placed on technologies relating to communication and power. This section aims to provide context to the upcoming sections on terrestrial and CubeSat communications. In many cases, there is a need to carefully assess potential communications strategies in the context of the current and emerging capabilities of the CubeSat platform.

Space-to-Ground Communication Systems

There is considerable variance in the implementation of S2G CubeSat communication sub-systems. The choice and design of a system is application dependent however, there are some broad patterns worth noting. For instance, the most common protocol for S2G communications is AX.25 [35]. Implementations using AX.25 at the link layer generally utilize UDP and IP protocols at higher protocol layers. CubeSat S2G communications sub-systems typically consume between 1W to 3W of power during transmission and can achieve data rates between 9.6kbps and 12kbps when using AX.25 [36].

There are some notable outliers to the trends in CubeSat S2G communications which denote advancements in the domain. In particular, NASA's Dynamic Ionosphere CubeSat Experiment (DICE) mission achieved a remarkable S2G maximum data rate of 3Mbps [37]. Such rates were achieved using a custom SDR based sub-system consuming approximately 9W of power and operating within the UHF band. The DICE mission holds the current record for the highest S2G data rate achieved by a CubeSat. At present there are few missions that attempt S2G rates in the order of Mbps, with the notable exception of JPL's ISARA mission [38]. The majority of upcoming missions aim to operate communication rates in the order of hundreds of kbps.

In order to approach protocol design for the PvTP trade-off, 'baseline' state of the art S2G characteristics were chosen. One of the primary inspirations for these characteristics is the Tianwang-1 (TW-1) mission [39]. This mission is an ideal candidate to use as a baseline for S2G communication modelling as the mission was designed specifically to test CubeSat inter-communications. As such, it is representative of the capabilities future CSN missions may achieve. TW-1 achieved S2G data rates of 125kbps. Details regarding the power consumption of the TW-1 S2G sub-systems are unavailable. However, by

examining previous and upcoming missions as well as work on energy budget analysis one may safely assume a peak transmission power consumption of 3W [40].

Satellite-to-Satellite Communication Systems

CubeSat S2S communications is by no means a new concept. The field began to gain popularity following a paper published in 2008 on the “Development of a Satellite Sensor Network for Future Space Missions” by Vladimirova et al. Despite this, CubeSat S2S communication remained purely conceptual until 2016 that the NASA Nodes mission made it a reality [12].

Unlike the case of S2G communications, there is comparatively little prior art regarding the S2S capabilities of CubeSats. This is unsurprising considering the age of the domain. There are effectively two cases which inform the state of the art of S2S CubeSat capabilities: The NASA’s Nodes mission and Gamalink [41]. As will be discussed, Gamalink is considerably more advanced than the systems employed on the Nodes mission. Nodes utilized a UHF transceiver and the AX.25 protocol to achieve S2S data rates of 1.2kps. As Nodes was a first in many regards it is unsurprising that the mission designers opted for a basic approach to CubeSat communications.

Gamalink is a proprietary SDR based technology developed by Tekever, a Portuguese Aerospace and Defence company. Gamalink is unquestionably the current state of the art in CubeSat S2S communications. Gamalink has been successfully tested on the TW-1 mission [23] it is also marked for use by several other missions such as i-INSPIRE II [42], DelFFi [43] and ESA’s Proba 3 [44].

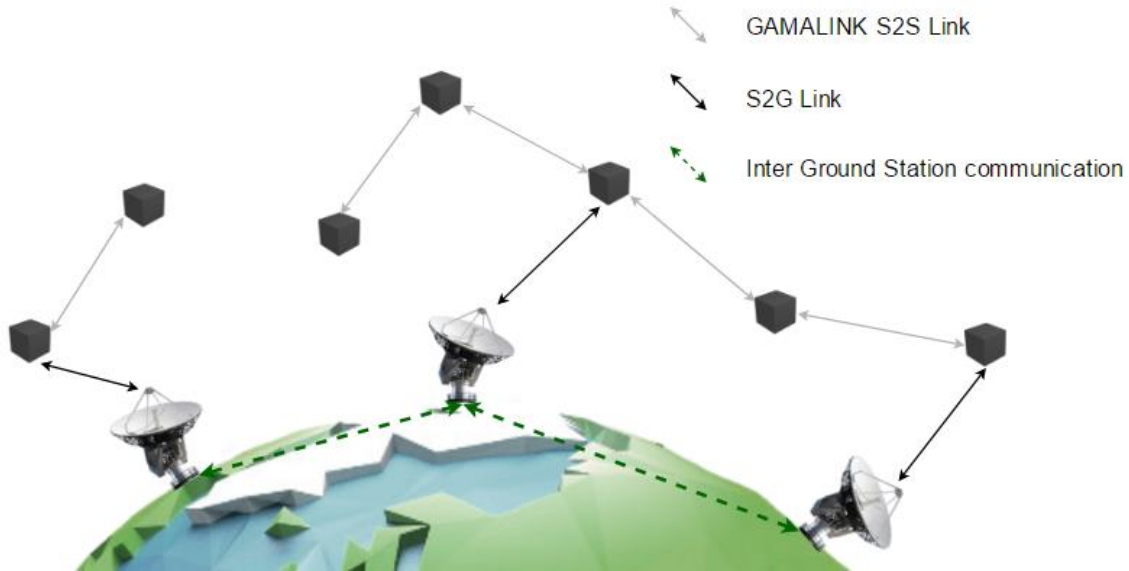


Figure 6 Unlike NASA's EDSN approach, Gamalink seeks to establish multi-hop CubeSat networks capable of communicating with multiple ground stations. Gamalink designers refer to such networks as GAMANETs.

Due to the Gamalink's proprietary nature and its likely use for military applications, details regarding Gamalink are exceptionally sparse. There appears to be no openly available information regarding protocol use or design. Tekever make several references to MANETs even stating that Gamalink implements an "SDR-based Ad hoc Space Network" (SASNET). This is an indication that the state of the art in MANET technology is integral to the design and development of Gamalink.

Despite the lack of protocol stack information, several key data points regarding Gamalink are available. According to promotional material presented to ESA, Gamalink is capable of achieving data rates up to 2Mbps. However, i-INSPIRE mission designers state that the maximum data rate as 1Mbps [42]. Gamalink operates in the S-Band, between 2.40 and 2.45GHz, with a bandwidth of 40Mhz making the advertised data rates appear feasible. Gamalink consumes a peak of 1.5W while transmitting and up to 200mW while receiving [43]. Using these details and assuming a maximum data rate of 2Mbps, a

state of the art baseline can be established for CubeSat S2S communications. This baseline is used alongside the S2G baseline in the parameterisation of this work's simulations.

Power capabilities

...

Other Capabilities

Although the most relevant CubeSat capabilities have been covered in the preceding sections, there are certain other capabilities worth discussing in brief. In general, the capabilities of CubeSats have progressed closer to those of larger satellites. Despite strict power, weight and size constraints effectively all major large satellite sub-systems have a corresponding CubeSat equivalent.

Attitude determination and control sub-systems (ADCS) are implemented to ensure correct spacecraft orientation. ADCS are critical to ensuring correctly positioned solar panels, antennae and or payload instruments. In almost all cases craft will also be required to 'de-tumble' prior to launch. There are numerous tested examples of ADCS technologies for CubeSats [7, 45, 46] several of which are available COTS (Figure 7). Along with ADCS some basic orbital control and manoeuvrability systems have also been tested at the CubeSat scale [6]. Such systems provide CubeSats with the basic capabilities to maintain regular orbits and formations.



Figure 7 An off-the-shelf CubeSat attitude control unit. The rotational velocity of the three reaction wheels shown above can be altered in order to adjust attitude. Image Credit: Clyde Space Ltd, All Rights Reserved.

The determination of accurate time and position are two classic challenges for spacecraft that have been solved in the domain of CubeSats. By communicating with Global Navigation Satellite System (GNSS) a CubeSat may acquire precise GPS time, velocity and position. Missions often require CubeSats to periodically update such information in order to coordinate in-orbit operations and S2G communications. One work by Glennon et al. on CubeSat time synchronization provides a clear overview of potential beneficial CubeSat capabilities within multi-CubeSat missions [47].

Finally, it is worth noting that the aforementioned Gamalink technology provides functionality beyond that of S2S communications. Gamalink also provides the secondary functions of GNSS receiving, attitude determination, ranging (5m resolution) and distributed clock synchronization.

2.1.2 Applications

This section examines a number of CubeSat missions. For the purposes of this work, references to CubeSat ‘applications’ should be considered synonymous with CubeSat mission objectives. Two categories of application are considered; sensing missions and CSN missions. This is not intended to assert that sensing missions and CSN missions are disjoint. CSN missions are highly suited to collaborative sensing applications.

Sensing Missions

When approaching the CSN PvTP trade-off, it is beneficial to establish a broad application case. As discussed, Earth observation is the most popular application of CubeSats to date. More generally, the majority of CubeSat missions have involved, to varying extents, some form of sensing. The application case chosen by this work is a simplified multi-CubeSat mission. Each CubeSat is assumed to carry some sensing instrument. This instrument is considered to be a black box which performs some sensing and produces some data. The goal of each CubeSat then becomes the communication of as much of its data to ground as possible. This scenario is detailed further in the Simulation chapter.

Two recent sensing missions are worth detailing: 3Cat-2 [48] and RAVAN (Radiometer Assessment using Vertically Aligned Nanotubes) [49]. 3Cat-2 is 6U CubeSat developed at the Universidad Polit cnica de Catalu na. It was launched in August of 2016 on-board a CZ-2D (Chang Zheng-2D) operated by the CNSA. 3Cat-2 represents a significant mission in the state of the art for Earth Observation missions. The extensive use of GNSS based systems also make 3Cat-2 an interesting case for the application of CSNs. 3Cat-2’s S2G downlink operates at a maximum of 115kps. This is a similar data rate as achieved by the Tianwang-1 mission which informs the baseline chosen for S2G communications.

3Cat-2's particular application case is ocean altimetry by means of GNSS-Reflectometry. 3Cat-2 performs altitude observations by examining the scattering and reflection of signals off of bodies of water. These are an 'active' form of measurement which depends on incident signals; Radar is another example of an active sensing.

Mission developers of 3Cat-2 have not stated a direct desire to pursue a multi-CubeSat mission. However, 3Cat-2's active sensing is uniquely suited to adaption for a CSN. Coordinated and synchronized measurement of signals by multiple craft in orbit could greatly improve observation fidelity and provide unique multi-dimensional data. Comparatively, 'passive' EO such as direct imaging benefits less from adaptation with CSNs.

RAVAN is a 3U CubeSat developed at the Johns Hopkins Applied Physics Laboratory. RAVAN was launched November of 2016 aboard an Atlas-5 as part of NASA's ELaNa (Educational Launch of Nanosatellites) program. RAVAN is highly relevant to this work as the mission designers clearly specify future intentions to develop a constellation of RAVAN craft. In satellite nomenclature a constellation is considered a formation of satellites evenly distributed over the surface of the Earth.

RAVAN carries an experimental carbon nanotube based radiometer. RAVAN's instrument performs multi-spectral measurements of outgoing radiation from Earth's surface. These measurements reveal trends regarding Earth's Radiation Budget (ERI) which are valuable to climate scientists. As mentioned, RAVAN is intended as a first test in a larger plan to develop a constellation of craft [50] (Figure 8). Unfortunately, the spacing of the forty proposed RAVAN craft prohibits S2S communication using current technologies. Nonetheless, RAVAN is strong example of the growing interest in multi-CubeSat missions.

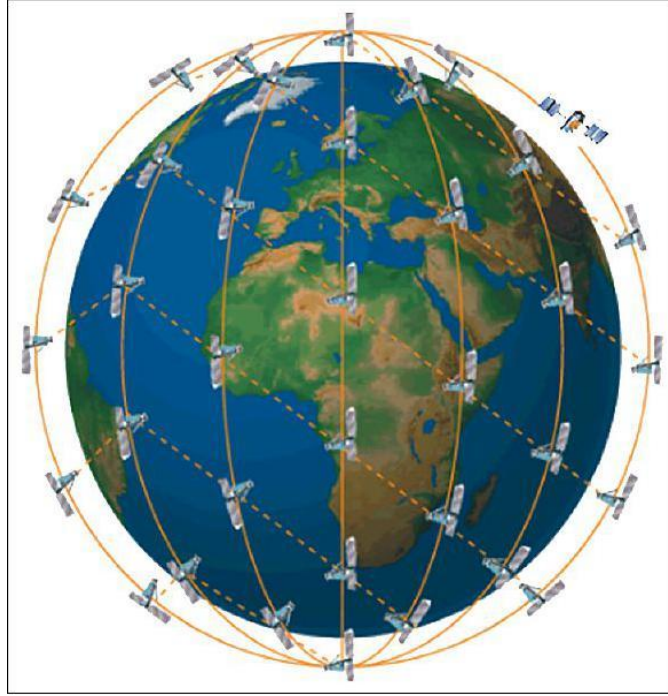


Figure 8 A conceptual illustration of the proposed RAVAN constellation.
Image Credit: John Hopkins University Applied Physics Laboratory.

It should be noted that CubeSat sensing is a deep and broad field. The examples of RAVAN and 3Cat-2 are in no way intended to illustrate a comprehensive study of the field. Notable upcoming missions such as CeREs (a Compact Radiation belt Explorer) [51], LAICE (Lower Atmosphere/Ionosphere Coupling Experiment) [52], and SOCON (Sustained Ocean Observation from Nanosatellites) [53] seek to advance the already diverse applications of CubeSats. In several regards, it is these advanced applications that drive interest in CSN based missions. As applications develop, so too does interest in advancements such as multi-point measurements, in-orbit interferometry [54] and synchronized observation. These advancements call for the introduction of CSNs. CSNs are, in many regards, an obvious next step in the development of CubeSat sensing missions.

CubeSat Network Missions

There are three major missions to consider in the area of CSNs: EDSN, Nodes, and Tianwang-1 (TW-1). Of these missions, both Nodes and TW-1 have successfully flown. NASA's eight CubeSat "Edison Demonstration of Smallsat Networks" (EDSN) mission was lost due to a failure during launch. The mission is still worth examining however as two of the remaining EDSN craft we used during the successful Nodes missions.

Two articles detailing the EDSN mission were published in 2014 prior to the loss of the mission in November 2015. The first work examines the inter-satellite communications architecture of the mission [55]. While the second work details lessons learned during development [56]. The primary objective of the EDSN mission was to implement autonomous communication and coordination of CubeSat's in orbit. Each EDSN craft is a 1.5U CubeSat weighing ~1.73kg. A Samsung smartphone provides activity scheduling and execution for each craft. Several secondary COTS micro-controllers handle various activities involving GNSS, data handling, scientific measurements, ADCS and additional sensor inputs. The scientific payload is an instrument designed to characterize radiation in LEO called the "Energetic Particle Integrating Space Environment Monitor" (EPISEM). Although the scientific objectives of the mission were secondary to the implementation of an in-orbit CubeSat network, it is clear that EDSN falls within the category of sensing applications. EDSN's sensing objectives are not entirely dissimilar from those of the RAVAN mission.

In terms of communications and power, the works published on EDSN provide a wealth of information. These works go as far as to state exactly which COTS components were used in the construction of EDSN craft. Each craft houses three primary radios: A MicroHard MHX2420 transceiver for S-Band S2G communication, an AstroDev Lithium

1 UHF transceiver for S2S communication and a StenSat UHF transmitter for beaconing (Satellite beaconing is required by North American Aerospace Defence Command (NORAD)). The AstroDev transceiver enables S2S communications at 9.6kbps with AX.25 as the link layer protocol. The Nodes mission scaled this data rate back to 1.2kbps, potentially to increase the maximum S2S communication range. Details regarding the MHX2420's S2G data rate capabilities are not provided.

Each EDSN craft carries four lithium ion batteries which combine to provide a maximum energy capacity of 5.2 Amp hours. The craft's bus operates at around 8 Volts. This implies a total energy provision of approx. 41.6 Watt hours. Six solar panels provide an average recharge of 1 watt during operation. These figures provide important context to the power consumption simulation results discussed in later chapters.

As mentioned EDSN employs an AX.25 link layer for S2S communication. The link layer, as detailed by the OSI reference model [57], is concerned primarily with medium access control (MAC). For the majority of CubeSat missions, AX.25 along with a basic application layer which communicates directly with the link layer is sufficient [36]. However, S2S communication introduces new challenges which warrant more involved approaches. In the case of EDSN a custom "Captain – Lieutenant" (Cpt/Lt) protocol was designed on top of AX.25.

The network formed by the EDSN craft is referred to as a "hub-and-spoke" (or star) network. One craft is designated as the "Captain" (Cpt) and all others are designated as "Lieutenants" (Lts). In general terms the Cpt acts as a central router to ground. All Lts send their data exclusively to the Cpt (Figure 9). The Cpt then communicates as much of this data to ground as possible. Lt communication in EDSN is triggered solely by the Cpt. The Cpt sends six pings over 50s seconds. Each set of pings specifies only one Lt from

which the Cpt is requesting data. Only after receiving a valid ping does a Lt forward its data to the Cpt. This scheme of Cpt request followed by Lt response suggests no overlapping communications can occur on the shared S2S frequency.

There is no acknowledgment scheme employed in EDSN's Cpt/Lt protocol. Lts send one "state-of-health" (SOH) packet followed by all queued science packets. The Cpt prioritizes the communication of these SOH packets to ground and treats science packets generated by Lts or by its own instrument in a FIFO manner. After ending a communication session with a Lt, the Cpt will then proceed to ping each remaining Lt in a fixed order (Figure 9). The Cpt will wait up to four minutes for a response from a pinged Lt before giving up and moving onto pinging the next Lt.

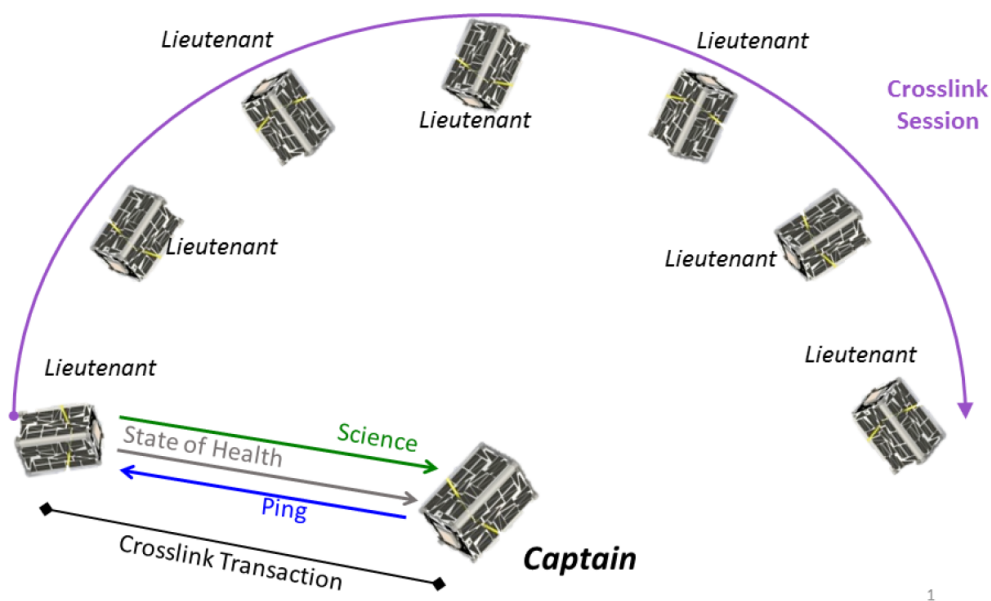


Figure 9 The Cpt/Lt protocol. EDSN designers refer to S2S communication as crosslinking. The Captain pings a Lieutenant before receiving state-of-health and science data packets. Image Credit: NASA Ames Research Centre

The Cpt role is "rotated" amongst the EDSN craft in a pre-defined fixed pattern. There is no real-time logic or election employed. Each craft periodically receives GPS time in order

to determine whose turn it is to assume the role of Cpt. If a craft cannot get GPS time it does not participate in either S2S or S2G communications. The duration a craft holds the Cpt role is referred to as a “minor cycle”. Each minor cycle lasts roughly 25 hours and includes 3-4 S2S sessions. Each session involves a single attempt by the Cpt to communicate with each Lt in the network. The Cpt self-determines one, and only one, S2G session during its minor cycle. Such sessions are scheduled by predicting the next ground station fly-over period (window) from the craft’s GPS location and velocity. As S2G communication occurs using a separate radio at a separate frequency to S2S communication, S2G sessions can take place in parallel with S2S sessions. Eight minor cycles, one for each craft, come together to form a major cycle. A major cycle lasts for roughly eight days. EDSN mission planners predicted that after three major cycles (three and one half weeks) the craft would have drifted too far apart (>120km) for S2S communication to be feasible.

During a minor cycle it is not feasible for all Lts to keep thier S2S transceivers continually on and in receiver mode. As such, along with the Cpt pinging scheme, all S2S sessions follow a fixed scheduled. A Lt will only begin listening for Cpt pings at a predetermined time during each minor cycle. During a minor cycle each craft updates it’s GPS time, position and velocity only once. Mission designers predict the maximum relative clock drift between a Lt and the Cpt to be 12 seconds. As such, an Lt will begin listening for Cpt pings 30 seconds before the scheduled time and will continue listening 30 seconds after the expected final, sixth ping. Each session start time is determined by a table of offsets. These offsets are relative to the start times of each minor cycle. Each craft is pre-programmed with the same minor-cycle start times and offset tables.

Following directly from the work on EDSN, the Nodes mission was successfully deployed from the ISS in May of 2016. Nodes employed two leftover CubeSat's from the EDSN mission which were used during the testing and development of EDSN. Effectively all aforementioned salient aspects of the EDSN mission remain. Despite only involving two craft, Nodes was able to achieve many of the objectives of EDSN. Some small changes were made to mission objectives and there were some notable changes to the custom Cpt/Lt communication protocol. These changes are detailed in a work published in 2016 following the mission's successful launch, deployment and conclusion [12].

Where EDSN was focussed purely on S2G communication, Nodes advances one step further by introducing to demonstration of Ground-to-Space (G2S) remote commands. In Nodes, the objective was set to communicate a command to the elected Cpt. The Cpt would then forward this command to the Lt for execution. Unlike with science and state-of-health packets, command packets are implemented along with specialized command acknowledgements and responses. Although this work focuses purely on S2G communications, it is worth noting that Nodes was the first demonstration of indirect command and control of CubeSats using only CubeSats.

There are several notable changes to the Cpt/Lt protocol introduced in the Nodes mission. Firstly, 12 pings over 110 seconds are used rather than 6 pings over 50 seconds. Rather than a fixed order of minor-cycles, the captaincy is negotiated between the Nodes craft (Figure 10). A default Cpt craft compares metrics relating to battery voltage, amount of science data collected and the predicated duration of the next ground station fly over. At the start of each minor cycle the selected default Cpt manages the negotiation. Once the default Cpt has compared its own metrics with that of the Lt, it will either continue as Cpt or send a "promote" command to the Lt. The default Cpt will only demote itself to the Lt

role once the promote command has been acknowledged. The Cpt will continue operations regardless of the presence of any communication with the Lt. All other aspects of the Cpt/Lt protocol are unchanged.

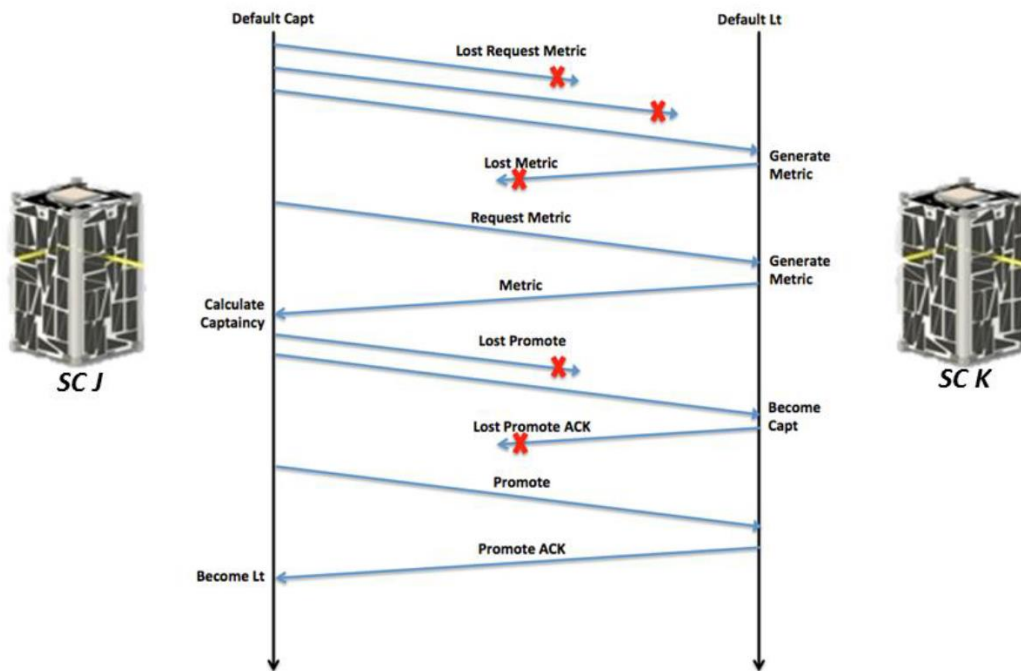


Figure 10 A timeline of the Captaincy negotiation process carried out between the two Nodes spacecraft. Image Credit: NASA Ames Research Centre

Over the course of its three week mission, Nodes completed and/or exceeded all five of its mission objectives:

- The collection and receipt of five ‘sets’ of science data
- Five successful S2G session
- One successfully executed indirect command
- Two successful captaincy negotiations
- The collection and receipt of 20 days of spacecraft state-of-health.

Of a total 470 science packets generated (science undisclosed) a total of 356 were successfully received at ground, ~25% packet loss. Five successful negotiations were carried out and 165 commands were executed by Nodes craft.

Nodes mission designers lay out numerous desirable enhancements: improved clock synchronization, inter-sat ranging, multi-hop routing, further acknowledge systems and/or delay tolerant networking principles, multiple ground stations and the interlinking of multiple Cpts to form clusters of clusters. Several of these suggested areas of future work are addressed in the protocols proposed and simulated in this work.

The CNSA's Tianwang-1 (TW1) mission, also referred to as STU-2, was a three CubeSat CSN mission involving numerous commercial and academic entities lead by the Shanghai Engineering Centre for Microsatellites (SECM). The majority of the published work relating to TW1 details its ADCS and novel propulsion systems [45, 58]. A presentation by Wu et al. during the 30th Annual AIAA/USU Conference on Small Satellites in 2016 offered a brief overview of the mission's Gamalink communication system [39]. However, as discussed, the details of the proprietary Gamalink technology are closely guarded. For the known details on Gamalink see section 2.1.1.2. In comparison to EDSN and Nodes, there is little information available regarding TW-1.

TW-1 was launched in late September of 2016, three months after the deployment from the ISS of the Nodes mission. TW-1 was, like Nodes, primarily a technology demonstration mission. TW-1's objectives were to flight test Gamalink, a novel ADCS and a novel propulsion (orbital control) system. TW-1 consisted of one 3U CubeSat (TW-1A) and two 2U satellites (TW-1B/C). TW-1A housed the experimental propulsion systems. This propulsion system allowed TW-1A to remain within range of TW-1B for a time far longer than would have been possible with an uncontrolled orbit. TW-1 collected data on

aircraft flight patterns using an on-board Automatic Dependent Surveillance (ADS) receiver. TW-1 also performed earth observation by using visual spectrum cameras to image polar regions. In line with its objective, TW-1 also carried out a number of in-orbit tests on its experimentation sub-systems. S2G communication of the mission's various results was demonstrated at a rate of 125kbps, far beyond the rates achieved by Nodes. Apart from this figure, there is no further relevant information available regarding TW-1's energy or communication capabilities.

There are several other missions, besides EDSN, Nodes, and TW-1, that can provide insight into the state of the art of CSNs: ESA's AIM COPINS [59], GomX-4 [60, 61] and Proba-3 [44], NASA's CPOD [62] and TROPICS [63], QB50 [64] and OLFAR [54]. With the exception of COPINS which was defunded, these missions are, at the time of writing, in development or awaiting launch.

2.2 Terrestrial Communications

Inspiration for the design of CSN communication protocols can be drawn from several terrestrial fields of research. In this section we examine two such fields, both of which bear numerous similarities to the domain of CSNs. These fields are Wireless Sensor Networks (WSNs) and Mobile Ad-Hoc Networks (MANETs). Both fields have considerable breadth and depth, as such a focus is placed on survey and review style publications. More focused publications are discussed in the subsequent "CubeSat Communications" section.

The following sections on WSNs and MANETs attempt to identify the most relevant sub-domains within each field. Within the field of WSNs works relating to data collection and energy conservation are of most interest. As discussed the primary application of CubeSat is sensing which leads to obvious similarities to terrestrial WSNs. MANET related works

are relevant in their treatment of the mobility of network members. Particular attention is paid to Flying Ad Hoc Networks (FANETs). FANETs share many of the same properties as CSNs. Like CSNs, FANETs are expected to experience intermittent, potentially predictable, access to a greater and more ‘static’ network. In the case of CSNs, this static network is represented by one or more ground stations.

2.2.1 Wireless Sensor Networks

Common WSNs challenges relate to the unpredictable or intermittent failure of network elements and resource and/or capability constraints. These challenges impact, to varying degrees, on a WSN’s ability to perform data collection or data dissemination. The sink (collection) or originator (dissemination) of data in the context of CSNs is the earth based ground station. In this work we are concerned with energy efficient data collection (the PvTP trade-off) with the added complication of node (CubeSat) mobility.

A survey by Rault et al. focuses on energy efficiency in WSNs [65]. The authors approach the domain by examining a number of areas of WSN application such as healthcare, transportation and industry. For each area the authors outline characteristic WSN application requirements such as scalability, mobility, security and so on (**Error! Reference source not found.**). Table 1 extends this table with a number of application cases relevant to CSNs.

Table 1 An extension of Rault et al.’s table shown in **Error! Reference source not found.** to include a number of CSN applications. ET – Extra-Terrestrial. ET Science examples: measuring solar radiation, performing astronomical measurements etc.

		Scalability	Coverage	RT Delay	QoS	Security	Mobility	Robustness
Space	CSNs – Earth Observation	-	+	--	+	-	++	+
	CSNs – ET Science	--	-	--	+	-	+	+

Rault et al. address several aspects of low-power WSNs and the trade-offs required when attempting to increase energy efficiency. Several low power WSN standards are explored such as IEEE 802.15.4 [66], Bluetooth Low Energy (BLE) [67], and Routing Protocol for Low power and lossy networks (RPL) [68]. No one standard presents itself as being an obvious choice for the CSN PvTP trade-off. However, clear benefits can be seen in elements of RPL and ISA100.11a [69], one of many extensions of 802.15.4.

Beyond low power WSN standards, several core WSN approaches are highlighted by Rault et al. Although all of these areas are relevant in some regard to CSNs, the areas of highest relevancy to this work are: Duty cycling schemes, cluster architectures, energy as a routing metric and sink mobility. Each of these areas are explored, to some extent, in the Proposed Protocols chapter.

Rault et al. discuss a number of approaches to the trade-offs involved in the implementation of energy efficient WSNs. Three techniques are discussed: Multimetric protocols, Cross layer approaches, or cross layer optimization (CLO), and Multi-objective optimisation (MOO). Of the three techniques, CLO bears the most relevance to CSNs.



Figure 11 An illustration of example information that may be passed between layers. This is a departure for the treatment of each layer as an isolated black box.

The authors define CLO as “solutions exploit(ing) interactions between different layers to optimise network performances” (Figure 11). Here ‘layers’ may refer to several communication models, but it can be safely taken as relating to the OSI reference model [57]. This work adopts Rault et al.’s definition of CLO from here on. Two particular surveys are cited as being authoritative on the WSN CLO domain [70, 71]. Rault et al. highlight the communication of radio energy consumption by the Physical layer to the MAC and network layers as an example of the application of CLO. Other examples given include: Communicating clustering formations achieved at the MAC layer and to higher layers and the explicit scheduling of communications at higher layers which allows lower layers to reduce activity.

No one approach or standard is singled out as the obvious choice for energy efficient WSNs. Rather, Rault et al. point to the importance of adapting approaches to applications. The authors state clearly that regardless of the application or approach, the introduction of greater CLO is expected to advance the field. CLO is a common and favoured theme throughout recent literature relating to WSNs and MANETs. The exact meaning of CLO differs somewhat depending on the author, application and domain. Despite this, it is clear that it is an active topic in both WSN and MANET energy efficiency research.

WSN data collection is examined by Francesco et al. in extensive survey paper published in 2011 [72]. The work is particularly useful as it focuses on WSNs with Mobile Elements (WSN-MEs). The authors concentrate on mobility while maintaining and referencing the existing relevant state of the art in WSN routing and energy management.

Francesco et al. lay out a number of WSN mobility scenarios the most relevant of which is the “Mobile Peer” scenario (Figure 12). With CSNs, there is no appreciable difference whether one considers the ground station as moving into range of a satellite or a satellite

moving over ground. In either case, a network of satellites in some formation must contend with a finite but generally predictable communication window with ground.

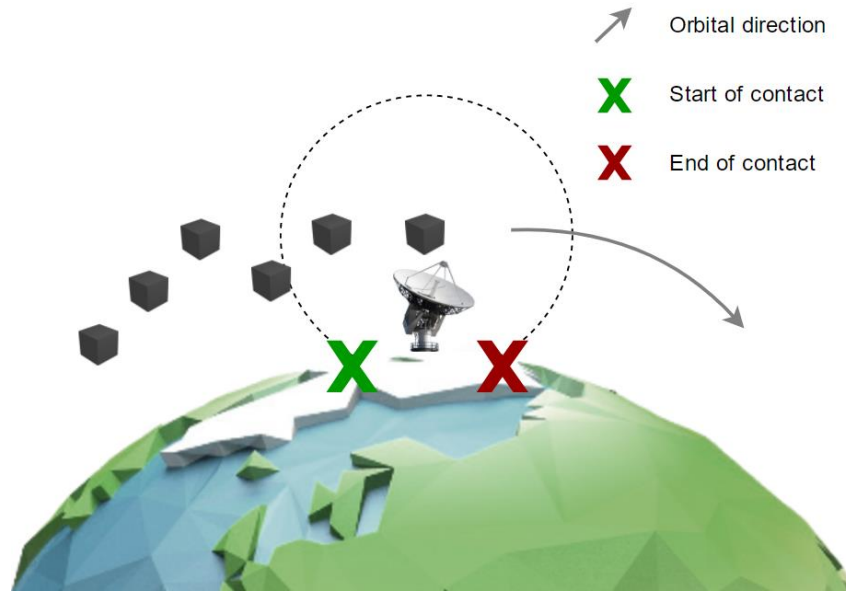


Figure 12 A mobile peer architecture bears clear similarities to CubeSats in orbiting passing over a ground station.

Data collection is divided by Francesco et al. into three distinct phases: 'Discovery', 'Routing', and 'Data Transfer'. In the discovery phase network members (nodes) attempt to identify their contactable neighbours. During the routing phase, nodes attempt to establish routes to unreachable nodes through their neighbours. Finally, given an established route or data destined for a neighbour, a node can begin the communication of data which constitutes the data transfer phase.

Francesco et al. authors outline that the treatment of the discovery phase is critical to the performance of WSN-MEs. The authors highlight issues with discovery protocols which rely on schedule based discovery of reachable nodes. The timeliness of the discovery carried out by a mobile element determines the maximum windows for communication. Schemes based on periodic or scheduled discovery attempts are liable to suffer from

reduced communication windows due to ill-timed discovery attempts. Francesco et al. highlight an approach that utilizes low-power short-range radios to asynchronously waken nearby nodes prior to data communication using a longer range higher power radio [73]. Other methods involving learning are noted for their potential to avoid the pitfalls of periodic/scheduled listening. Such methods may, with or without prior heuristics or topology knowledge, converge on optimal discovery schedules over time [74]. In the case of CSNs, CubeSats can easily determine their next communication window with ground through the use of GPS time, position and velocity. However, discovery of S2S windows closely mirrors the challenge of WSN mobile element discovery.

The data transfer phase, which follows the discovery phase, is primarily concerned with the quality of communication and MAC schemes. The authors state that WSN-ME data transfer is a field that requires further work. The authors note a stop-and-wait protocol [75] as well as an automatic repeat request (ARQ) scheme [76]. Further references to specific well-established MAC schemes are sparse. Francesco et al. make a clear recommendation that network coding schemes require greater attention in relation to WSN-ME data transfer [77]. MANETs and WSN-ME bear numerous similarities. Work on data transfer within the domain of MANETs tends to be broadly applicable to WSN-MEs. Why Francesco et al. avoid the direct evaluation of relevant MANET related work is unclear. It is possible the authors intend to stress the importance of approaches specifically adapted for WSNs.

Routing in WSN's, as highlighted by Francesco et al., is considerably more developed than discovery or data transfer. The authors make an assumption that the motion of network elements is not controlled. In this regard, the works discussed are relevant to S2S

communications of CubeSats which are assumed to have no orbital control capabilities. Routing for uncontrolled mobile WSN elements is classified by the authors into flat routing and proxy-based routing. In flat routing schemes all nodes behave in the same fashion whereas in proxy-based schemes certain nodes may take on routing or proxy style roles.

Several approaches to routing are discussed by Francesco et al. Three are worth noting in brief: A modified Optimized Link State Protocol (OLSR+) [78], Energy-Aware Routing to Mobile gateway (EARM) [79], and a cluster based approach by Somasundara et al. [80]. OLSR+ adapts to the mobility of element by passing velocity information between nodes. With this added information nodes can estimate the future stability of an inter-element link and avoid unnecessary abandonment or predict route switches. EARM implements an adaptive approach wherein nodes may boost their transmission power to a mobile node as the node moves out of range. EARM's core logic allows nodes to decide the point at which a direct link should be abandoned in favour of a multi-hop route. Somasundara et al.'s approach employs adaptive clustering. Nodes collaboratively and dynamically form clusters and elect cluster heads. These heads are elected based on their distance from some mobile sink. Cluster heads act as routers to the mobile sink and manage inter-cluster communication. Somasundara et al.'s approach is similar to the cluster based MAC approach developed by this work. Although, this work does not employ a cluster based routing strategy such as the Cluster Based Routing Protocol (CBRP) [81] (Figure 13). Rather, this work adapts a routing protocol known as the "Dynamic MANET On-demand routing protocol" (DYMO) [82] for use with a cluster based MAC approach.

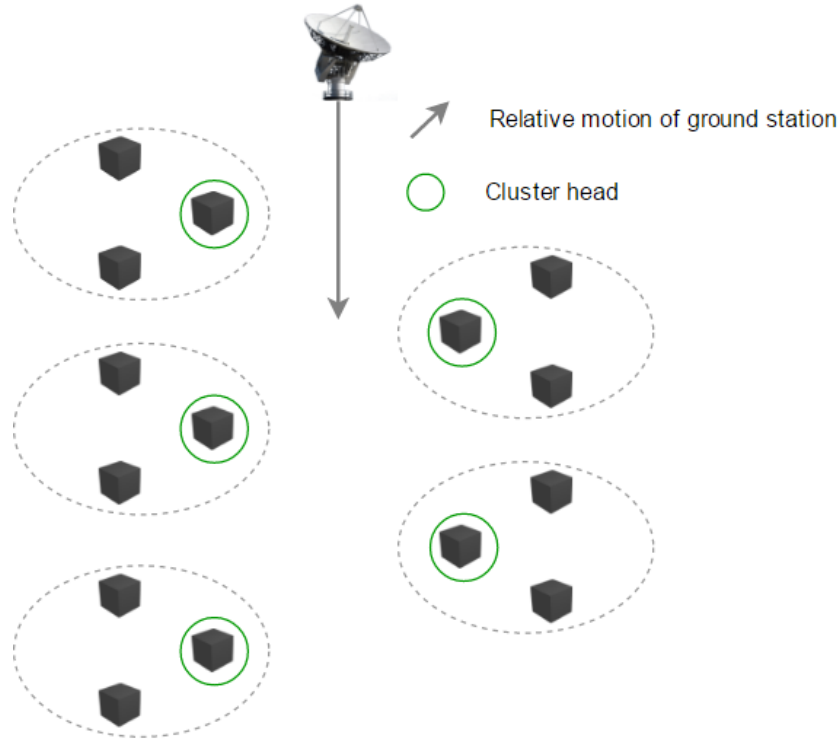


Figure 13 An example of network cluster forming with the election of cluster heads. With the introduction of a mobile sink (ground station), clustering can be adapted to insure election of cluster heads with longest contact duration and/or most resources [80].

Finally, to compliment the discussed WSN survey materials a more targeted work is considered. "Efficient data collection in wireless sensor networks with path-constrained mobile sinks" by Gao et al. approaches the PvTP trade-off directly through the design of novel routing protocol [83]. Increasing the relevancy of this work to the CSN domain is the focus by Gao et al. on WSN's with mobile sinks. Each mobile sink can be considered analogous to a CSN ground station. The proposed protocol is given the name "Maximum Amount Shortest Path" (MASP). The authors determined the formal properties of MASP and used OMNeT++ to simulate and analyse its performance.

MASP outperforms a common approach to the determination of the most efficient routes to adopt called Shortest Path Tree (SPT). There are several implementations of SPT, but a

common approach is to construct a tree of possible routes to a destination. This tree can then be search with an algorithm such as A* with a heuristic cost function which represents the energy cost of using a given route. MASP outperforms SPT by utilizing a genetic algorithm which solves a multi-dimensional optimization problem. This problem is based on known and computable routes and route heuristics.

The work of Gao et al. represents a state of the art approach to the PvTP trade-off in WSNs. Energy as a routing metric was clearly noted by Rault et al. as critical when attempting to optimize data collection with energy efficiency. MASP develops on this concept of energy as a routing metric and introduces a complex, yet performant, solution. However, MASP can only determine efficient routes given successful and timely discovery. Without the introduction of an appropriate discovery scheme MASPs performance is fundamentally limited. This lack of development in the area of discovery is further discussed by Francesco et al.

From the analysis of WSN based prior art one can identify several similarities to the CSN PvTP trade-off. In the area of energy efficient data collection, there are clear recommendations for further development of CLO and node discovery. Although the CSNs diverge in certain regards from WSNs, prior art therein provides a considerable basis for the development of the CSN domain.

2.2.2 Mobile Ad-Hoc Networks

MANETs bear obvious similarities to CSNs. MANET research tends to be less concerned with the constraints, such as power and bandwidth, placed upon network members. Rather, a focus is placed on an approach's ability to efficiently and reliability enable self-organization and communication.

In MANETs no initial shared knowledge of the network is assumed. With CSNs, although several CubeSats may be deployed together, countless factors could cause unpredictable failures or orbit perturbations. LEO is an environment of extremes with intermittent flares of radiation, hazardous solar winds, fluctuating magnetic activity and temperatures ranging from -170 to 150 degree Celsius. The MANET technologies which will prove most valuable are those which are mature, adaptive to topology changes and energy efficient.

The most discussed and active topic within MANET research is that of routing. MANET routing protocols are generally divided into three primary classes: reactive, proactive and hybrid. Reactive protocols attempt to establish routes only as required whereas proactive protocols attempt to establish routes in-advance of communication. Hybrid protocols implement a mix of reactive and proactive approaches, generally by restricting reactive or proactive behaviours to certain areas of a network. The methods by which protocols maintain and discover routes may or may not differ between protocols of differing classes.

This section places a focus on reactive protocols. Generally, proactive and hybrid protocols are unsuitable for CSNs. Proactive protocols are typically designed for low mobility networks with reliable links. Hybrid protocols can provide a 'best of both worlds' between reactive and proactive protocols. However, hybrid approaches are typically best suited for larger networks where the additional overheads of more complex protocols scale more favourably.

Several routing protocols are examined by Mohseni et al. in their survey of MANET routing protocols [84]. The conclusions of the authors reinforce the assertion that reactive protocols are better suited for CSNs. The authors state that, comparatively, proactive protocols tend to require more power, bandwidth and incur larger overheads. The

primary benefit of proactive protocols is the constant availability of routes which lowers latency and increases the consistency of communication throughout the network.

Mohseni et al. discuss a number of the most well-known reactive protocols such as “Dynamic Source Routing” (DSR) [85], Ad hoc on Demand Distance Vector (AODV) [86], Temporally Ordered Routing Algorithm (TORA) [87] and CBRP [81]. Of these it is worth providing a brief overview of DSR and AODV. In DSR, a node (the originator) broadcasts a route request (RREQ) for a given target. Non-target nodes add their address to an incoming RREQ packet and rebroadcast it. Loops are avoided by nodes dropping RREQ packets to which they have already added themselves. Once an RREQ reaches its target it contains a list of nodes representing one possible route from originator to source. An attempt is then made to use this route to send a unicast route response (RREP) from the target to the originator. Through this approach, DSR is capable of building up multiple routes for a given target. In DSR, due to the construction of routes within packets, packet sizes grow in proportion to overall network size.

AODV builds upon concepts from DSR and a proactive protocol known as Destination-Sequenced Distance-Vector routing (DSDV) [88]. Unlike DSR, AODV only specifies that the originator and target addresses must be contained within a route packet. AODV borrows route packet sequence numbers from DSDV in order to avoid infinite loops and determine the ‘freshness’ of a route. AODV also introduces features such as ‘intermediate-RREPs’ and ‘Hello’ messages. An intermediate-RREP may be generated by an intermediate node in response to incoming RREQs if said node has a route to for a given target. ‘Hello’ messages may be periodically generated in order to detect broken routes and maintain up to date route costs.

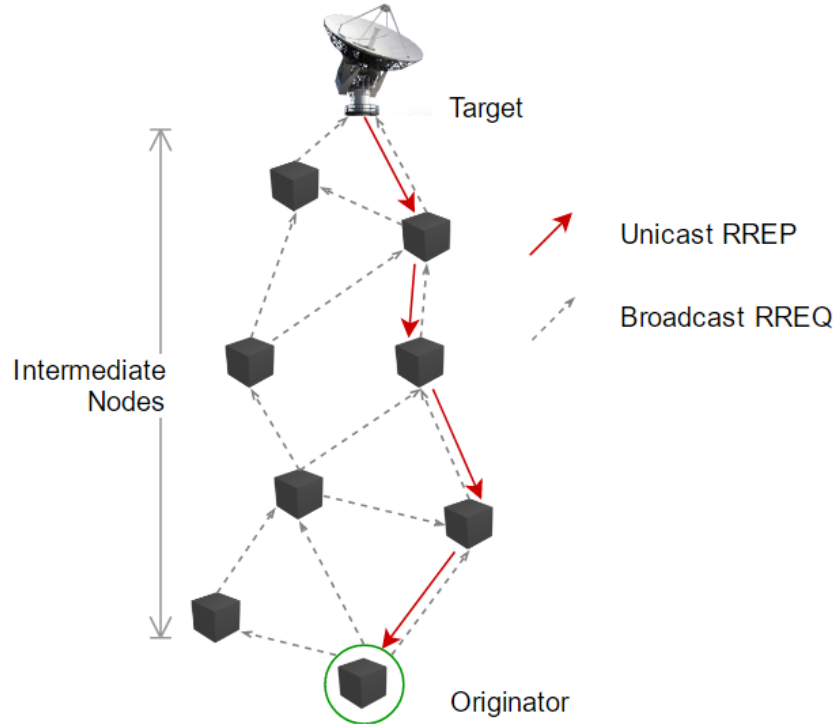


Figure 14 An illustration of the broadcast RREQ unicast RREP approaches used by DSR and AODV. An added feature above allows any node receiving a route message to add a 'reverse' route entry for the message's last hop.

The routing protocol utilized by this work, DYMO, is a modification of AODV. In fact, recent IETF specification drafts have begun to refer to DYMO as "AODV version 2". This work's implementation of DYMO within the network simulator OMNeT++ is based on an older IETF specification draft [82] which uses the DYMO naming convention. The primary differences between DYMO and AODV relate to implementation. The core route discovery and maintenance approaches of AODV are largely unchanged by DYMO. Details of DYMO are discussed further in the Proposed Protocols chapter.

Vehicular Ad-hoc Networks (VANETs) are a sub-domain of MANETs. There are notable parallels between CSNs and VANETs such as the separation of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) concerns. Despite initial similarities, a survey of the

protocol stacks in the domain by Mohammad et al. reveals several undesirable facets of VANETs [89]. As mentioned, there is generally less focus in the field of MANETs placed on resource constraints. VANETs follow this trend as the majority of applications focus on planes, trains, and automobiles [90]. Also, security is a central topic in recent VANET research. Although security has relevance to CSNs, secure communication is beyond the scope of this work.

Mohammad et al.'s survey shows the strong preference of VANET protocol stacks towards mobile IPv6 capable technologies and flavours of the 802.11 stack. Implementing a stack which supports IPv6 enabled RPL [68] for CSN communications is potentially feasible. However, in broad terms, the power use, overheads and low mobility basis of 802.11 flavours are generally unfavourable. Mohammad et al. point to C2CNet (Car-to-Car Net) as the emerging state of the art standard within VANETs. C2CNet's protocol stack makes heavy use of 802.11 standards.

The field of FANETs (Flying Ad-Hoc Networks), a sub-domain of VANETs, is more relevant to CSNs. A survey by Bekmezci et al. [91] introduces FANETs in the context of MANETs and the VANETs. The authors deal primarily with unmanned aerial vehicles (UAVs). Capability and resource constraints are more central within the field of FANETs. This is particularly true where long haul UAVs and small scale drone based applications are concerned. Bekmezci et al. highlight the departure from the use of standard 802.11 MAC layer protocols for FANET applications as the field has developed. The authors point to the development of tailored FANET MAC protocols, many of which aim to take advantage of advancements in directional antennae design. At the network layer, both DSR and a modified AODV [92] are noted along with other well-known MANET routing protocols such as GPSR [93] and OLSR [94]. Interestingly, due to the use of time slots for

route discovery, the aforementioned modified AODV protocol has similar properties to the combination of protocols proposed by this work.

The work of Bekmezci et al.'s provides an overview of FANETs which details the use of customized MAC protocols and mainstream MANET protocols. One FANET related project cited by Bekmezci et al. called the Cooperative Autonomous Reconfigurable UAV Swarm (CARUS) project [95] is worth mentioning briefly. The project focuses more on application layer coordination and formation flying. Due to several notable parallels with CSNs the project merits investigation for future developments on CSN formation flying and cooperative observation.

Bekmezci et al. place value on cross-layer architectures (CLO) within in the field of FANETs. CLO, as discussed, is an important topic within WSNs and its applicability within FANETs further reinforces its importance to the future of CSNs. Bekmezci et al. note works that take advance of cross-layer clustering and scheduling and the cross-layer sharing of attitude and antennae related information in order to improve performance [96, 97].

The state of art in MANETs provides an insight into several potential routing protocols which may be employed in addressing the CSN PvTP trade-off. It is clear that the FANET sub-domain contains the highest relevancy work. Overall, examining MANET prior art augments and reinforces the findings relating to WSNs. Such prior art provides important context to the existing state of art in CubeSat communications and this work's proposed protocols.

2.3 CubeSat Communications

Prior to the development and flight of the first CSN related mission several publications examined the inter-communication and networking of CubeSats. Most notably Challa and McNair of University of Florida provide explorations of distributed applications implemented upon CSNs [36, 98-101]. These works are out of the scope of this project as they deal more with applications running upon CSNs rather than communications within the CSN itself.

The most relevant work in this area is a survey by Radhakrishnan et al. [9]. This survey provides the primary source for the exploration of the CubeSat communications state of the art. This work proposes MAC (data link layer) and routing (network layer) protocols to address the PvTP trade-off. Radhakrishnan et al. detail several relevant works relating to MAC, routing and energy efficiency.

Radhakrishnan et al. provide an overview of some of the common terms used when referring to CSN formations. A ‘trailing’ formation, sometimes referred to as ‘leader-follower’, involves a single orbit chain. A ‘cluster’ formation has a broader mean. “A cluster of satellites” generally implies a collection of satellites in multiple orbits which maintain some topology or formation. The term ‘constellation’ is often misused. A constellation formation focuses on coverage of the earth’s surface. Communications and GNSS constellations typically seek to achieve complete coverage or ‘visibility’ of the earth’s surface. The term ‘swarm’, is also often misused. A swarm is not a satellite formation in the same sense as a cluster or constellation. To quote Sundaramoorthy et al. “a satellite swarm is a group of identical, minimal, self-organised (self-functioning) satellites in space that achieve a common objective with their collective behaviour” [102]. Radhakrishnan et al. adopt the same definition as Sundaramoorthy et al. Similar to the

concept of a swarm is the concept of ‘fractionated’ satellites wherein “the functionalities of a single large satellite are distributed across multiple modules, which interact using wireless links” [102] (Figure 15). In this work we consider CSNs which fly in cluster formations and operate as swarms.

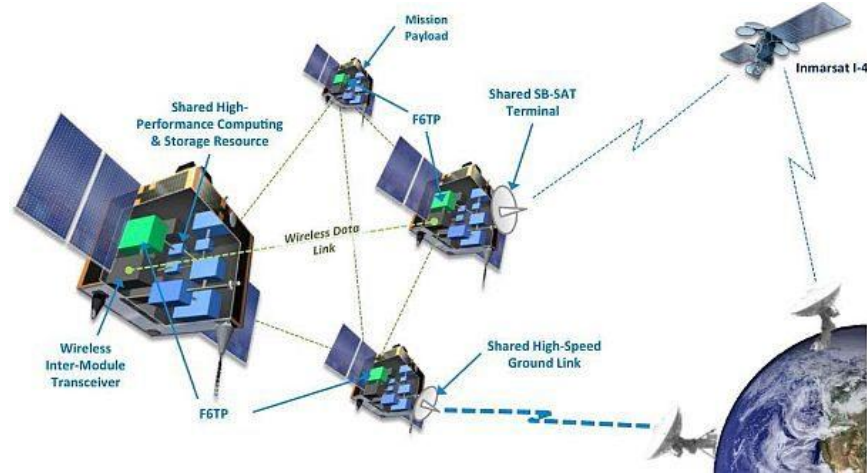


Figure 15 DARPA fractionate satellite concept. It can be seen that the mission payload exists independently of other core systems such as compute and communications. Image Credit: DARPA

2.3.1 Physical Layer

Although the physical layer of OSI model is not of core interest in this work it is nonetheless worth noting some of the findings of Radhakrishnan et al. When referring to the maximizing data rates Radhakrishnan et al. recommend the increase of bandwidth rather than reducing the signal to noise (S2N) ratio. They also cite that higher S2G data rates can be achieved by transmitting in bursts rather than continually [103]. This intermittent S2G communication approach is adopted by this work. In terms of modulation and coding schemes employed at the physical layer, Radhakrishnan et al. cite Binary Phase Shift Keying (BPSK) as the current state of the art for small satellites. Quadrature Phase Shift Keying (QPSK) and offset-QPSK are noted for potential future

development provided additional bandwidth balances out increased power requirements.

In comparison to other areas of CubeSat communications, there has been notably more work published relating to antenna design. Radhakrishnan et al. point to Gamalink [41] for its use of an advanced antenna. Unfortunately, no further information regarding Gamalink is offered. Single patch S-band (2 – 4 GHz) antennae are highlighted as the current state of the art. Also, the authors state that “a maximum distance of 1000 km between satellites can be achieved using a 3 W transmit power” using UHF (300 MHz and 3 GHz) radios [104]. This assertion seems dubious considering the work previously discussed in relation to CubeSat communication capabilities. Radhakrishnan et al. express doubts regarding the applicability of complex MIMO and multi-patch antenna, instead recommending the use of multiple simple antennae. Radhakrishnan et al. state that links between satellites are typically full duplex typically favouring Time Division Duplex (TDD) over Frequency Division Duplex (FDD).

2.3.2 Data Link Layer

Thus far the data link, or simply ‘link’, layer has been referenced in relation to Medium Access Control (MAC). Although the focus in this work will continue to be placed on MAC protocols, it is worth noting some of the other duties of the link layer such as: framing, synchronization, error control, flow control and MAC addressing. MAC protocols determine how multiple agents share a common radio medium. As such, MAC protocols have a considerable effect on energy efficiency, network scalability, channel utilization, latency and throughput. There are two main classifications of MAC protocol: contention based and contention free. Contention based protocols such as Carrier Sense Multiple Access (CSMA) rely on detecting when the medium is in use and when two

signals have collided on the medium (collision). Contention free protocols seek to completely avoid the need to detect medium use or collisions. Such protocols generally achieve this by allowing multiple agents to communicate at once without collision such as with Code Division Multiple Access (CDMA) or by logically dividing the medium such as with Time Division Multiple Access (TDMA) or Frequency Division Multiple Access (FDMA) (Figure 16).

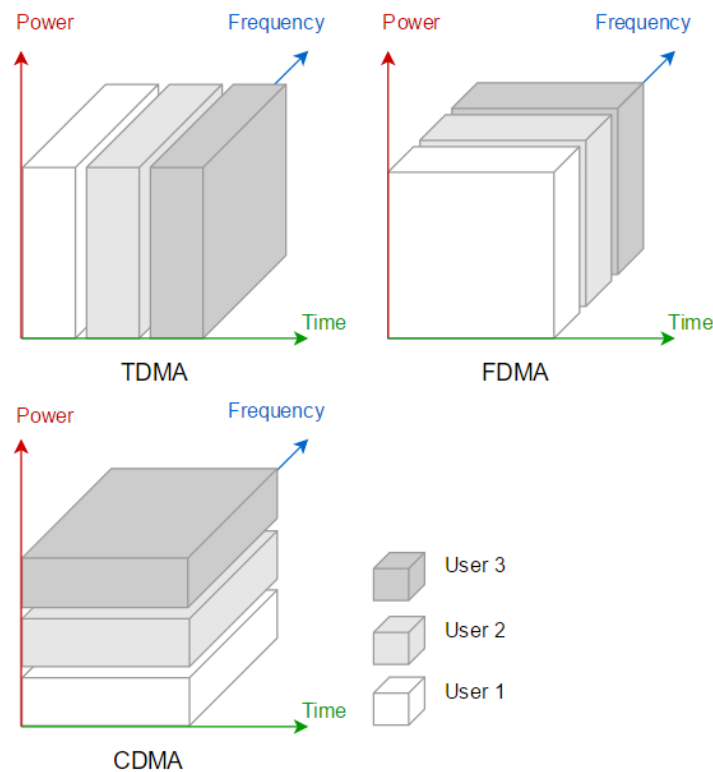


Figure 16 A comparison of common contention free MAC schemes. In CDMA a 'chip' is used to insure that signal on the medium are orthogonal and therefore cannot collide.

Several MAC protocols are discussed by Radhakrishnan et al. in relation to small satellite communication. One approach attempting to adjust the IEEE 802.11 physical and MAC standard for communications in LEO [105]. As discussed in relation to VANETs several aspects of 802.11 based standards are unsuitable for space based communications. The

modified 802.11 based approach addresses issues relating to inter-frame spacing (IFS). In LEO networks propagation delays may be in the order of milliseconds and can often be difficult to predict prior to communication. Using known propagation models and GNSS based information the modified 802.11 approach adjusted contention windows to fit communication delays. The distributed adjustment of IFSs and contention windows introduces considerable complexity at scale. The designers of this modified 802.11 approach assert its feasibility for use in LEO communications. However, it is this Author's opinion that the modifications introduced solve a problem left-over from terrestrial communications that need not exist in the first place.

A work, led by Radhakrishnan, explores the use of a CSMA style MAC protocol [106]. A protocol using CSMA with Collision Avoidance (CSMA/CA) is examined. The proposed protocol makes use of control packets to address problems such as the hidden node problem. In the hidden node problem some node B can hear nodes A and C. However, nodes A and C cannot hear one another. Both A and C may see the medium as free and attempt to communicate to B. Request To Send (RTS) and Clear To Send (CTS) packets control packets can ameliorate this issue. A and C can avoid communication with B until receipt of an appropriate CTS packet from B. In their assessment of this CSMA/CA approach Radhakrishnan et al. conclude that the protocol is best applied in situations with low frequency communications within tightly grouped formations. Generally, CSMA and other contention based schemes are liable to introduce unnecessary and wasteful communications adversely affecting power consumption.

Researchers at the University of Delft propose a CDMA based MAC protocol for use in "Precision Formation Flying" (PFF) missions [107]. The proposed protocol employs a form of half-duplex CDMA which allows for the networks to adaptively scale and reconfigure

as new members are introduced. The CDMA scheme was shown to have adverse effects on the ranging and navigation functions required for PFF missions. As such, the protocol's designers recommend the use of adaptive transmission power control mechanisms. PFF missions require high frequency low latency communications. Given this, CDMA may be the best approach. However, for the CSN PvTP trade-off, the additional power requirements of CDMA are not matched by obvious benefits in throughput.

Radhakrishnan et al. discuss a protocol proposed by Chen et al. called "Load Division Multiple Access" (LDMA) [108]. LDMA is a hybrid MAC protocol which utilizes a mix of TDMA and CSMA in an attempt maximize channel (medium) utilization. LDMA allows network elements to operate in two different modes High Contention Level (HCL) and Low Contention Level (LCL). HCL mode is used in response to high levels of communication which may result in numerous collisions and vice versa for LCL. In HCL mode TDMA is used. TDMA protocols rely on a schedule of time slots shared among network members (Figure 16). Each time slot within a schedule is typically assigned an owner. During a time slot only the slot owner may transmit. There are many different flavours of TDMA. For instance schedule and slot ownership may be fixed or may be negotiated between nodes in a distributed manner. LDMA uses a fixed TDMA scheme.

In LCL mode, a version of CSMA is used. Nodes in LDMA may be in differing modes at any one time. As such, when a node is in LCL mode it gives priority to the owner of the current time slot whenever collisions are detected on the medium. In response to collisions nodes will generate conflict frames. Nodes switch from LCL to HCL based on the number of conflict frames received. Nodes will revert to LCL mode as their communication demand drops and the number of overhead conflict frames reduces.

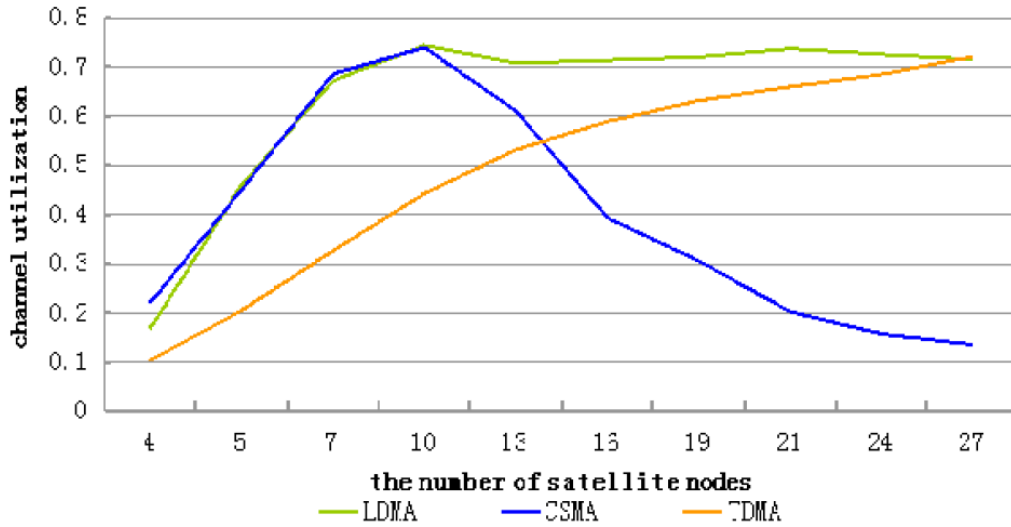


Figure 17 Channel utilization (Vertical axis), measured from 1.0 (100%) to 0, compared to network size (Horizontal axis) for LDMA, pure CSMA and pure TDMA. Image Credit: [108]

Channel utilization generally refers to the percentage of time for which the common radio medium is used for communication of data. The communication of data in this case being distinct from the communication of protocol control information. Channel utilization is best measured under steady state heavy loading conditions wherein a node always has a data packet queued to send. When correctly measured channel utilization is a key indicator of the overall throughput performance of a network. Chen et al. compared LDMA to pure TDMA and CSMA protocols through simulation. One graph representing their findings is shown in Figure 17.

LDMA presents itself as strong candidate for use in attempting to optimize throughput for CSNs. As communication activity falls the protocol mirrors a pure CSMA approach and it follows that the protocol approaches pure TDMA as activity increases. The power consumption profile of the protocol merits further investigation. The protocol requires nodes to be promiscuous in order to overhear conflict frames and mode change broadcasts. Leaving radios constantly in receiver mode will incur a power consumption

penalty over time. In comparison, certain TDMA protocols allow nodes to completely sleep their radios for periods under certain conditions.

Two further hybrid approaches are discussed by Radhakrishnan et al.: An FDMA/TDMA (F/TDMA) hybrid and a CDMA/TDMA (C/TDMA) hybrid. The F/TDMA protocol, based on WiMedia [109] and proposed by Heidari et al. is worth mentioning briefly. The protocol introduces the distributed management of heterogeneous network state as well as two-dimensional super frames in place of TDMA time slots. The C/TDMA protocol takes a cluster based approach to the division of CDMA and TDMA [110]. Comparatively the C/TDMA protocol provides many of the same properties as the aforementioned F/TDMA protocol without the prohibitive levels of complexity.

The C/TDMA protocol was chosen as the starting point for the MAC protocol proposed by this work. As such, the details of the protocol will be discussed at length in the Proposed Protocols chapter rather than here. The protocol was chosen as it makes explicit allowances for energy awareness and the improvement of throughput. For a given scenario involving a trailing formation, simulations of the protocol show C/TDMA outperforming the previously discussed CSMA/CA approach (Figure 18). The protocol's formation of clusters is based on the energy available to nodes. Nodes with higher levels of remaining power are elected as cluster "Masters" which act as routers between clusters. This allows other nodes within the cluster to conserve energy.

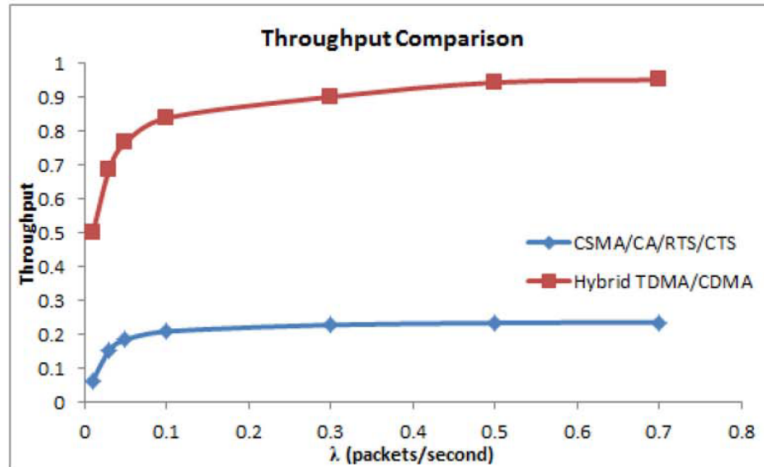


Figure 18 ??? Throughput here is measured as the amount of time spent transmitting data divided by the amount of time available to transmit data. Considering the findings in Figure 17 It is likely that the experimental setup and the traffic simulated naturally favoured the T/CDMA protocol. Image Credit: [110]

2.3.3 Network Layer

The primary responsibility of the network layer is routing, although packet forwarding and address handling are also important activities. Routing protocols affect the discovery and selection of optimal routes within a network. There are several generally applicable approaches to the discovery of routes discussed in the context of MANETs. Unlike discovery, the selection of optimal routes is often highly application dependent. For instance, in the context of this work, there may be benefits in avoiding routes which rely on nodes with a low amount of remaining power. The approaches to determining and maintaining optimal routes differentiate many of the routing protocols proposed for small satellites. It should be noted that there has been less published work relating to the routing protocols than to MAC protocols. As evident from the Nodes and Tianwang missions, the current state of the art for CubeSats are small one-hop networks. Such networks don't need to perform route discovery or chose between multiple routes. As such, the network's

performance is predominantly determined by layers below the network layer, such as the link layer.

Radhakrishnan et al. discuss a number of routing protocols in their survey of inter-satellite communication for small satellites [9]. The authors highlight routing approaches which have been adapted in the past for use with larger satellites such as the Border Gateway Protocol (BGP) [111] and Multi-Layered Satellite Routing (MLSR) [112]. Unfortunately, the authors make no comment on the suitability of these approaches for small satellites. This is a theme throughout the authors' discussion of routing protocols which reflects the scarcity of relevant research in this area. For instance the authors discuss delay tolerant networking (DTN) at length. DTN has been proposed recently by some for small satellite applications but no notable works on this have yet to be produced.

An approach proposed by Bergamo et al. involves each node classifying its neighbours as either 'new' or 're-occurring' [113]. The approach is similar to that of AODV however the classification of neighbouring nodes reduces the overall frequency of route discovery. The protocol mainly focuses on synchronization of larger satellites with regular control orbits. However, this protocol may have benefits in situations involving CSNs composed of multiple swarms in disparate orbits.

Of the routing protocols discussed by Radhakrishnan et al. few are obviously suitable to CSNs and none deal directly with the balance of energy efficiency and throughput. As such, prior art relating to MANETs and WSNs informs the choice of proposed routing protocol.

Radhakrishnan et al. recommend further work on cross-layer optimization (CLO) and the introduction of protocols which adapt naturally to predictable topology changes. No recommendation that further work be carried out on routing is provided however, there

is clear need for such work. Radhakrishnan et al.'s survey is a valuable resource when approaching the domain of CubeSat communications. The authors cover the physical and data link layers well but fail to discuss and identify the gaps in relation to network layer protocols. Also, the authors provide considerably more content for works on which one or more of the survey's authors were involved which may indicate a degree of bias.

2.3.4 Other Works

Wong et al., operating mainly out of NASA's Goddard Flight Centre, examine a potential future for CSNs. Wong et al. propose that space to ground communications be performed indirectly through relay with existing space bound communication networks [10]. This concept is further explored for deep space missions in the preliminary development of the ESA's COPINS mission [59].

Another survey style paper on inter-satellite link for CubeSats by Budianu et al. [8] published in 2013 provides a somewhat out of date overview of the field. However, Budianu et al. pay more attention to antenna design and link budget analysis than Radhakrishnan et al.

2.4 Other Areas of Note

Alongside the primary areas of concern there are secondary areas which are deemed to be less relevant to this work. These secondary areas illustrate further the considerable context which must be considered when approaching issues relating to CubeSat networking and communications.

2.4.1 Energy Aware Scheduling

Energy aware scheduling is an active area of research in the terrestrial domain, especially in relation to WSNs [114]. Despite the growing popularity of small form factor satellite missions, there are few related notable publications in the domain. However, new insights into the area was produced in 2016 as a result of the technology demonstration focused GomX-3 mission [115]. GomX-3 was designed by private Danish company GOMSpace and flown by ESA. Following the mission's success mission designer published a work entitled "Battery-Aware Scheduling in Low Orbit The GomX-3 Case" [60]. The work outlines the approach taken to adaptively model and predict battery usage in order to produce a activity schedule which optimized power consumption. In relation to long duration CSNs, one might consider an energy conservation scheme which entails generating an optimal schedule for each CubeSat. Through distributed consensus these schedules could then be used in order to generate final desired activity schedules.

2.4.2 Delay Tolerant Networking

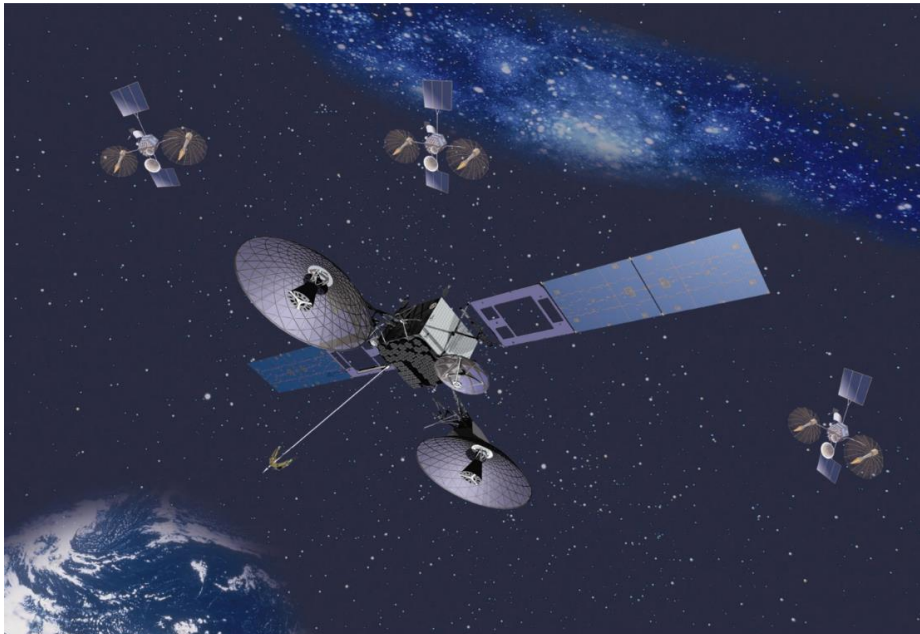


Figure 19 An illustration of several NASA Tracking and Data Relay Satellites (TDRS) which make the primary backbone of NASA's deep space network. Some of the most advanced space bound radios enable the relay of signals from spacecraft throughout the solar system. Image Credit: NASA

Interplanetary communication presents many novel problems. Delay tolerant networking (DTN) approaches have been employed successfully solved many of these [116]. A notable success case is NASA's deep space network [117] (Figure 19). Although CSN's don't face the same magnitude of challenges presented by inter-planetary communication many, including Radhakrishnan et al., point to DTN as important to the future of CSNs. DTN has the obvious advantage in this regard of being tested on-board larger satellites and developed by experts within the space industry.

Chapter 3: Proposed Protocols

One-hop network topologies are the current state of the art for CSNs. In such networks, once a node has discovered its neighbours there is no need to establish routes. Consider EDSN's "hub-and-spoke" topology and Cpt/Lt protocol. EDSN's approach involves no multi-hop routing. If CubeSats are not in range of one another they cannot communicate. Given this relationship, the performance characteristics of an S2S link are primarily determined by aspects of the physical layer and link layer. Section 2.31 briefly discusses the state of the art of CubeSat physical layer implementations, as assessed by Radhakrishnan et al. Addressing and modifying such implementations is beyond the scope of this work. This work's simulations are configured with idealized physical layers, the behaviour of which is considered immaterial to gathered results.

At the link layer, we are primarily interested in MAC protocols. Although, as discussed in section 2.3.2 there are other aspects of the link layer which may affect the PvTP trade-off. In any wireless network the method by which the medium is shared amongst nodes will impact various aspects of performance. Approaches such as TDMA and CSMA are commonly used for simple one-hop networks. Even in such simple networks there are numerous dimensions to consider. If nodes are closely grouped with each node having multiple neighbours then TDMA is likely to outperform CSMA due to its avoidance of collision. However, if the network is tightly clustered but nodes only sporadically generate traffic and contention is low then CSMA may outperform TDMA. The complications inherent in even the most basic of networks highlights that the choice of optimal MAC protocol is rarely obvious. This choice is affected by application, network

topologies, physical layer capabilities, node mobility, the network's environment and numerous other factors.

Two MAC protocols stand out from the reviewed literature as potential candidates for the basis of this work's proposed MAC protocol: LDMA [108] and C/TDMA [110]. Ultimately C/TDMA was chosen. LDMA offers the potential for the best aspects both CSMA and TDMA. C/TDMA on the other hand, has the ability to selectively reduce the energy consumption of certain nodes through cluster formation. The final choice was motivated primarily by the greater simplicity of and availability of information for C/TDMA rather than the properties of its implementation. As routing protocols are also of interest in this work, the expected time required to implement C/TDMA was of significant importance.

This work makes a several small changes to C/TDMA as specified by Radhakrishnan et al. These changes primarily relate to the enabling nodes to periodically conserve energy in response to network activity. The final protocol is referred to as "CubeMac" for convenience. CubeMac's operation remains founded in C/TDMA with many of the changes made drawing inspiration from EDSN's Cpt/Lt protocol.

As discussed, due to the current CSN state-of-the-art, routing protocols are currently of less significance to the PvTP trade-off. Despite this, it is clear from the development of Gamalink [41] and the recommendations of several domain experts [9, 12, 56], that multi-hop networks are the next step for CSN applications. As such, this work proposes the use of DYMO [82] as a routing protocol for use with CSNs. DYMO is introduced briefly in section 2.2.2. This choice was primarily motivated by the availability of an existing implementation of DYMO for OMNeT++. Without this implementation it was unlikely that a suitable routing protocol could have been fully addressed by this work. As with C/TDMA, some modifications are made to DYMO. These modifications relate mainly to

resolving issues with the existing OMNeT++ implementation. This work's implementation of DYMO will be referred to as D3 (DYMO Cubed).

This work places the majority of its focus on CubeMac. Compared to CubeMac the customization of and the choice of basis for D3 are less informed by the current state of the art. This is partially a result of the scarcity of prior art relating to CSN routing protocols. Nonetheless, DYMO represents suitable candidate for the basis of D3. As discussed in section 2.2.2, DYMO is a reactive MANET routing protocol based on AODV. DYMO's reactive nature allows the examination of the effects of intermittent ground access through on-demand discovery of S2G links. Without a protocol such as DYMO this aspect of CSN's would have required an idealized simulation approach reducing the fidelity of results. Ultimately, both D3 and CubeMac combine through elements of cross-layer optimization to form the primary contribution of this work.

3.1 CubeMac

...

3.1.1 TDMA

At the core of CubeMac is TDMA. TDMA, as discussed in section 2.2.2, can be implemented with varying degrees of adaptivity. TDMA time slot owners can be dynamically assigned through distributed negotiation [118]. Approaches also exist which allow nodes to opportunistically share time slots and dynamically adjust the length of time slots [119]. Radhakrishnan et al. take a purely static approach to the TDMA aspects of C/TDMA. In this static approach time slots cannot be shared and have fixed lengths and owners. For convenience this static approach will be referred to as 'pure' TDMA.

Although the avoidance of more adaptive approaches is not discussed directly by C/TDMA's designers, pure TDMA has clear desirable properties. Introducing adaptivity to TDMA introduces the need for distributed decision making and consensus. These added requirements introduce varying degrees of overheads and delay. Propagation delays in LEO may be in the order of milliseconds. Also, node clocks may drift apart by up to 12ms [55]. Distributed decisions making and consensus in such environments, although not impossible, are likely to incur potentially unacceptable time and energy costs. Although pure TDMA may be outperformed in certain regards by various adaptive schemes, it avoids the need for such decisions making phases. The Cpt/Lt protocol, although adaptive in its assignment of roles, uses a pure TDMA approach. The schedule and duration of time slots and their respective owners is fixed and known to all network members.

3.1.2 Cluster Formation

Pure TDMA is coupled with CDMA to form Radhakrishnan et al.'s C/TDMA. This integration is made possible by the introduction of clustering. Using a "centrality algorithm", which is not specified further by Radhakrishnan et al., a CSN is divided into a number of clusters as in (*previous*). Within each cluster a "master" is selected. The remaining nodes in the cluster assume to roll of "slaves". Inter-cluster communication is handled solely by cluster masters (*new*). Master-to-master and master-to-slave communication takes place using pure TDMA. Slave-to-slave and slave-to-master communications on the other hand uses CDMA. In short, a special time slot is dedicated to for the use of CDMA allowing all cluster slaves to simultaneously communicate with their cluster master. This behaviour is discussed further in the following section.

It is worth noting that Radhakrishnan et al. specify two distinct type of TDMA frames, referred to by the authors as “channels”. A frame in TDMA nomenclature is a group of time slots which follows some fixed repeatable pattern (*new*). In the specification of C/TDMA, one frame is dedicated to ‘uplink’ and the other to ‘downlink’. The use of the term ‘downlink’ here should not be confused with S2G communication which is often referred to as ‘downlinking’. During uplink frames slots are dedicated to allow slaves transmit to their cluster master and cluster masters to transmit to one another. During the downlink frame cluster masters focus on transmitting to slaves within their clusters.

As discussed in section 2.2.1 this work is only concerned with data collection and not data dissemination. As such, CubeMac only implements a modified version of the Radhakrishnan et al.’s uplink frame as packets destined for ground will never be routed to slaves (...). This is a result of the behaviour of which manages the selection of which nodes may perform S2G communication. In this work’s simulated scenarios, only CubeMac masters may be selected for S2G communication. A selected master is referred to as a “ground master” (GM). Slave packets are routed first to a cluster master. If this master is not a GM, these packets will be forwarded to through neighbouring masters until reaching a GM (...). CubeMac’s operation is unaffected by D3’s behaviour, D3 is simply aware of the CubeMac’s master and slave roles.

3.1.3 CDMA

Simulations of CubeMac take a highly simplified approach to the treatment of CDMA. In simulations only the additional energy costs and parallel communication of CDMA are modelled. This approach was deemed sufficient for the purposes of the exploring CubeMac’s effects on the PvTP trade-off. Nonetheless, it is worth briefly discussing the general operation of CDMA.

CDMA based protocols combine ‘chip’, or ‘code’, with messages in order to form a final signal which is spread across a greater bandwidth (...). The primary benefit is that multiple signals may share a wireless medium at once without collision or interference, provided appropriate codes are used. There are numerous other benefits to CDMA approaches including enhanced resistance to certain types of noise, fading and jamming. By sharing codes between nodes, messages may be retrieved from spread signals using the code which initially spread the message.

Parallel communication through CDMA comes at the cost of added complexity and overheads. As shown in (...), spread signals require greater bandwidth than the ‘raw’ message which they contain. There are also added computational overheads due to message encoding and decoding. These bandwidth and computational overheads often result in a several factor increase in energy consumption. CDMA also generally requires more complex and costly energy demanding modulation schemes such as those based on Quadrature Amplitude Modulation (QAM) [120].

In this work’s simulation of CDMA, coding, modulation and spreading are ignored. Nodes simply receive messages unencoded and ignore any potential collisions. Transmitters using CDMA incur a higher energy cost which is proportional to the amount of time taken to transmit the unencoded message. This simplification was motivated largely by the absence of an existing suitable CDMA implementation for OMNeT++.

3.1.4 Frame Structure

As alluded to, CubeMac’s behaviour changes based on the role of a node. Masters uses TDMA for communication whereas slaves use CDMA. Assuming a network containing NM clusters, and therefore NM masters, each CubeMac frame will consist of NM+1 time

slots. CubeMac assigns each cluster an ID from 1 to NM. Each cluster master will own the time slot corresponding to its cluster ID. For example, the master of the cluster with ID 1 (M1) will occupy time slot 1 (T1). The final time slot (NM+1), named the “uplink” slot, is reserved for cluster slaves. During this slave uplink slot all slaves with each cluster use CDMA to transmit data to their cluster master. The completion of uplink slot marks the end of a CubeMac frame after which a new frame starts again at slot 1. During each frame nodes transition between transmitting, listening and sleeping states. (*New*) Illustrates the behaviour of nodes during one complete frame.

The behaviour illustrated in (...) represents a slight departure from Radhakrishnan et al.’s ‘uplink’ frame. Cluster master may use their slots to transmit to any node which is listening. This represents a departure from Radhakrishnan et al.’s C/TDMA specification. In C/TDMA, each master would instead reserve a separate time slot for each neighbouring cluster master and each slave within its cluster. This change was necessary due to the omission of Radhakrishnan et al.’s ‘downlink’ frame without which no transmissions from master to slave can occur. The second change relates to the ability of slaves to sleep during certain slots.

3.1.5 Energy Conservation

CubeMac introduces two additional features to C/TDMA both of which are intended to allow nodes to conserve energy. The first feature, which is illustrated in (...), allows slaves to sleep during certain slots. Slaves only communicate with their cluster master. As such slaves may sleep during any slot which is not the uplink or their cluster master’s slot. Each slave maintains knowledge of the cluster within which it resides. As cluster ids correspond directly the slot owned by the cluster masters, it is a simple process for slave to determine the slot in which they may sleep.

The second feature allows nodes to sleep during slots under various conditions. In implementing this feature CubeMac introduces a special field with packet headers. This field, the “last” field, allows nodes to indicate that a packet is the last they intend to send during a slot. CubeMac also introduces a timeout period, which is not specified in C/TDMA. Furthermore, CubeMac requires each cluster master to track the number slaves present in its cluster. This is achieved by the addition of a packet header field which contains the ID of the cluster to which the packet originator belongs. Finally, CubeMac introduces the concept of an optional “no data” (ND) packet and an ND header field. A node, without packets to send, may broadcast an ND packet to those listening to indicate that it has no packets to send during this slot. ND packets are made as small as possible and are important in cases where multiple nodes are waiting in listening mode. Using these additions and the general structure of CubeMac, there are several conditions under which a node may sleep during a slot.

In the master role:

1. If no packet is received from a slot owner or from any slave during an uplink slot for a period longer than the configured timeout period
2. If, during a non-uplink slot, the slot owner sends an ND packet or a packet marked as its last (...)
3. If, during an uplink, all slaves transmit an ND packet or a packet marked as their last
4. Following the transmission of an ND packet or a packet marked as “last” (...)

In the slave role:

1. If no packet is received from the cluster master during the cluster master’s slot for a period longer than the configured timeout period

2. If, during the cluster master's slot, the cluster master transmits an ND packet or a packet marked as their last (...)
3. Following the transmission of a packet marked as "last" or an ND packet (...)

These added energy saving features almost incur a low cost as a result of C/TDMA's static TDMA approach and cluster based architecture. The only tangible overheads introduced are those of the added packet header fields. The "last" and ND fields only require one bit and the cluster ID field only requires $\log_2(NM)$ bits. Crucially these energy saving features do not come at the cost of throughput. These features take advantage of the correspondence between slots and cluster IDs as well as master-slave relationship with requiring any explicit reduction in traffic.

3.1.6 Drawbacks

There are two notable drawbacks of the added energy conservation features to consider. The first relates to the master-slave relationship. The restriction of slave communications to cluster masters is fundamental to CubeMac. To lift this restriction would require introducing additional slots for each slave within a cluster at which point CubeMac essentially approximates pure TDMA. Where data collection is concerned, allowing slaves to communicate with one another offers no obvious benefit as only masters may communicate to ground stations. The second major drawback is one inherent in effectively all TDMA or schedule based systems. If a new packet is generated the moment after a node makes the decision to go to sleep it will have to wait for an entire frame before being sent (...). This is fundamentally the same issue as occurs with packets generated immediately following the end of a node's slot. Once again however, this drawback is of little concern in the context of this work. Ill-timed packet generation will affect latency but

overall will not affect throughput. In this regard CubeMac sacrifices on latency in order to reduce energy consumption.

Similar to the ill-timed packet generation issue is that of slot saturation. Consider a scenario where only one node, a master for instance, is generating packets. If this node consistently generates more packets than it can send in a time slot then the medium is being poorly utilized. In CubeMac this master could not borrow time from another time slot in order to transmit its additional packets; even though all other time slots are effectively empty. LDMA overcomes this issue through the use of CSMA in such low contention scenarios. However, yet again, the impact of this issue is reduced as a result of context of this work. The chosen hypothetical mission assumes that all nodes carry scientific instruments and generate packets containing science data at the same rate.

The use of global time slots also creates an issue in relation to CubeMac. It is inevitable that nodes within CSN will be widely spatially distributed. If all nodes are within range of one another TDMA will succeed in avoiding all potential conflicts without any 'waste'. In spatially disparate networks such as CSNs, several nodes may observe a time slot with owners who are multiple hops removed. In such cases these nodes are 'isolated' from the time slot and cannot perform any useful communication. This is a result of global time slots. In case where time slots are dynamically allocated at a local scope this isolation can be avoided. Once again this issue is addressed by the low contention level mode of LDMA. A protocol known referred to as Lightweight MAC (LMAC) uses a dedicated slot assignment stage in order to assign time slots at a local rather than a global [118]. LMAC uses some basic carrier sense based approaches during the assignment stage to overcome issues inherent with local time slot assignment such as hidden node problem discussed in section 2.3.2.

The general nature of CubeMac makes it ill-suited for node failures. For instance, If a master's instrument fails then an entire time slot will remain unused regardless of the saturation of other slots. If a master as a whole dies and entire cluster will be isolated from the network and ground (...). CubeMac has no facilities for acknowledging the receipt of packets and as such slaves cannot recognize when their master has died or drifted out of range. The lack of acknowledgement scheme is not fundamental to CubeMac and one could be added. Acknowledgments are omitted as this work concentrates on the PvTP trade-off without consideration of node failures.

Clustering and master selection are important in addressing challenges related to node mobility and failure. Neither are implemented although both could be added to CubeMac. As masters have considerably more work to perform than slaves, periodic master selection and re-clustering based available node battery capacities constitute crucial features of any real-world implementation of CubeMac. Otherwise, master nodes would quickly exhaust their available energy supplies and all slaves would become isolated from ground. Radhakrishnan et al. mention master selection and clustering briefly but provide no tangible specification of either. This work examines CubeMac during steady state operation following clustering and master selection. Both dynamic clustering and master selection are also discussed in the context of future work in section 6.2.

3.2 D3

D3 does not diverge significantly from that of DYMO the behaviour of which is highly similar to that of AODV. Like AODV, DYMO makes use of three different "route messages": route requests (RREQs), route responses (RREPs), and route errors (RERRs). DYMO's general use of these message is the same as that of AODV which is detailed in

section 2.2.2 and (2.2.2 AODV Figure). The remaining relevant aspects of DYMO's operation are discussed in the following sections.

3.2.1 Intermediate RREPs

In section 2.2.2. the concept of intermediate RREPs was briefly introduced in the context of AODV. DYMO employs the same approach as AODV in implementing this feature. An node receiving an RREQ for a given target may check whether it has a valid route for the target. If the node has a valid route it may generate an RREP to be sent to the originator. This RREP will be identical to an RREP a target receiving the same RREQ would generate.

Allowing intermediate RREPs can significantly reduce the amount of traffic generated during each route discovery attempt. Intermediate RREPs are especially impactful in the case of CubeMac. All CubeMac slaves only communicate with cluster masters, as such the next hop for any route held by a slave will be the cluster master. If the cluster master has a route for a target it can generate an RREP thus reducing the need for slave RREQs to traverse the network. In simulation scenarios, as this work is concerned S2G throughput, all packets are routed to address assigned to ground. In CubeMac master time slots occur before the slave uplink slot. This arrangement increases the likelihood that a master will be able to generate an intermediate RREP and reduce traffic. However, there is one notable drawback to consider with intermediate RREPs. Reducing the propagation of RREQs to target nodes increases the likelihood that suboptimal routes held by intermediate nodes are used.

3.2.2 RERRs

RERRs are generated in response to link breakages. RERRs often rely on packet acknowledgement schemes. As discussed, CubeMac does not utilize acknowledgement

schemes and node failure is not within the scope of this work. Also, the specification of this work's hypothetical mission in section 1.1 states that nodes are assumed not to move relative to one another. A node's motion relative to ground is the only source in this work of link breakages. As will be discussed further in section 4.2.2, master nodes use a separate network interface for S2G communication. This interface uses a MAC protocol based on CSMA rather than CubeMac. This CSMA protocol does utilize acknowledgements. As ground masters move out of range of ground this CSMA protocol signals DYMO that the link has broken. This is the only mechanism by which links break within this work's simulation scenarios. In response to a link break signal DYMO will generate a RERR message. This message is broadcast throughout the network allowing all nodes to discard any routes to ground which they maintain.

DYMO is capable of behaviour similar to that of DSR. In DSR every node receiving a RREQ for which it is not the target adds their address information to the RREQ packet prior to forwarding. DYMO mimics this behaviour by allowing nodes to optionally add their "intermediate node" information to RREQ packets prior to forwarding. In the same manner as AODV RREQs, DYMO RREQs only contain information for the originator and target nodes by default.

3.2.3 Sequence Numbers

Like AODV, DYMO utilizes sequence numbers (SNs) to determine the freshness of a route. DYMO records the SN associated with each incoming route. Prior to replacing an existing route DYMO checks a candidate route's SN against the SN of the existing route. If the candidate's SN is lower than the existing SN then the candidate route is considered stale. By default, stale routes are discarded regardless of whether the route has a lower cost than the existing route. As a result of the relative motion of nodes such stale routes

have a higher likelihood of being broken. It follows that routes with higher SNs have been established more recently and as such routes are less likely to have been affected by topology changes. SNs can also be used to detect routing loops. However, this is not the approach taken by DYMO.

3.2.4 Route Costs

As alluded to, DYMO allows node to record each route's 'cost'. These costs are central to DYMO's approach to routing loop detection. Route costs are established within RREP packets. As an RREP is transmitted unicast from the target node to the originator node, each intermediate node adds some greater-than-zero value to the RREP's 'cost' field (*new*). As the name suggests the final value in this field represents the cost of the route described within the RREP. The value added by an intermediate node represents the cost of the link between the intermediate node and the node previously visited by the RREP (...).

The most common and basic cost value is that of "hop count". When using a hop count cost function nodes simply add '1' to the current cost field within RREPs. Originators will then favour routes which represent the shortest path to targets (...). As discussed in section 2.2.2, energy metrics are a common and impactful source of link cost values. An energy aware cost function may utilize information regarding a node's remaining battery capacity. For example, a node may use the following logic to determine a link's cost: if battery capacity < N add '10' to link cost – else add '1' to link cost. Provided all nodes use the same cost logic originators will favour routes which avoid nodes with battery capacities below the threshold of N. If all nodes for all possible routes have battery capacities above this threshold then the shortest route will be used.

Costs may also be based on other metrics such as those relating to: the signal strength of a received RREP, the number of packets a node has waiting for transmission, and the time remaining before a node will be in range of ground. All of these approaches represent some degree of cross layer optimization, a common topic discussed in chapter 2. It is also possible for CubeSats to use the GomX-3 energy aware scheduling approach discussed in section 2.4.1. Through such an approach CubeSats could forecast upcoming energy availability and demands in order to determine the potential cost of announcing a ‘cheap’ link cost.

DYMO is capable of detecting routing loops using route costs as a result of its focus on opportunistically establishing routes. Unlike some other routing protocol, DYMO route entries only require a target address and a next hop address. For instance, when a node X receives an RREQ from another node Y and the RREQ has an originator of Z it follows that a route is available to Z and that Y is the next hop (...). When X first sees an RREQ it will add this route to Z and then broadcast the RREQ. Y will see this RREQ from X however Y has enough information to avoid rebroadcasting this RREQ and causing a broadcast loop. Y already has a route to Z and the RREQ from X will have a higher cost as all added cost value must be greater than zero. This allows Y to drop the RREQ. This approach avoid routing loops and the same logic can be used to avoid loop of arbitrary lengths (...).

3.2.5 Discovery and Maintenance Patterns

Through the use of route messages, sequence numbers and route costs DYMO has suitable capabilities to discovery, chose and detect the breakage of routes. The remaining salient aspects of DYMO relate to its timing of route discovery and maintenance. By default DYMO includes no explicit route maintenance. Like AODV, DYMO has a specification for

the “Hello” messages which are intended to help maintain up to date routes and route costs. Such messages are discussed briefly in section 2.2.2.

Inactive routes in DYMO are subject to a timeout. This timeout insures that routes that are unused for a given period of time are ‘refreshed’ before use. This refreshing process is identical to a route discovery attempt utilizing RREQs and RREPs. During route discovery attempts originator nodes also have a timeout to handle ‘lost’ RREQs. After sending an RREQ a node will only wait a fixed amount of time for an RREP before considering the discovery attempt a failure. Multiple attempts may take place within one discovery cycle. Each failed attempt incurs a binary exponential back-off. This back-off between attempts increases the likelihood of the nodes moving into a more favourable configuration and avoids repeatedly flooding the network with unnecessary RREQs. After exceed a maximum number of failed attempts to discovery a rout to a target a node will cease any further attempts and enter a “holddown” state. The duration of this holddown state is configurable. Holddown states only affects route discovery for ‘failed’ targets. To summarize, the following aspects of DYMO’s behaviour are configurable: The time to wait for an RREP, the maximum number of discovery attempt for a target and the duration of a holddown (*timeline*).

3.2.6 Other Features

DYMO’s specification includes numerous feature, several of which are not considered relevant to this work and as such are not discussed. In this section some of the less impactful features of DYMO’s are briefly noted.

Router Clients

All DYMO nodes discussed thus far are routers, in that they are capable of forwarding traffic from other nodes. DYMO allows for nodes to have associated clients. Clients generate no route messages. Clients forward their packets blindly to a DYMO router. The router then carries out any necessary route discovery and forwards any client packets as required. Routers will also announce routes for their clients in a manner similar to that of an intermediate RREP. Allowing routers to have clients reduces route message traffic and the workload of clients. However, as with the CubeMac's master slave relationship, clients can become isolated from the network if their associated router fails.

Multicast RREPs

DYMO uses multicasting in place of AODV's broadcasting. In this work all network interfaces are configured to belong to the same multicast group. This configuration makes DYMO's multicast behaviour identical to that of broadcasting. As such, this behaviour of DYMO has been ignored up to this point. As there is no appreciable difference between multicasting a message to all neighbours and broadcasting the same message this work refers to DYMO's multicasting as broadcasting.

RREQs are broadcast by default. RREP can also be optionally sent via broadcast rather than unicast. The primary benefit of this feature is that it reduces the overall number of RREQs that network nodes will generate for the given target. As with RREQs, nodes which pass on RREPs will use the RREP to opportunistically populate its routing table. As such, this feature may result in nodes adding routes they will never use. This can be especially wasteful in large networks.

Hop Limits

As DYMO RREQs traverse a network via broadcasting nodes update a hop count field. DYMO can be configured to limit the number hops that an RREQ can make. DYMO's ability to detect RREQs which are travelling in a loop combats one aspect of wasteful route discovery. However, RREQs may continue to traverse a network even when an RREP has been generated by its target. Hop limits combat this by allowing network designers to specify a maximum hop limit. If the longest possible, or acceptable, route within a network is N then any RREQ which makes more than N hops without reaching a target can be safely dropped.

3.2.7 D3 Modifications

In the case of D3, DYMO is comparatively unaltered in when compared to CubeMac and C/TDMA. D3 introduces two additions: The concept of a ground master and special treatment of multiple interfaces (I/Fs). In simulation all CubeSats with a CSN have two interfaces, one for S2S and the other for S2G communications. DYMO broadcasts RREQs and RERRs on the first available network interface within the appropriate multicast group. This presents an issue. If CubeSats only ever broadcast RREQs on their S2S I/F a route to ground will never be discovered. Similarly, if only the S2G I/F is used then no S2S routes will be discovered. Also, the I/F on which an RREP arrives is used as the I/F for all traffic for the RREP's target and originator. As such, if a route to ground is discovered using the S2G I/F the discoverer will attempt to use its S2G I/Fs for S2S communication, which is of course not desirable. These issues are overcome by introducing the role of ground master GM. As mentioned, only CubeMac masters may participate in S2G communications. It follows that only CubeMac masters may take the role of GM. The

method used to designate a master as GM and the impact thereof is discussed in section 4.1.1.2.

The GM role allows D3 to overcome the aforementioned multiple interfaces issues by changing the default behaviour of GMs. When a GM receives an RREQ for the address of ground this RREQ is sent out via the S2G I/F. Any RREP returning from ground on the S2G I/F is forwarded out via the S2S I/F. GMs must also change their default route storage approach to insure that the next hop address in routes to ground reference the correct I/F. Non-GMs effectively ignore their S2G I/F entirely and use their S2S I/F for all communications. The final implementation of the GM role required many small changes to the existing OMNeT++ implementation of DYMO. All salient effects of these changes relate to the aforementioned I/F behaviours.

3.2.8 Drawbacks

D3 has several undesirable characteristics. Most notable among these is its inability to retain redundant routes. This was a clear design choice by the designers DYMO as there are other AODV derivatives which allow nodes to maintain redundant routes for a given target. Protocol designers must weigh the cost of a node of storing redundant routes against the overall network cost of requiring a complete route discovery attempt every time a route breaks. Within this work's hypothetical mission only routes to ground can break. As will be discussed further in relation to simulation scenarios, it is assumed that only one master may be a GM at any one time. As such all valid routes to ground at any one time will share the same final hop, from the current GM to ground. It is also assumed that nodes do not move relative to one another. Given these two assumptions there are no benefits in maintaining redundant routes as each S2G link break invalidates all possible routes to routes to ground. Nonetheless, maintaining redundant routes may be of value

in more realistic scenarios wherein the motion of nodes relative to one another must be considered.

As discussed in section 2.1.1, most missions will require CubeSats to periodically check the current time and their position and velocity via GNSS networks. Such information, when opportunistically shared amongst nodes, can greatly benefit the efficiency of route discovery, maintenance and choice. In particular position based routing approaches have become popular in recent VANET related research [121]. As discussed in section 4.1.1.2, position information is utilized in GM designation. Aside from this use, D3's lack of exploitation of readily available GNSS information represents a missed opportunity. Such information could be used to predict link breakages, avoid the use of certain unstable routes and avoid unnecessary route discovery attempts.

Designating a given master as GM adds a potentially undesirable degree of complexity to D3 and introduces the need for distributed decision making. This work's implementation of the GM solution is presented as sufficient rather than optimal. For instance an alternative solution may be to conduct route discovery over both interfaces from all nodes. This approach requires the use of a link cost function that reflects the difference between a link created using an S2G I/F and S2S I/F. An appropriate cost function should reflect the difference in bandwidth and energy consumption of the two interfaces. In general, nodes would be expected to favour using their S2S I/Fs for S2S links and vice versa. However, this solution could allow an isolated node to use its S2G radio to achieve some degree of communication with the CSN when no other link is available. Also, multiple nodes may establish S2G links. The cost functions for these links could reflect a node's remaining battery capacity, number of queued packets and expected S2G communication. In this case there is a clear case to maintain redundant routes to ground. The primary

drawback of this approach is the added energy expense of conducting route discover over multiple S2G I/Fs when in reality only a small number of these will succeed. Ultimately, without an implementation of this approach to compare to the GM approach it cannot be stated empirically which is likely to be more performant.

... Will have to do some more reading and thinking before adding any further points to this section, perhaps a note on handover ...

Chapter 4: Simulation

... OMNeT++ ...

4.1 Implementation

...

4.1.1 Proposed Protocols

...

CubeMac

...

D3

...

4.1.2 Space-to-Ground communication

...

4.1.3 Scenarios

...

Scenario 1

...

Scenario 2

...

Scenario 3

...

4.1.4 Implementation Issues

... So many wonderful issues

NS-3

...

CDMA

... Radio multiple reception, Broken narrowband radio

AODV

... Predicting timing, instability, other routing protocols no worky

DYMO

... Metrics, loop freedom, RERRs, Multicast

Multiple Interfaces

... Oh so many headaches

4.2 Discussion

... Fidelity, why simulation etc.

Chapter 5: Results

...

5.1 Key Metrics

...

5.2 Simulation Results

...

5.2.1 Scenario 1

...

5.2.2 Scenario 2

...

5.2.3 Scenario 3

...

5.3 Discussion

...

Chapter 6: Conclusions

6.1 Discussion

...

6.1.1 Contributions

...

CubeMac

...

D3

...

CSN Simulation

...

6.1.2 Space Junk

...

6.1.3 Mission Design

Those familiar with mission planning and satellite technology will note that there is little treatment of mission operational requirements and technology readiness. These omissions allow a focus on the PvTP trade-off by ignoring many of the realities mission

design. In reality, space bound technology must be thoroughly understood, tested and generally as simplified as possible. There are hints of this in the design of NASA's EDSN and Nodes missions. Mission designers opted to design a protocol rather than implement an existing one. This has two clear benefits; It allows for complete knowledge of the protocol by mission designers and operators and it greatly reduces the risk of the existence of unknown bugs or flaws.

Once a satellite is in orbit there is little that can be done to fix issues. As such, mission designers tend to focus on simple, easily-understood and if possible space-tested approaches. Both CubeMac and DYMO are less complex than some current MAC and Routing protocols but, there are definitely less complex approaches in existence. For instance, why not adopt pure TDMA and DSR? The approach may not be as performant but it would be simple to test in simulation prior to launch and issues in orbit would be easily understood.

This preference towards simplicity provides part of the motivation behind protocol design choices, especially with CubeMac. There are several protocols such as LDMA which could have provided a basis for CubeMac [9]. In many cases it was felt that the added complexity of these protocols was not accompanied by sufficient benefits.

6.2 Future Work

...

This section gets auto formatted by EndNote on updates, it will be the last section to get edited and properly formatted.

References

- [1] L. Brennan and A. Vecchi, *The business of space: The next frontier of international competition*: Palgrave Macmillan, 2011.
- [2] G. Johnson. (2012, January, 15). *Revised, Expanded Launch Cost Data*. Available: <http://exrocketman.blogspot.ie/2012/05/revised-expanded-launch-cost-data.html>
- [3] A. Scholz and J.-N. Juang, "Toward open source CubeSat design," *Acta Astronautica*, vol. 115, pp. 384-392, 2015.
- [4] S. Padmanabhan, S. Brown, B. Lim, P. Kangaslahti, D. Russell, and R. Stachnik, "Airborne Deployment and Calibration of Microwave Atmospheric Sounder on 6U CubeSat," in *AGU Fall Meeting Abstracts*, 2015.
- [5] V. Hernandez, P. Gankidi, A. Chandra, A. Miller, P. Scowen, H. Barnaby, *et al.*, "SWIMSat: Space Weather and Meteor Impact Monitoring using a Low-Cost 6U CubeSat," 2016.
- [6] U. Kvell, M. Puusepp, F. Kaminski, J.-E. Past, K. Palmer, T.-A. Grönland, *et al.*, "Nanosatellite orbit control using MEMS cold gas thrusters," *Proceedings of the Estonian Academy of Sciences*, vol. 63, p. 279, 2014.
- [7] X. Sun and X. Wu, "A cubesat attitude control system with linear piezoelectric actuator," in *Piezoelectricity, Acoustic Waves, and Device Applications (SPAWDA), 2014 Symposium on*, 2014, pp. 72-75.
- [8] A. Budianu, T. J. W. Castro, A. Meijerink, and M. J. Bentum, "Inter-satellite links for cubesats," in *Aerospace Conference, 2013 IEEE*, 2013, pp. 1-10.

- [9] R. Radhakrishnan, W. W. Edmonson, F. Afghah, R. M. Rodriguez-Osorio, F. Pinto, and S. C. Burleigh, "Survey of Inter-satellite Communication for Small Satellite Systems: Physical Layer to Network Layer View," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 2442-2473, 2016.
- [10] Y. F. Wong, O. Kegege, S. H. Schaire, G. Bussey, S. Altunc, Y. Zhang, *et al.*, "An Optimum Space-to-Ground Communication Concept for CubeSat Platform Utilizing NASA Space Network and Near Earth Network," 2016.
- [11] W. Harrington and J. Heath, "Development of a Low-Cost, Open Software/Hardware Command, Control and Communications Module for CubeSats," in *AIAA SPACE 2016*, ed, 2016, p. 5616.
- [12] J. Hanson, A. G. Luna, R. DeRosee, K. Oyadomari, J. Wolfe, W. Attai, *et al.*, "Nodes: A Flight Demonstration of Networked Spacecraft Command and Control," 2016.
- [13] A. Tatomirescu, G. F. Pedersen, J. Christiansen, and D. Gerhardt, "Antenna system for nano-satellite mission GOMX-3," in *Antennas and Propagation in Wireless Communications (APWC), 2016 IEEE-APS Topical Conference on*, 2016, pp. 282-285.
- [14] M. Swartwout, "The first one hundred CubeSats: A statistical look," *Journal of Small Satellites*, vol. 2, pp. 213-233, 2013.
- [15] K. Kelley, "Launch systems to support the booming nanosatellite industry," in *Aerospace Conference, 2015 IEEE*, 2015, pp. 1-6.
- [16] D. Hitt, K. F. Robinson, and S. D. Creech, "NASA's Space Launch System: A New Opportunity for CubeSats," 2016.
- [17] D. Masutti, T. Banyai, J. Thoemel, T. Magin, B. Taylor, and D. Kataria, "Investigating the Middle and Lower Thermosphere using a Cubesat Constellation: the QB50 Mission and its Particular Challenges," in *EGU General Assembly Conference Abstracts*, 2015, p. 9016.
- [18] M. Tsay, J. Frongillo, K. Hohman, and B. K. Malphrus, "LunarCube: A Deep Space 6U CubeSat with Mission Enabling Ion Propulsion Technology," 2015.

- [19] R. W. Ridenoure, D. A. Spencer, D. A. Stetson, B. Betts, R. Munakata, S. D. Wong, *et al.*, "Status of the Dual CubeSat LightSail Program," in *AIAA SPACE 2015 Conference and Exposition*, 2015, p. 4424.
- [20] R. Glumb, C. Lietzke, S. Luce, and P. Wloszek, "Cubesat Fourier Transform Spectrometer (CubeSat-FTS) for Three-Dimensional Global Wind Measurements," in *American Meteorological Society Annual Meeting*, (January 2015), 2015.
- [21] S. Nag, J. L. Rios, D. Gerhardt, and C. Pham, "CubeSat constellation design for air traffic monitoring," *Acta Astronautica*, vol. 128, pp. 180-193, 2016.
- [22] D. Westley, A. Martinez, and A. Petro, "Edison Demonstration of Smallsat Networks," 2015.
- [23] R. Barbosa. (2015, September, 24). *China debuts Long March 11 lofting Tianwang-1 trio*. Available: NASASpaceFlight.com
- [24] A. Varga, "OMNeT++," *Modeling and tools for network simulation*, pp. 35-59, 2010.
- [25] H. Helvajian and S. W. Janson, *Small satellites: past, present, and future*: Aerospace Press, 2008.
- [26] H. Heidt, J. Puig-Suari, A. Moore, S. Nakasuka, and R. Twiggs, "CubeSat: A new generation of picosatellite for education and industry low-cost space experimentation," 2000.
- [27] M. Swartwout, "Cubesat database," *St. Louis University*. [Online]. [Accessed 7 February 2015], 2015.
- [28] C. Boshuizen, J. Mason, P. Klupar, and S. Spanhake, "Results from the planet labs flock constellation," 2014.
- [29] R. A. Deepak and R. J. Twiggs, "Thinking out of the box: Space science beyond the CubeSat," *Journal of Small Satellites*, vol. 1, pp. 3-7, 2012.
- [30] J. Puig-Suari, J. Schoos, C. Turner, T. Wagner, R. Connolly, and R. Block, "CubeSat developments at Cal Poly: the standard deployer and PolySat," in *Proceedings of SPIE-The International Society for Optical Engineering*, 2000, pp. 72-78.

- [31] J. Farkas, "CPX: Design of a standard cubesat software bus," *California State University, California, USA*, 2005.
- [32] J. Straub, C. Korvald, A. Nervold, A. Mohammad, N. Root, N. Long, *et al.*, "OpenOrbiter: A low-cost, educational prototype CubeSat mission architecture," *Machines*, vol. 1, p. 1, 2013.
- [33] A. K. Nervold, J. Berk, J. Straub, and D. Whalen, "A Pathway to Small Satellite Market Growth," *Advances in Aerospace Science and Technology*, vol. 1, p. 14, 2016.
- [34] K. Hayward, "The Economics of Launch Vehicles: Towards a New Business Model," in *Yearbook on Space Policy 2015*, ed: Springer, 2017, pp. 247-256.
- [35] W. A. Beech, D. E. Nielsen, J. T. Noo, and L. K. Ncuu, "AX. 25 Link Access Protocol for Amateur Packet Radio, Version: 2.2 Rev," in *Tucson Amateur Packet Radio Corp*, 1997.
- [36] P. Muri and J. McNair, "A survey of communication sub-systems for intersatellite linked systems and CubeSat missions," *JCM*, vol. 7, pp. 290-308, 2012.
- [37] C. Fish, C. Swenson, T. Neilsen, B. Bingham, J. Gunther, E. Stromberg, *et al.*, "Dice mission design, development, and implementation: Success and challenges," 2012.
- [38] R. Hodges, B. Shah, D. Muthulingham, and T. Freeman, "ISARA–Integrated Solar Array and Reflectarray Mission Overview," 2013.
- [39] S. Wu, W. Chen, and C. Chao, "The STU-2 CubeSat Mission and In-Orbit Test Results," 2016.
- [40] S. S. Arnold, R. Nuzzaci, and A. Gordon-Ross, "Energy budgeting for CubeSats with an integrated FPGA," in *Aerospace Conference, 2012 IEEE*, 2012, pp. 1-14.
- [41] A. Oliveira. (2015, 02-02). *Final Report Summary - GAMALINK (Generic SDR-bAsed Multifunctional spAce LINK)*. Available: http://cordis.europa.eu/result/rcn/172006_en.html
- [42] (2016, April 14th). *CubeSat Design Overview Report*. Available: <http://sydney.edu.au/engineering/aeromech/AERO3760/private>

[/CDR/1%20%20Critical%20Design%20Overview%20i-INSPIRE%EF%BC%92.pdf](#)

- [43] J. Guo, J. Bouwmeester, and E. Gill, "From Single to Formation Flying CubeSats: An Update from the Delft Programme," 2013.
- [44] M. Focardi, V. Noce, S. Buckley, K. O'Neill, A. Bemporad, S. Fineschi, *et al.*, "The shadow position sensors (SPS) formation flying metrology subsystem for the ESA PROBA-3 mission: present status and future developments," in *SPIE Astronomical Telescopes+ Instrumentation*, 2016, pp. 99044Z-99044Z-17.
- [45] G. Sun, X. Xia, S. Wu, Z. Wu, and W. Chen, "Attitude Determination and Control System Design for STU-2A CubeSat and In-Orbit Results," 2016.
- [46] J. Li, M. Post, T. Wright, and R. Lee, "Design of attitude control systems for cubesat-class nanosatellite," *Journal of Control Science and Engineering*, vol. 2013, p. 4, 2013.
- [47] E. Glennon, J. Gauthier, M. Choudhury, A. Dempster, and K. Parkinson, "Synchronization and syntonization of formation flying cubesats using the namuru V3. 2 spaceborne GPS receiver," in *Proceedings of the the ION 2013 Pacific PNT Meeting, Honolulu, HI, USA*, 2013, pp. 23-25.
- [48] A. Cortiella, D. Vidal, J. Jané, E. Juan, R. Olivé, A. Amézaga, *et al.*, "3CAT-2: Attitude Determination and Control System for a GNSS-R Earth Observation 6U CubeSat Mission," *European Journal of Remote Sensing*, vol. 49, pp. 759-776, 2016.
- [49] W. H. Swartz, S. R. Lorentz, P. M. Huang, A. W. Smith, D. M. Deglau, S. X. Liang, *et al.*, "The Radiometer Assessment using Vertically Aligned Nanotubes (RAVAN) CubeSat Mission: A Pathfinder for a New Measurement of Earth's Radiation Budget," 2016.
- [50] W. H. Swartz, L. P. Dyrud, S. R. Lorentz, D. L. Wu, W. J. Wiscombe, S. J. Papadakis, *et al.*, "The RAVAN CubeSat mission: advancing technologies for climate observation," in *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, 2015, pp. 5300-5303.

- [51] S. Kanekal, P. O'Brien, D. N. Baker, K. Ogasawara, J. Fennell, E. Christian, *et al.*, "Radition belt dynamics: Recent results from van Allen Probes and future observations from CeREs," in *41st COSPAR Scientific Assembly, abstracts from the meeting that was to be held 30 July-7 August at the Istanbul Congress Center (ICC), Turkey, but was cancelled. See <http://cospar2016.tubitak.gov.tr/en/>, Abstract PRBEM. 2-1-16.*, 2016.
- [52] J. Westerhoff, G. Earle, R. Bishop, G. R. Swenson, S. Vadas, J. Clemmons, *et al.*, "LAICE CubeSat mission for gravity wave studies," *Advances in Space Research*, vol. 56, pp. 1413-1427, 2015.
- [53] J. M. Morrison, H. Jeffrey, H. Gorter, P. Anderson, C. Clark, A. Holmes, *et al.*, "SeaHawk: an advanced CubeSat mission for sustained ocean colour monitoring," in *SPIE Remote Sensing*, 2016, pp. 100001C-100001C-11.
- [54] M. Bentum, A. Meijerink, A.-J. Boonstra, C. Verhoeven, and A.-J. v. d. Veen, "OLFAR: the orbiting low frequency array, how a cube sat swarm becomes a novel radio astronomy instrument in space," *De Vonk*, vol. 25, pp. 1-5, 2010.
- [55] J. Hanson, J. Chartres, H. Sanchez, and K. Oyadomari, "The EDSN intersatellite communications architecture," 2014.
- [56] J. Chartres, H. Sanchez, and J. Hanson, "EDSN development lessons learned," 2014.
- [57] G. Bora, S. Bora, S. Singh, and S. M. Arsalan, "OSI reference model: An overview," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 7, 2014.
- [58] K. I. Parker, "State-of-the-Art for Small Satellite Propulsion Systems," 2016.
- [59] O. Barnouin, J. Biele, I. Carnelli, V. Ciarletti, A. Cheng, A. Galvez, *et al.*, "The Asteroid Impact and Deflection Assessment (AIDA) mission: Science Proximity Operations," in *LPSC 2016 47th Lunar and Planetary Science Conference*, 2016, p. 1427.
- [60] M. Bisgaard, D. Gerhardt, H. Hermanns, J. Krčál, G. Nies, and M. Stenger, "Battery-Aware Scheduling in Low Orbit: The GomX-3 Case," in *FM 2016: Formal Methods: 21st International Symposium*,

- Limassol, Cyprus, November 9-11, 2016, Proceedings 21, 2016, pp. 559-576.*
- [61] B. Niels, "ESA and GomSpace Sign Contract to Launch Advanced Nanosatellite," ed. Web: GOMspace, 2016.
 - [62] M. Villa, A. Martinez, and A. Petro, "Cubesat Proximity Operations Demonstration (CPOD)," 2015.
 - [63] D. Cecil, "Potential Future NASA Satellite Data and Applications for Tropical Cyclones," 2016.
 - [64] E. Gill, P. Sundaramoorthy, J. Bouwmeester, B. Zandbergen, and R. Reinhard, "Formation flying within a constellation of nanosatellites: The QB50 mission," *Acta Astronautica*, vol. 82, pp. 110-117, 2013.
 - [65] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks*, vol. 67, pp. 104-122, 2014.
 - [66] J. Zheng and M. J. Lee, "A comprehensive performance study of IEEE 802.15. 4," ed: IEEE Press book Los Alamitos, 2004.
 - [67] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology," *Sensors*, vol. 12, pp. 11734-11753, 2012.
 - [68] N. Accettura, L. A. Grieco, G. Boggia, and P. Camarda, "Performance analysis of the RPL routing protocol," in *Mechatronics (ICM), 2011 IEEE International Conference on*, 2011, pp. 767-772.
 - [69] P. T. A. Quang and D.-S. Kim, "Throughput-aware routing for industrial sensor networks: Application to ISA100. 11a," *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 351-363, 2014.
 - [70] L. D. Mendes and J. J. Rodrigues, "A survey on cross-layer solutions for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 34, pp. 523-534, 2011.
 - [71] G. Miao, N. Himayat, Y. G. Li, and A. Swami, "Cross-layer optimization for energy-efficient wireless communications: a survey," *Wireless Communications and Mobile Computing*, vol. 9, pp. 529-542, 2009.

- [72] M. Di Francesco, S. K. Das, and G. Anastasi, "Data collection in wireless sensor networks with mobile elements: A survey," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, p. 7, 2011.
- [73] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *IEEE Transactions on mobile computing*, vol. 99, pp. 70-80, 2002.
- [74] L. Bölöni and D. Turgut, "Should I send now or send later? A decision-theoretic approach to transmission scheduling in sensor networks with mobile sinks," *Wireless Communications and Mobile Computing*, vol. 8, pp. 385-403, 2008.
- [75] A. Kansal, A. A. Somasundara, D. D. Jea, M. B. Srivastava, and D. Estrin, "Intelligent fluid infrastructure for embedded networks," in *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, 2004, pp. 111-124.
- [76] G. Anastasi, M. Conti, E. Gregori, C. Spagoni, and G. Valente, "Motes sensor networks in dynamic scenarios: an experimental study for pervasive applications in urban environments," *Journal of Ubiquitous Computing and Intelligence*, vol. 1, pp. 9-16, 2007.
- [77] L. Pelusi, A. Passarella, and M. Conti, "Encoding for Efficient Data Distribution in Multihop Ad Hoc Networks¹," *Algorithms and Protocols for Wireless and Mobile Ad hoc Networks*, p. 87, 2009.
- [78] K. Dantu and G. S. Sukhatme, "Connectivity vs. control: Using directional and positional cues to stabilize routing in robot networks," in *Robot Communication and Coordination, 2009. ROBOCOMM'09. Second International Conference on*, 2009, pp. 1-6.
- [79] K. Akkaya and M. Younis, "Energy-aware routing to a mobile gateway in wireless sensor networks," in *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*, 2004, pp. 16-21.
- [80] A. A. Somasundara, A. Kansal, D. D. Jea, D. Estrin, and M. B. Srivastava, "Controllably mobile infrastructure for low energy

- embedded networks," *IEEE Transactions on Mobile Computing*, vol. 5, pp. 958-973, 2006.
- [81] S. A. Awwad, C. K. Ng, N. K. Noordin, and M. F. A. Rasid, "Cluster based routing protocol for mobile nodes in wireless sensor network," in *Collaborative Technologies and Systems, 2009. CTS'09. International Symposium on*, 2009, pp. 233-241.
 - [82] C. Perkins and I. Chakeres, "Dynamic MANET on-demand (DYMO) routing," *draft-ietf-manet-dymo-26 (work in progress)*, p. 127, 2013.
 - [83] S. Gao, H. Zhang, and S. K. Das, "Efficient data collection in wireless sensor networks with path-constrained mobile sinks," *IEEE Transactions on Mobile Computing*, vol. 10, pp. 592-608, 2011.
 - [84] S. Mohseni, R. Hassan, A. Patel, and R. Razali, "Comparative review study of reactive and proactive routing protocols in MANETs," in *Digital ecosystems and technologies (DEST), 2010 4th IEEE international conference on*, 2010, pp. 304-309.
 - [85] D. Johnson, Y.-c. Hu, and D. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4," 2070-1721, 2007.
 - [86] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," 2070-1721, 2003.
 - [87] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE*, 1997, pp. 1405-1413.
 - [88] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM computer communication review*, 1994, pp. 234-244.
 - [89] S. A. Mohammad, A. Rasheed, and A. Qayyum, "VANET architectures and protocol stacks: a survey," in *International Workshop on Communication Technologies for Vehicles*, 2011, pp. 95-105.

- [90] T. L. Willke, P. Tientrakool, and N. F. Maxemchuk, "A survey of inter-vehicle communication protocols and their applications," *IEEE Communications Surveys & Tutorials*, vol. 11, 2009.
- [91] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, pp. 1254-1270, 2013.
- [92] J. H. Forsmann, R. E. Hiromoto, and J. Svoboda, "A time-slotted on-demand routing protocol for mobile ad hoc unmanned vehicle systems," in *Defense and Security Symposium*, 2007, pp. 65611P-65611P-11.
- [93] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, 2000, pp. 243-254.
- [94] A. I. Alshabtat, L. Dong, J. Li, and F. Yang, "Low latency routing algorithm for unmanned aerial vehicles ad-hoc networks," *International Journal of Electrical and Computer Engineering*, vol. 6, pp. 48-54, 2010.
- [95] S. Chaumette, R. Laplace, C. Mazel, R. Mirault, A. Dunand, Y. Lecoutre, *et al.*, "Carus, an operational retasking application for a swarm of autonomous uavs: First return on experience," in *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*, 2011, pp. 2003-2010.
- [96] W. Huba and N. Shenoy, "Airborne surveillance networks with directional antennas," in *IARIA International conference on Computers and network Systems, ICNS*, 2012.
- [97] A. I. Alshbatat and L. Dong, "Cross layer design for mobile ad-hoc unmanned aerial vehicle communication networks," in *Networking, Sensing and Control (ICNSC), 2010 International Conference on*, 2010, pp. 331-336.
- [98] O. N. Challa, "CubeSat Cloud: A framework for distributed storage, processing and communication of remote sensing data on cubesat clusters," 2013.

- [99] O. N. Challa and J. McNair, "Cubesat torrent: Torrent like distributed communications for cubesat satellite clusters," in *MILCOM 2012-2012 IEEE Military Communications Conference*, 2012, pp. 1-6.
- [100] O. Challa and J. McNair, "Distributed Computing on CubeSat Clusters using MapReduce," in *Proceedings of the 1st Interplanetary CubeSat Workshop*, Cambridge, MA, 2012.
- [101] O. N. Challa and J. McNair, "Distributed Data Storage on CubeSat Clusters," *Advances in Computing*, vol. 3, pp. 36-49, 2013.
- [102] P. P. Sundaramoorthy, E. Gill, and C. Verhoeven, *Systematic Identification of Applications for a Cluster of Femto-satellites*: International Astronautical Federation, 2010.
- [103] M. de Milliano and C. Verhoeven, "Towards the next generation of nanosatellite communication systems," *Acta Astronautica*, vol. 66, pp. 1425-1433, 2010.
- [104] P. Rodrigues, A. Oliveira, R. Mendes, S. Cunha, R. Garcia Von Pinho, C. Salotto, *et al.*, "GAMANET: Disrupting communications and networking in space," presented at the 64th International Astronautical Congress, Beijing, China, 2013.
- [105] K. Sidibeh and T. Vladimirova, "Wireless communication in LEO satellite formations," in *Adaptive Hardware and Systems, 2008. AHS'08. NASA/ESA Conference on*, 2008, pp. 255-262.
- [106] R. Radhakishnan, W. Edmonson, and Q. Zeng, "The performance evaluation of distributed inter-satellite communication protocols for cube satellite systems," in *The 4th Design, Development and Research Conference, Capetown, South Africa*, 2014.
- [107] R. Sun, J. Guo, E. Gill, and D. Maessen, "Potentials and limitations of CDMA networks for combined inter-satellite communication and relative navigation," *Int J Adv Telecommun*, vol. 5, 2012.
- [108] B. Chen and L. Yu, "Design and implementation of LDMA for low earth orbit satellite formation network," in *Embedded and Ubiquitous Computing (EUC), 2011 IFIP 9th International Conference on*, 2011, pp. 409-413.

- [109] W. Alliance, "WiMedia logical link control protocol," *WLP Specification Approved Draft*, vol. 1, 2007.
- [110] R. Radhakrishnan, W. W. Edmonson, F. Afghah, J. Chenou, R. M. Rodriguez-Osorio, and Q.-A. Zeng, "Optimal multiple access protocol for inter-satellite communication in small satellite systems," in *4S Small Satellite Systems and Services Symposium*, 2014.
- [111] E. Ekici, I. F. Akyildiz, and M. D. Bender, "Network layer integration of terrestrial and satellite IP networks over BGP-S," in *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, 2001, pp. 2698-2702.
- [112] I. F. Akyildiz, E. Ekici, and G. Yue, "A distributed multicast routing scheme for multi-layered satellite IP networks," *Wireless Networks*, vol. 9, pp. 535-544, 2003.
- [113] M. A. Bergamo, "High-Throughput Distributed Spacecraft Network: architecture and multiple access technologies," *Computer Networks*, vol. 47, pp. 725-749, 2005.
- [114] C.-T. Cheng, K. T. Chi, and F. C. Lau, "An energy-aware scheduling scheme for wireless sensor networks," *IEEE Transactions on vehicular technology*, vol. 59, pp. 3427-3444, 2010.
- [115] D. Gerhardt, M. Bisgaard, L. Alminde, R. Walker, M. A. Fernandez, A. Latiri, *et al.*, "GOMX-3: Mission Results from the Inaugural ESA In-Orbit Demonstration CubeSat," 2016.
- [116] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, *et al.*, "Delay-tolerant networking: an approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, pp. 128-136, 2003.
- [117] D. J. Mudgway and R. Launius, *Uplink-Downlink: A History of the Deep Space Network, 1957-1997*, 2001.
- [118] L. F. Van Hoesel and P. J. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches," 2004.
- [119] F. N. Ali, P. K. Appani, J. L. Hammond, V. V. Mehta, D. Noneaker, and H. Russell, "Distributed and adaptive TDMA

- algorithms for multiple-hop mobile networks," in *MILCOM 2002. Proceedings*, 2002, pp. 546-551.
- [120] W. T. Webb and L. Hanzo, *Modern Quadrature Amplitude Modulation: Principles and applications for fixed and wireless channels: one*: IEEE Press-John Wiley, 1994.
- [121] S. M. Bilal, C. J. Bernardos, and C. Guerrero, "Position-based routing in vehicular networks: A survey," *Journal of Network and Computer Applications*, vol. 36, pp. 685-697, 2013.